COMPARISON OF EXPLICIT AND IMPLICIT KEYWORDS TO CATEGORIZE GEOGRAPHIC INFORMATION SYSTEM PROCEDURES

by

Roland J. Viger

B.A. University of Toronto, 1992

M.A. University of Colorado, 2004

A thesis submitted to the Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirement for the degree of

Doctor of Philosophy

Department of Geography

2011

This thesis entitled:

Comparison of Explicit and Implicit Keywords to Characterize

Geographic Information System Procedures

written by Roland J. Viger

has been approved for the Department of Geography

_____

Professor Barbara P. Buttenfield, committee chair

_____

Lauren Hay, Ph.D., committee member

Date _____

The final copy of this thesis has been examined by the signatories, and we

find that both the content and the form meet acceptable presentation standards

of scholarly work in the above mentioned discipline

Viger, Roland J. (Ph.D., Geography)

Comparison of Explicit and Implicit Keywords to Characterize Geographic Information System Procedures

Thesis directed by Professor Barbara P. Buttenfield

The author designs and implements an approach that exploits semantically important information that is not ordinarily included in traditional information retrieval approaches to improve the handling of Geographic Information System (GIS) procedural software. In this approach, what are termed here *implicit keywords*, descriptors designed to recognize characteristics not explicitly recorded within the GIS procedure source code, are created and used in an automated, inductive process to organize a large set of GIS procedures to reveal meaningful groupings. The process uses the Self-Organizing Maps (SOM), a specialized artificial neural network, to create a two-dimensional representation of an input data set wherein topological properties of the input data set are preserved. Such maps are important tools for helping visualize, browse, filter, and evaluate a set of GIS procedures . Browsing, filtering, and evaluation help to improve human understanding of available GIS resources. By facilitating mechanisms for improved software sharing and exchange, the methods described here may guide future researchers in the selection of more appropriate procedures for a given task.

Through experiments of this dissertation, the author demonstrates that while using GIS commands as explicit keywords can produce helpful organizations of GIS procedures, development of implicit keywords can be used to moderate, improve, and specialize the results of the explicit keyword process. The results of the different experiments not only show the impacts of applying different keyword schemes, but bear witness to the fact that GIS functionality can be organized with consistent methodological rigor in potentially very different ways to reprioritize specific types of functionality.

Dedication

For my family, who have prepared me for and supported me through this journey. I think of this as their accomplishment as much as my own.

Acknowledgements

I thank my advisor and the rest of my committee for their efforts. I acknowledge the value of the encouraging environment provided by my employer, the U.S. Geological Survey. In particular, I thank my supervisor, Lauren Hay, for her support. I also acknowledge helpful collaborations with my fellow students, Jochen Wendel and Jeremy Smith, in the early days of this research.

Contents

List of Tables

List of Figures

1 Semantic Views of GIS Functionality

As the breadth of geographic data and procedures for manipulating them continue to explode, many users are overwhelmed by choices. As the sizes of sets of GIS resources (corpora) continue to grow, users are increasingly less likely to exhaustively search for the optimal resource for their needs, nor are users likely to resolve whether a *subset* of suitable resources is available. Further, there is generally no way for a user to analyze and organize a set in a way that is specifically relevant to their own needs. With the emergence of online repositories of data (including GIS procedures) that may be added to by any member of a community, traditional (static) indexes and catalogs become quickly obsolete, effectively making it harder to discover newer information.

This problem is significant for several reasons. Logistically, users may simply fail to find the appropriate resource and be forced to expend effort to develop something that already exists. A possible outcome of this redundancy is the creation of an unintended variation on a standard. Further, there is the possibility that the user is unable to support this creation, either due to cost or lack of technical capacity, with the result that the user goal is never realized at all. At a data processing and a scientific level, a user may not discover potentially superior or simply informative alternatives. In the case where the user attempts to redundantly recreate a pre-existing resource, the impact of a variation in methodology may not be understood and thereby negatively impact scientific understanding and the realization of the user goal. All of these scenarios represent failure to exploit pre-existing knowledge. This directly results in a hindrance to better management of geographic resources, as well as an impediment to the improvement of scientific understanding.

Therefore, there is a need for a way to analyze and organize these resources that can be adapted to different user communities, or even individuals. With this, a user can query resources to get a concise and relevant set of offerings. This problem of geographic information retrieval is analogous to

those associated with corpora of more traditional information that are expressed in natural language, either in a printed or digital form. To address the retrieval problem for natural language text, research in information and library sciences has produced methodologies to organize, filter, and compare large bodies of text-based documents. These methods approach the problem by attaching keywords to each element in a collection (i.e., each document in a set). The keywords are used to delineate an information space and locate individual documents within that space. User-submitted queries (formed from sets of keywords) can then be compared to sets of document keywords. Documents that are located near to the query in the information space are considered similar or matching.

This dissertation argues that traditional keyword-driven approaches for analyzing and organizing sets of text-based information such as journal articles are insufficient for dealing with software procedures used in geographic information systems (GIS), such as user-written scripts. These approaches use only keywords that are explicitly embedded in the source material. This dissertation designs and implements an approach that exploits semantically important information that is not included in traditional approaches. It does so by creating what are termed here as *implicit keywords*, descriptors designed to recognize characteristics not explicitly recorded within the source code of GIS procedure scripts. The author defines alternate sets of implicit keywords, each reflecting a different view of GIS functionality, and uses them to drive analysis of the same set of GIS procedures. Results are evaluated individually and compared with each other. By organizing a set of GIS procedures into an information space, clusters can be delineated and measures of similarity can be calculated, both of which are important tools for helping users visualize, browse, filter, and evaluate the set. This helps to improve human understanding of available GIS resources and the selection of more appropriate procedures for a given task. The approach will be assessed using GIS procedures pulled from a number of software libraries.

1.1 The Insufficiency of Traditional Keywords to Organize GIS Resources

Keywords are, of course, a critical component of the solution. They can be manually assigned by the document author or another person such as a cataloguing librarian, or can be automatically detected and extracted from the body of a document. An important concept underlying this type of approach is that keywords taken from a natural language have a meaning that is shared across all users, and this meaning is unique relative to other possible keywords. In fact, every keyword is usually assumed to be equally "distinct" from all other keywords. These assumptions are not always met, as will be discussed below. Another problem with traditional keyword-driven approaches is that the meaning of keywords can vary widely depending on the user or the user context. These approaches rely on the *de facto* standard provided by a shared natural language (such as English) to help ensure consistent understanding of the meaning of keywords. For users whose needs are not similar to those of the "norm" from which keywords are drawn, the utility of these approaches is greatly diminished. For example, the term "ice" is almost uselessly general to an Inuit person who differentiates dozens of kinds of ice.

Another problem with explicit keyword strategies is their application to natural language types of resources. Although keywords have been assigned to images and recorded sounds, they must be assigned manually or derived from some quantifiable characteristics because they cannot be directly extracted from the "document" itself. This significantly limits the ability to apply explicit keyword techniques to large corpora of non-text items. Using keyword-driven approaches for pieces of software is a special domain where this problem occurs. Although software is written in a computing language, the meanings of the tokens of such languages (for example, "if..then", "go to", and so on) have not been rich enough to support approaches based on keywords explicitly found within pieces of software. Alternate approaches use keywords extracted from documentation associated with the software to analyze and organize a set of software components, using the documentation as analogs for software

components and relying on traditional information retrieval approaches. This is a substantial limitation because many smaller GIS procedures do not have associated documentation. New approaches to handling non-traditional types of information, specifically GIS procedures, need to be developed. These approaches should be capable of being able to integrate user-specific valuations about GIS functionality because users from different scientific domains may have very different needs.

1.2 Research Question

The proposed research develops a strategy to assign and exploit *implicit keywords*, that is, keywords that reflect implicit, context-specific semantics of GIS software procedures. The implicit-keyword strategy will be applied to organize sets of GIS procedures, that is, script source codes that run within a GIS environment, into an information space. The question is whether the use of implicit keywords leads to a substantially different and possibly richer organization of GIS procedures. If so, this research will examine the structure and usability of changes in the resultant information spaces derived from different sets of implicit keywords.

1.3 The Vagaries of Language in Forming Definitions

Determination of the similarity between documents (or discrimination between them) is one of the rudimentary tasks that a robust organization of a document set must support. If three individuals ask scientific questions involving watersheds, it is not necessarily the case that the meaning of the term "watershed" is identical in all cases. A common way to ascertain whether the term has the same meaning is to examine the definition of the term in a natural language such as English. There is potential for multiple natural language definitions for single geographic feature concept. For example, a watershed is defined variously as "a divide" (http://dictionary.reference.com/browse/watershed, accessed November, 2008), "a region or area bounded peripherally by a divide and draining ultimately to a particular watercourse or body of water" (http://www.merriam-

webster.com/dictionary/watershed?show=0&t=1295295888, November, 2008), “the region or area drained by a river, stream, etc.; drainage area” (http://dictionary.reference.com/browse/watershed, accessed November, 2008), or “an area of high ground from which water flows down to a river” (http://dictionary.cambridge.org/define.asp?key=89360&dict=CALD, November, 2008). The first definition is obviously distinct, and each offers at least slight variation.

This demonstrates the problem of a polynym. Although an English speaker will likely be able to determine whether “a divide” is appropriate based on context, automated methods of assessment may not. With regard to the subsequent definitions, even human speakers/listeners may not be able to discern which is appropriate because of partial similarity between the definitions. Further, an important question in interpreting these definitions is the usage of terms that may themselves need definition. For example, what is the definition of “flow” and by which computational method should one assess which direction flow will follow at different points on the land surface in delineating a watershed?

Even if all three definitions are not worded identically, one might expect the term and the concept it represents to be consistent across the contexts of all three individuals. Alternatively, the definitions might use obviously different words and grammatical arrangements (what Chomsky (1965) calls “surface structure”) to describe the same concept (“deep structure”). Here, too, the term and concept might be expected to interoperate across the three contexts. There may also be the case where the definitions overlap for the most part, but exhibit some differences. This might result in a degree of interoperability that is partial or incomplete. As a final alternative, the definitions associated with the shared term, "watershed," could be very different, indicating different concepts altogether. In this case, there would be no shared conceptual understanding and there would be no explicit basis for interoperability across the three individuals using the watershed. In all cases, the similarity of the definitions of the watershed term is assessed by some kind of text analysis, carried out by algorithmically or by human interpretation.

Although it suffers many of the same problems as algorithmic approaches, human interpretation as a method for document discrimination is set aside as a topic in this dissertation. The justification for researching ways to algorithmically discriminate between objects based on natural language descriptions (for example, (Deerwester and others, 1990; Ampazis and Perantonis, 2004)) is obvious simply because of the wealth of text-based information that exists. One class of problem in this field of research includes the lack of a controlled or at least limited vocabulary with which to form descriptions, the likelihood of inconsistent usage of the vocabulary (what concepts do the vocabulary terms themselves represent?), and the existence of polynyms (terms with more than one meaning). While proper interpretation and reasoning about grammar and sentence structure is another problem, dealing with incomplete definition of a concept is perhaps the most difficult issue for automated methods to handle. By operating with a definition that is not minimally sufficient or explicit, only partial knowledge is available for reasoning about a concept. The development of adequate definitions capable of overcoming partial knowledge relies on many difficult requirements, including a consensus on what kinds of reasoning one can expect to apply (that is, what kind of questions can be asked?).

Returning to the scenario: if three individuals use GIS to model with watersheds and, more specifically, to create watersheds then there might be at least one and possibly multiple procedures for manufacturing representations of a watershed within the digital GIS environment. Even if the individuals concur as to the natural language definition and the ultimate intension of this definition, they may disagree about the specific procedure used to render a representation within a GIS. In this case, several new questions can be asked about the similarity among “watersheds.”

- At what point do differences in the procedural definitions indicate different concepts? Can these differences be used to reason about, compare, or even organize the concepts?
- Can differences in the manufacturing procedures be used to anticipate substantive differences (or assess the similarity) between the output digital representations?

This research focuses on developing a framework wherein the manufacturing procedure

becomes a definition of a geographic feature concept. This definition is articulated using the commands of GIS as its vocabulary. More than a tactic merely to avoid the difficulties of automating the analysis of natural language definitions, this alternative definition is a more specific basis for describing procedurally what a geographic feature is. Further, it may provide a more precise basis for organizing a set of geographic concepts into a mathematical framework that enables, among other things, the *assessment of similarity* between GIS procedures (and by proxy, the geographic concepts that the procedures make representations of) that would otherwise not be possible with natural language definitions.

1.4 Differences Between Natural and Procedural Languages for Creating Definitions

For the purpose of this dissertation, *natural language* is simply defined as human language (AAAI, 2008) or "a language that is spoken or written by humans for general-purpose communication" (Wikimedia Foundation, 2008b). Natural language is usually contrasted with *formal language*, which is defined as a set of symbols (constituting the "alphabet") and a grammar for assembling sequences of symbols into strings (Sakharov, 2008). In formal languages, some think of grammar as a set of functions that returns an output (for example, (Hill, 2008)). This dissertation proposes using a procedural language, a type of formal language, in place of natural language. A procedural language is defined as a type of computer programming language that explicitly gives a sequence of steps to carry out (Howe, , accessed January 2009). In the case of this dissertation, a step is a GIS command. An example would be a command that accepts a raster elevation data set and derives a new raster data set where every cell indicates flow direction. This might be referred to as the "flow direction" command.

An example of a procedural definition of a watershed might be:

- outlet = getpoint(user_mouse_input)
- flowdir_raster = flowdirection(elevation_raster)

- wshed_raster = influx(flowdir_raster, outlet)

The actual words used in this definition are not critical to understanding this example, but the grammar is. The reader should focus on the fact that each line contains one GIS command (found immediately to the right of the = sign) that produces an output (specified to the left of the = sign), based on inputs (specified within the parentheses of the GIS command). For this type of definition to work, there must be shared agreement on what the available commands are (that "getpoint," "flowdirection," and "influx" exist).

Several differences in natural and procedural definitions are worth noting. The first, and perhaps most obvious, is that natural language is so much richer and subtle than most (any) procedural or other computational language. Natural language definitions have the potential to be much more detailed and subtle than those given in a computational language. It should be acknowledged that the aspects of a concept that cannot be expressed in a computational language are simply lost in a digital environment. Natural languages, in addition to having very large vocabularies, are typically wielded based on unstated assumptions about the knowledge already held by both the speaker/writer and the listener/reader. The assumption of external knowledge allows for the use of implied meanings and incomplete specifications. As alluded to previously, when all parties actually do not hold the assumed knowledge, ambiguity about creation and interpretation of a definition increases. A procedurally-specified definition is, by design, complete and is to be evaluated solely based on the explicit content of the definition.

Although the reduced descriptive power afforded by a procedural definition can reduce the accuracy of a user's understanding of an underlying concept, it can be argued that within a GIS environment the procedural definition is a direct and complete definition of the concept. No interpretation of a procedural definition is needed because the method recorded by a procedure is complete. The procedure is the only thing used to create a GIS representation of the concept. This procedural definitions is proposed to be sufficiently meaningful to enable automated methods of

organizing and differentiating geographic information concepts.

Another significant difference is that the language of the GIS procedure is a logically and computationally rigorous syntax, whereas the natural language syntax can be open-ended. The only open-endedness with the language of the GIS is the meaning (implementation) of the words (commands). If two GISs provide substantially different versions of a command, like "flow direction," then this is likely to create confusion. For the sake of this research, it is assumed that this variability does not exist across GIS implementations (such as ArcInfo, GRASS, or GeoTools). Within natural language, words can have synonyms or near-synonyms, homonyms, or even context-variant meanings. In addition, words and definitions can be used to imply much larger, but unstated concepts or assumptions. It is not likely that these implications are universally understandable to other individuals, let alone to automated mechanisms used to compare definitions. Reasoning in a computational environment is much more constrained in that it is limited to explicit content. Although more limited, the increased rigor of GIS procedural language definitions provide the possibility of using GIS procedural definitions as the basis from which to analyze and organize geographic information concepts.

The two types of definitions (natural language and procedural) could be considered endpoints along a continuum of possible definitions types. It is certainly possible to express procedure using natural language. For example, one could translate the procedural definition above as: first, define the outlet through which all surface water drains from the intended watershed; next, using the flow direction raster, designate all cells whose flow ultimately passes through the outlet location as being part of the watershed. While this middle-ground type of definition might ease the burden of interpretation for the human speaker/listener, handling this using automated software tools is still problematic because the appropriateness of mapping text like “flow ultimately passes through the outlet” to the use of the “influx()” command may not be recorded or inferable by a computerized system.

Using procedural definitions as indicators of the meaning of concepts, methods for systematically organizing sets of procedural definitions are developed in this dissertation. Although the absolute meaning of individual GIS commands may not be given by the resulting organization, the relative meaning and differentiation of types of procedures is; and it is this differentiation that could be used to discriminate between the resulting features (and their underlying concepts). This structure is envisioned as a way to help human users understand, reason about, and select appropriate GIS procedures, and perhaps more importantly, geographic concepts. The research seeks to determine whether using user-selected (implicit) knowledge to inform the organization enhances the ability to reason about the organization and meanings of the geographic concepts.

1.5 Example Scenario

An example scenario is given here to provide a high-level description of the research design. The scenario examines the procedure used to delineate instances of a geographic concept (that is, to create a feature within a GIS). The individual steps in the procedure are referred to as *commands*. The ultimate set of procedures used for the dissertation research will be much larger than what is described here, in terms of number of procedures, the complexity of the individual procedures, and the number of commands. Table 1-1 lists three hypothetical GIS procedures for delineating a watershed. Each procedure uses three commands. Examples of a command are: 1) a function to derive a depression-filled version of an input raster elevation data set, 2) a function to derive a raster data set depicting flow direction, and 3) a function to derive a raster data set of flow accumulation. To use the jargon of information science, the procedures are “documents” and the commands are “keywords.” A keyword is referred to as “explicit” because it is actually present within the specification of the procedure.

Table 1-1 Hypothetical GIS procedures for delineating instances of the watershed geographic feature concept.

GIS Procedure	GIS command	GIS command	GIS command
$WatershedProcedure_1$	A	B	C
$WatershedProcedure_2$	A	B	D
$WatershedProcedure_3$	A	B	E

All three procedures (*$WatershedProcedure_1$*, *$WatershedProcedure_2$*, *$WatershedProcedure_3$*) use the same explicit keywords, "*A*" and "*B*," for their first two steps. Each procedure uses a unique third step. If one were to simply determine the proportion of explicit keywords common to each pairing of procedures, one might conclude that each procedure showed a 66.7 percent similarity with the other procedures. While this may be useful for a broad, synoptic characterization, it is not likely to help discriminate between procedures. Many other characteristics, such as the sequence of keywords, which could be added to the analysis, are not included for the sake of this example. This type of approach is essentially how traditional approaches to information retrieval work. In this application, the GIS commands serve as explicit keywords.

Missing from the example scenario defined so far is consideration of the similarity between the explicit keywords themselves. For example, if the keywords "*C*" and "*D*" were more similar to each other than to the keyword "*E*", then it should be the case that procedure *$WatershedProcedure_1$* and *$WatershedProcedure_2$* are more similar to each other than to *$WatershedProcedure_3$*.

There are a number of techniques for quantifying and presenting similarity between explicit keywords (such as GIS commands). One common approach is to derive quantitative measures of similarity and report the values for each pair-comparison in a table. While this is relatively simple to do and understand, the utility of this approach decreases drastically as the number of explicit keywords grows beyond the human reader's ability to keep track of the results. If an information space, such as the one shown in Figure 1-1, could be constructed and used to locate each of the GIS commands in the example scenario then measures of similarity between them could be quantified and more easily

understood by a human reader. From this kind of information space, one could gain a quantitative indicator that explicit keywords "*C*" and "*D*" are in fact more similar to each other than to the keyword "*E*." From this, one could justify the assessment that procedures $WatershedProcedure_1$ and $WatershedProcedure_2$ are more similar to each other than to $WatershedProcedure_3$.

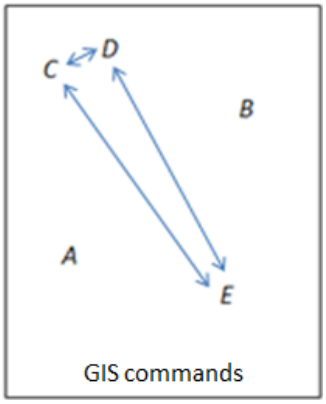


Figure 1-1 Hypothetical clustering of GIS commands where proximity indicates similarity.

By adding a layer of description to the explicit keywords, an information space can be constructed and similarity of explicit keywords can be explored. The assessment of the similarity of entire sequences of GIS commands could be built on top of an assessment of similarity between individual GIS commands (the explicit keywords). Further, this new layer of description can be used to change how GIS procedures (built from GIS commands) are themselves described and compared. The new layer of description serves as a way for a user to add arbitrary judgments about explicit keywords to the process of evaluating similarity.

In typical keyword-driven approaches, there is no “layer of description” about the explicit keywords which can be used to organize or relate them to each other. Therefore, developing this new layer to improve understanding about explicit keywords and then using that understanding to better organize a set of GIS procedures are two contributions that this dissertation seeks to make. Another is to

organize the GIS procedures by directly analyzing the procedures themselves to establish GIS commands as explicit keywords, instead of relying on documentation about the procedures.

1.6 Dissertation Structure

The structure of the dissertation is as follows. Chapter 2 places the dissertation research in the context of relevant current and historical literatures. Three major areas are used to carry this out. The first describes the field of Information Retrieval and how that field thinks about creating models of information space. The second is a broad review of inductive methods for organizing information, dealing with both classic techniques of inferential statistics as well as methods of machine learning for working with information spaces. This subsection finishes by bringing the discussion of information spaces into geographic information science by addressing research in the area of spatialization. The third topical area focuses on thinking from within geographic information science about how GIS functionality should be viewed.

The third chapter provides a detailed description of the experimental design and the methods used in the dissertation. The focus of this chapter is on the definition of implicit keywords that better capture semantics and the description of how to use these keywords to develop the "new layer of description" of GIS commands with which to drive the analysis and organization of information spaces of GIS procedures. The results are presented in chapters 4 and 5. Chapter 4 will document the analysis of a set of GIS procedures using the GIS commands found within the procedures as explicit keywords. This set of results will serve as a datum against which to gauge the impact of developing and using descriptions of GIS commands to aid the analysis of the set of GIS procedures, which will be described in chapter 5. Because many of the methods used to produce the results in the two chapters are the same, chapter 5 will refer to chapter 4 for specification of technical details whenever possible. The final

chapter interprets and discusses the results, assessing the success of the framework and where it might be improved. This concluding chapter will also discuss the prospects for future use of the framework, including a discussion of how it might be put into use by various agencies and user communities.

2 Review of Strategies for Organizing Information Spaces

This chapter focuses on approaches for organizing information into structures that allow for computational methods for reasoning about geographic procedures. The chapter begins with a review, in section 2.1, of how current theory on information retrieval (IR) has approached developing information spaces from large bodies of weakly structured and unevenly described text documents to help with query and recall of relevant documents. The data used in the dissertation experiments, GIS procedures, are analogous to the documents, used in IR.

This review will demonstrate that although IR has been or is extending to other forms of data (sounds, images, and even geographic data), there have been no contributions from IR or geographic information science developing these ideas to deal with procedures for creating geographic information. Included in this review of IR are descriptions of various models of information space, a discussion of typical problems for these approaches—some of which were introduced in chapter 1, and motivations for simplification of information spaces.

The next section, 2.2, introduces a number of inductive techniques for carrying out simplification of an information space. This section distinguishes between classic inferential statistics (such as principal components analysis) and newer methods of machine learning (such as neural networks). The rationale for using machine learning as the dominant analysis technique in this dissertation is provided in a critique of the assumptions associated with classic inferential statistics. Spatialization is then introduced. Spatialization exploits concepts from geography to enhance the organization and visualization of data that may not actually have any spatial or positional characteristics. Spatialization has been adopted as a powerful tool for exploratory data analysis (EDA) in areas beyond the field of geography. This topic will briefly revisit the topic of simplifying information spaces, but from a more geographically-motivated perspective.

Section 2.3 describes efforts to manually define bases for organizing GIS functionality presented in the literature. These efforts provide theoretically or, in one case, empirically grounded frameworks for grouping and thinking about GIS commands. One focuses on the transformation of data content from one of four basic data models (images, themes, fields, and features) (Gahegan, 1996). Another is driven by a focus on how a GIS command transforms data based on the roles of space, time, and attributes (Chrisman, 1999). The final contribution has, through user surveys, developed groupings of "universal elementary GIS functions" (Albrecht, 1999). These differing views on GIS functionality are used to demonstrate that there is "more than one way to skin the cat." These contributions are presented to emphasize that there is no consensus on how to "best" organize GIS functionality.

2.1 Information Retrieval and Models of Information Space

2.1.1 Introduction

Although having a fully specified, enforced way to organize information that is contextually relevant is the ideal, the set of data to be used in the experiments described by this dissertation does not have such an organization. Therefore, an approach must be selected to create an information space based on readily defined characteristics of the data being used. Because one of the goals of these experiments is to compare the effectiveness of alternative sets of characteristics, the approach used to define the information space must be generic to some degree.

The problem of how to organize information has long been studied in traditional library and information science (LIS). Two major approaches for organizing information (in libraries) are 1) cataloging and classification and 2) alphabetical indexing languages (Chan, 1981; Wikimedia Foundation, 2008a). Bibliographic classification theory has been developed in conjunction with descriptive cataloguing techniques to help assign an item or document an index that serves as a locator or address within some kind of organizational, usually hierarchical, structure. Alphabetical indexing attaches one or

more entries from a list of subject headings or a thesaurus to a document. The former approach is not as important in the digital environment because it was largely developed for the physical placement of a book on a shelf, which can only be at a single location—essentially a crisp classification. The latter approach is more directly useful for digital applications because it allows multiple terms to be associated with a book, and because it does not necessarily entail a rigid organizational scheme, can be reorganized relatively easily.

Even armed with these contributions from LIS, it is apparent that the size of digital data sets or libraries are far too large, diverse, and dynamic for manually-defined catalogs, ontologies, or other kinds of hard-coded knowledge-based systems to be feasible. Some researchers contend that the amount of data collected doubled between 1999 and 2002 (Lyman and Varian, 2003). Separately, computer science, and in particular database theory, gave rise to the application of inductive statistical methods for data mining by the mid-1990s (Chen and others, 1996), enabling machine-learning approaches to organizing information.

The remainder of the section will be organized into three areas: 1) an overview of information retrieval that includes the major approaches defined in IR for organizing information, 2) a discussion of problems commonly found with these approaches, and 3) an introduction to how researchers in this field began to realize the need for simplification of information spaces. The theory encompassed by this review does not assume that the set information to be worked with has already been organized into a structure or that an externally defined organizational structure is used to define interaction with that set.

The primary focus of all the reviewed approaches is to organize a set so that relevance between any possible query and the contents of the set can be established efficiently and effectively. Efficiency refers to the time to return an answer and how many items are recalled by the query (more responses are considered more efficient). Effectiveness refers to the proportion of the recalled items are actually

useful with regard to the query. Effectiveness is generally improved by more accurately measuring the similarity of what the query describes with items in the set.

2.1.2 Information Retrieval

The text in the following section covers the theory specifically dedicated to working with unstructured, text-based data sets, an area that is somewhat under-referenced in the geographic information science literature. Specifically, IR is an interdisciplinary field related to library and information science, computer science, linguistics, statistics, and cognitive psychology, among others. IR overlaps with several other types of retrieval research (data, document, text, etc.). The crux of all information retrieval is taking terms from a user-specified request and matching them against those keywords assigned to or found within the text of a collection of documents. Various statistics have been devised to weight the value of query terms and document keywords, to assess the relevance of retrieved documents. The assumptions of these statistics rely on the assumptions of the particular conceptual model of the information retrieval space being used.

IR has defined or adopted several approaches for matching queries against collections of documents, using the term *model* or *space* to refer to the approach used. Van Rijsbergan (2000) states that there are four major models of information space: the vector space (Salton, 1989), the probabilistic, logical, and Bayesian net models. Each of these models have been implemented in working systems: vector space in the SMART system (Salton, 1971), probabilistic in the Okapi system (Mitev and others, 1985; Robertson, 1997), logical in the Hyspirit system (Rolleke, 1999), and inference nets in the INQUERY system (Callan and others, 1992).

Others also point to the earlier Boolean model, which is useful for teaching, because it is very simple to implement and understand (Soboroff, 2002). It does not appear to be an actively considered model within the IR research community because it retrieves only exact matches of the terms in a query,

provides no relevance ranking of retrieved documents (everything is just yes/no), provides no partial matches, can result in prohibitively complex query expressions (because of its Boolean logic language), and does not allow terms to be weighted (all terms are equally important). Despite this, the Boolean model is still the "standard' model for large operational information retrieval systems (Belkin and Croft, 1992). A Boolean query language can be used on top of a non-Boolean retrieval model.

The remaining (non-Boolean) models are sometimes referred to collectively as "best-match" models (Belkin and Croft, 1992) because they offer an alternative to the "exact match" approach of the Boolean model. The Vector Space Model (VSM) (Salton, 1989) has been one of the more popular examples (Skupin, 1998). The VSM expresses both the document and query as vectors in a multidimensional space, with keywords serving as dimensions of the space. The VSM has been fairly prominent in geographic information science because of its overt spatial metaphor. Statistics, such as the cosine similarity coefficient, can be calculated to evaluate the similarity between the two vectors. The statistical values can then be used to rank the degree to which documents match the query. The VSM has been extended to allow keywords to be weighted based on how often they appear in a document or the collection containing it. One of the earliest such statistics is IDF (inverse document frequency) term weighting (Sparck-Jones, 1972), usually applied as part of one of many possible forms of tf*idf (term frequency-inverse document frequency) weight (e.g., Salton and Buckley, 1988). The thrust of these statistics is to measure the importance of a term in a document, but also to account for the scarcity of the term across the entire collection. This captures not only the match of the document to a query term, but also how unusual that term is over the entire collection (indicating that the term match is a better way to discriminate against documents matching other, more common query terms). Advantages of the VSM are that it produces partial matches, as well as rankings of matches.

Probabilistic models rank documents in decreasing order of relevance to the query, which is based on the Probability Ranking Principle (PRP) (Robertson, 1977). The defining feature of this model is

similarity measures are based on probability theory, as opposed to using a more geometric analysis like the cosine similarity measure between documents. User feedback can be used to refine measures. The probabilistic model explicitly accounts for uncertainty in the retrieval process (Soboroff, 2002), specifically about how well the query represents the user's need and how well the text surrogates (that is, the keywords) represent a document. Other models, such as VSM, only account for the uncertainty about the match between the query terms and the document keywords, and even then derive weightings based on frequency distributions. PRP can use other forms of evidence to calculate uncertainty.

The inference net (IN) model is a type of probabilistic model based on Bayesian inference networks (Pearl, 1988). Belkin and Croft (1992, p. 33) describe it as a "directed, acyclic dependency graph in which nodes represent propositional variables or constants and edges represent dependence relations between propositions." They state that the major difference between INs and other types of probability models is that INs use multiple sources of information to determine probabilities.

2.1.3 Problems with Keywords

These traditional approaches rely primarily on keywords that are, in most cases, manually assigned to each document. While this is effective for organizing large lists of documents, it suffers a variety of weaknesses. Perhaps first and foremost is the cost of having someone assigning the keywords to a document and cataloging it. Next is the fact that documents (or other kinds of items) are being created more rapidly than they can be catalogued. The approach described in this dissertation seeks to overcome this limitation by assigning characteristics to a relatively small set of GIS commands (like those identified by Albrecht (1999) as "universal elementary GIS functions"), and then use this information to characterize a potentially large set of procedures built out of those commands. These procedures are analogs to the documents typically handled in the IR literature, but do represent a substantially new

kind of data. Developing an understanding about this new kind of data for the purpose of information retrieval is part of the purpose of this research.

An additional problem with traditional keyword driven approaches is one of user interface. Fabrikant and Buttenfield (2001) cite Maudlin (1991) as giving the term *keyword barrier* to the fact that keywords must be known in advance of the query construction. This has been noted by other researchers, such as (Furnas and others, 1987). If a user does not know the appropriate keywords to use, their query may not be successful. Once appropriate keywords are employed by a user to form a query, relevant results must identified and, typically, used to further refine keywords and subsequently narrow the result set (Shneiderman, 1998). Put another way, the user (or system designer) faces the problem of being overwhelmed by a query being too successful—possibly because terms in the query are too commonly occurring.

Keywords also suffer a variety of weaknesses related to linguistics. Although not explicitly discussed in Fabrikant and Buttenfield (2001), language is constructed based on mental abstractions of reality, it contains the inherent structures, biases, and omissions of these abstractions (Jackendoff, 1992; Tversky and Lee, 1998). One result of this is that language can constrain questions that can be asked. It is interesting to note the circular relationship that cognition and language have to each other, at least as demonstrated through notable pieces of literature. The title of Tversky and Lee's 1988 contribution, "How space structures language", inverts that of Talmy's (1983) "How language structures space." Many authors have noted issues arising from the ambiguity of vocabulary, such as synonymy, polysemy, anaphora (a grammatical substitute, such as a pronoun), metaphors, and analogies (Furnas and others, 1987; Deerwester and others, 1990; Marchionini, 1995; Fabrikant and Buttenfield, 2001). Furnas and others (1987) contended keywords are not adequate discriminators of semantic content. Bartell and others (1992) note the oftentimes many-to-many relationship between keywords and the things they refer to or represent. These factors combine to place a heavy burden on the selection of

keywords, both by the information seeker and the information cataloger. They result in poor retrieval precision (Deerwester and others, 1990).

As a result, research on information retrieval and handling has looked to supplant or at least augment the shorthand of keywords with means that are semantically richer, and therefore intuitive, to the information seeker. One approach is by the explicit inclusion of human cognitive principles into the interfaces to information stores; designers are exploiting higher-level cognitive concepts of the information consumer (Dervin, 1983; Marchionini, 1995). Within the area of integrating GIS and environmental modeling, Viger (2004) has proposed presenting geoprocessing methodologies in ways that correspond more directly to the abstract concepts of the user and the environmental models that they seek to use.

Use of an alternate, lower-level language based on the commands of GIS as the list of possible keywords for describing items (in this case, geographic procedures instead of text-based documents) is proposed by the dissertation author. Although augmentation of explicit GIS commands keywords with what are termed “implicit keywords” will be evaluated, the author will also test whether using only the explicit GIS command keywords improves the ability to discriminate GIS procedures. This can be done because GIS command-keywords are more explicit and precise than natural language-keywords. At a conceptual level, GIS commands correspond more directly to the digital representation of geographic concept (that the GIS procedure attempts to depict) than do keywords derived from natural language that describe the mental abstraction of the concept.

As is the case in the field of IR, the experimental approach used in this dissertation moves away from *a priori* taxonomic conceptualizations, common to the geographic information science literature. This design is distinct from traditional approaches in IR in that keywords are weighted not only by the frequencies of their occurrence within a single procedure or over the entire set of procedures, but based on context-specific attributes expressed through implicit keywords. Put another way, the experimental

design of this dissertation uses more adaptive and inductive approaches to the development and usage of keywords.

2.1.4 Simplification of Information Spaces

Traditional approaches for document retrieval in IR use linear associative methods (Jardine and van Rijsbergen, 1971) that examine each document in a set individually, calculate the degree of its association with the submitted query, and then rank the results. Calculation of these associations is costly and possibly intractable for very large numbers of documents. The concept of clusters or document groups was developed as a tactic to reduce the number of items that needed to be compared with a query (e.g., Salton, 1971), thereby improving the efficiency of an algorithm. According to this concept, a query would not be compared with all documents individually. Rather, it would be compared with a representative document, or a synthetic document (representing a *centroid*), for each group. Once the "best-match" group for the query was identified, all the documents associated with the group could be retrieved and linear associative retrieval methods could then be applied to the documents associated with the winning group.

Researchers realized that the use of groupings not only reduced computation times, but also improved the relevance (sometimes expressed as the *effectiveness*) of the retrieved documents. Jardine and Rijsbergen (p. 219, 1971) proposed the cluster hypothesis, "that the associations between documents convey information about the relevance of documents to requests." This was an important evolution in that prior to this idea of "joint-relevance," documents in a set were only individually compared with the query and were not compared with each other. The characteristics of documents were beginning to be seen as indicating relative position, grouping, and separation of documents. IR was moving to the use of document characteristics to construct an information *space*. Jardine and Rijsbergen (1971) proposed a hierarchical organization of their groupings (they referred to the groups as

clusters). This graph structure can also be referred to as a taxonomy or dendrogram. Another important point about this evolution is that the distance between the points in the information space provided some indication of similarity between the points.

Salton (1989) mentions that in addition to the inefficient spreading of a document description across a number of keyword index files, traditional approaches for storing documents (or even descriptions of the documents) make no guarantee that similar documents will be positioned in any sort of proximity in the file system (nor, therefore, "will similar documents be presented "nearby" to each other in the output lists shown to the user?"). Salton (1989) continues to point out that if users are to browse a collection, then similar items should appear close together. He (and others) suggested that browsing becomes possible when the set is grouped or clustered. For geographic information scientists, these thoughts are echoes of Tobler's Law (1970): everything is related to everything else, but near things are more related than distant things (p. 236).

Most of these authors were working in a period of constrained computing resources and did not have readily available software for spatialization or visualization. As a result, they were not contemplating visualization metaphors, cartographic or otherwise, for users of a document collection. Nor were they thinking explicitly of database theory that espoused the separation of the view of a data set from the structure used to store the data set, as described in ideas such as the three-tiered schema model (Tsichritzis and Klug, 1978) or designs for federations of autonomous databases (Sheth and Larson, 1990). Despite the fact that there was not an explicit idea to separate presentation from data structure, the grouping concept was a step in this direction. Regardless of the IR model used, it signaled thinking about methods that simplify an information space. The idea of browsing pointed toward exploratory data analysis.

2.2 Inductive Methods for Organizing Information

Usages of the term *induction* have implied a number of meanings. Pierce defined it as the conclusion of "facts, similar to observed facts, are true in cases not examined," meaning to characterize something without prior knowledge (Gahegan, 2000). In terms of statistical implementation, induction is used to refer to a confusingly diverse array of terms, including clustering, data clustering, cluster analysis, segmentation analysis, taxonomy analysis, and unsupervised classification, to name a few (Gan and others, 2007). As used here, inductive techniques are distinguished from techniques that need to be trained using data that has been previously classified. Although some inductive techniques may also require training, there is no definition of type or grouping in the training data. Terms describing the types of techniques alluded to here include unsupervised, empirical, a posteriori, data-driven, pattern-recognizing. Some of these techniques are self-optimizing, while others are applied using trial-and-error (or at least require an optimization method that is external to and controlling the technique).

The dissertation author seeks to develop ideas that help differentiate types of GIS procedures based on different sets of descriptions of individual GIS commands. Rather than using techniques to find procedures of a specific pre-defined type, author seeks to use techniques that exploit descriptions to discover the number and characteristics of the types. This is motivated by the desire to test whether an inductive approach can be used to generate meaningful groupings of GIS procedures based on context-specific descriptions. Supervised classification schemes, by their nature, require an a priori definition of types and are therefore not relevant for this work.

As is the case for IR, these methods use a set of data (such as documents or GIS procedures) as input. Each data item has a set of descriptive attributes. The set of data and the associated attributes are usually organized into a data matrix, with attributes functioning as the columns in the table. It is convenient to think of creating an information space where each attribute column constitutes a dimension of the space. Because similar data points will have similar values for given attributes, the

position of the points in the information space will be closer than points with dissimilar attribute values. An information space of this type can have a large number of dimensions, too many for a person to be able to assimilate easily. Simplifying the information space by reducing the number of dimensions is one of the most common approaches to improve ease of use for humans.

The following subsections (2.2.1 and 2.2.2) describe several multivariate techniques for Exploratory Data Analysis. EDA developed as way to use a data set to constrain the development of a hypothesis about that data set to those that are actually testable using that data set. This was motivated by the belief of John Tukey (1977), among many others, that formulating hypothesis before examining the data limits the hypotheses formulated and could yield to false assertions of truth about a hypothesis. The dissertation experiments use EDA techniques to construct an information space about GIS procedures.

2.2.1 Classic Inferential Statistics

The following two subsections highlight two important inferential statistical techniques. While there are of course many other unsupervised techniques available from inferential statistics that could be discussed, notably K-means, ISODATA, and maximum likelihood clustering (although supervised) methods, the two discussed below were selected because of their focus on the problem of the reduction of the dimensionality of information spaces. They also have a history of usage in the geographic information science community. Principal component analysis is used in the dissertation experiment. The second technique discussed is Multi-Dimensional Scaling.

Principal Component Analysis

Principal component analysis (PCA; Pearson, 1901; Hotelling, 1933) is a one of the simpler unsupervised vector space transformation techniques. It is commonly used as a tool for EDA to reduce the number of dimensions associated with a multivariate data set, especially when the dimensions are

related to each other (dimension independence is usually one of the more important, but unrealistic assumptions of inferential statistics). It seeks to retain as much of the variation in the original data as possible, while reducing the number of dimensions as much as possible. Conceptually, this is done by ranking the dimensions based on the proportion of the variance in the data explained by each and then eliminating the lesser-ranked dimensions (which have less explanatory power). The analysis produces a list of principal components (the reduced dimensions) that are uncorrelated with each other and are ordered in terms of ability to explain the variation in the input data (Gan and others, 2007). PCA does this by using eigenvalue decomposition of a data matrix. An important characteristic of this analysis is that it assumes the dimensions to be orthogonal.

Factor analysis and the Karhunen-Loeve tranforms (KLT) are computationally similar to PCA, in that these methods attempt to describe the variation in an input data set using a smaller number of dimensions (termed *factors*). They differ in that they provide an error figure for each input dimension, where PCA assumes a constant error value across all dimensions.

The formulation of PCA implies assumptions that can have important impact on the results, the most important of which is that it assumes the output is a linear function of the attributes of the inputs. If non-linear phenomena are being analyzed, then some pre-processing should be applied to the data set prior to its input to PCA. Another important characteristic is that in calculating Eigenvectors, it subtracts the matrix mean of the distribution from the data set—which can impact the technique's ability to perform clustering because the magnitude of the distribution.

Multi-Dimensional Scaling

Multi-dimensional scaling (MDS) (Kruskal and others, 1978) was developed from the field of psychometrics. Torgerson (1952) is credited with the initial specification of the method and giving the term (Young, 1983; Kotz and others, 1988; Arabie, 1991). There are several types of MDS, usually categorized as *classical, replicated,* or *weighted*. Classical MDS uses a single, unweighted matrix.

Replicated MDS develops several unweighted matrices. Weighted MDS uses several weighted matrices. These types can have both metric and non-metric implementations. Metric MDS describe similarity by quantitative means and non-metric versions of MDS use qualitative descriptors. Metric MDS was initially proposed by Torgerson (1952) based on ratio data. This idea has since been broadened to handle interval data. The similarity measure used to populate the information space used as input to MDS is usually based on Euclidean distance between points within the information space. Although other measures can be used, Skupin and Buttenfield (1996) note that the selection of the proximity measure can have a large impact on the result of the entire MDS process. A popular alternative to the Euclidean similarity measure is the cosine angle, which uses the orientation from the information space's origin to an item's position within it (Wise, 1999; Pike and Gahegan, 2003).

ALSCAL (Alternating Least Squares SCALing) (Young and Lewyckyj, 1987) is one of the most common software implementations of MDS, used in other software such as the SPSS statistical package (Skupin and Fabrikant, 2003a). Fabrikant (2000) provides a detailed description of the entire MDS-based spatialization process in her dissertation. Another notable implementation of MDS has been developed at the US Pacific Northwest National Laboratory (PNNL). There the Spatial Paradigm for Information Retrieval and Exploration (SPIRE) project developed several transformation and visualization techniques. Their Galaxies and ThemeScape products built their information spaces differently, but both use the Anchored Least Stress (ALS) algorithm. ALS was developed as a workaround for "bottlenecks" with the traditional MDS approach that hindered its ability to work effectively on extremely large datasets. In this approach, the similarity of each document is not analyzed in pair-wise relation to every other document in the set, as normal, but relative to pre-processed topical clusters. This drastically reduces the computing costs of this algorithm.

Multidimensional Scaling (MDS) is suitable for handling relatively small data sets. Wise (1999) specifies an operational limit of 1,500 data items. MDS is normally used to generate scatter plot

displays. Skupin and Fabrikant (2003) note that although it is rare to see any other types of MDS-generated displays, using the output of MDS to create alternate displays (e.g. raster surfaces or Thiessen polygons) is possible.

Summary

Peter Gould's (1970) "Is Statistix Inferens the Geographical Name for a Wild Goose?" reviews some of the main problems with using inferential statistics, with a particular focus on their application to geographical problems. He points out that the form of the function between variables is most likely oversimplified. He also calls assumptions about the randomness, bias, and representativeness of a sample into question, as well as assumptions about the independence of observed variables, error terms, and the shapes of their distributions. Gahegan (2003) revisits (Gould, 1970) based on progress of thirty years that brings improvement in inferential statistics, the advent of machine learning techniques, as well as the rise of the availability of machines "to bludgeon apart difficult problems" (Gould, 1970, p. 442). Gahegan (2003, p. 70) quotes (Openshaw and Openshaw, 1997, p. 3), "Sadly, nearly all of the available methods for analysis, modeling and processing to extract value date from an earlier period of history where data were scarce and the analyst had to rely on his or her intuitive skills aided by an intimate knowledge of what little information was available to formulate analysis tasks".

Although Gahegan (2003) certainly does not recommend that inferential statistics are a fixture of the past, he does note that there are at least two substantial issues surrounding the usage of dimension reduction techniques such as MDS or PCA. Both of his criticisms of traditional inferential statistics echo Gould's concerns. The first is that these techniques assume that the phenomenon of interest (such as a classification) can be discerned using a relatively small number of variables, which may not be true. He uses examples of several social-change processes to illustrate that many different attributes may be required to explain changes. Likewise, a minimum number of data points (GIS

procedures, in the case of this dissertation) may also be required to allow the algorithm to resolve what is important and thereby appropriately reduce dimensions in the information space. His second criticism, while not as problematic for the type of data being used in this dissertation, is that these techniques assume that the variance in the relation between the attribute values of the training data and their corresponding classifications is constant. This has more importance if the data used for learning is actually spatial. In this case, the variation in the relation of the attribute values of a data point to their ultimate classification is likely to vary as a function of the data point position in space. Because this spatial variation is likely to be of fundamental interest to geographers, this assumption by these techniques essentially discards important information available in the data.

Creation of high-dimension information spaces were once computationally intractable due to the large and complex input data setes, although advances in computing power have reduced this burden. Gahegan (2003) states that techniques designed expressly for the purpose of reducing the dimensionality of an information space were often motivated by a now "outdated need for computational simplicity". Instead, he promotes "inductive machine learning" as a modern alternative to these classical inferential statistics. Although he reasons (correctly) that the "classic" kinds of simplification inevitably cause the loss of explanatory power of the data, it is not apparent that more "modern" techniques of machine learning entirely avoid this problem.

2.2.2 Methods of Machine Learning

Although term *inductive machine learning* is somewhat loosely defined in Gahegan's (2003) text, it corresponds directly with the term *heuristics methods* as used in the computer sciences and, in particular, in Artificial Intelligence (AI). The fields in which machine learning research is carried out have yielded powerful algorithms for heuristic methods, such as (rule-based) decision trees, neural networks, and genetic algorithms, that have overcome issues outlined in the previous subsection. Heuristic

techniques are “robust in the presence of noise, flexible in the statistical types that can be combined, able to work with attribute spaces of very high dimensionality, require less training data and make fewer prior assumptions about data distributions and model parameters” (Gahegan, 1999, p. 205).

This section will focus on artificial neural networks because these methods have particular strengths not only for dimension reduction and finding groupings of similar data, but because they also have a strong capability for visualization (which has also been popular within the geographic information science literature). The particular form of artificial neural network will be used in this dissertation is the self-organizing map, which will be detailed in its own subsection.

Decision trees are not necessarily suitable for the problem being addressed in this dissertation for a number of reasons. The chief one is that the number of close/near/similar relation that a data item can have with other data items is severely constrained by the network topology of a hierarchical tree. Genetic algorithms, which specialize in iterative search/optimization, are not generally used for data reduction, classification, or visualizations.

Artificial Neural Networks

An artificial neural network, also referred to as a neural network (NN), is a mathematical structure inspired by biological neurons. Kohonen (1988) credits the original theory to McCulloch and Pitts (1943), with additions for adaptive stimulus-response by Farley and Clark (1954). The idea was expanded by Rosenblatt’s (1958) concept of the *perceptron*, among many others. Current research and software development approaches to artificial neural networks have moved away from maintaining fidelity to the original biological structure as an analogy. A NN is a set of nodes (also referred to as neurons, processing elements, etc) arranged into at least two, and usually three, layers: an input and output layer, plus one or more hidden layers (Figure 2-1). Neurons are connected in a massively parallel form. The primary function of a node (neuron) is to transform input signals it receives and send the transformed signal to other neurons. This function is non-linear, and so are the functions that control

the signals that it receives, causing the whole NN to be highly non-linear. Each connection into a neuron has a weight ranging between zero and one. The capability/knowledge of the NN is contained in the set of weights associated with each connection (Zahedi, 1991). The weight of a particular connection indicates to what degree the "from" neuron excites the "to" neuron.

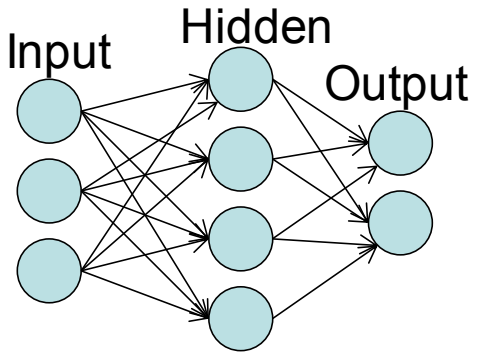


Figure 2-1 A neural network showing nodes (neurons) in the input, hidden, and output layers. The arrows depict the connections between nodes (neurons).

Before using a NN, neuron connections and their weights must be trained through an iterative training process. The process ends when there is no change in the network or it falls below a set threshold. It is this training process that is the strength of this approach and the central research goal of neural network theory. Many different learning processes exist. Because the system is so complex, most use a form of *backpropagation*. The backpropagation concept finds the difference between expected and actual outcomes of the NN. That error figure is propagated as a signal that moves back into the network from the output layer into preceding layers, resulting in the adjustment of neuron weights. Kohonen (1988) points out that this corrective action penetrates several layers into the NN, where classical approaches merely adjust the previous layer—and that such a localized solution fails to exploit the massively connected nature of the network.

Most flavors of neural networks start by using a subset of data items as a training set from which to iteratively develop weightings associated with the connection between each neuron in a layer

and a neuron in the next neuron. It should be noted that NN can be trained in a supervised fashion, as well as an unsupervised one. Unsupervised learning variants are desired for usage in this dissertation because the "types" or "categories" of GIS commands and procedures are not known ahead of time (or if they are, the experiment seeks to determine whether their existence can be corroborated inductively).

Therefore the general type of unsupervised learning process that will be highlighted here is characterized as adaptive learning because the connections and the weightings change as the individual data items in the training data set are evaluated by the NN in its training phase. The resultant network is described by some as a set of *learned responses* (Kohonen, 1988). Once a neural network has been trained, it can be used to group items in a set (in the case of this dissertation, GIS procedures).

Self-Organizing Maps (SOMs)

Self-Organizing Maps (SOMs; Kohonen, 2001), also known as Kohonen Networks, are a type of neural network. Though there have been many variations on SOM (see Kaski and others, 1998b; Oja and others, 2003), the original form of SOM shares important characteristics with PCA and MDS, chiefly that it can be used to visualize results in a spatialized display. Kohonen (p.1, 1998) states, "[SOM] implements an orderly mapping of a high-dimensional distribution onto a regular low-dimensional grid. Thereby it is able to convert complex, nonlinear statistical relations between high-dimensional data items into simple geometric relations on a low-dimensional display. As it compresses information while preserving the most important topological and metric relations of the primary data items on the display, it may also be thought to produce some kind of abstractions. These two aspects, visualization and abstraction, can be utilized in a number of ways…"(Kohonen, 1998, p. 1)

The SOM algorithm will organize the spatial arrangement of neurons in the network so that neurons of similar character are closer to each other. Kohonen (1998) describes the SOM algorithm as a nonparametric, recursive regression process that yields a graphical output that can be interpreted as a similarity graph. The distinguishing feature of SOM, aside from being a relatively simple algorithm, is

that the network will *self-organize*. The learning concepts in NN, referred to as learning vector quantization (Ahalt and others, 1990), have no concept of adjacency or neighborhood between neurons (Cottrell and others, 1998) as the SOM technique does. Cottrell and others (1988) describe the SOM learning process as occurring in two phases: the first is the arrangement of neurons, the second in which the weights are adjusted to achieve convergence (an optimal solution).

During the first phase, an adaption or *plasticity* parameter (Kohonen, 1988) is relatively large, allowing the SOM network to reconfigure itself. Simply, when a neuron is found to match an input data point and is having its weights adjusted to more closely mimic the dimensions of the data point, neighboring neurons are also adjusted. The intensity of the adjustment is parameterized by this plasticity parameter, representing a distance-based decay of the influence modification carried out at the matching neuron. This function, referred to here as the neighborhood function (composed of a neighborhood radius and a distance decay coefficient), tends to create a spatial separation or grouping within the reduced-dimension map space of "types" of neurons in the resultant information space. Once the spatial arrangements are set and the learning process is in its second phase, the plasticity parameter decreases to zero so that all adjustments are occurring at a single neuron only—thereby preserving the spatial ordering but adjusting the weightings at that neuron. Although NNs in general have the idea of plasticity, it is only usually applied to a single neuron.

Once training is complete and the network is defined, each element (node) in the map space can be viewed as a discrete category (although treating these as points a continuum would be more biologically faithful). This forms the SOM. The remainder of the original data set can then be quickly associated with a single neuron in the SOM. The SOM can be visualized a variety of ways, colorizing the attribution of each node, such as cluster number. An important characteristic of the SOM is that the individual data items from the original data set are easily placed directly into the display, allowing visual assessment of how well the differentiated regions in the SOM correspond with individual data points. In

addition, the spatial arrangement of neurons in the SOM preserves all of the topological relations found in the original high-dimension information space, such as relative distance. This means that not only do like items appear in the same groups, but neighboring groups exhibit some degree of similarity in the SOM-space as they did in the original information space defined by the dimensions of the inputs data set.

Although SOM displays are map-like, the technique does not attempt to create geographic representation (Skupin, 2002a). A SOM should be understood as a relative-location or conceptual (Douglass, 2004). As mentioned above, the SOM is usually visualized as a two-dimensional display that fully tessellates the display space, effecting a raster-like presentation. The neurons can be thought of as cells in a grid, or pixels on a screen. The geometric form of the ultimate SOM visualization (that is, grid, honeycomb, and so on) can drive the type of neighborhood that is needed during the training of the SOM map. As with other visualizations of high-dimensional visualization techniques, each axis in a SOM display is potentially associated with a large number of attributes within the (training) data set. Interpreting and labeling the dimensions can be one of the more difficult parts of the user experience for this (and many other) dimension-reducing approaches.

The SOM process is well suited for handling large data sets (in contrast with MDS), both because its computational cost does not necessarily grow exponentially as the number of items increases and it is an unsupervised algorithm. Although not needed for this dissertation, SOMs are especially well suited to handling temporally changing data sets because the map, once trained, only has to classify new data according to the previously developed network. While the cost of the training phase is important, it is usually a one-time task. In the experiments of this dissertation, SOMs will first be trained using explicit keywords associated with GIS commands, and then with implicit keywords.

The SOM technique, although somewhat complex, is one of the more flexible data reduction techniques that are widely used. Not only does the SOM approach provide a wealth of “local”

information about which items are neighboring (that is, the most similar), it also provides a good sense of global organization across an entire data set. Typical clustering techniques tend to emphasize the local aspects of an information space at the expense of the global (Douglass, 2004).

It should be noted that arguments have been made in the literature (e.g., Bartell and others, 1992) that the SOM technique is functionally equivalent to several other techniques. Conceptually, this makes sense, especially if the SOM is being used as a black-box clustering tool and is being compared with techniques that attempt to achieve similar goals while facing similar constraints, such as the preservation of certain properties across a transformation. For example, although the approaches of SOM and PCA are very different, they both merge dimensions and minimize the error between the map dimensions and the actual data, and attempt to preserve the variations and relations between the original data items in the output. What SOM does do is dimension-reduction, grouping, and visualization using data sets with potentially noisy data with large numbers of records and attributes. PCA is used in this dissertation as a secondary analysis to aid in the interpretation of the SOM results.

2.2.3 Spatialization

Although the concept of an information space where the distance between points is an indicator of similarity is clearly not unique to geographic information science, this field has much to add to this conversation. Of particular relevance is spatialization, which geographic information scientists define as "the transformation of high-dimensional data into lower-dimensional, geometric representations on the basis of computational methods and spatial metaphors. Its aim is to enable people to discover patterns and relationships within complex *n*-dimensional data while leveraging existing perceptual and cognitive abilities. Spatialization can be applied to various types of data, from numerical attributes to text documents and imagery" (p. 418, Skupin, 2008). In particular, Fabrikant (2003) cites Kuhn and Bluementhal (1996), Couclelis (1996), Dogdge and Kitchin (2000), and Fabrikant and Skupin (2005) as

examples of researchers that have addressed information space handling and visualization. Skupin and Fabrikant (2003a) also review work by cartographers to bring geographic principles and cartographic techniques to the visualization of non-geographic information. They distinguish between research into computational approaches to visualization and those that deal with human-computer interactions (HCI) (for example, the cognitive processes of the user interpreting the results). Spatialization really encompasses (at least) two components, the more obvious visualization using a geographic metaphor, and more important geographically-informed approaches to knowledge organization discovery (Miller and Han, 2001; Skupin and Fabrikant, 2003a). A simplified point to make about the quantitative methods used in spatialization is that they (usually) seek to reduce the dimensionality of a complex information space in ways that preserves important relationships about the data points within it.

This chapter's review of PCA, MDS, NN, and especially SOM describes some of the techniques used in knowledge discovery. MDS has been used by a number of geographic information scientists (Skupin and Fabrikant both discussed or used this method in their dissertation work). Agarwal and Skupin (2008) just released a book comprised of essays on the use of SOM for applications in geographic information science. Because of this background, the current subsection will not revisit geographically-informed methods for knowledge discovery.

It is worth visiting the topic of data quality with respect to spatialization. Skupin and Fabrikant (2003a) specify three forms of data: structured, unstructured, and semi-structured. While these forms present crisp distinctions between the types of data, these categories are really members of a spectrum. The degree to which a data set is structured significantly impacts the process of spatialization. Spatialization itself entails a way to structure a set of data. A data set may eliminate some spatialization options and force others by the nature of its structuring or ordering. An example of structured data can be found in a table. Each row is a discrete item, and has a limited and exhaustive set of fields (attributes) of known value.

Although the spatialization of an unstructured data set has the extra burden of creating a structuring scheme and then locating data within that structure, it is the most flexible form in terms of choices related to spatialization. Unstructured data sets are essentially free form text documents or records. There is no attribute information (such as keywords) about the document, other than what can be inferred from the document's content. In addition to providing these examples, Skupin and Fabrikant (2003a) give as an example of semi-structured data a store of conference abstracts that isolate attributes such as title, keywords, and author from the body of the abstract text. The attribution of an individual abstract does not imply an organizational structure across the entire store of abstracts, although it does facilitate the construction of such a structure. A semi-structured data set is one to which generic templates can be applied to create a structure.

Creation of a High-Dimensioned Information Space

The term *characteristic* is used here as an umbrella term to include the extrinsic relations (such as cross-references) found in structured data sets, the intrinsic attribution (such as keywords) of semi-structured data sets, as well as information that can be surmised from the content of unstructured data sets. Generally speaking, a characteristic that an item has in common with another item indicates a similarity between the two items. The more shared characteristics, the greater the similarity.

Characteristics of each data point are collected, analyzed, and associated with some kind of attribute that can be recorded into a matrix. Each row of the matrix corresponds to a data item. Each column corresponds to an attribute. A keyword, as described previously, is a type of attribute. The matrix is used to construct a representation of the items in the data set as points in a multi-dimensional information space. The dimensions or axes of the information space correspond to the different attributes found in the matrix. The number of dimensions can be extremely large. For example, the SPIRE Galaxies visualization, using a thesaurus, yielded a 200,000 dimension information space (Wise, 1999)!

The vector space model (Salton and McGill, 1983; Salton, 1989) is a concept that is used to build this high-dimension information space. The vector space model is generally described as providing a metric space where proximity between items is a surrogate for similarity between the two. The location of a point (i.e. a data item) is plotted based on the value associated with each dimension.

In the case of a semi-structured set of conference abstracts (text documents), keywords associated with each document can be analyzed to detect term overlap. Where keywords associated with different abstracts match, there is a semantic similarity between the documents (Fabrikant and Buttenfield, 2001). Matrices can be constructed for all keywords associated with the abstracts in a set. If the data set is an unstructured set of text documents and no keywords exist, then a content analysis of each document can be carried out to extract the meaning bearing terms, which can then be treated as keywords. Latent Semantic Indexing (Deerwester and others, 1990) is a widely used example of this kind of content analysis (Pike and Gahegan, 2003).

Although details on how matrices are constructed can vary, they are generally used to derive some kind of similarity or dissimilarity measure. Some of the more popular metrics within the geographic information science literature include a squared Euclidean distance measure (Fabrikant, 2000), a cosine similarity coefficient (Pike and Gahegan, 2003), and a vector of a singular value decomposition (Deerwester and others, 1990). Calculating such a measure usually involves comparing keyword vectors at different locations within the vector space.

Wise (1999) states that, conceptually, “the exact means [of constructing the information space] is not important as long as a statistically ‘rich,’ reasonable [sic] sized dimensional representation results.” The resultant information space will contain a quantitative metric of similarity. Despite Wise’s contentions, the techniques employed can have obvious ramifications for efficiency and scalability. With 200,000 dimensions (as in the case of Wise’s SPIRE system), even today’s computing power can become scarce relatively quickly. Although the details are beyond the scope of this review, an additional caveat

is that the technique selection for the rest of the spatialization and visualization process can dictate how the information space should be assembled.

Geographic Perspectives on Data Reduction

Graphical rendering is inherently limited to two spatial dimensions, maybe three, depending on the selected presentation metaphor. Symbology (e.g. size, shape, texture, orientation, etc.) may also be used to give expression to a handful of additional dimensions (Fabrikant and Buttenfield, 2001). In order to use the newly created information space to render a visualization, the dimensionality of that space must be reduced to conform to the drastically limited dimensionality of the graphics space. This essentially means merging of large numbers of dimensions in the information space. This reduction is, obviously, a transformation of the information space. This transformation defines the coordinate system of the graphics space and the mathematics for projecting items from the information space into the graphics space.

The reduction of dimensionality must be done in such a way as to ensure that the spatial relationships between items in the high-dimensional space are in some way preserved in the space that will be used to generate a graphic (Wise, 1999). This process is analogous to a cartographic projection (Skupin, 2002b). Because distortion is inherent in any projection, the selection of a projection procedure will inevitably preserve some spatial relationships while deteriorating others. Skupin and Fabrikant (2003b) raise the complexities of this selection. They relate that a standard map projection at least has the benefit of inherent, physical, meaningful, ordered properties. The spatialization process deals with a space that has many more dimensions, none of which necessarily have any directly interpretable meaning. In addition, the natures of the alternative projection techniques are very different, including the methods for data preparation.

Prior to projection, an option is to use clustering techniques to adjust the positions of items within the original (high-dimension) information space into topical groupings, effectively simplifying the

distribution of items within the space. This is essentially a technical refinement that allows the data processing sequence to leverage the richer semantics of the original vector space to emphasize or concentrate groupings, possibly to minimize the pull of outlying data items during projection. The SPIRE visualization, for example, used K-means clustering and complete-linkage hierarchical clustering at different points in its history (e.g. Wise 1999).

Visualization

The purpose of visualization is to exploit the principles of cognition and vision that are essentially "hard-wired" into the human brain, enabling human users to better understand the wealth of relationships within the set where more traditional presentations of statistical summaries might overwhelm (Gahegan, 1999). By presenting a visualization that uses a spatially-informed metaphor, the user can view and explore an information space in a manner that is similar to viewing and exploring the physical landscape. This type of spatialized visualization not only leverages the general visual strengths of human cognition, but also a well-developed sense of geographic space and the meaning of relative positions.

Part of the purpose of this subsection is to emphasize that visualization is only part of the spatialization process, one that is relatively less important for this dissertation. Despite this, there are several fields of literature that have developed around visualization. Amos Tversky is at the intellectual center on the topic of evaluation of cognitive theory about the human understanding of space, for example (Tversky and Teiffer, 1976; Tversky, 1977; Tversky, 1981; Tversky and Hemenway, 1983, 1984; Tversky, 1993; Tversky and Lee, 1998). More recently, a field related to carrying out laboratory measurement of human responses to given visualizations has been growing. Examples of these activities include (Hartley, 1977; Goldstone, 1994; Fabrikant and others, 2006; Fabrikant and Montello, 2008; Fabrikant and others, 2008). Beyond the existence of journals like International Journal of Human Computer Studies, there have been special issues dedicated to empirical evaluation (Chen and

Czerwinski, 2000a) and attempts to test Tobler's Law in visualizations (Montello and others, 2003). The recentness of this last contribution is surprising because it seems to be a needed theoretical foundation to justify the spatial metaphor not only for visualization of information spaces, but for all of cartography.

What follows is a very brief listing of some of the earlier work that put forward visualization techniques. Rather than delving deeply into specific visualization paradigms, a survey will be made. Most of this work occurred in the early 1990s, coincident with increases in the power of computing platforms. Although there were many players in this time, there were a few centers of activity, notably the Xerox Palo Alto Research Center (PARC) and the campuses of the Bell Laboratories.

Ahlberg and Shneiderman (1994) developed Starfield displays, which were designed to provide overviews of data sets. The display was actively updated as a result of queries made by the user, allowing them to see "where" their results lay in relation to the rest of an entire database. This combined with the ability to incrementally adjust a query provided a highly interactive interface to a data set. Later work by Johnson and Shneiderman (1991) led to the development of TreeMaps for showing hierarchically organized data in a single, planar display.

Stuart Card, George Robertson, and Jock Mackinlay of Xerox PARC have created a large number of visualization concepts, including the three dimensional room (Card and others, 1991), the perspective wall (Mackinlay and others, 1991), the document lens (Robertson and Mackinlay, 1993), and cone trees (Robertson and others, 1991). The perspective wall was designed to create a smooth visual transition between a detailed view and the larger context of information situated around it. This was supposed to be an improvement over the already established concept of the fisheye view of Bell Laboratories' Furnas (1986). The cone tree was a three dimensional version of a traditional hierarchy. The integrating of the third dimension enabled more differentiation within a generation (i.e. level) of offspring. These researchers also developed ideas about moving the viewer's perspective through the visualization space in a rapid, but controlled manner. The document lens was a planform paneling display of concepts as

sheets of paper. It allowed a detailed view of multiple pieces of information simultaneously. In this era, researchers from Columbia University proposed Worlds within Worlds (1990), which featured nested three-dimensional visualizations. Each visualization was tied to the display from which it descended.

The SPIRE project developed the Galaxies display, which was effectively a scatter-plot. Another of their products, ThemeScape, used a landscape metaphor. Frequency of a topical cluster is indicated by elevation, as opposed to more points. ThemeRiver (Havre and others, 2000) is another visualization product from PNNL that has been designed to illustrate the linear progression of information over time and provide a visual metaphor for pattern detection and trend analysis. The Name Voyages interface is a similar visualization (http://www.babynamewizard.com/voyager, accessed November, 2008). Topic Islands (Miller and others, 1998) are yet another product from this group. It is worth mentioning because creates its graphics space using wavelet-analysis instead of more traditional approaches like MDS or SOM.

Visualization interfaces have been developed specifically for SOM (e.g. WEBSOM (Kaski and others, 1998a)). Perhaps more broadly interesting, GeoVISTA Studio (Gahegan and others, 2002) from the Pennsylvania State University is one of the more comprehensive efforts to assemble various transformation techniques, including SOM, with geocomputation, exploratory spatial data analysis, cognition, collaboration, machine learning, and visualization into a unified object-oriented framework. While still under development, it seems to be at the forefront of improving the human-computing interface for the creation and sharing of geo-scientific concepts. This platform goes far beyond the traditional target of visualization of matching geometric primitives against cognitive categories that underlie human understanding and representation of space.

The above selections represent some of the more influential ideas in visualization, but many more exist. Omitted were mention of icon displays, network, and Process Flow Diagram displays. A variety of researchers have attempted to provide taxonomies of visualization techniques (see Skupin,

2002b for a list), while others have attempted to discuss future research directions (Gershon and Eick, 1998; Skupin and Fabrikant, 2003b). Some researchers have focused on the quantitative evaluation of visualization techniques (Chen and Czerwinski, 2000b; Morse and others, 2000).

Again, the experimental design of this dissertation uses a spatial metaphor to guide the organization and interpretation of information spaces more than developing new ideas about visualization. It is usually possible to take the output of organization techniques and visualize them in a number of alternative forms. While some may be more valuable or understandable to users, the author seeks to develop better inputs to those organization techniques so that their output is more meaningful and will, in turn, yield more meaningful visualizations.

2.3 Frameworks for Organizing GIS Functionality

Because one of the research goals of this paper is to improve the organization of GIS functionality, this section will review approaches for organizing GIS commands, provided by several other authors, as a starting point. This will help assessing the quality of the alternative organizations of procedures (sequences of GIS commands) produced in the dissertation experiments. Chrisman (1999) provides a good overview of this topic that will be used here. He notes that several efforts in the 1980s yielded lists of commands that were considered fundamental or part of a minimally sufficient set of functionality needed to constitute a GIS (e.g., Guptill, 1988; Berry, 1989). Others include Tomlinson and Boyle (1981), Rhind and Green (1988) , and Goodchild and Bruesgard (1989). Several efforts enumerated major headings or groupings of commands, such as Dangermond (1982).

Map Algebra (Tomlin, 1983) is highlighted by Chrisman as an early taxonomy of at least some GIS functionality. Map Algebra also defined a language that allowed GIS commands to be treated as symbolic variables. It also extends mathematical operators to include spatial neighborhoods (local, focal, zonal, etc.), which form the dominant basis for the organization of Map Algebra commands. An

important characteristic of this approach is that it only deals with raster data sets, and therefore omits discussion of commands that operate on other data models and those that transform content from one data model to another. Chrisman points out that the value of Map Algebra as a taxonomy is diminished because of its focus on the syntax of its language and that the meaning of the commands is left unspecified. The ability to characterize or classify sequences of commands (like the procedures to be used in this dissertation) is weak if Map Algebra is the only available logic with which to work.

Subsequent efforts amended Tomlin's ideas (Berry, 1989) but are too software-specific to be considered generic. Others provided more generic schemes, but still suffered from being too concerned with the syntax used to specify sequences of commands (Chrisman cites (Hadzilacos, 1996) as an example). Goodchild (1987) proposed a framework for organizing GIS commands based on the data model of the GIS and the primitives (e.g., point, line, polygon) enabled by the model. More specifically, he felt that the types of primitives that are input to and output from a command should influence the placement of the method within the organizational scheme. He also proposed characterizing a command based on six different criteria that describe whether attributes and/or position are used and whether more than one type of object is involved, among other things. Goodchild (1987) explicitly notes that the "definitions [of the GIS commands] derive from operations on the data model, rather than from the cartographic meaning of those operations" (p.II-72).

Chrisman (1999) criticizes Giordano (1984) for using a hierarchical tree to organize GIS commands. Chrisman contends a tree is not an appropriate metaphor because it does not account for the transformation of data (in fact, Giordano assumes a constant data structure, as Map Algebra did). He also dismisses other attempts, such as (Burrough, 1992), as merely manufacturing a list of function names that could generically apply to different commercial products. To be fair, creating a generic inventory was likely the intended purpose of these efforts, rather than providing a pedagogical device or a way to understand the meaning of content created by sequence of GIS commands. Chrisman's main

point is that previous taxonomies, by focusing on "mechanical" characteristics such as what kind of math is used and depending on the metaphor of a math-like language (as was the case with Map Algebra) too heavily, miss the point. He proposes that the meaning derived from a sequence of commands is created by the transformations of the data.

The absence of Gahegan's (1996) "Specifying the transformations within and between geographic data models" in Chrisman's otherwise comprehensive review is conspicuous because it not only presents ideas that are somewhat similar to those of Goodchild (1987), but because Gahegan also explicitly invokes transformation as a central concept. Gahegan's piece stems from a different intellectual source than the cartography of either Goodchild (1987) or Chrisman (1999), instead extending the work of Pascoe and Penny (1995) on using communication interfaces for improved data exchange. This work appears to have developed using a perspective grounded in database theory. Chrisman's work is closer to "philosophical cartography" (if such a term can be coined).

As Chrisman did in his later publication, Gahegan (p. 137, 1996) points out that "any meaning inherent within a dataset is intrinsically connected to the model by which it was captured" and that the process of creating information, abstracting it from raw data, is formalized by the sequence of transformations (that is, the GIS commands) applied the data models. He characterizes the process of creating information as transforming imagery into thematic data and then to either or both field or feature/object data. The four different data models (image, thematic, field, and feature/object) he uses are in fact conceptual models of geographic space. These different data models do not refer directly to raster, vector, or any other geometric types or file structures. This is a helpful evolution of Goodchild (1987), in that it organizes GIS commands on the basis of a *view* of the content as opposed to the way the content was encoded into a database or file on a computer hard drive. Although Goodchild acknowledged the distinction between the view and the underlying file structure, his approach was still relatively tied to a list of geometric primitives (such as points, lines, polygons).

Perhaps another illustration that Gahegan (1996) and Chrisman (1999) are coming from different places intellectually is the fact that Tobler's (1979) "A transformational view of cartography" is cited only by Chrisman. Despite this, the two authors do agree that the impact of how data is modified by a command is an important characteristic for organizing a taxonomy of GIS commands. Chrisman specifically focuses on what he terms measurement frameworks, which is effectively an ontology for measurement. He develops this idea to formalize the spatial, temporal, and attribute characteristics of a measurement. From these, he argues, one can begin to assess the whether or how data gathered under one measurement framework might relate to data in another measurement framework, or how data might be transformed appropriately into an alternate measurement framework. Gahegan does not explicitly discuss the measurements or frameworks by which raw data such as imagery are created, as Chrisman does. The measurements that yielded what Gahegan treats as "raw" image data are treated as externalities that are unvarying in their nature, although he does acknowledge that temporal and uncertainty characteristics of the various forms of data are important and should be treated.

Jochen Albrecht attempted to identify a set of "universal analytical GIS operations" in a series of publications (e.g., Albrecht, 1994, 1995; Albrecht, 1999). He used user interviews and surveys to identify the operations (commands) in a GIS that are used for analysis, as opposed to data management. His exclusion of data management operations, representing something on the order of 80% of the work done with GIS (Albrecht, 1999), could be a subject for debate if some of these are considered transformative. The impact of this exclusion is not readily discernible because it is not clear how many of the 144 operations he originally identifies remain in his set. The operations are intended to be analysis concepts that are independent of particular data structures or GIS platforms. Details of Albrecht's organization are also presented in chapter four.

Note that all of these efforts attempt to organize individual GIS commands. None of them deal with GIS procedures (sequences of commands), as this dissertation seeks to. Albrecht (1999) probably

comes the closest with his definition of a task. He describes his adaptation of Huxhold's (1991) information pyramid, that represents *goals* (such as land management), is supported by *tasks* (like risk analysis). Tasks are built out of sets of *functions* (like spatial query). Functions in turn are built on *data*, referring to geometric types. His functions are analogous to the use of term *commands* in this dissertation. His use of the term task is (very) roughly analogous to use of *procedure* in this dissertation, although it is not an exact match. Albrecht's definition of task seems to change across publications, sometimes being defined (1999, p. 579) only on the basis of actions that require human input or "knowledge about context." In other cases, he (1995, p. 236) expands a task to include "all combined actions that requires some…knowledge about semantic and spatial relations." This earlier definition appears to be the closest any of the geographic information science literature comes to trying to make inference about GIS procedures. Regardless of his intended meaning, Albrecht's research does not attempt to realize new ideas about the characterization of procedural sequences.

Albrecht (1999) attempts to delineate a semantic net based on survey results. The motivation for this is to build an expert system to aid users of a GIS platform which he was developing. An important part of his implementation is assign weights to the edges in his net, presumably as one might weight commands. It is unfortunately unclear in his (conference) paper how the weights were assigned, although it is assumed values were manually attached based on survey results or operational constraints such as data availability. The weighting of links in the semantic network then apparently allows automated evaluation of a sequence of GIS commands (elementary GIS operators) to accomplish a task. He contends that the path a task will follow through the net will always be application-dependent, reflecting the goal specific to that endeavor. Exactly how this context specific behavior is enabled by his design is not evident, but sounds tantalizingly close to the point of this dissertation—the ability to organize GIS procedures flexibly based on context-specific valuation. It is not apparent in any of his publications that he ever realized this.

All of the authors featured in this section do recognize that the file structure or the geometry used to represent geographic information should not be the basis for creating a taxonomy of GIS functions. Albrecht and Gahegan explicitly deal with the process of transformation. Chrisman does too, but as indirectly resolved through the measurement frameworks associated with different commands. As Goodchild (p. 712, 2004) points out, GIS has historically been data-driven. He says "whether the phenomena are static or dynamic, these efforts remain largely focused on form…,in practice the dominant emphasis…is on objects that form the basis of geographic description and representation, rather than on the processes that are the primary goal." Albrecht comes the closest to handling procedures, but his literature never indicated a successful implementation or a complete theoretical specification of an approach.

Previously published taxonomies and perspectives on GIS commands and procedures provide a basic theory of information science for GIS, chiefly in the form of classification theory. This theory is used in this dissertation to help guide the application of inductive methods for organizing GIS procedures, and perhaps even the interpretation of the results, in ways that have not been presented in the geographic information science literature.

2.4 Emerging Trends

The creation of sets of implicit keywords could be informed by almost any logic. This author focuses on types of geoprocessing as the basis for discriminating between GIS procedures. For some users, other bases could be more helpful. It could be the case that a user would want to discriminate not between the GIS procedures but by the type of geographic features produced. In such cases, there are a number of relatively recent contributions to the literature that could be used to define alternative sets of implicit keywords. While thought might be required to exploit these bases within the approach outlined in this dissertation, they are described briefly here. In some cases, it is not immediately

apparent that these alternative bases are an appropriate match for the approach developed in this dissertation.

Bittner and others (2009) discriminate between types of features, groups of features, and specific instances of features, as well as the relationships between features to define an ontology of broadly defined entities. Others have also recognized the value of being able to track understanding about individual features rather than for entire sets of them. Peng (2005) presented a prototype for sharing vector transportation data on the internet that relied on the concept of feature-level metadata. Schwering (2008) has discussed using human perception of types and relations between them ("tiny is more similar to small than to huge.", p. 8) as a means for directly assessing similarity of geographic features.

The problem with using these types of ontological views about features in conjunction with the method presented in this dissertation is that it may not be apparent how these views should be applied on a per-GIS command basis, which is the focus of the methodology developed in this dissertation. Doing so requires an understanding of the entire GIS procedure (which is not resolvable when evaluating individual GIS commands). It could also be argued that these alternative approaches do not appear to assess the methodology (i.e. the GIS procedure) by which geographic data is produced as one of their organizing principles, but rather an externally defined type. This does not devalue these efforts to develop formal ontologies; it serves to emphasize the difference in focus of this dissertation from this increasingly popular topic of ontologies based on human cognition currently in the literature.

As pointed out by Reitsma and others (p. 707, 2009), although ontologies are indeed powerful, they are limited because they typically focus on the "what," as opposed to the "why" or the "how." They note that key institutions, such as NASA, have altered the data structures that they use to highlight the "how" to help make integration of information from different disciplines easier. Although beyond the scope of this dissertation, comparative analysis could be applied to alternative sets of implicit keywords,

or the associated results, in order to develop cross-walks between implicit keywords from different sets. Although developed for particular geographic locations, Sharma and others (2010) have proposed a method for developing such conflations. An important ingredient for enabling such comparisons is a consistent form of expression (syntax), such as proposed by Tanasescu (2007).

Perhaps the most exciting method for developing implicit keywords is through crowd-sourcing. Whether a critical mass of user-generated description of GIS commands could be developed is uncertain, but this could be a way to generate a truly general description of GIS commands along the lines of what Albrecht sought to develop. Authors, such as Nguyen and others (2008), have begun to organize the considerable volume of totally unstructured metadata tags collected through social media (for example, web sites such as Del.icio.us and Flickr.com) into semantic clouds. These data could be mined for more relevant implicit keywords. While this would not necessarily reduce the burden of assigning a value for an implicit keyword-GIS command combination, it could still result in a superior SOM. Hotho and others (2006) have proposed a formal model for these user/crowd-defined conceptual structures ("folksonomies") and search metrics, including ranking.

Comber and others (2008) argue for the inclusion of semantic information in metadata. Characterization of the manufacturing procedure by which data was created is semantically relevant for assessing similarity of different data and possibly even interpreting the meaning of an individual dataset. Doing it in a machine-readable way would add substantial new information with which to improve information retrieval in general.

2.5 Summary

The research presented in this dissertation seeks to synthesize aspects of many of the literatures described in this chapter. The characteristics of GIS commands presented by (Gahegan, 1996; Albrecht, 1999; Chrisman, 1999) for organizing GIS commands into taxonomies will be evaluated for their

effectiveness as a way to inductively organize a taxonomy of individual GIS commands. Their schemes will be used to generate typological groupings of GIS commands, based on command-attribute matrices developed from their respective views and used as input to inductive analyses. In addition, once the inductive methods are trained, their results will be used to generate groupings of GIS procedures (sequences of commands).

Thinking from the field of information retrieval has been reviewed extensively because it informs thinking about how to create information spaces. This dissertation will extend that work by applying it to a new type of data, which is the GIS procedure. Additionally, it will look at developing a number of different keyword-attribution schemes, where keywords are used as attributes that refer to GIS commands, based on the taxonomies like those alluded to above.

This keyword-development scheme is also a departure from previous applications of IR methods, in that they are not based on natural language. An important aspect of this approach is that it presents and develops the idea of using the keyword-development scheme as a way to integrate characteristics that are otherwise only implicitly understood about GIS commands. Further, by demonstrating the impact of different keyword schemes, this dissertation demonstrates that GIS functionality can be organized in potentially very different ways with consistent methodological rigor but using different ways to value functionality.

The main method of analysis will be the self-organizing map (Kohonen, 2001) type of neural network. To help validate and interpret the results of the SOM work, classic inferential statistics, such as PCA, will also be used to identify the important characteristics of the set and to aid in the interpretation of the SOM results. These computational techniques allow value judgments from different perspectives (like those of Albrecht) to alter the ultimate organization of an information space by exploiting the different keyword-development schemes. Although this review has attempted to rationalize the selection of particular inductive techniques as appropriate, this dissertation does not intend to present

these as the only, or even necessarily the best, techniques for this kind of work. The integration of implicit semantics into the application of these techniques is the more important point.

As with the choice of inductive technique, the dissertation author does not seek to extend theory in spatialization, other than to apply these concepts to a new type of information. By choosing the SOM as the main inductive technique, spatialization is an active part of this dissertation. This type of approach to information space enables the author to make quantitative interpretations about the similarity of GIS procedures and the geographic concepts that they represent in a quantitative fashion using distance as a metric. In addition, being able to graphically present a more global view of the results enables an important (qualitative) way to corroborate the quantitative analysis.

An important point is that most of the efforts described within the geographic information science literature attempt to provide classifications (and derived interpretations) of a set of text-based documents, as opposed to the procedural specifications being used in this dissertation. Even when these techniques are used in the domain of computer science for organizing or searching software repositories, which is more analogous to the data type being used in this dissertation, these efforts are built to use documentation about the software (as opposed to the software itself). In all cases, keywords are used to summarize or characterize the data items in the analysis. The meanings of the keywords themselves are never actually examined or adjusted based on the user's context, which could provide a valuable enhancement to the process. The absolute and even relative meanings of the keywords are entirely implied.

3 Methods for Discovering Patterns of GIS Procedures

The author addresses the problem of differentiating GIS procedures in order to support reasoning about their meaning. Software implementing GIS procedures is available through software vendors, as freeware libraries, shared between colleagues, as subscription sources, and other sources. As the quantity of available GIS software procedures continues to grow, many users are overwhelmed by choices. Organizing these resources in a way that is relevant to different user communities would allow a user to quickly discover a selection of GIS software procedures with common meaning in a given application context. The proposed experiment employs inductive methods for discovering patterns of GIS software procedures and creates an information space within which a user can locate GIS software procedures. The information space is structured in such a way that similar GIS software procedures are placed near to each other in groups. For this reason, it is called a spatialized catalog.

This experiment is novel for two reasons. First, as determined by a literature review, traditional IR efforts have not been applied to GIS software procedures, nor have they defined context-specific keywords from which to create information spaces. Although other authors have presented efforts to use inductive methods that allegedly have been applied to organize repositories (non-GIS) software (e.g. Ye and Lo, 2001; Tangsripairoj and Samadzadeh, 2006), these efforts have ultimately relied on natural-language documents, such as users manuals, that describe the software and not the software itself. The keywords in these and all other cases have been derived from the documents. Secondly, in addition to analyzing a new type of data (GIS procedures), this experiment compares the effectiveness of using GIS commands found within them as explicit keywords with attributes used to describe the GIS commands (implicit keywords) as a basis for the inductive analysis. Prior to the research developed here, there has been no way to capture and exploit the implicit understanding associated with a “custom” world view about a set of GIS procedures into the information retrieval process. The definition of these implicit keywords is a way to capture those world views.

3.1 Goals of Research Design

This experiment seeks to address the following goals:

- To develop an automated method for distinguishing among GIS software procedures
- To explore the use of explicit and implicit keywords as a way to distinguish GIS software procedures
- To test the sensitivity of the method to different lists of keywords
- To assess the effectiveness of the method for evaluating the similarity among GIS software procedures
- To consider the utility of the approach for developing geographic and domain-specific sets of implicit keywords.

3.2 Overview

This experiment has two major components. The first is to use a traditional IR approach to organize GIS procedures using GIS commands found within the procedure source code as keywords. Because the GIS commands are actually present in the source code of the GIS procedure and are readable by scanning software, they are referred to as *explicit keywords*. This requires that the frequency of a set of GIS commands is tabulated within each of a set of GIS procedures. This tabulation (referred to as the "procedure matrix") is then used as input to an inductive statistical technique, a machine learning algorithm, for clustering. It demonstrates the effectiveness of traditional explicit keyword-driven approaches to describing documents when applied to GIS procedures. This set of results serves as a datum against which to evaluate the extensions to traditional IR techniques developed in this dissertation.

In order to ensure that the explicit keyword experiment performs to the best of its ability, two additional experiments will be carried out. The first will re-run the machine-learning algorithm with the same input data, but will optimize the training parameters that the algorithm uses. The second will apply pre-processing to derive a simplified version of the procedure matrix which is then used as input to the machine-learning algorithm (using optimized training parameters).

In the second component, attributes are defined by the author for all GIS commands found within the GIS procedures and used as keywords. Because these attributes are not actually present within the source code of the GIS procedure, they are referred to as *implicit keywords*. The machine learning algorithm is then re-run using modified versions of the procedure matrix. More specifically, two sets of implicit keywords are developed and applied independently to produce two sets of output within this component of the experiment. One set is used to represent a generalized view of GIS functionality while the other is a view specialized to a particular domain. The two different sets of implicit keywords are used to test whether the approach is sensitive to the different semantics of the two views. Therefore, there will be five sets of outputs from the two components of the experiment. Various metrics for evaluating the results are calculated to help compare and interpret the results of each set of implicit keywords with the explicit keywords and with each other. This serves to reveal the sensitivity of the clustering analysis to the choice of implicit keywords.

One of the two sets of implicit keywords is developed based on the author's interpretation of published literature relating to the creation of taxonomies of GIS functionality and is intended to represent a broad, generic view of GIS functionality as might be useful in educational contexts. The other set of implicit keywords is devised based on the author's expertise to represent functionality used in environmental modeling. The knowledge for this second set of implicit keywords is drawn based on the author's expertise in this area. Differences in the results created using each set of implicit keywords is then examined in the context of their effectiveness for thinking about GIS procedures.

The purpose of the second component of the experiment is to demonstrate the concept of substituting more semantically meaningful attributes for the explicit keywords, and using those substitutes as the basis for deriving an organization scheme. As described previously, the input data set is composed of GIS procedures expressed in a scripting language. Although the vocabulary in this language (i.e., the GIS commands) may refer to specific algorithms for processing spatial data, the

meaning of a term may be different depending on the user or how they use it. The addition of these attributes (i.e. implicit keywords) is an approach to better inform the process of organizing the set.

The effectiveness of this new approach is judged by comparing its organization of GIS procedures with those produced by the explicit keyword experiment. The alternate sets of implicit keywords reflect the semantics of a specific perspective or view of GIS functionality. The semantics behind these sets of implicit keywords can be thought of as an epistemology (to use the meaning given by Philosophy) or a domain-specific ontology (to use the meaning given by geographic information science). Although this experiment relies heavily on the use of a particular statistical technique to carry out the clustering, the idea of augmentation of explicit keywords with context-relevant attributes (implicit keywords) demonstrated by this experiment could be implemented using any of a large number of other statistical techniques.

The question driving this research is whether information retrieval (IR) can be improved by using descriptive attributes that are not derived directly from the items being searched, but that are arbitrarily defined to capture user understanding or perceptions about the items being organized and searched. While this dissertation will define multiple sets of keywords, the intent is not to define the "ultimate" or "correct" set of keywords. The dissertation seeks to develop a framework for adaptively organizing a body of information that integrates implied understanding of a specific user or community of practice about a corpus into standard IR approaches. This research seeks to explore whether the approach can be used to allow a user or user community's views on GIS functionality to tailor the organization of a set of GIS functionality to better suit their needs. This can be thought of as allowing a user to view a set of functionality through their own "rose-colored glasses."

3.3 Experimental Design

Before beginning the description of the experiment, a few terms that have been used generically are given more specific definitions. An example of a GIS software procedure that is used for this experiment is a program written in the Workstation ArcInfo GIS (ESRI, 2001) scripting language, referred to as an Arc Macro Language (AML) script. The set of GIS software procedures in this experiment consists of approximately 1500 AML scripts found in the USGS GIS Weasel (Viger and Leavesley, 2007) and in the ArcTools library of AMLs distributed by ESRI. An AML script is built from a sequence of GIS commands. A command is piece of GIS software that carries out a single conceptual function, such as "copy," "union," or "flow direction." An AML script may contain a large quantity of other text (such as comment strings) that does not actually carry out spatial analysis. This additional matter is not considered in the experiment. Rather than arbitrarily defining the set of Workstation ArcInfo commands to consider, all (~1400) were scanned for within the 1500 GIS procedures. It should be noted that components of the Arc Macro Language referred to as directives and functions are excluded from the experiment.

The first part of this experiment, referred to as the explicit keyword experiments, addresses the question of whether a traditional IR approach—that organizes corpora of documents based on keywords associated with each document—is effective when applied to GIS software procedures. In this case, the GIS commands found within each AML script are used as keywords. These keywords are referred to as *explicit* because they appear within the scripts. To apply this type of approach, a matrix is created with as many rows as there are scripts (~1500) and as many columns are there are unique keywords (~1400 GIS commands). The value of a cell in the matrix indicates the frequency with which the associated explicit keyword appears in the script being described. The matrix defines an information space whose dimensions correspond to the explicit keywords, with individual scripts plotted as points in the space based on the frequency of keywords within each script. Table 3-1 shows an example procedure matrix

that has a limited set of GIS procedures (AML scripts, shown in the row labels) and uses only five GIS commands as keywords (shown as the column headings). The *flow direction* command is used three times in the source code for the *zone_down-id.aml* GIS procedure, therefore the value assigned to the cell at the intersection of the corresponding keyword and procedure is set to 3. The remaining values are set in the same manner.

Table 3-1 Procedure matrix that indicates the frequency of commands within procedures.

GIS Procedure	GIS Commands				
	Flow direction	Flow accumulation	Zonal centroid	Zonal statistics	Euclidean distance
Zone_accumulation.aml	1	2	0	0	0
Zone_centroid.aml	0	0	1	1	0
Zone_dist_euclidean.aml	0	0	0	0	1
…	…	…	…	…	…
Zone_down-id.aml	3	1	0	0	0

In order to build the procedure matrix, software is written to scan all the GIS procedures for the GIS commands. Chapter 4 provides a more detailed description of this software and how it was checked to ensure accuracy of its results. Once the initial version of the procedure matrix is built, it is cleaned. The GIS commands that appear fewer times than an arbitrary threshold are dropped from the matrix. GIS procedures with less than an arbitrary minimum number of GIS commands are also dropped from the matrix. Details about thresholds and their impact on the procedure matrix are presented in Chapter 4. After the cleaning of the procedure matrix, it is expected that the number of GIS commands still present should drop to ~200-300. This expectation is based on the author's > 15 years of experience in writing GIS procedures.

While the example shown in Table 3-1 defines a low (five) dimensioned space, the actual experiment will result in a much more complex space (because there will ~200-300 GIS commands used, the information space will have ~200-300 dimensions). To be more useful, the information space needs to be simplified and set items need to be grouped. This dissertation uses the Self-Organizing Map

technique (SOM, Kohonen, 2001), a type of neural network algorithm, to organize of the set of GIS procedures. The map visualizations produced by the SOM technique are used in this dissertation as an illustration of an organization of a set. The map visualizations produced by the SOM technique are 2-dimensional displays wherein the neurons that make up the map each represent coordinates within the SOM information space. These coordinates are defined according to the dimensions of the procedure matrix. The coordinates of a neuron is sometimes referred to as the *signature* of the neuron. The spatial arrangement in which the neurons are positioned can be used to visually discern groups or clusters of similar GIS software procedures. Neurons that are adjacent or near to each other are expected to be similar, while neurons that are separated by a greater distance are presumed to be dissimilar. As with individual neurons, the spatial proximity of clusters (groupings of neurons with similar signatures) is also interpreted as an indicator of similarity between the clusters. As noted above, while the SOM technique has several appealing features (most notably a powerful visualization), the experiment could have been designed to use other statistical techniques.

As described in the "Overview" section, the explicit keyword experiment develops three SOMs based directly on GIS commands, plus two additional SOMs to compare with the explicit keyword results. For each alternative, instead of relying solely on commands found in the GIS procedure (*explicit keywords*), a set of *implicit keywords* is used to describe the GIS commands found within the GIS software procedures. While the quality of the additional SOMs is likely to be extremely sensitive to the definition of the associated implicit keyword set, it is again noted that it is not the intent of the experiment to produce a definitive set of implicit keywords for each alternative.

Each of the SOMs is examined in a number of ways. Several statistics, including a unified distance matrix and the best-matching unit mapping (Ultsch and Siemon, 1990), will be used to evaluate the quality of the trained SOM. Although these analyses do not indicate whether the result is semantically meaningful, they do provide an indication of confidence in the SOM training process. Once

a level of confidence in the overall SOM is established, the placement of points, each representing a GIS software procedure, is examined in two ways. First, the Euclidean distance between pairs of points will be evaluated as an indicator of similarity and overall SOM quality. These "similarity distances" will be compared with the author's expectations of similarity between pairs of procedures. The second method of evaluation will be visual examination of the groupings shown by the SOM. The patterns shown by the visual output of the SOM will be heuristically examined to assess whether procedures that are perceived by the author as similar are located in groups or clusters in the SOM-space, whether the number and spatial arrangement of groupings corresponds with the author's expectations, and whether the degree of separation between groupings within the SOM-space is consistent with expectations.

3.3.1 Modifying the Procedure Matrix with Implicit Keywords

In order to make use of a set of implicit keywords, the procedure matrix (for example, Table 3-1) for the explicit keyword experiment is modified. To do this the set of implicit keywords are defined and then expressed in what is termed here the *command matrix*, an example of which is shown in Table 3-2. The rows of the command matrix enumerate all the explicit keywords (GIS commands) found in the columns of the procedure matrix. Each column of the command matrix corresponds to one of the heuristically defined implicit keywords (described in Chapter 5). Each GIS command is evaluated as to whether or not it conforms to each keyword. "1" indicates a conformance and "0" indicates non-conformance. As stated elsewhere, although the experiment does not seek to define the optimal set of implicit keywords, the definition of the keywords and the evaluation of a GIS command with regard to them is the crux of the method developed in this dissertation. The nature and number of the implicit keywords that are used are the means by which the world view of a user or a community of practice is represented within the IR process. The implicit keywords in the command matrix are used as input to the SOM training process.

Table 3-2 shows an example command matrix that has a limited set of GIS commands (shown in the row labels) and uses seven implicit keywords (shown as the column headings), adapted from work previously initiated by the dissertation author to define implicit keywords for hydrologic modeling (Buttenfield and others, in preparation; Wendel and others, 2008a; Wendel and others, 2008b).

Table 3-2 Command matrix describing GIS commands based on implicit keywords.

GIS Commands	Implicit Keywords						
	Raster Only	Data Mgmt	Geometric	Terrain Flow	Local	Regional	Changes Spatial
Flow direction	1	0	1	1	1	0	0
Flow accumulation	1	0	1	1	1	0	0
Zonal centroid	0	0	1	0	1	0	0
Zonal statistics	0	0	0	0	0	1	0
Euclidean distance	0	0	1	0	1	0	0

As stated previously, two sets of implicit keywords are developed (resulting in two tables like the one shown above). To develop a table for each like the one shown above, a conformance/non-conformance value must be set for each implicit keyword for each of the GIS commands (e.g., "does the flow direction GIS command handle data that is 'Raster Only' or not? Is it a 'Data Management' function?", etc.). The results are recorded into a corresponding command matrix. Detailed description on the definition of each set of implicit keywords is provided in Chapter 5. Tables of the valuations for each implicit keyword-GIS command combination are provided in Appendices A and B.

It is not immediately possible to use a command matrix to influence the SOM analysis of the GIS procedures—the real objective. In order to do this, the procedure matrix needs to be modified ("augmented") so that it reflects the information in the command matrix. To transfer this information from the command matrix to the procedure matrix, a database "relate" operation between the procedure matrix and the command matrix, using the GIS commands as the relate key, is used. Table 3-3 shows an example of an augmented procedure matrix, where the rows correspond with the procedures listed in the rows of Table 3-1 and the columns correspond with the implicit keywords (columns) shown in Table 3-2. To illustrate how the values in Table 3-3 are set, the derivation of values

for the zone_accumulation.aml GIS procedure is described as an example. In the source code of this procedure, according to Table 3-1, there is a single occurrence of the flow direction command and two occurrences of the flow accumulation command. The row of values for flow direction, as shown in the command matrix (Table 3-2), is (1,0,1,1,1,0,0). The frequency of occurrence of the flow direction command, 1, is multiplied by this vector. In this case, the vector is the same as the input, (1,0,1,1,1,0,0). The values of the implicit keywords for flow accumulation is (1,0,1,1,1,0,0). Multiplying the frequency of occurrences of flow accumulation, which is 2, produces a vector, (2,0,2,2,2,0,0). To set the implicit keywords for the zone_accumulation.aml procedure, these two vectors are summed to yield (3,0,3,3,3,0,0). The result is recorded in the first row of Table 3-3. The implicit keyword values for the remaining procedures are calculated in the same manner. The augmentation of the procedure matrix will be carried out once for each set of implicit keywords, which is then be used to create a new SOM. More detailed description of the augmentation procedure is provided in Chapter 5.

Table 3-3 Procedure matrix augmented with implicit keywords.

GIS Procedure	Implicit Keywords						
	Raster Only	Data Mgmt	Geometric	Terrain flow	Local	Regional	Changes Spatial
Zone_accumulation.aml	3	0	3	3	3	0	0
Zone_centroid.aml	0	0	1	0	1	1	0
Zone_dist_euclidean.aml	0	0	1	0	1	0	0
…	…	…	…	…	…	…	…
Zone_down-id.aml	4	0	4	4	0	0	0

The same methods of evaluation applied to the explicit keyword results are used to evaluate the quality of each set of implicit keyword SOM. In addition to checking whether the results are logically consistent with the author's expectations given the implicit keyword set being featured, the outputs are compared to each other and to the explicit keyword outputs by visual examination. The visual examination will determine whether there are substantial differences in each output, whether the

number, separation, spatial arrangement, and meaning of the groupings of the same set of GIS software procedures across the implicit keyword sets change in substantive ways.

The following sub-sections provide greater detail on the theory behind each of the sets of implicit keywords. Although presentation of the actual keywords will be reserved for chapter 5, the theoretical bases for the implicit keyword sets are described in enough detail here to provide the reader with a sense of the qualities that the author will use to define and evaluate implicit keywords. The first set of keywords defines characteristics that have been posited in the literature as useful for classifying or organizing GIS functionality into generally useful taxonomies. The second set of implicit keywords is based on environmental modeling.

3.3.2 Implicit Keywords for General Classification

This section discusses Albrecht's (1999) ideas on "universal" analytical GIS functions. These ideas are used to define a set of implicit keywords that might be treated as universally or generally useful. Albrecht (1999) organized GIS functionality according to how users perceive, learn, and organize GIS commands. To gain understanding of user perceptions, he carried out a series of surveys. He identified data-related and analytical types of GIS commands. Data-related commands are involved in: map making, data entry, item selection, map display, and attribute classification. Analytical commands are involved in: search, locational analysis, terrain analysis, spatial distribution or neighborhood analysis, spatial analysis, measurement. Albrecht (1999) provided a description for each of these analytical types, which eases the work of creation of implicit keywords and a command matrix representing his ideas for application within this dissertation. More detail is provided in the following chapter.

3.3.3 The Environmental Modeling Domain

A second set of implicit keywords is developed and used to classify the set of GIS software procedures based on implicit keywords that reflect ideas about organizing GIS functionality that are

used by (GIS-literate) environmental modelers, which is an example of an applied user community. The world view that these keywords reflect is not intended to be comprehensive. In fact, the value of this intentionally limited perspective is to test the sensitivity of the organization method to a keyword list that is likely very different than any those produced by the general frameworks discussed above. Further, by using a relatively specialized and therefore constrained world view, this portion of the experiment demonstrates that the approach of using inductive methods with implicit keywords shows a degree of robustness and flexibility.

Initial development for this portion of the experiment was carried out by the author under a grant from the U.S. Geological Survey Center for Excellence for Geographic Information Science in collaboration with the Professor Barbara Buttenfield. In this work, they assigned implicit keywords to a subset of GIS commands and input this information into a SOM analysis. Several conference presentations were developed and a manuscript describing this work is currently in preparation (Buttenfield and others, in preparation; Wendel and others, 2008a; Wendel and others, 2008b).

Examples of implicit keywords developed in this initial research indicated whether commands:

- handled raster, vector, or both formats of data
- dealt with data management tasks (such as copy, delete) or with hydrology tasks (this set of keywords were used for experimental control)
- modified the data geometry, attributes or both
- dealt with flow over topographic terrain
- analyzed information local to individual cells (in rasters)
- analyzed information over a region, but not over entire domain
- changed spatial relationships

A subset of the GIS commands used in that work is shown in the rows of Table 3-4. The implicit keywords are shown in the columns.

Table 3-4 Command matrix describing GIS commands based on implicit keywords.

GIS Command	Implicit Keyword						
	Raster_Only	Data_Mgt	Geometric	Terrain_flow	Local	Regional	CSR
aggregate	0	0	1	0	0	1	0
area weight	0	0	0	0	0	1	0
aspect	1	0	1	1	1	0	1
combine	1	0	0	0	1	0	1
conditional	1	0	0	0	1	0	0
copy	0	1	0	0	0	0	1
cost alloc	1	0	1	1	1	0	1
cost backlink	1	0	1	1	1	0	1
cost distance	1	0	1	1	1	0	0
cost path	0	0	1	1	1	0	0

3.3.4 Dimension Reduction to Train Organization Schemes

The rows and columns of a procedure matrix define a high-dimensional information space. While the modification of the procedure matrix to use implicit keywords does drastically reduce the number of dimensions associated with the procedure matrix, there are still too many dimensions to be able to effectively visualize the information space created by the procedure matrix effectively. In order to make it easier to examine, interpret, and visualize, the number of dimensions are reduced. The technique used is the self-organizing map (SOM), introduced in the preceding chapter. The specific implementation of the SOM technique used in this experiment is the SOM Toolbox for Matlab, published by the Helsinki University of Technology's Laboratory of Computer and Information Science (HUT-CIS) (http://www.cis.hut.fi/projects/somtoolbox/, accessed November, 2008). HUT-CIS is the institution with which Professor Emeritus Teuvo Kohonen, the inventor of SOM, is affiliated. SOM functionality has been implemented and made available by other authors, most notably Jun Yan of the University of Iowa for the *R* open-source statistical package. An advantage of the SOM Toolbox from HUT-CIS is that it also provides implementations of PCA and other dimension reduction techniques.

A number of factors can influence the accuracy of a neural network analysis, and a SOM analysis in particular. This experiment relies on two influential papers to help recognize and deal with factors

related to the fundaments of neural network approaches (Kohonen, 1993; Foody and Arora, 1997). Other authors provide guidance on the definition of keywords (for example, (Lagus and Kaski, 1999; Azcarraga and others, 2004). Fabrikant and Buttenfield (2001) caution against inadvertently creating a "keyword barrier" but that problem should not arise because the GIS commands used in this experiment constitute a fixed vocabulary—in other words, there is no real choice of explicit keywords because the set depends solely on whatever GIS commands are found in the set. Perhaps the most difficult aspect of using a technique that aggregates dimensions is interpreting the meaning of the resultant clusters and dimensions of the reduced-complexity information spaces. Merkl (1998) and Azcarraga and others (2005) help to inform the interpretation of results.

One of the most common techniques for interpreting a SOM's groupings and dimensions is to compare SOM outputs with the results of alternative techniques, such as Principal Component Analysis. PCA is helpful because in addition to reducing an information space's complexity by merging dimensions, it ranks the power of each dimension to explain the variance in the data and also reports, via Eigenvectors, how much each principal component relies on specific dimensions in the input data matrix. The PCA results provide valuable insight about what characteristics (dimensions) of the input information space dominate the process of delineating a cluster. For each of the developed SOMs, a PCA is carried out.

3.3.5 How to Build a SOM

This section provides an extremely brief description of spatializations produced by the SOM technique. All the graphics presented in this section are taken from an excellent tutorial provided by the Peltarion company web site (http://www.peltarion.com/doc/index.php?title=Self-organizing_map&oldid=1512, accessed January, 2009) or created as a product of the hydrological modeling experiment undertaken by Buttenfield and others (in preparation) and (Wendel and

Buttenfield, 2010). A SOM is a network of neurons. Each neuron is connected to a set of neighbors. Each neuron is described by as many attributes as there are dimensions in the input command matrix. As the network is trained, the attributes of the individual neurons are adjusted. Because the attribute values are used to define the neuron position, the network topology is distorted by the training process. A trivial two-dimensional example of the distortion is shown in the example in Figure 3-1a). The network always maintains the topology (that is, connections) between neurons. This enables the network to be projected to a flat, two-dimensional "maplet," as shown in Figure 3-1b). Note that the labels in the cells of the maplet correspond to those associated with the neurons in Figure 3-1a).

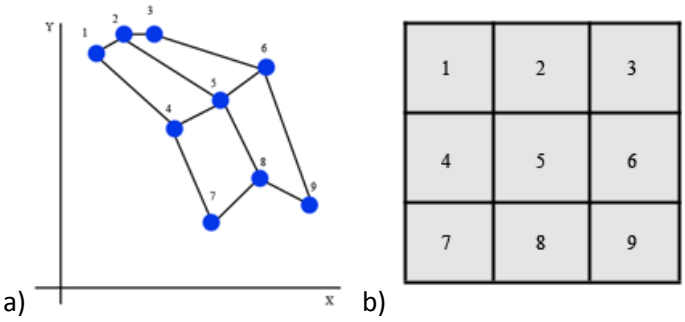


Figure 3-1 Views of a SOM. (a) A SOM network is a planar arrangement of connected neurons. (b) The result can be presented as a regularized matrix. From http://www.peltarion.com/doc/index.php?title=Self-organizing_map&oldid=1512 (accessed January, 2009).

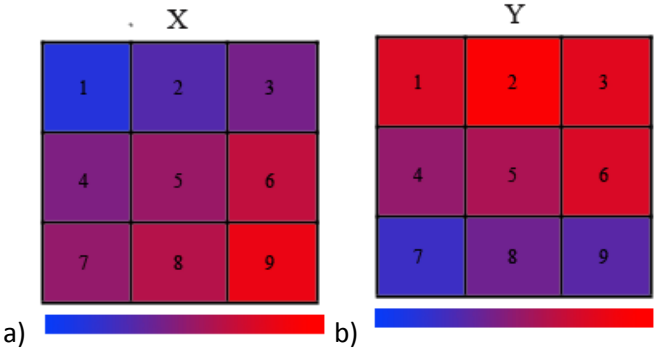


Figure 3-2 Maplets illustrating values for dimensions in Figure 3.2. (a) Values for the X dimension. (b) Values for the Y dimension. From http://www.peltarion.com/doc/ index.php?title=Self-organizing_map&oldid=1512 (accessed January, 2009).

One can indicate the values associated with each dimension in the original information space. In Figure 3-2a), the color hue associated with each cell varies according to the position of the neuron in Figure 3-2a) along the X dimension. For instance, cells colored in shades of blue lie closer to the origin along the X-axis than red neurons. Figure 3-2b) shows a similar colorization to represent the values in the Y dimension.

A more realistic example is shown in Figure 3-3a), also taken from the Peltarion URL provided at the top of this section. This example features a ten-dimensional input data set, where dimensions refer to demography, landcover, landmarks, and access to water, for example. The Peltarion authors make an example of the cells at the top right of the Churches maplet, noting that areas with higher numbers of churches correspond to areas with more cows, but also with higher populations that have a lower median age (the lower left maplet).

Figure 3-3b) shows results from the hydrology experiment. The input data utilized eight dimensions (only seven are shown) where each dimension describes processing characteristics of the input commands. For example, “local” and regional” identify whether the GIS command operates on single pixels or on a neighborhood. “Geometric” indicates if a command operates on geometry (as opposed to operating on attributes).

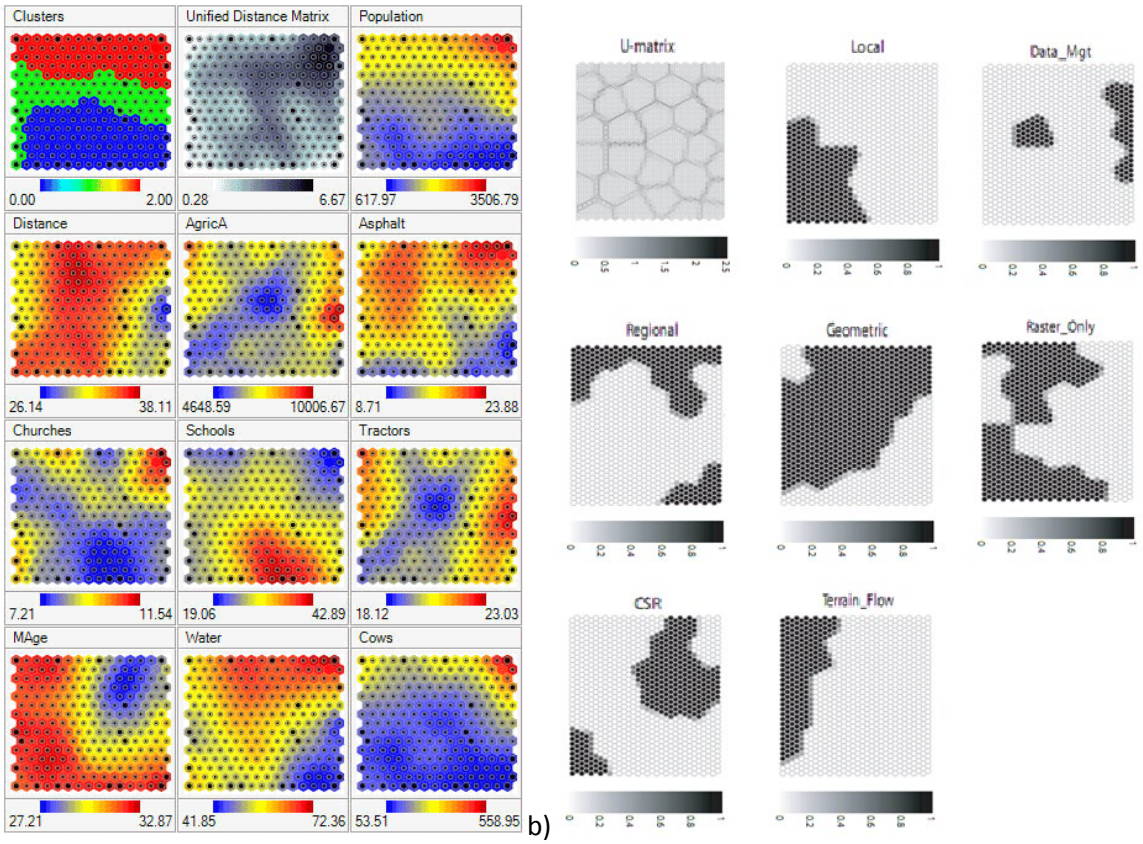


Figure 3-3 Example visualizations of SOM analysis. (a) From http://www.peltarion.com/doc/index.php?title=Self-organizing_map&oldid=1512 (accessed January, 2009). (b) From (Buttenfield and others, in preparation) work that tested applicability of SOM to GIS commands.

In addition to the maplets, the SOM determines a unified distance matrix (or U-matrix) which indicates the degree to which the neurons in the SOM are separated from their neighbors. The darker the color in the U-matrix graphic is, the greater the separation of clusters on either side of a given neuron. From a more mathematical perspective, the unified distance measure indicates the Euclidean distance separating the neurons in the output network. Looking back to Figure 3-1a), we can see that the neurons are not uniformly spaced. The maplet representation eliminates this distance separation by regularizing the spacing of the neurons in the graphical representation of the SOM. The U-matrix restores this information by colorizing a maplet based on distance. Adjacent neurons that are separated by a greater distance are relatively dissimilar and therefore are likely members of different clusters. The

example U-matrix in Figure 3-4 shows relatively strong separation between three different groupings of neurons. Note that the crispness of this example is not necessarily typical when compared with the U-matrix shown in Figure 3-3a). The U-matrix shown in Figure 3-3b), obtained in the work of (Buttenfield and others, in preparation), is even crisper than that shown in Figure 3-4.

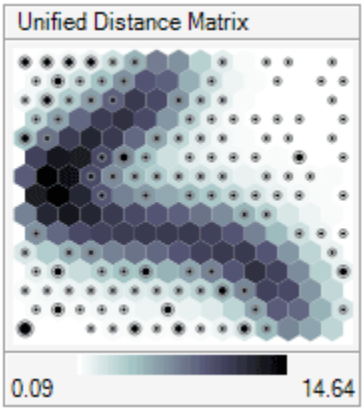


Figure 3-4 Unified Distance Matrix (U-matrix) showing the separation of groups. From http://www.peltarion.com/doc/index.php?title=Self-organizing_map&oldid=1512 (accessed January, 2009).

Evaluation of SOM networks

The explicit keyword portion of the experiment establishes whether implicit information is needed at all in organizing and characterizing geographic procedures. Then, each of the SOM networks is analyzed by derivation of and visual inspection of a corresponding U-matrix (Ultsch and Siemon, 1990) and by calculating the quantization error and topographic error (http://www.cis.hut.fi/somtoolbox/package/docs2/som_quality.html, accessed January, 2011) . *Quantization error* describes the average distance between each data point in the command matrix and the associated Best Matching Unit (BMU) in the SOM network. This statistic is interpreted as an indication of the appropriateness of the SOM resolution (i.e., the numbers of rows and columns), given the distribution (density/sparseness) of the data in the original network. The second statistic, referred to as the *topographic error*, indicates, in a trained SOM, what fraction of data points' first and second BMUs are *not* adjacent to each other. This statistic provides a quantification can be interpreted as a

measure of Tobler's Law (Tobler, 1970)), that states that nearer neurons should be more similar than those that are farther away. This statistic would be better named as the topological error, as it is indicative of how well the SOM describes proximity (or connection) between neurons. Large values imply that the SOM creation process failed to produce a global organization where neighbors are similar, regardless of whether the SOM shows a low quantization error.

To provide an additional source of information to help interpret the SOM, the same inputs are analyzed using PCA. PCA results can be compared with other techniques to test for statistically significant variations in individual principal components (Aldenderfer and Blashfield, 1984). Other interpretation aids not used here include psychological profile assessment (Skinner, 1978) which describes elevation, scatter, and shape of clusters.

3.4 Summary

The experiment outlined here is designed to test alternative bases for implementing a self-organizing map of GIS software procedures. An initial explicit keyword experiments simply uses the explicit presence of GIS commands within GIS software procedures as keywords to drive the analysis. While treating GIS commands as "keywords" and GIS procedures as "documents" in a traditional IR-type of approach is a novel application, in that it uses GIS procedures in lieu of documents, it is only carried out to demonstrate the inadequacy of traditional approaches relying on explicit keywords. In the subsequent portions of the experiment, implicit keywords are used to drive the organization of the set. Two sets of implicit keywords are derived from several different conceptual frameworks, and then applied to the same set of GIS procedures used in the explicit keyword. The five sets of results (the explicit keyword SOM, the optimized explicit keyword SOM, the PCA-driven version of the explicit keyword SOM, the Albrecht universal function implicit keyword SOM, and the hydrological modeling implicit keyword SOM) are examined, evaluated and compared for effective organization of GIS

functionality. It was expected that the semantically enhanced implicit keywords would capture otherwise unused understanding that the explicit keywords miss, and in so doing, improve the ability to isolate meaningful clusters of similar GIS software procedures. Chapter 4 reports the results and analysis of the SOM derived using explicit keywords, and presents details of the creation of the procedure matrix and the visualization of the SOM results. Chapter 5 documents results and analysis of the SOM derived using two different sets of implicit keywords.

4 The Explicit Keyword Experiment

This chapter documents the specifics of the methodology executed for the explicit keyword experiment. In addition, this chapter provides information upon which the following chapter, which describes the implicit keyword experiments, builds. In addition, the results of the explicit keyword experiment are presented within this chapter. Beyond providing content that aids in the assessment of the effectiveness of the explicit keyword experiment, these results provide a datum against which the results of the implicit keyword approach can be differentiated and evaluated.

The structure of this chapter is as follows: Section 4.1 describes the creation of the procedure matrix (introduced in Table 3-1), including the techniques to accurately scan the set for explicit keywords (i.e. GIS commands) and to filter unused or underused commands and empty procedure scripts from the SOM training process. Section 4.2 describes the training of Self-Organized Maps using training parameters set by the software provided by the developers of the SOM Toolbox for Matlab (Vesanto and others, 1999). This SOM is referred to as the "default" SOM. Section 4.3 introduces a number of clustering techniques used to help describe and interpret the default SOM and. Section 4.4 describes the SOM that was produced when the parameters for SOM training were optimized based on experimentation and guidelines published by others (e.g. Wendel and Buttenfield, 2010). This SOM is referred to as the "optimized" SOM. The description of both the default and the optimized SOM examples includes statistics and figures.

Section 4.5 examines whether a widely-used technique, Principal Component Analysis (PCA), can improve SOM results when used to pre-process the inputs to the explicit keyword experiment. In addition, it is also introduced as alternate method for visualizing the SOM. Section 4.5.1 provides a brief overview of PCA. Section 4.5.2 describes the analysis of the procedure matrix with PCA. The results of this more traditional analysis, presented in Section 4.5.3, are used to identify important characteristics of the set and thereby aid in the interpretation of the SOMs in the final chapter. In addition to

Eigenvectors and Eigenvalues, the PCA results produced a new set of dimensions for each data point (i.e. GIS procedure). These dimensions could themselves be used as input to the SOM training process. The results of this process are presented in section 4.6.

4.1 Pre-processing of the Set

The set of GIS procedures consisted of 1558 GIS scripts, all written in the Arc Macro Language (AML; ESRI, 1999). Each script was scanned for the number of occurrences of any of the 1440 GIS commands that serve as explicit keywords for the explicit keyword experiment. Because this process of detecting and tabulating the frequency of keywords is fundamental to the entire dissertation, great care was taken to avoid counting the occurrence of the keyword in non-valid contexts. Examples of keyword occurrences that were treated as non-valid include when the keyword falls between quotation marks, as part of a text string, or in an AML comment, and when the keyword text appears as a part of the invocation of another GIS command. For example, the word "line" is a GIS command, but can also appear as part of the syntax for using the LISTFILE command. An important characteristic of the frequency tabulation is that keywords are counted only if they appear within the GIS procedure itself; occurrences of keywords in procedures invoked by a GIS procedure are not added to the tabulations for the original procedure. The accurate processing of the currently defined tabulation rules was verified using a specially constructed synthetic data set that produced frequency tabulations that were consistent with manually calculated totals. In addition, the tabulations of the full set were extensively spot checked by hand.

Tabulations were recorded in a matrix form and ingested into a MATLAB procedure matrix (an example is shown in Table 3-1). Each explicit keyword in the procedure matrix defines a dimension in the initial information space. Reducing the number of dimensions prior to developing a SOM was important because this reduces the complexity of the input data set and computation times. Substantial

numbers of keywords (i.e. explicit keywords) fail to appear in any GIS procedures or appear only rarely. Unused and underused explicit keywords were considered superfluous and were eliminated from the procedure matrix. “Underused” commands were defined using an arbitrary threshold, requiring that a command occurs in at least 1 percent of the GIS procedures, or at least 15 times across the entire set. The threshold selection was arbitrary and conceivably could be adjusted according any of a number of conditions. This filtering action reduced the initial set of 1440 explicit keywords to 148, which aligns with the expectations formed during the experiment definition. Table A-2 in Appendix A lists the GIS commands that were selected. Figure 4-1 shows the impact of varying this threshold on the number of keywords. Increasing the threshold and removal of dimensions not only reduces the workload associated with the development of implicit keywords, discussed in the next chapter, but also eliminates dimensions where non-zero values will be sparse. Sparsely used dimensions make the SOM training process more expensive computationally without adding information.

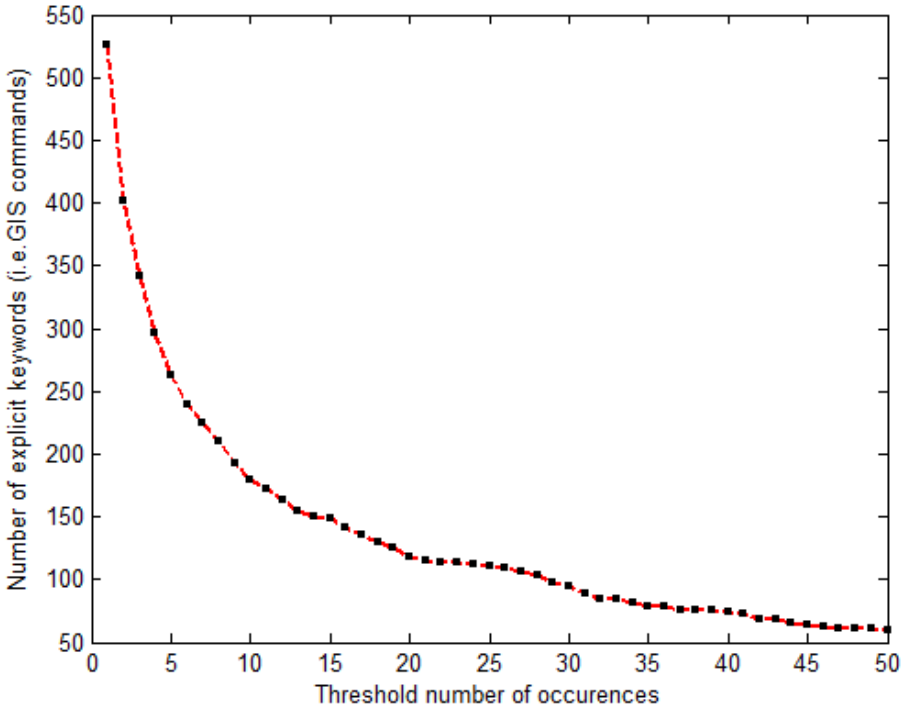


Figure 4-1 Number of explicit keywords exceeding frequency thresholds within the set of GIS procedures.

In addition, a number of AML procedures in the set included no GIS commands. These procedures carried out non-spatial operations, such as creating GUIs, checking logical conditions, or calling for the execution of other GIS procedures. Procedures with no GIS commands were obviously unwanted for this analysis. Scripts with a single command were thought of as being identical to a single GIS command, no more than a container for a single explicit keyword. These scripts were also eliminated from the matrix. Procedures that invoked two or fewer GIS commands were eliminated. Again, the setting of such a threshold could be set as a function of some kind of logic. This reduced the initial set of 1558 scripts to 738.

Table 4-1 shows a subset of the procedure matrix built from explicit keywords. The first column lists the names of the GIS procedures that were selected; the subsequent columns correspond to a subset of the GIS commands that are used as explicit keywords. The values in the cells of the table indicate the number of times the GIS command appeared in the GIS procedure. The entire procedure matrix was not included because its size (738 rows by 148 columns) made presentation cumbersome in a page-based display. The overall procedure matrix is relatively sparsely populated with non-zero values because many GIS procedures have only a small number of occurrences of a small number of GIS commands.

Table 4-1 Subset of the explicit keyword procedure matrix used in explicit keyword experiments.

GIS Procedure	GIS Command				
	'ARC'	'CALC'	'ASELECT'	'CURSOR'	'RESELECT'
'flyby'	12	0	3	51	5
'param_oregon-calibration-assign'	0	142	0	0	1
'la'	4	0	37	0	36
'address_select'	0	0	21	0	21
'fly_around'	8	0	1	24	2
'ascheckout'	6	0	3	0	9
'routing'	2	0	29	12	20
'spatialsel'	0	0	16	0	15
'shutoff'	0	4	55	0	17
'camera'	0	0	0	0	0

This table indicates that the ‘flyby’ GIS procedure invokes the ‘ARC’ command 12 times, the ‘ASELECT’ command three times, the ‘CURSOR’ command 51 times, and the ‘RESELECT’ command, 5 times. The 'param_oregon-calibration-assign' procedure invokes the ‘CALC’ command a whopping 142 times and the ‘RELECT’ command once. There is no unifying theme to the selected GIS procedures. ‘flyby’ generates data for surface fly-through visualizations; ‘fly_around’ generates a flight path around a user-specified location; ‘camera’ controls viewing settings for three-dimensional visualizations; ‘shutoff’ carries out a trace analysis of a network; ‘routing’ solves vehicle routing problems; ‘spatialsel’ allows mouse-based selection of a vector feature; ‘address_select’ finds a feature based on street address; ‘ascheckout’ is for selecting and possibly extract features from an ArcStorm database; ‘la’ is a driver for a suite of location-allocation analyses; and ‘param_oregon-calibration-assign’ generates a set of parameters for an input map of features.

4.2 Default SOM Training Results: Working with Explicit Keywords

The explicit keyword procedure matrix was then used to build three SOMs. The first relied on the default parameters associated with the SOM creation and training process. While this was not expected to yield very good results, it was generated as a datum from which to evaluate improvement in the SOM performance as a function of refined parameterization of the SOM training process. This chapter presents two additional variations on this “default” SOM, both of which were also generated using the procedure matrix. The alternate SOMs are used to ensure against incomplete or inaccurate characterization of the effectiveness of the explicit keyword procedure matrix for SOM generation because of potentially inappropriate “default” training parameters. These SOMs will be compared (in Chapter 6) with those developed in the next chapter using procedure matrices augmented with implicit keywords.

The SOM Toolbox for Matlab provides a helper function, som_make(), that can be executed using only the procedure matrix as input to the SOM training process. All parameters pertaining to the training are either hard-coded to defaults or defined by an automated analysis (carried out by som_make()) of the input procedure matrix. As mentioned above, the "default" SOM produced by this tool is used as a datum against which to compare other results developed in this dissertation. The som_make() tool is presumed to encapsulate the "best practices" for developing a SOM because it was developed by the same institute, the Laboratory of Computer and Information Science and Adaptive Informatics Research Center at the Helsinki University of Technology (LCIS-AIR), that originally published and has consistently developed the concept of Self Organizing Maps. The only values of the parameters set by som_make() reported here are those used to describe and interpret the output. Detailed discussion or analysis of these parameters is not presented here as this experiment is not intended to serve as an exploration of the best SOM training parameters. Instead, the author relies on recommendations established by others (such as Kaski and others, 1998a; Kohonen, 1998; Lagus and others, 2004; Wendel and Buttenfield, 2010) to optimize these parameters in the next explicit keyword experiment.

The default SOM was configured as a sheet of hexagonal neurons, and built using the standard Gaussian function to find and adjust neighboring neurons during the batch mode execution of the training process. The process resulted in 135 neurons, arranged in 15 rows and 9 columns. Alternative SOM shape options include a cylinder or a toroid. Figure 4-2a) shows the neurons at the corners of the SOM labeled with identification numbers. Note that a "column" is composed of a zig-zagged vertical sequence of adjacent neurons. Identification numbers begin from 1 at the top left corner and increase down the first column, then continuing to the subsequent columns. For example, the neuron at the top left, labeled with a "5" in Figure 4-2a), is the number 1 neuron and the adjacent neuron, labeled with a

"1," is the number 2 neuron. The number 3 neuron is adjacent to number 2 and is labeled with a "7" in Figure 4-2a).

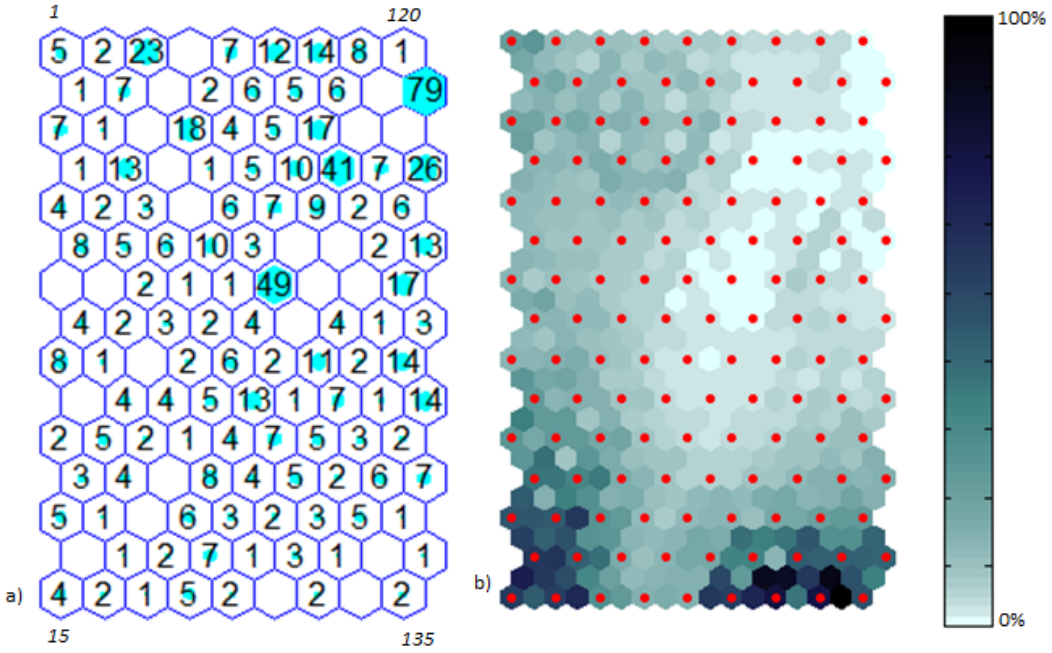


Figure 4-2 The SOM trained with explicit keyword procedure matrix and default parameters from som_make(). (a) Neurons labeled with frequency of best-matching GIS procedures. The size of the blue patch also indicates match frequency. Corner neuron identification numbers posted in italics. Neurons are numbered sequentially from top to bottom, left to right. (b) U-matrix for the SOM, showing dissimilarity as a darker color. Color indicates percent of range in U-value within SOM. Actual values range from 0.26 to 18.13. Regions of lighter colors indicate clusters of similar neurons and darker values indicate separation between clusters. The red dots indicate locations of neurons.

The choice of SOM shape can be used to avoid edge effects. The sheet shape tends to show the most edge bias, but is the easiest visualization to interpret. One can generate a toroid shape, a continuous three-dimensional surface, to avoid this, but this will ultimately require transformation in order to visualize as a two-dimensional map. This transformation can result in a cluster being split and shown as straddling the seam (i.e., member neurons will appear at opposite edges of the SOM visualization). Understanding this is important for properly interpreting the visualization. The individual neurons can be treated as lattice of square cells instead of hexagons. While hexagons are more difficult to process using standard GIS or matrix processing tools, hexagonal SOMs have the advantage of

consistent spacing between a neuron and all neighbors (i.e., consistent spacing in both cardinal and diagonal directions).

The "batch mode" parameter of the SOM training process indicates that the adjustment of the output neurons is done in a single step using the entire input set of data points, as opposed to the sequential approach which sequentially adjusts the output neurons based on each data point. While this should not result in any improvement in the quality of the resultant SOM, it does exploit built-in efficiency of MATLAB to carry out the SOM training process as a matrix operation and drastically reduces computing times (Vesanto and others, 1999). Vesanto and others (1999) reported that typical computing times were reduced by an order of magnitude when the batch training method was used in lieu of sequential training and that the reduction in computing times increased as larger input data sets were used.

The training process usually consists of two phases, one for "rough" fitting of neurons to the input data that is followed by a "fine tune" fitting. During each phase, the BMU for each data point (i.e., GIS procedure) is determined, the BMU is then adjusted to more closely match the data point, and then the neighbors of the BMU are adjusted according to a (Euclidean) distance decay function to more closely match the BMU. Part of that process includes specification of the radius that defines the neighborhood, which is expressed in the unit distance between adjacent neurons. During the rough phase of the automatic training process, the search radius initialized at 2 neurons and decreased to a size of 1 neuron. For the fine tuning phase, the search radius was held constant at 1. The inverse type of alpha function, which defines the rate of distance decay rate for the adjustment of neighboring values, was applied. Somewhat surprisingly, the actual alpha value did not appear to be specified in the online documentation or to be modifiable. The training history associated with the output SOM indicates that only 2 iterations were used during the rough phase, and 8 iterations during the fine tuning phase.

The numbers for both the search radius and the number of training iterations are surprisingly low given the recommendations of Wendel and Buttenfield (2010), who indicate a radius of at least half the size of a side of a SOM network and thousands of iterations. The SOM toolbox online documentation indicates the data points in the input data set are presented to the SOM training process as listed in the input data set (referred to by the online documentation as the data set's "linear" order) in the batch mode. It seems that even if the data set sort order was randomized, the result should be the same because the batch mode presents the entire input data set to the SOM training process in a single step to derive a globally optimal set of neuron adjustments.

In Figure 4-2a), each neuron of the SOM is labeled to indicate the number of GIS procedures for which it was selected as the BMU after all training is completed. This display is referred to here as the "hit histogram." This display reveals where data points are concentrated on the map. For instance, there are three neurons with BMU frequencies exceeding forty. Further, this display reveals that in addition to a large number of neurons with a frequency lower than 10, a handful of neurons were never considered to be a BMU. This display is useful for examining whether the matrix was appropriately sized (although this is a subjective interpretation) and indicates the degree of spreading of the data. A matrix that has too few neurons results in poor separation of data points. A matrix which has too many neurons results in each data point getting its own neuron—which is might be viewed a failure to cluster the data points in any way, although there may still be useful information in the relative spatial placement of GIS procedures across an over-large SOM. Neurons with a zero hit frequency was included 17% of the SOM. Table C-2 in Appendix C lists the identification numbers for the SOM neurons and the GIS procedures that associated the corresponding neuron as the best matching unit.

A U-matrix was derived from the default SOM in order to calculate the separation between neurons (which is interpreted as dissimilarity) and to help visualize the groupings of GIS procedures. High U-matrix values indicate that neighboring neurons are dissimilar. Figure 4-2b) shows the result of a

fixed color ramp from light blue to black to represent U-matrix values that range from 0 to 18.1 (signifying “similar” to “dissimilar”), respectively. The upper right corner and center not only have very little similarity but also a very high frequency of procedures. Down at the bottom of the SOM, where many empty neurons occur, there is a high degree of dissimilarity.

Note that the U-matrix has twice as many cells in its display as the original SOM has neurons. This does not imply that the U-matrix has more neurons than the SOM. The extra cells (which are not neurons) are used to show separation between adjacent neurons (shown as red dots).The red dots indicate the position of the original neurons within Figure 4-2b). Although these neurons also have a U-matrix value assigned to them, presentation of the values assigned to the cells separating the original neurons is the purpose of Figure 4-2b).

The quantization error, which indicates the fit between the data points and their BMUs, is 4.285 for this SOM. This metric is analogous to those used in surface interpolation to understand how tightly a derived surface honors the input data (or whether the surface passes through the actual data points). Quantization error is computed as the average discrepancy between each data point of the input matrix and the neuron within the SOM which most closely resembles the respective data point (the BMU). In creating a SOM, the dimension values associated with a data point (i.e. a GIS procedure) are treated as coordinate positions. Taken as a vector, the explicit keywords constitute coordinates that position the data point within the information space. Each neuron in the SOM also has a vector of coordinates. The Euclidean distance between any two points is used to quantify similarity (discrepancy). Summed across all data points, the mean discrepancy between each data point and its BMU defines the quantization error.

A second measure evaluating the SOM is the topographic error. In contrast to quantization error, topographic error characterizes the continuity of character from one neuron to the next. First, this technique determines the first and second Best Matching Units (BMUs) for each data point in the input

data set. Because these two BMUs are similar to the same data point, it is expected that these two BMUs should be located near to each other in the output SOM. The topographic error statistic computes the proportion of first and second BMUs that are *not* adjacent to each other in the SOM. This statistic is intended to convey how well the SOM organizes information across the neurons. The topographic error for the default SOM training was 0.047. The topographic and quantization errors are not really meaningful in absolute terms, but are more useful when compared with those of the subsequent experiments.

4.3 Deriving Clusters from the Default SOM

The SOM results in this dissertation still require interpretation in order to infer groupings and to characterize these groupings. The SOM arranges the GIS procedures across an information space, but does not delineate clusters with crisp boundaries. For example, the U-matrix indicates a degree of similarity, but makes no definitive statement about which neurons contain procedures of the "same" or "different" types. In order to support the author's subjective cluster interpretations of the SOMs, several automated clustering analyses are carried out and presented in this section. The SOM results were clustered by inputting them into K-means clustering and Ward's Linkage clustering analyses. Resultant clusters are presented and discussed. While none of the objective analyses can truly indicate whether a SOM provides the "right" or "semantically meaningful" answer, they are useful in looking for patterns or characteristics of a SOM.

K-means determines the distance between each neuron and neurons that have been automatically selected as the centroids of possible clusters. Assignment of a neuron to a cluster is made in order to minimize the Davies-Bouldin Index (DB Index; Davies and Bouldin, 1979), a Euclidean measure of cluster separation. The process iterates in order to assess whether relocating cluster centroids can further reduce the error figure. According to the MathWorks documentation, "this

iterative partitioning minimizes the sum, over all clusters, of the within-cluster sums of point-to-cluster-centroid distances" (http://www.mathworks.com/help/toolbox/stats/kmeans.html, accessed March, 2011).

The optimal solution, shown in Figure 4-3a), finds that there are 11 clusters in the SOM. The optimal solution is determined based on a set of assignments that minimizes the Davies-Bouldin Index. Identification numbers have been used to label clusters in the figure. The bright green neurons shown in the figure are the centroids of the corresponding cluster. The black neurons have a BMU frequency of zero and have therefore been excluded from the cluster assignment process. Note that although no GIS procedures are associated with these neurons, they still have a valid set of dimension values in the same manner as neurons that do have associated GIS procedures.

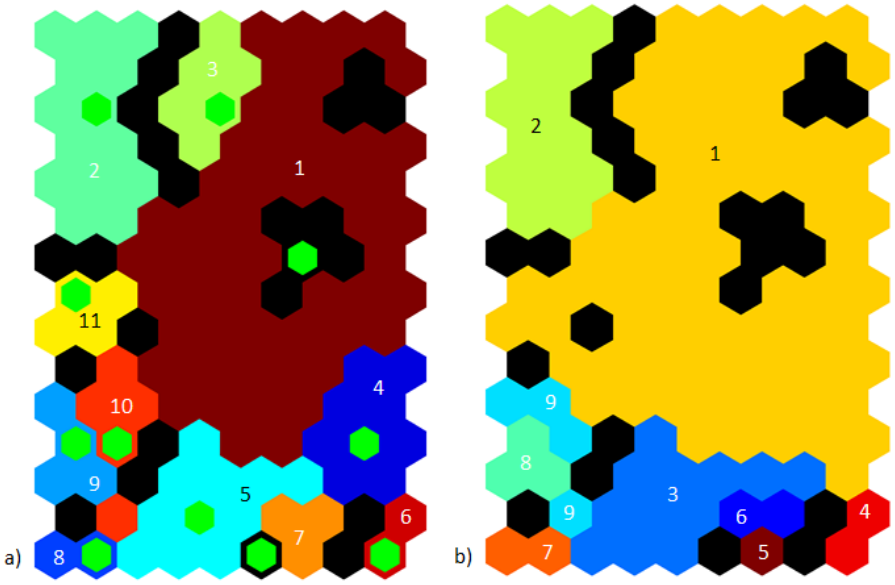


Figure 4-3 Optimized clustering of SOM neurons trained with explicit keyword procedure matrix and default parameters from som_make(). (a) K-means created 11 clusters. Green neurons are cluster centroids. (b) Ward's Linkage created 9 clusters. The number of clusters was optimized using the Davies-Bouldin Index in both sub-figures. Black neurons have no GIS procedures associated with them.

Figure 4-3b) shows the optimized number of clusters according to Ward's Linkage analysis, which is developed as an alternative to K-means. The optimum level of clustering is again determined by

minimizing the Davies-Bouldin Index. Linkage analysis creates an agglomerative hierarchical tree. Clusters are derived based on the single linkage algorithm, which seeks to minimize separations between groupings of neurons. The separation (or distance) metric used was Ward's method, which is defined by the MathWorks as follows: "Ward's linkage uses the incremental sum of squares; that is, the increase in the total within-cluster sum of squares as a result of joining two clusters. The within-cluster sum of squares is defined as the sum of the squares of the distances between all objects in the cluster and the centroid of the cluster. The sum of squares measure is equivalent to the following distance measure d(r,s), which is the formula `linkage` uses:

$$d(r,s) = \sqrt{\frac{2n_r n_s}{(n_r + n_s)}} \left\| \bar{x}_r - \bar{x}_s \right\|_2 ,$$

Equation 4-1

where $\| \ \|_2$ is Euclidean distance, $\bar{x}_r$ and $\bar{x}_s$ are the centroids of clusters r and s, and n_r and n_s are the number of elements in clusters r and s" (http://www.mathworks.com/help/toolbox/stats/linkage.html, accessed March, 2011).

Detailed interpretation of the clusters will be presented in the following section. It is worth comparing the two results briefly now. Despite the fact that the Ward's Linkage clustering yields a different number of optimal clusters than the K-means analysis, the two sets of results are similar. In most cases, the additional clusters found by K-means are subdivisions of the Ward's Linkage clusters (as opposed to forming from neurons from multiple Ward's Linkage clusters). For example, the neurons in the cluster numbered 1 in the Ward's Linkage are largely the same as those in the K-means cluster numbered 1, although it also includes the K-means clusters number 3, 4, and 11. The clusters numbered 2 are the same and cluster 3 in Ward's Linkage matches cluster 5 in the K-means results (with one extra neuron). There are some differences in the smaller clusters found at the bottom of the respective displays.

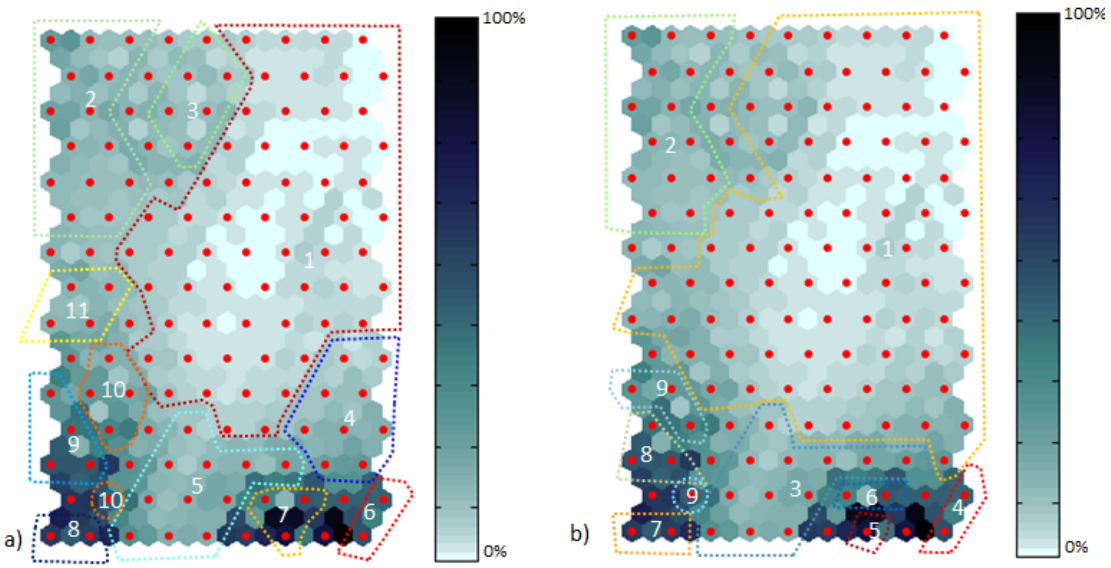
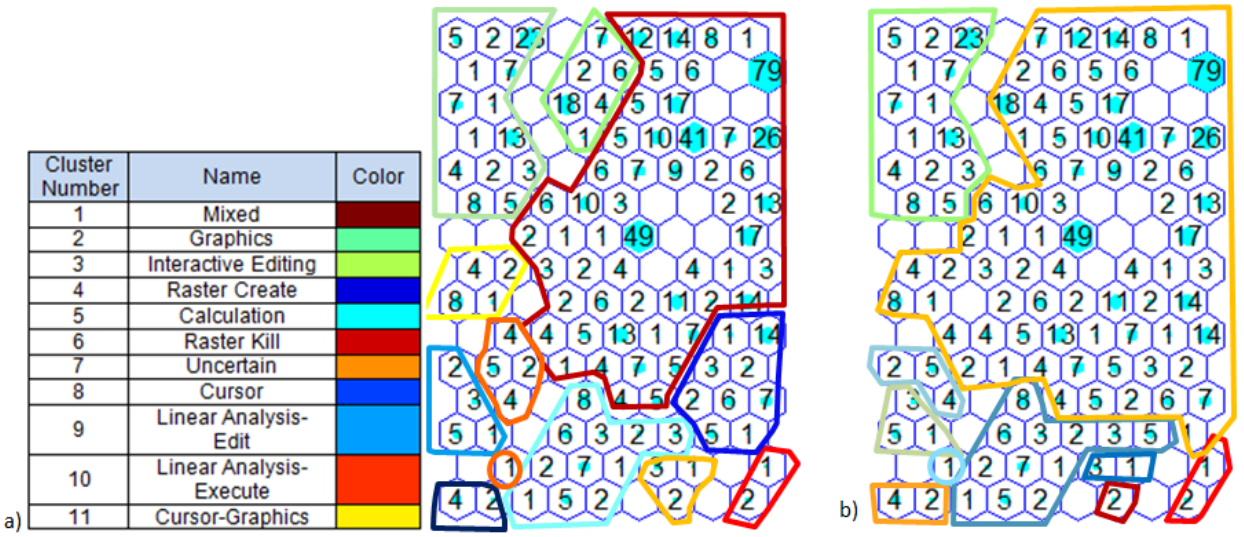


Figure 4-4 Boundaries of optimized clusters superimposed on the U-matrix. Clusters derived from SOM neurons trained with explicit keyword procedure matrix and default parameters from som_make() using (a) K-means and (b) Ward's Linkage clustering. The U-matrix was derived from the same SOM. Boundary colors correspond to those in Figure 4-3.

In order to assess the validity of these automatically generated clusters, it is helpful to superimpose the cluster boundaries onto the U-matrix in order to understand the degree of dissimilarity between neurons within the clusters, the degree of dissimilarity between clusters, as well as SOM-wide patterns of dissimilarity. In Figure 4-4, the cluster boundaries are represented as lines with colors corresponding to those shown in Figure 4-3a). The cluster numbers from Figure 4-3a) are also posted within each cluster boundary in Figure 4-4. Note that if a cluster is composed of more than one patch, each patch is labeled.

Figure 4-5 shows the same boundaries superimposed on the hit histogram shown in Figure 4-2a). This display is useful for understanding whether there is correlation between patterns of similarity between neurons and the number of GIS procedures associated with those neurons. For example, some of the lowest U-matrix values in the SOM are within cluster 1 in both cluster maps in Figure 4-4, but that these values are associated with neurons that were not best matching units for any GIS procedures.



Cluster Number	Name	Color
1	Mixed	
2	Graphics	
3	Interactive Editing	
4	Raster Create	
5	Calculation	
6	Raster Kill	
7	Uncertain	
8	Cursor	
9	Linear Analysis-Edit	
10	Linear Analysis-Execute	
11	Cursor-Graphics	

Figure 4-5 Boundaries of optimized clusters derived from SOM neurons trained with explicit keyword procedure matrix and default parameters from som_make() using (a) K-means and (b) Ward's Linkage clustering superimposed on the hit histogram. The hit histogram was derived from the same SOM. Boundary colors correspond to those in Figure 4-3. Names assigned to K-means clusters shown in sub-figure a) are given in the table

4.3.1 Interpretation of the Default SOM

The meanings with which the clusters are interpreted by examining the GIS procedures contained within each cluster and the signatures of the neurons that make up the cluster and, in turn, by looking at the GIS commands associated with those procedures. Based on this analysis, the K-means clusters are given the names in the table shown in Figure 4-5a) and are described in this section. Interpretation of clusters is important to establish whether the spatial structure of the SOM and the procedure matrix used to create them are semantically meaningful. All such labeling is subjective, as is assessment of semantic meaning. Both rest on human cognition and so cannot be evaluated by statistical or mathematical measures alone. Unless otherwise noted, the clusters referred to in the subsequent description refer to the K-means clusters.

The K-means cluster, numbered 1, is by far the largest in the SOM. It encompasses 39% of all neurons in the SOM and 46% of neurons with non-zero hit frequencies. It is associated with more GIS procedures than any other cluster in the SOM. Because the GIS procedures in this cluster are not

especially dominated by any one command or any one purpose, cluster 1 is named a “Mixed” cluster. Within the cluster, there are regions where different types of GIS procedures concentrate. Quite a few of the GIS procedures associated with neurons at the lower left of the cluster deal with mathematical analysis and graphics generation. At the right edge, many of the associated procedures deal with various kinds of raster processing. Examples include those for generating raster surfaces (such as a digital elevation model), for generating features or zones (such as watersheds or streams), and for generating tables of information based on raster inputs. At the upper left edge, database management functionality is common and so is interactive editing. GIS procedures associated with neighboring neurons show some commonality in the GIS commands they share, but the specific command varies substantially across the spatial extent of the cluster (i.e., math at the lower left, interactive editing at the upper left, raster processing on the right). This leads to drift in the types of GIS procedure as across the cluster.

The Mixed cluster is anchored at its upper right in an area with a low uncertainty pattern (that can be seen in light blue colors at the upper right of the U-matrix in Figure 4-4a). Towards the lower left edges, uncertainty values tend to increase. High U-matrix values found within a cluster indicate potential weakness of a grouping. Generally, the consistency of uncertainty values within a cluster can be thought of as cluster stability. By this assessment, the stability of the Mixed cluster is the highest of all clusters in the SOM. This is interesting because, in addition to the range of procedure types, it also encompasses by far the largest number of neurons. One would assume that neuron signatures would change across the cluster (and this is corroborated by the examination of the distribution of GIS procedures across the SOM). Even though the range of signatures was large, the rate of change from one neuron to its neighbor is never abrupt enough to elevate U-matrix values (which indicate dissimilarity between pairs of neurons). The lack of boundaries appearing in the U-matrix values within the Mixed cluster emphasizes that the U-matrix is a local indicator of change, as opposed to a SOM-wide indicator of change or variability.

Rather than focusing on inter-neuron separation, the K-means clustering iterates to minimize the separation between neurons and automatically selected cluster centroids that can be many steps away from any given neuron. Despite this more global view of the SOM, it did not delineate more than one cluster in this area. The fact that the automated clustering was unable to demarcate a boundary within this part of the SOM indicates a failure of the explicit keyword matrix to differentiate the included GIS procedures.

Cluster 2, named "Graphics," shows the most narrowly focused type of functionality of any cluster in the SOM, containing large numbers of GIS procedures used to carry out the visualization of geographic data. The K-means Graphics cluster is also mirrored in cluster 2 of the Ward's Linkage result. The Graphics cluster is clearly separated from its neighbors. Its eastern boundary is underlain by higher U-matrix values (darker colors), as is the southern edge of the cluster. Within the Graphics cluster, there are a number of U-matrix values that indicate the certainty of association between member neurons is sometimes no better than with neurons located outside the cluster boundary, indicating relatively poor stability of the cluster.

Cluster 3, named "Interactive Editing," is associated with GIS procedures for interactive editing (mostly in the Workstation ArcInfo module, ArcEdit). These procedures also feature some selection related functionality. Several neurons to the right of this cluster, along the upper edge of the SOM are just outside of the cluster, but are very similar. The stability of the Interactive Editing cluster seems poor when cross-referenced with the U-matrix. The U-matrix shows a ring of relatively high uncertainty values (the left portion of which helped define the boundary for the Graphics cluster) which seems to segregate at least the two upper right neurons from the rest of the cluster.

Cluster 4, named "Raster Create," is associated with raster processing procedures that are used for a mix of needs, including creation of raster surfaces, raster zones, and tables derived from raster. While this and several other clusters in the map show a mix of these raster types of procedures, the

Raster Create cluster procedures all have a heavy data management component to them (evidenced by the consistent invocation of commands like 'COPY', 'RENAME', and particularly 'KILL').

Cluster 5, interpreted as "Calculation," heavily associates with tabular calculation of new information. This usually means that new attributes are added to pre-existing vector features or raster zones by the GIS procedures in this cluster. Although there is usually some selection process associated with these calculations, the primary characteristic of the GIS procedures in this cluster is an aspatial operation. There is also an association with raster processing types of GIS procedures. After these two types of functionality within the procedures of the cluster, there are a number of subordinate types of functionality that cloud the interpretation of this cluster's meaning. The Calculation cluster appears to be the second most stable cluster in the SOM after the Mixed cluster, to which it is adjacent. The separation of these two clusters, while not strong, appears to be driven by slightly elevated U-matrix values (slightly darker colors) in the inter-neuron spaces where the two clusters meet. Uncertainty values within the Calculation cluster are higher than for the Mixed cluster, but do not show major variability. The Calculation cluster is shown by the U-matrix to be substantially dissimilar from clusters to its right (numbered 6 and 7) and left (numbered 8 through 10). This separation is stronger than for its separation from cluster 1.

To the right and to the left of the Calculation cluster are regions that are designated to be very unstable. These regions generally feature very high uncertainty values, both across and within clusters. In addition, the hit histogram (Figure 4-5a) shows relatively low numbers of GIS procedures associated with these clusters. A low hit frequency for a single cluster is not inherently bad. This could indicate that best matching GIS procedure(s) for this neuron were different enough from the rest of the set that allocating the procedure a distinct neuron was better than adjusting other neurons that represented different GIS procedures (thereby improving the quantization error, for instance). At the very least, a low hit frequency indicates that there was little competition to adjust the neuron signature.

Cluster 6, named “Raster Kill,” again isolates raster processing procedures with a data management component to them (like the Raster Create cluster), but is much more focused on the data management aspect than the Raster Create cluster. There are only three GIS procedures in this cluster, but they invoke the ‘KILL’ command 93 times. The U-matrix shows relatively high values within this small cluster are a little surprising given the apparent similarity in the set of GIS procedures. At a semantic level, the separation of this cluster from the Raster Create cluster seems unnecessary.

Cluster 7, named “Uncertain,” is only slightly larger than the Kill cluster, having three neurons and 6 associated GIS procedures. The procedures carry out an incongruous set of functions. Two pertain to the creation of annotation, an advanced form of labeling for vector features in graphics. Three derive tabular parameters from raster data, and the last converts and reorganizes vector maps into a raster format. The commonality in this cluster is the invocation of the ‘CALC’ command. One procedure alone invokes it 142 times. This cluster abuts and includes some of the maximum U-matrix values in the entire SOM. Based on the small incongruous mix of GIS procedures and these values, this semantic meaning of this cluster considered to be very low. This example shows that although the K-means optimization was improved by the demarcation of this cluster, the cluster is not necessarily semantically meaningful. This is interpreted as a weakness in the automated clustering algorithm.

Cluster 8, named “Cursor,” is again very small (2 neurons) and unstable (with high U-matrix values), similar to the Kill and Uncertain clusters. In the Cursor cluster, there are procedures for topographic analysis for visualization, preparation of tables for raster creation, and for editing associated with features. The commonality is that these routines all use the ‘CURSOR’ command. It is similar to the Calculate cluster in that it is focused on creation or manipulation of tabular information, but seems to be more focused on pre-processing. This interpretation is somewhat weakly supported by the data. The similarity between the Cursor and Calculation clusters is supported by the fact that these two clusters are adjacent within the SOM.

Cluster 9, named “Linear Analysis-Edit,” includes four neurons and 11 GIS procedures that pertain to analysis of linear features; most pertain to editing vector features, such as stream networks or street system, in anticipation of location analysis and network segmentation. The procedures of cluster 10, named “Linear Analysis-Exec,” are similar to those of the Linear Analysis-Edit cluster in that they rely heavily on the same type of selection GIS commands and the overall purpose of the procedures is analysis of linear features. The separation of these two clusters is difficult to rationalize. The only thing the author can infer is that the “Linear Analysis-Exec” cluster leans slightly more towards the actual execution of linear analyses, as opposed to preparation of input for linear analysis.

Cluster 11, named “Cursor-Graphics,” is a square of four neurons that include GIS procedures that span a number of purposes, from spatial analysis (line of sight, contouring) to interactive editing of spatial features, to tabular modification. Although not dominated by any single type of processing, most of the GIS procedures in this cluster use the ‘CURSOR’ command. This is somewhat surprising because it is positioned many neurons away from the Cursor cluster, which one might expect to be a basis for collocation of the two clusters. These procedures also tend to use some kind of graphics to accomplish their purposes. The Cursor-Graphics cluster shows better similarity among its member neurons than most clusters, although this is a relatively small cluster. Relative to its internal uncertainty, the U-matrix indicates that this cluster is more dissimilar to the neurons adjacent to the cluster boundary shown in the K-means figure.

Although the Calculation and Graphics clusters appear to have similar (medium) levels of uncertainty, interpretation reveals a much clearer purpose for the Graphics cluster than for the Calculation cluster. The region bounding the lower edge of the SOM (featuring the Kill, Uncertain, Linear Analysis-Edit, and Linear Analysis-Execute clusters) shows high degrees of dissimilarity between neighboring neurons, regardless of whether the neighbor is part of the same cluster. The clustering results drive home the same point shown by the high U-values in the area, which is that neighboring

neuron values in this SOM have very different signatures. It is also informative to note that the areas with relatively high U-matrix values are often centered around neurons with a zero hit frequency and neighbors with low hit frequencies, forming “bulls-eyes of uncertainty.” The space between the Kill, Uncertain, and Raster clusters is an example.

In summary, although the SOM technique is reasonably successful in using the explicit keyword procedure matrix to create a spatial pattern to differentiate GIS procedures, there seems to be large degree of overlap between clusters (for instance, mixtures of raster processing was common in many). The result is dominated by a single, very large cluster that is not considered to be meaningful. The SOM training algorithm is not able to differentiate equally well between all GIS procedures using the explicit keywords in the procedure matrix. The large homogenous region of the Mixed cluster could indicate that the training algorithm is not sensitive enough to variations in the explicit keywords or that the explicit keywords do not have enough/appropriate content. On the other hand, the region at the lower edge of the SOM could be argued to be over-fragmented (evidence by many clusters and the highest U-matrix values).

In order to ensure that this single attempt does not inappropriately characterize the explicit keyword matrix, the parameters with which the SOM is trained are optimized in the next section. Notable among these parameters are specifications of the matrix size. By enlarging the matrix, it is possible that at least the smaller clusters seen in the default SOM will be able to organize more effectively. Another parameter whose adjustment might improve the SOM is the size of the neighborhood of influence around a neuron over which adjustments are applied. The default SOM was developed using a neighborhood radius that was relatively small. While larger neighborhoods tend to smooth the result, essentially enforcing a trend outwards from a given neuron, they also tend to create a more logical spatial arrangement of clusters. A small neighborhood effectively allows a neuron to be adjusted with little or no impact on its neighbors, physically independent. The small neighborhood size

used by the som_make() function could account for the rapid change in neuron signatures (indicated by very high U-matrix values) at the lower edge of the SOM. Another possible cause of the poor SOM organization is that the small number of training iterations used (specified as a parameter to the training process) constrained the degree to which the training algorithm could adjust the randomly initialized state that the SOM begins with.

4.4 Optimized SOM Training Results

The training process for the default SOM was optimized based on recommendations outlined in Wendel and Buttenfield (2010). The procedure matrix was used to create a second SOM, optimizing the training process by specifying size and shape of the output SOM and the number of training iterations. The SOM size was specified at 1653 neurons, according to the Vesanto (2005) equation (5*sqrt(nrows*ncols)), cited by Wendel and Buttenfield (2010). The SOM shape was specified as a toroid of hexagonal cells. It was further specified that for the two-dimensional visualization, the neurons were to be arranged so that the ratio of the sides of the SOM would be 1.5 : 1, as defined in Wendel and Buttenfield (2010) producing 51 rows and 33 columns, as shown in Figure 4-6a).

For the rough training, a Gaussian neighborhood function was specified. The knowledge about the size of the sides of the two-dimensional representation of the SOM was used to set the initial search radius equal to the long dimension (51) at the initial time step and decayed linearly to half that length (26) by final training iteration. 10,000 iterations were used for the rough training phase. 5,000 iterations were used for the fine tuning phase.

The quantization error after the rough training phase was 6.1743. The topographic error was 0.0352. The fine tuning phase began with a search radius of half the length of the long side of the SOM (i.e., 26) and decreased to a single neuron over 5000 iterations. This work reduced the quantization

error to 4.3628 (slightly greater than for the default SOM) and also improved topographic error to 0.0244 (roughly half the value for the default SOM).

Figure 4-6a) shows the frequency that each neuron was a BMU in a hit histogram. In contrast with the default SOM training results (shown in Figure 4-2a), there are a very large number of neurons with frequencies less than 5, and many more which have no data points associated with them at all. 75% of all neurons have a zero hit frequency (compared with 17% for the default SOM). There are still a handful of neurons with very high frequencies, such as one with a value of 47 (near the upper left of the SOM, at row-column position (1,6)). Table C-3 in Appendix C lists the neuron identification numbers and the associated GIS procedures that found the associated neuron to be the best matching.

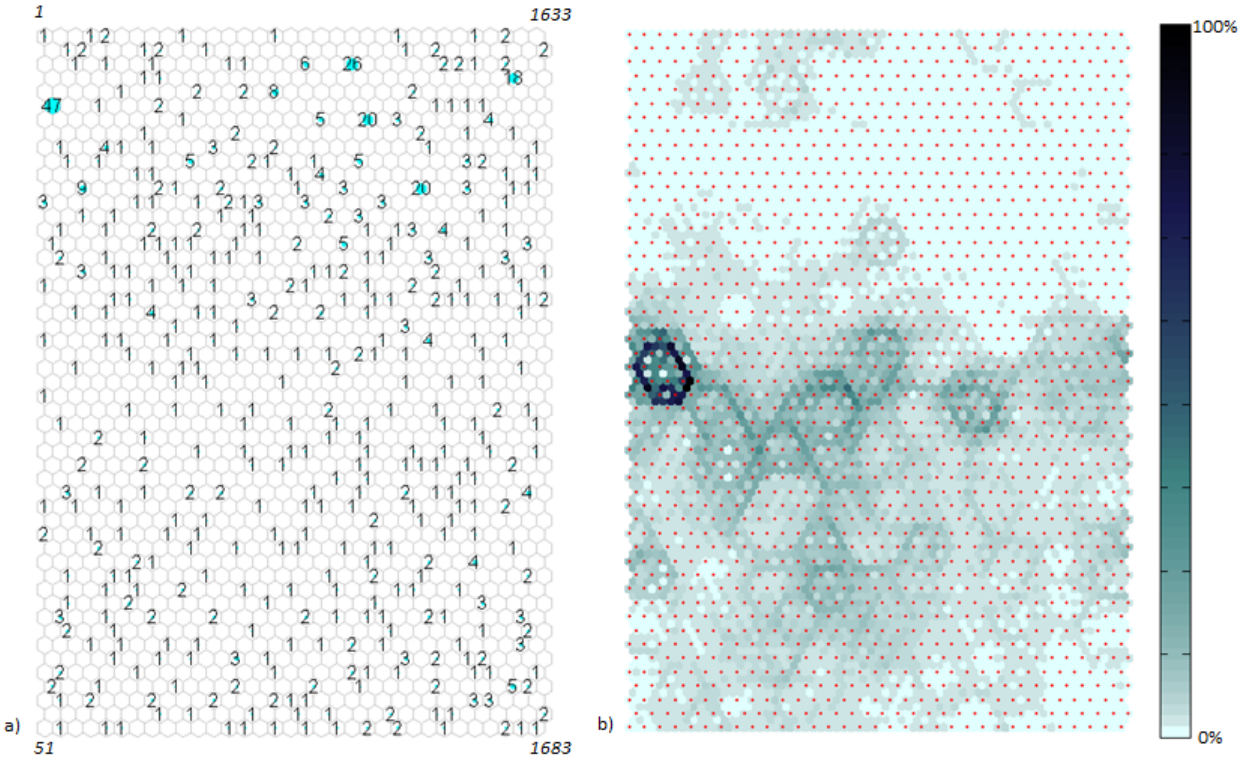


Figure 4-6 The SOM trained with the explicit keyword procedure matrix and parameters derived from Wendel and Buttenfield (2010). (a) Neurons labeled with frequency of best-matching GIS procedures. The size of the blue patch also indicates match frequency. Corner neuron identification numbers posted in italics. Neurons are numbered sequentially from top to bottom, left to right. (b) U-matrix for the SOM, showing dissimilarity as a darker color. Color indicates percent of range in U-value within SOM. Actual values range from 0.01 to 85.64. Regions of lighter colors indicate groups of similar neurons and darker values indicate separation between groups. The red dots indicate locations of neurons.

Figure 4-6b) shows the U-matrix for the optimized SOM. The average dissimilarity value in the SOM increased from 2.6233 in the default SOM to 3.260 (the median was more stable, but still increased from 1.7444 to 1.8202), the standard deviation nearly doubled in going from 2.8526 to 5.3543. The visualization is similar to those in Figure 4-2b), in showing a large area of relatively low U-values (in this case, a horizontal band across the top of the matrix). In contrast with Figure 4-2b), a number of ring-shaped features are also visible (for example, at the center left). Although the areas demarcated by the rings indicate a relatively strong separation from neurons beyond the ring, there are relatively high uncertainty values among neurons within the rings. This could be interpreted to signify that the strength of the similarity of the within-ring neurons is relatively weak. Although GIS procedures are associated with the "best-matching units," it appears that the training process had difficulty organizing the overall structure of the SOM. This could be caused by the matrix being so large that it allowed GIS procedures to be spread out farther than the neighborhood of adjustment used in the training process and resulted in over-fitting around individual data points.

The large number of neurons in the optimized SOM appears to be problematic for both the K-means and the Ward's Linkage results, shown in Figure 4-7a) and b), based on the relatively low optimal number of clusters and the fact that the only two clusters, outside of the one that encompasses the lion's share of the matrix, is concentrated in relatively small area of the SOM (which incidentally shows a relatively high degree of separation according to the U-matrices shown in Figure 4-6b). These results do not convey any real meaning.

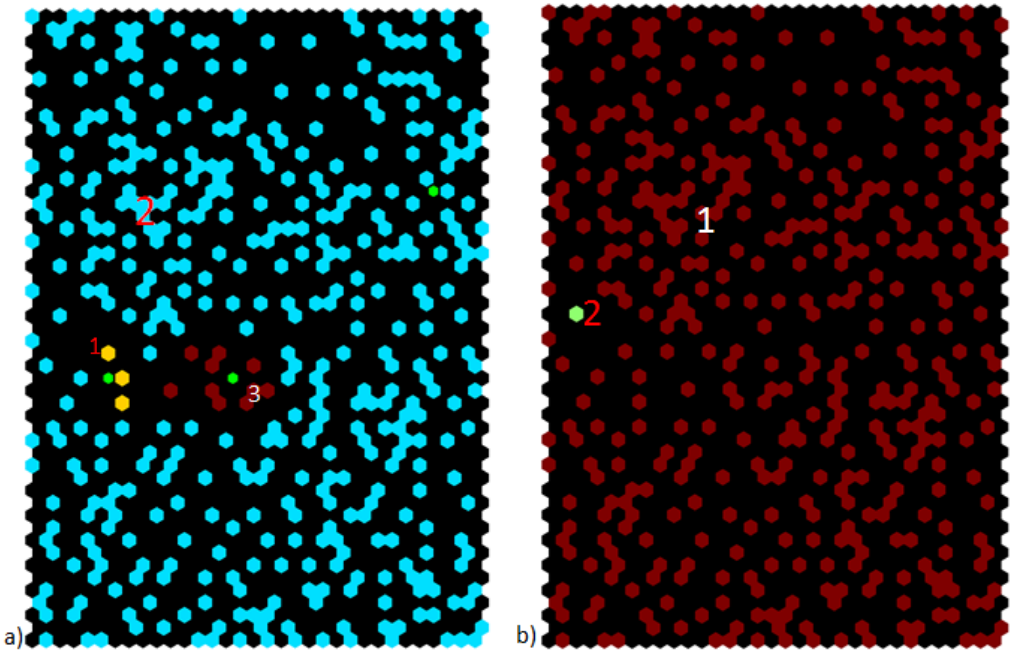


Figure 4-7 Optimized clustering of SOM neurons trained with the explicit keyword procedure matrix and parameters derived from Wendel and Buttenfield (2010). (a) K-means created 3 clusters. Green neurons are cluster centroids. (b) Ward's Linkage created 2 clusters. The number of clusters was optimized using the Davies-Bouldin Index in both sub-figures. Black neurons have no GIS procedures associated with them.

Figure 4-8 shows the boundaries of the K-means clusters superimposed on both the U-matrix and the hit histogram. The logic for delineation of these clusters is not immediately apparent when looking at the U-matrix. There are a number of rings of high U-matrix values which one would intuitively interpret to indicate a boundary around a region of similarly typed neurons that are distinct from neighboring areas. Mathematically, one might expect that clusters would be defined based on these artifacts because having neurons from opposite sides of these boundaries would increase the DB-Index (which the cluster optimization seeks to minimize). The very dark ring at the left edge of the SOM, pointed out by the yellow arrow, is an example. The hit histogram provides the answer, indicating that there is only a single GIS procedure in this region of SOM. Because neurons with zero hit frequencies are excluded from the clusterings, there is only a single viable neuron to be considered. The global optimization of the DB-Index for the K-means clustering concludes that defining a new cluster for this single anomalous neuron is too expensive (although this single neuron is isolated as a cluster in the

Ward's Linkage analysis). Any other neurons forced into the new cluster would be much more similar to almost any other neurons in the SOM than this one. This ring also is interesting because it shows the size of the neighborhood used in the SOM training process. This points out that, as mentioned in the description of the default SOM, the clustering algorithms are not driven by U-matrix patterns, but by broader trends in the signatures of the neurons. The GIS procedure that has caused this feature is itself unusual because it uses the 'CALC' command 142 times, and almost no other GIS commands.

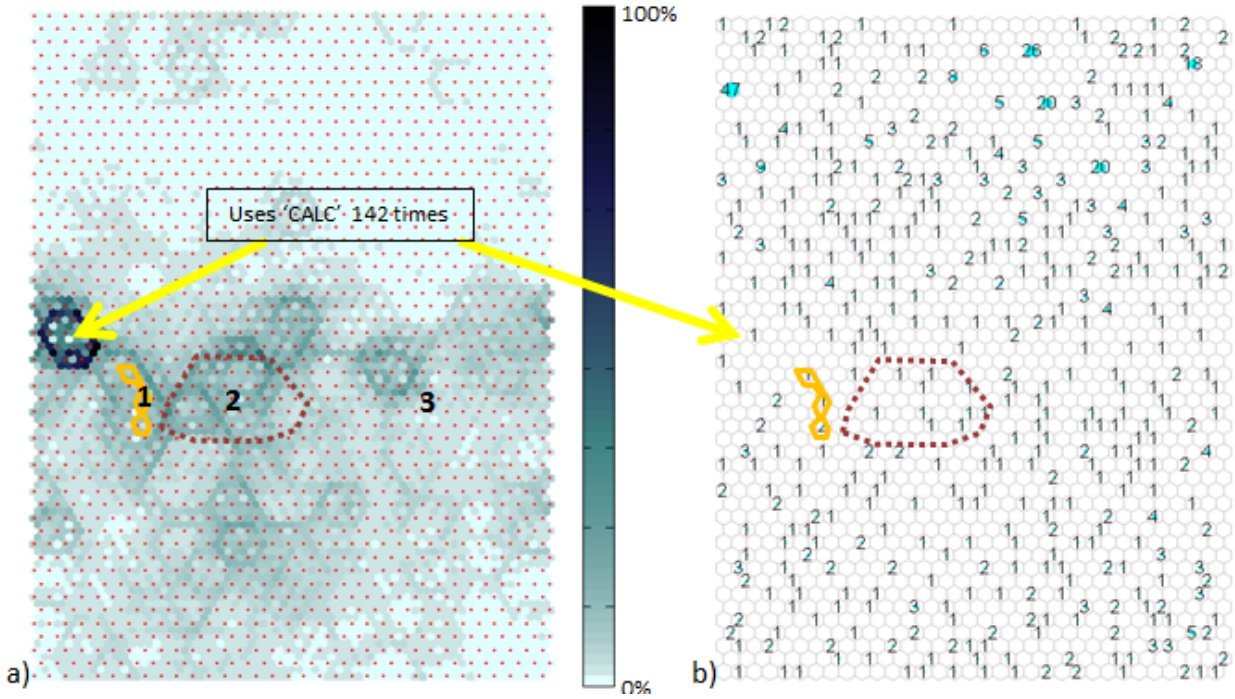


Figure 4-8 Boundaries of optimized K-means clusters shown in Figure 4-3 superimposed on a) the U-matrix and b) the hit histogram for the SOM trained with the explicit keyword procedure matrix and parameters derived from Wendel and Buttenfield (2010). Cluster identification numbers are posted to sub-figure a).

Cluster 1, named "Cursor" in Table 4-2, has a somewhat darker ring of dissimilar values surrounding it. This ring is larger than the cluster itself. The cluster is composed of only neurons with non-zero numbers of hits within the ring. Because this cluster is so small, it is difficult to infer anything about the stability of the cluster. Given that it represents only four GIS procedures out of the entire set ("fly_around", "flyby", "reclass", and "slice"), compared to just about every other procedure (in cluster

2), the validity of cluster 1 is limited. All four of these procedures in this cluster show very high numbers of GIS commands invoked (121, 166, 57, and 63, respectively) and presumably exert a relatively strong impact on the best-matching neuron during the training phase. While the first two GIS procedures are indeed very similar (their purpose is for fly-through visualizations) and so are the second two (their purpose is for reclassification of features), the two pairs are dissimilar from each other. When the GIS commands that these procedures have in common are examined, 'CURSOR' and a variety of graphics-generating commands are found.

Table 4-2 Names given to K-means clusters (Figure 4.7a) of SOM trained with explicit keyword procedure matrix and parameters derived from Wendel and Buttenfield (2010).

Cluster Number	Name
1	Cursor
2	Linear Analysis
3	Everything Else

Cluster 2, named "Linear Analysis" encompasses more neurons than the Cursor cluster, but is still composed of a very small number relative to the overall matrix size and number of clusters. The neurons in this cluster are each associated with a single GIS procedure (the cluster is associated with 10 procedures in total). There is a degree of dissimilarity within the limits of this cluster according to the U-matrix. The purposes of GIS procedures within the cluster include interactive editing of vector datasets within the ArcEdit module of Workstation ArcInfo and execution of spatial analyses such as location-allocation analysis. There are nearby neurons that are also associated with GIS procedures that the author considers to be similar to those within the cluster, and so the relatively small extent of the cluster is considered inaccurate. The GIS procedures in this cluster most frequently invoke GIS commands relating to selection and, to a lesser degree, graphics generation. Two of the neurons at the right of the cluster refer to spatial analysis and mathematical operations.

Cluster 3, named "Everything Else," is so large that it essentially forms the background on which the other two clusters are located. It encompasses 97% of the neurons with non-zero hit frequencies.

Clearly there is such a broad range of GIS procedures within this cluster that it is not a meaningful grouping. The Ward's Linkage result generates only two clusters, one of which was comprised of the single neuron with the GIS procedure that uses the 'CALC' command 142 times. This result is judged to contain no relevant breakout of the SOM structure at all (other than to indicate a problem with the SOM) and so its clusters are not discussed.

There are definite regions of types of GIS procedures in the SOM, although apparently these regions were too diffuse to be recognized by the clustering algorithms. A large horizontal band of the various raster-processing GIS procedures (zone feature creation, surface creation, tabular parameterization) appears above the Linear Analysis cluster. From the right of the Linear Analysis cluster to the right-hand edge of the SOM, there are many neurons associated with GIS procedures whose purpose is related to interactive editing within Workstation ArcInfo's ArcEdit module. This region also features a number of neurons associated with database management operations. At the boundary towards the Linear Analysis cluster, this region shows a mix of both interactive editing and spatial analysis GIS procedures. As noted above, this same mixture is seen at the adjacent part of the Linear Analysis cluster, which calls the meaning of this boundary of the Linear Analysis cluster into question. Below the Linear Analysis cluster is another large region of neurons associated with graphics and visualization. The lower right corner of the SOM featured a mixture of types of GIS procedure associations. Although the automated extraction of clusters was largely a failure in this experiment, the organization of the types of GIS procedures across the optimized SOM definitely provides some information content.

The premise behind optimizing the SOM training parameters was to better leverage the content of the explicit keyword procedure matrix. It is arguable that the effect was the opposite because of the negative artifacts in the SOM described above. Increases in statistics of the U-matrix could be good signs if they indicated a clear set of cluster boundaries, but this is not the case for the optimized SOM. The

extra space in the matrix enabled the creation of silos and generally increased the dissimilarity between neighboring neurons. This dissimilarity and the sparse nature of the hit histogram resulted in vastly reduced ability to resolve groupings of neurons using the K-means and Ward's Linkage analyses.

The overall quality of the SOM is interpreted to be extremely poor. The U-matrix reveals that sparse distribution of BMU hits causes the SOM surface to be highly variable at a local (inter-neuron) scale and the under-differentiated clustering results reveal that the SOM is highly smooth at broader scales. In addition to impacting the extraction of clusters, the optimization of the training parameters definitely has an effect on the overall organization of the SOM. The quality of this organization is weakened by the size of the matrix. By allowing the GI S procedures to spread so far apart, often beyond the size of the SOM training neighborhood, the capacity of the SOM algorithm to organize the data spatially is weakened. One possible way to compensate for the apparent failure of the SOM is to use a different clustering algorithm that is not confused by such a sparse matrix.

4.5 Dimensional Redundancy

Because the procedure matrix has so many dimensions (i.e. 148 explicit keywords), there is an increased likelihood that some describe the same characteristic and are therefore redundant. For example, a command for filling depressions in a DEM might be considered very similar to a command for deriving flow direction from a DEM because both look at the direction of steepest descent away from each cell. This kind of redundancy essentially multiplies the weight of the characteristic, which may be inappropriate for some datasets. In such cases, many researchers seek to reduce or eliminate redundancy. On the other hand, some redundancy might be desirable to allow data points that are similar, but not identical, to be grouped near to each other in the SOM even if they do not share the same BMU. Although elimination of redundancy may ease SOM training costs as well as the identification of clusters, some redundancy may allow the SOM technique to more effectively arrange

the clusters relative to each other across the SOM visualization into a coherent pattern. Ultimately, the need for a reduction in the number of dimensions is case-specific and likely a subjective judgment.

In order to explore this issue within the set, a dimension reduction technique, Principal Component Analysis (PCA, described in Section 4.5.1), was used. The results of the PCA are used in several ways. First, the Eigenvalues and Eigenvectors produced by the PCA are be used in Section 4.5.2 to better understand the explanatory power of the explicit keywords in the procedure matrix. This information can provide insight into which explicit keywords (i.e. GIS commands) tend to replicate each other, and which have the most power to delineate groupings of GIS procedures. Even though the two dimensions that define the SOM visualization are not directly connected with the principal components that one can generate from the same data set, knowledge gained from PCA (such as which GIS commands were important overall) can be used to interpret the results of the SOM more readily.

Second, the PCA was used to produce a set of coordinates derived from the input data points that locates the points in PCA space. PCA coordinates derived for both the input data points of the procedure matrix and for the neurons in the related SOM are plotted to provide an alternate visualization that, like the U-matrix, helps communicate how the data points are separated. This display is described in Section 4.5.3.

Lastly, a subset of the PCA results (the first 15 principal components) derived from the procedure matrix is used as input to the SOM training process to test whether a substantially different SOM would be produced as a result of the PCA dimension reduction process. This last output, referred to as the PCA-driven SOM, is the final attempt of the explicit keyword experiments to improve the SOM produced by the explicit keyword procedure matrix. This work is described in Section 4.6.

4.5.1 Overview of Principal Component Analysis

The columns of the procedure matrix (i.e., GIS commands) define the dimensions of an information space. The vector of dimension values associated with a data point (i.e., GIS procedures) gives its position in that space. The values themselves indicate the frequency of the corresponding GIS command within the GIS procedures. It is assumed that no single dimension is necessarily independent of other dimensions. The PCA technique clusters sets of data points by projecting the dimensions of the input data set into a new information space with a reduced number of dimensions, all of which are uncorrelated. In essence, the projection into a new information space is analogous to looking at a reflection of a multi-dimensional data set from a new, more informative vantage point. The constructed view is based in fewer dimensions which are based on the strongest (i.e., "principal") structural components. Component strength is based on the amount of variance in the data set which is explained by the principal components, and quantified in the form of Eigenvalues.

In this study, the dimensions are defined by the frequencies of the 148 GIS commands. The PCA assigns a weight (the Eigenvectors) to each dimension of each of the original data points, for each principal component. These weights can be thought of as the specification of a transformation or re-projection of the original procedure matrix into a new information space. They can also be thought of as an indicator of how much influence each of the original dimensions has in defining a given principal component.

Although the PCA in fact produces an information space with as many dimensions as the input (although the keyword dimensions have been replaced by principal components), it also produces information (in the form of Eigenvalues) that provides clear indication on the explanatory power of each of the new dimensions and thereby supports the selection and application of a threshold by which the data analyst can throw out a large number of the new dimensions while still explaining most of the

variation in the input procedure matrix. There are rules-of-thumb for setting this threshold, but it is ultimately a subjective decision or at least one to make through data exploration.

4.5.2 Results of the Principal Component Analysis of the Input Data

The Eigenvalues (the strength) of the first sixteen principal components generated from the procedure matrix are shown in Table 4-3. Eigenvalues are useful for evaluating how many principal components are sufficient to constitute a new information space that is capable of explaining enough of the variance in the initial data to appropriately separate the data points into relevant clusters. A rule of thumb is that principal components with Eigenvalues of less than 1.0 are lack sufficient explanatory power to warrant carrying them into subsequent clustering processes (p. 424, Cohen and Cohen, 1983). Applying this rule of thumb to create the optimized SOM, the first 15 principal components were used.

Table 4-3 Principal components derived from the procedure matrix of explicit keywords.

Principal Component	Eigenvalue	Cumulative Percent (total variance explained)	Percent Change
1	29.868	29.868	
2	16.2524	46.1204	0.455859
3	9.3345	55.4549	0.425654
4	7.4092	62.8641	0.206256
5	4.6473	67.5114	0.372766
6	4.1358	71.6472	0.110064
7	3.2844	74.9316	0.205861
8	2.9364	77.868	0.105955
9	2.5493	80.4173	0.131828
10	1.6966	82.1139	0.334484
11	1.5526	83.6665	0.084876
12	1.4751	85.1416	0.049916
13	1.2248	86.3664	0.169683
14	1.1272	87.4936	0.079686
15	1.1025	88.5961	0.021913
16	0.9093	89.5054	0.175238

An alternative way to look at the Eigenvalues is to visualize them in a scree plot (Figure 4-9). The plot shows a relatively consistent rate of degradation in the explanatory power of principal components

with increasing order. That is, there appears to be a regular rate of change in the slope of the trend line that has been fitted to the data points. Visual interpretation of this plot suggests that the explanatory power for principal components after the 5th exhibit a much reduced rate of reduction, and the absolute values of the Eigenvalues of these principal components are so low that they probably do not enhance the effectiveness of subsequent analyses. Referring back to Table 4-3, however, shows that although the first five principal components explain most of the total variance (~67 percent), by maintaining the first 15 explains an additional 21 percent (for a total of ~88 percent) of the total variance.

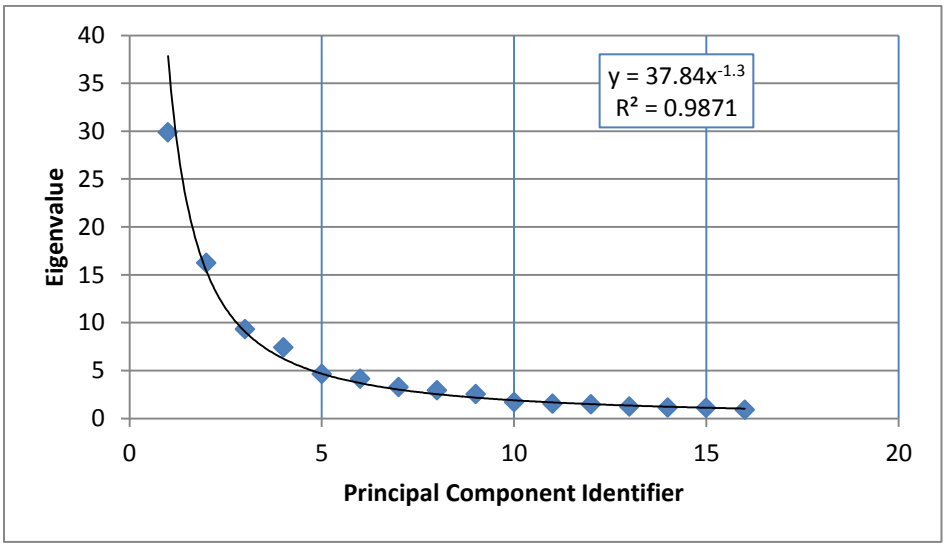


Figure 4-9 Scree plot showing the eigenvalue of principal components derived from the procedure matrix.

The 148 Eigenvectors associated with each principal component indicate how heavily each component relies on each of the explicit keywords to explain variation in the input procedure matrix. By looking at the Eigenvectors, the reader can interpret the meaning or type of GIS functionality associated with the principal components. Table 4-4 shows the ten explicit keywords (the rows) that are most important for the top 5 principal components (the columns); the remaining Eigenvectors for principal components are not shown because their importance to any component is relatively small. The table

contents are discussed briefly here in order to impart to the reader how the reduced-dimension PCA coordinate space condenses the original information space of the procedure matrix.

Table 4-4 Ten highest loading explicit keywords (GIS commands with highest Eigenvectors) to the first five principal components derived from the procedure matrix. Component loadings are shown in parentheses below each keyword. Highest positive loadings are shown in red and lower loadings in black. Negative loadings are shown in blue.

Explicit Keyword	Principal Component				
	1 Data transformation	2 Selection (graphics)	3 Searching	4 Data mgt (Raster Creation)	5 Launch GIS (Relational links)
Importance 1st loading	'calc' (0.9988)	'ASELECT' (0.7262)	'CURSOR' (0.8876)	'KILL' (0.8057)	'ARC' (0.8772)
2nd loading	'ARC' (0.0414)	'RESELECT' (0.4878)	'ASELECT' (-0.2316)	'CON(' (0.4710)	'CURSOR' (-0.2440)
3rd loading	'KILL' (0.0097)	'CURSOR' (0.2472)	'ARC' (0.2274)	'ARC' (0.2370)	'CON(' (-0.1950)
4th loading	'CLEARSELECT' (0.0079)	'READSELECT' (0.2331)	'CALCULATE' (0.1373)	'SIN(' (0.1833)	'ASEXECUTE' (0.1742)
5th loading	'COMBINE(' (0.0067)	'WRITESELECT' (0.1743)	'RESELECT' (-0.1324)	'ISNULL(' (0.0674)	'KILL' (-0.1421)
6th loading	'SORT' (0.0061)	'MARKERSYMBOL' (0.1096)	'MARKERCOPY' (0.1031)	'SETMASK' (0.0659)	'RELATE' (0.1104)
7th loading	'ADDITEM' (0.0056)	'LINECOLOR' (0.1044)	'READSELECT' (-0.0994)	'ASELECT' (0.0598)	'SIN(' (-0.0747)
8th loading	'MAPEXTENT' (-0.0051)	'MARKERSET' (0.0912)	'MARKER-SYMBOL' (0.0714)	'EDIT-FEATURE' (-0.0589)	'SORT' (0.0714)
9th loading	'CALCULATE' (0.0050)	'NSELECT' (0.0871)	'MARKER' (0.0698)	'COPY' (0.0555)	'RESELECT' (0.0417)
10th loading	'CURSOR' (0.0048)	'MARKERCOLOR' (0.0818)	'WRITESELECT' (-0.0692)	'RENAME' (0.0527)	'READSELECT' (0.1064)

The first principal component list keywords that transform data (‘calc’, ‘CALCULATE’, ‘ADDITEM’, ‘COMBINE(‘, and ‘SORT’). It is interesting to note that all but the first keyword show loadings that are very close to zero. This indicates that these keywords are in fact not strongly correlated (positively or negatively) with this component. Note that although ‘calc’ and ‘CALCULATE’ explicit keywords refer to the same command, but that they are tracked as different GIS commands. If the frequency tabulation software used to develop the procedure matrix were more sophisticated, the totals for ‘calc’ and ‘CALCULATE’ keywords would be combined into a single figure. When tracked separately, these two GIS commands have drastically different loadings. ‘calc’ accounts for nearly all of the loading in this component. This most likely reflects that the frequency with which ‘calc’ appeared (460 times) within the GIS procedures set greatly exceeded that of ‘CALCULATE’ (106), the third most frequently appearing GIS command in the entire set of GIS procedures.

The second principal component, which explains an additional 16% of the variance, relies on two types of GIS commands. The most important type of GIS commands are those related to selection (‘ASELECT’, ‘RESELECT’, ‘CURSOR,’, ‘READSELECT’, ‘WRITESELECT’, ‘NSELECT’). The five highest ranked explicit keywords pertain to selection, i.e., making subsets of information. The sixth through tenth most important keywords pertain to visualization or graphics, save the ninth, which again is a selection keyword. These remaining keywords have near-zero loadings.

The third principal component is strongly and positively correlated with the ‘CURSOR’ command which is used to search through tables of data items. The high positive correlation of this command dominates this component, which explains an additional 9% of the total variance in the data set. The component shows consistent negative correlation with selection type commands ('ASELECT', 'RESELECT', 'READSELECT', and 'WRITESELECT'), with the loading for ‘ASELECT’ relatively high. This combined with the other negative loadings is interpreted to mean that this component reflects searching, but not selection. No other types of GIS commands show a negative correlation with this principal component.

GIS commands for dealing with symbology in graphics ('MARKER', 'MARKERSYMBOL', 'MARKERCOPY') show low positive correlations with this component.

The fourth principal component is related with several types of GIS commands. It is most strongly correlated with data management types of GIS commands and explains 7.4%, about the same quantity of variance as the third principal component. Three explicit keywords for data management ('KILL', 'COPY', 'RENAME') create or modify (delete) data associated with this principal component. The 'KILL' command has the highest positive correlation (0.8057), while the correlations for other two are much lower. The component also emphasizes creation of new raster data sets shown by positive correlation with the GIS commands 'CON(', 'SIN(', 'ISNULL(', and 'SETMASK', although the strength of the correlation on the last two commands is relatively weak. The 'CON(' command has the second highest positive loading (0.4710). The 'SIN(' command, which derives a raster surface of sine values from the input raster, is relatively weakly correlated (0. 1833). The command 'ARC,' which launches Workstation ArcInfo, loads somewhat highly (0.2370).

The fifth principal component emphasizes the 'ARC' command, which launches Workstation ArcInfo, with a 0.8772 correlation. This component is (weakly) positively correlated with selection commands ('RESELECT', 'READSELECT') and relational data access ('ASEXECUTE', 'RELATE), but is negatively correlated with 'CURSOR'. This component has the strongest negative correlations with searching, logic, and data management. It is also negatively correlated with creation ('CON(', 'SIN(') and destruction of datasets ('KILL').

4.5.3 Presentation of Default and Optimized SOMs Using PCA Projections

As mentioned above, the traditional SOM display, like those used in Figure 4-2, shows all neurons as uniformly spaced. While this makes the display orderly and easy to interpret, it implies that all neurons are equally similar to each of their adjacent neighbors. This is of course, not the case. The U-

matrix provides an analysis and visualization that attempts to overcome this problem. An alternative approach is to abandon the uniformly spaced visualization altogether. While one could generate a plot of the data point (and SOM neuron) positions using all the original dimensions, visualizing the result (in this case, a 148-dimensioned plot) would be impossible. Selecting just two of the dimensions is possible, but because one presumes that all dimensions are equally important or at least that there is no clear way to establish the two most important dimensions, this is not a useful approach. Because the PCA indicates the two most important principal components, these can be used to project both the data point and neuron positions of the default SOM (Figure 4-10a) into the PCA space (using the Eigenvectors). Because the positions of these points are not forced to be uniformly spaced, as in the traditional display, this provides an effective visualization of how well the explicit keywords separate the data points in the two dimensions that explain the most variation in the dataset and how well the SOM neurons approximate the distribution of the data points.

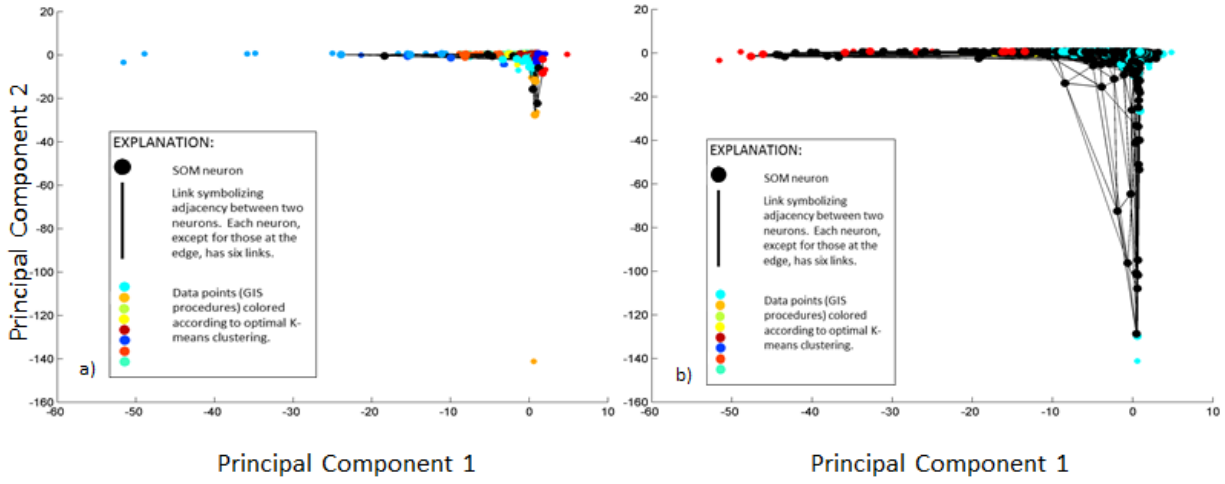


Figure 4-10 Data points and SOM neurons plotted using coordinates of the first two PCA components. (a) SOM neurons trained with explicit keyword procedure matrix and default parameters from som_make(). (b) SOM trained with the explicit keyword procedure matrix and parameters derived from Wendel and Buttenfield (2010). PCA analysis of the explicit keywords explained 42.23 percent of the variance in the explicit keyword procedure matrix.

In the Figure 4-10a, the data points are colored according to the groupings in the optimal K-means clustering scheme (shown in Figure 4-3a). Although the actual colors used to differentiate clusters are not consistent across the two figures, the separation of GIS procedures into clusters is. In addition, the PCA coordinates for the default SOM are used to plot the neurons, shown as black dots. This display shows that the explicit keywords are not successful at separating data points because, aside from a few cyan colored data points, most are concentrated close to the origin. While generating a display using more than the first two principal components might be better, the improvement would likely be marginal because the explanatory power of higher-number principal components decreases by definition (and the decrease is rapid in this case, as shown by Figure 4-9). Although the adjacency between neurons in the SOM is visualized here as connections between the black dots, this is not apparent because the SOM derived from these data is similarly compacted. The two principal components of the data points' original dimension values explain approximately 42 percent of the variance in the data points' spread.

Figure 4-10b) shows the same data points plotted according to their PCA coordinates, but plots the positions of the neurons of the *optimized* SOM instead of the default SOM. In addition to more black dots (because there are more neurons in the optimized SOM), the spread of the neuron's dots more closely mimics that of the input data (although the spread of the data points has not changed). The improvement of the optimized SOM skill in separating types of GIS procedures based on this metric is negligible. On the positive side, this display illustrates that having more neurons available in the SOM allows the training process to accommodate the data points whose principal coordinates place them relatively far from the center of the region around which all data points congregate. For example, there are a number of neuron dots that extend to approximately -45 on the X-axis (where the PCA of the default SOM, shown in Figure 4-10a), indicates that no neurons reached below -25).

4.6 PCA-Driven SOM

The coordinate position for each GIS procedure within a PCA-produced information space was then used to define a new procedure matrix and to derive a new SOM. The PCA-driven procedure matrix, derived from the explicit-keyword procedure matrix, is defined by the first 15 principal components. The current section presents SOM results created using the PCA-derived coordinates for each GIS procedure as input to the SOM training process. The SOM training process was configured as specified for the optimized SOM training, described in Section 4.4. According to the Vesanto (2005) equation (5*sqrt(nrows*ncols)), the 738 GIS procedures and the 15 principal components equated to a matrix of 140 neurons, arranged in 10 columns and 14 rows, as shown in Figure 4-11a). Note that the total number of neurons is greatly reduced because of the reduced number of dimensions (15, from the original 148 explicit keywords).

Figure 4-11a) also shows the hit histogram for the SOM generated using the PCA coordinates for the procedure matrix. There are a handful of neurons in the upper left of the SOM with very high BMU frequencies. Outside of a frequency of 32 in the lower left of the SOM, there are a small number of frequencies with magnitudes in the 10-15 range distributed across a relatively sparse matrix. This spread is obviously very different than that for the optimized SOMs (Figure 4-6), although it shares a number of characteristics with the hit histogram for the default SOM (Figure 4-2). Both the default and PCA-driven results show the same grouping of the 3 very high hit frequencies, albeit in different regions of their respective SOMs. In addition, the 4th highest hit frequency is clearly separated from the top 3 in both SOMs, but not by a large distance if one wraps from one edge of the SOM to the opposite. This indicates that the structure of the PCA-driven SOM is similar to the default SOM. 24% of the neurons in the SOM have a zero hit frequency (i.e., the neuron was the best match for none of the GIS procedures), which is about 7% higher than for the default SOM (and about 1/3rd of the value for the optimized SOM). Table

C-4 in Appendix C lists the neuron identification numbers and the associated GIS procedures that found the associated neuron to be the best matching.

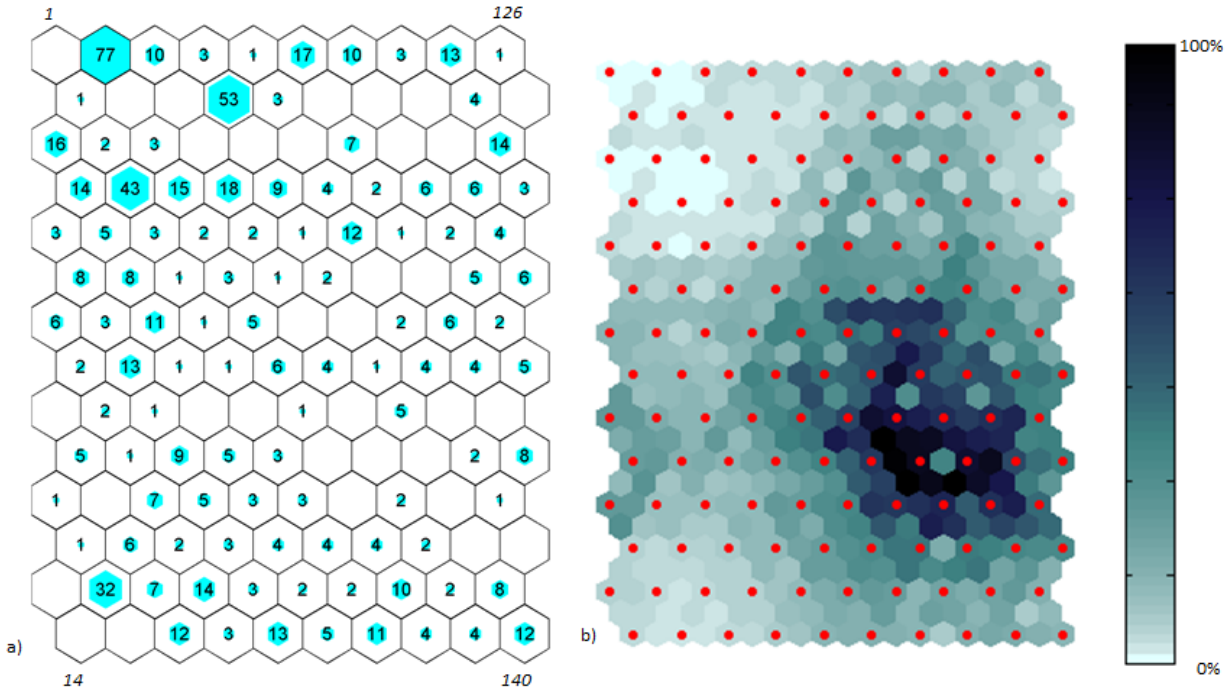


Figure 4-11 The SOM trained with PCA coordinates and parameters derived from Wendel and Buttenfield (2010). (a) Neurons labeled with frequency of best-matching GIS procedures. The size of the blue patch also indicates match frequency. Corner neuron identification numbers posted in italics. Neurons are numbered sequentially from top to bottom, left to right. (b) U-matrix for the SOM, showing dissimilarity as a darker color. Color indicates percent of range in U-value within SOM. Actual values range from 0.12 to 13.30. Regions of lighter colors indicate clusters of similar neurons and darker values indicate separation between clusters. The red dots indicate locations of neurons.

The U-matrix (Figure 4-11b) has the lowest maximum of any of the explicit keyword SOMs, at 13.30. There is a minor increase in the amount of separation between groups of neurons relative to the default SOM. Two patches of cells with relatively low U-matrix values, at the upper left and lower left corners of the SOM, in contrast with the previous SOMs which are dominated by a single large patch with low U-matrix values. It is not clear whether these patches indicate “clusters” or just that the procedure matrix was not informative enough to allow the SOM to separate groups of GIS procedures. Another possibility is that these patches are part of the same cluster in the actual three-dimensional

(toroidal) form of the SOM, but that it is cleaved as a result of the transformation process used to visualize the data structure on the two-dimensional display used here. There is a notable region of high dissimilarity at the right-central area within the SOM.

The optimal K-means and Ward's Linkage clustering of the PCA-driven SOM is shown in Figure 4-12. Names are given to the K-means clusters by the author (in the table shown in Figure 4-12) and are used in the subsequent interpretations. The optimal K-means cluster map and the optimal Ward's Linkage map are extremely similar. Both yielded the same number of clusters and the populations of these clusters are identical except for approximately nine neurons. Interpretations of individual clusters are discussed below. Figure 4-13 shows the K-means cluster boundaries superimposed on the U-matrix and Figure 4-14 shows the same boundaries superimposed on the hit histogram.

a)

Cluster Number	Name	Color
1	Mixed	
2	Interactive Editing	
3	Linear Analysis-Edit	
4	Linear Analysis-Execute	
5	Cursor	
6	Raster Processing	
7	Graphics	
8	Uncertain	
9	Calculation	

b)

Figure 4-12 Optimized clustering of SOM neurons trained with PCA coordinates and parameters derived from Wendel and Buttenfield (2010). (a) K-means created 9 clusters. Green neurons are cluster centroids. Clusters are named in the table at left. (b) Ward's Linkage created 9 clusters. The number of clusters was optimized using the Davies-Bouldin Index in both sub-figures. Black neurons have no GIS procedures associated with them.

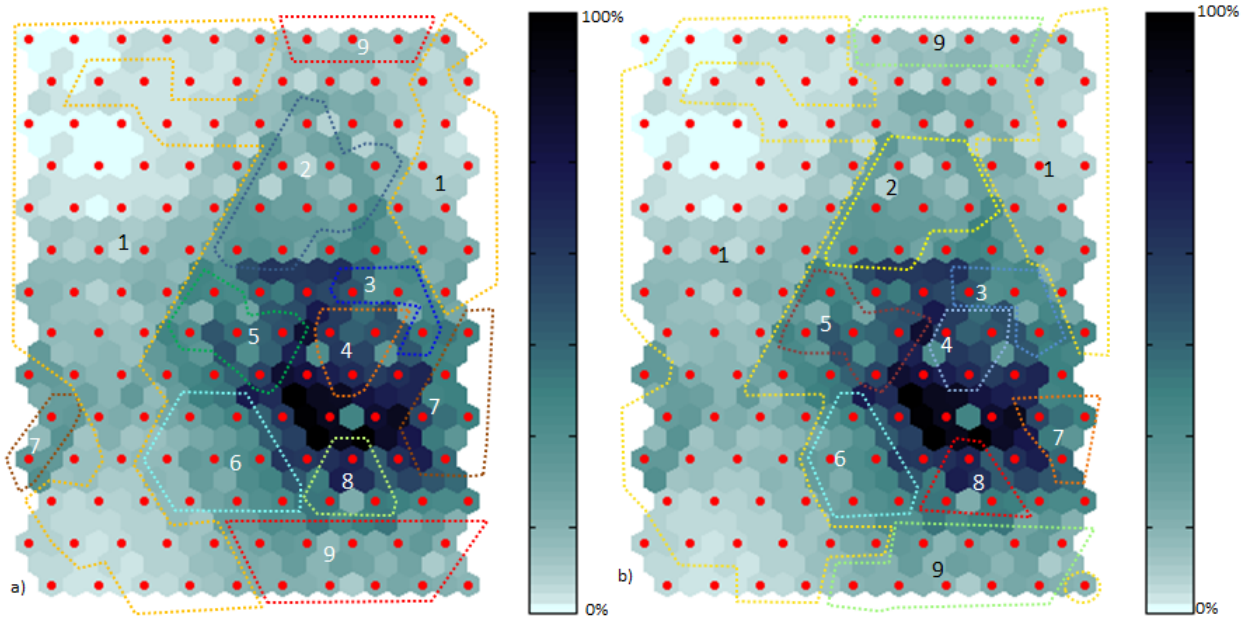


Figure 4-13 Boundaries of optimized clusters superimposed on the U-matrix. Clusters derived from SOM neurons trained with PCA coordinates and parameters derived from Wendel and Buttenfield (2010) using (a) K-means and (b) Ward's Linkage clustering. The U-matrix was derived from the same SOM. Boundary colors correspond to those in Figure 4-12.

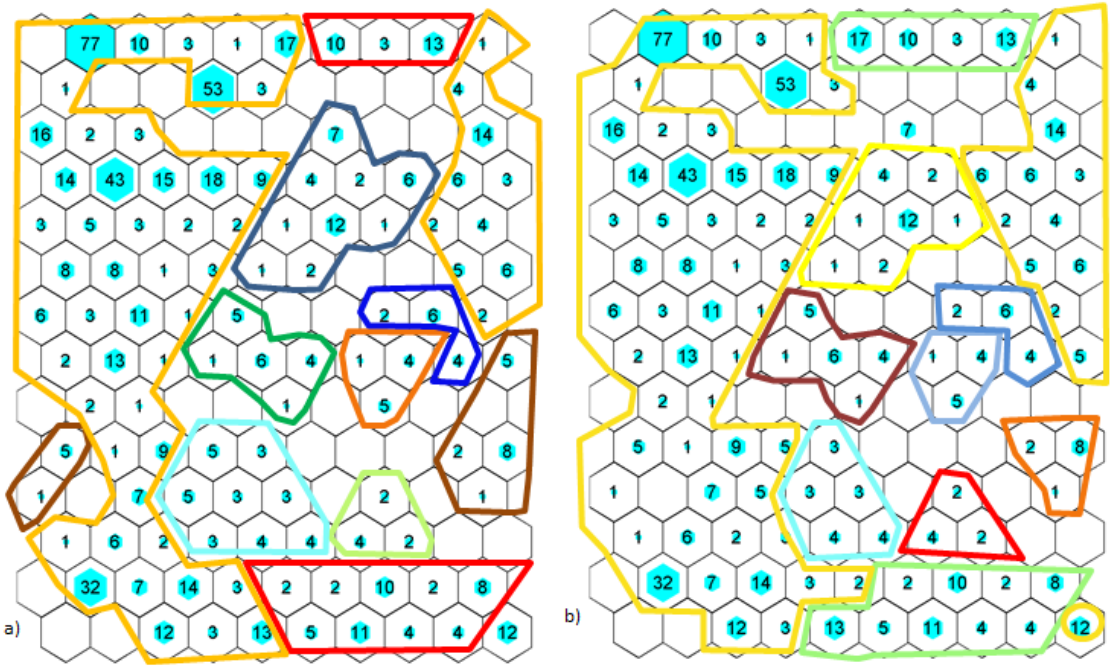


Figure 4-14 Boundaries of optimized clusters superimposed on the hit histograms. Clusters derived from SOM neurons trained with PCA coordinates and parameters derived from Wendel and Buttenfield (2010) using (a) K-means and (b) Ward's Linkage clustering. The hit histogram was derived from the same SOM. Boundary colors correspond to those in Figure 4-12.

In general the K-means clusters of the PCA-driven equate directly with those of the default SOM. Consistent cluster names have been used here to facilitate comparison. In some instances, the PCA-driven SOM clusters are associated with two of the default SOM clusters. This resulted in the derivation of two fewer clusters for the PCA-driven SOMs. These instances are explicitly noted in the following descriptions. The K-means clusters are associated with the principal components in Table 4-4 in the interpretations.

In the K-means results, the Mixed cluster (numbered 1) is by far the largest cluster. This cluster occupies 41% of all neurons and 54% of the neurons with non-zero hit frequencies, an increase of approximately 8% from the default SOM. The Ward's Linkage results has a nearly identical cluster (also numbered 1), although it encompasses a slightly smaller number of neurons. The Mixed cluster encompasses GIS procedures with a broad range of purposes. Three major sub-regions exist within its boundaries. At the top and bottom edges of the SOM, the neurons associate with GIS procedures that use a variety of raster-processing types of commands. There is overlap with clusters 9 and particularly 6 in this regard. Towards the center left of the SOM, graphics-related GIS commands are dominant. Above this patch, towards cluster 2 (named "Interactive Editing," described below), are a number of neurons that deal with interactive editing and spatial analysis that seem like they would be better included in the Interactive Editing cluster. Several of the neurons in this area associate with a mixture of GIS procedures, potentially indicating that specific GIS commands do not dominate in this area. This is also true of the portion of the Mixed cluster to the right of the Interactive Editing cluster.

Despite being the largest and including GIS procedures for a wide range of purposes, the Mixed cluster shows the lowest average internal dissimilarity of any of the clusters (much like the Mixed cluster in the default SOM. The upper left of the cluster has a patch of low dissimilarity values and features a number of neurons that have hit frequencies exceeding 10 (including the neurons with the top three hit frequencies of 43, 53, and 77). This cluster is split by the transformation of the SOM toroid to the two-

dimensional visualization and appears at both the left and right edges of the SOM. The cluster narrows to the right of the cluster 7 (named “Graphics,” described below), and experiences some of the highest U-matrix values, which could indicate that neurons on opposing sides of the area are of relatively different types. There are few hits in this region which, as discussed in the interpretation of the optimized SOM, allows nearby neurons to maintain substantially different signatures. The general shape of this cluster wraps around a region of the highest uncertainty values at the center-right of the SOM.

Cluster 2, named “Interactive Editing,” is associated mostly with GIS procedures for interactively modifying vector features in the Workstation ArcInfo module called ArcEdit. There is some heterogeneity of secondary types of functionality in the cluster, with more database management associated with the neurons at the lower edge of the cluster, mathematical operations at the top, and selection activities to the right. Dissimilarity values decrease in a fairly smooth trend with distance away from the central region. The Interactive Editing cluster shows this, with higher U-matrix values at its lower edge. Although the Interactive Editing cluster does not show particularly good internal stability, the logic of where its boundaries are defined corresponds with elevated dissimilarity values in the U-matrix. The mixture of GIS commands found within the GIS procedures of the Interactive Editing cluster indicates the likelihood of influence from multiple principal components in defining this cluster. Because the 4th component features editing type commands and the 2nd features selection, these are the most likely candidates. The 1st principal component’s heavy weighting on ‘CALC’ also seems to be a good match for this cluster.

Five of these clusters (numbered 4-8, which are named “Linear Analysis-Execute,” “Cursor,” Raster Processing,” “Graphics,” and “Selection,” respectively) encircle the region of highest dissimilarity, each with an edge towards the peak in the dissimilarity region. This is the first example of high-dissimilarity values constituting a barrier or boundary between one or more clusters seen in the SOM

clustering results. Most of these clusters show a relatively high degree of internal dissimilarity, although this decreases as the distance from the peak of dissimilarity increases.

Cluster 3, named "Linear Analysis-Edit," which is further from the region of high uncertainty, is somewhat mixed. Most GIS procedures use selection type functionality and, to a lesser degree, mathematical operators. The most common purpose of the procedures in this cluster is for linear types of analyses such as location-allocation and dynamic segmentation. One of the neurons of this cluster that is adjacent to a database management influenced neuron in the Interactive Editing cluster also shows this influence. On the whole, this cluster is relatively similar to cluster 4, named "Linear Analysis-Execute." The Linear Analysis-Edit cluster is wrapped around the Linear Analysis-Execute cluster, so the similarity in the types of functionality in these two clusters is supported by their relative positions in the SOM. The Linear Analysis-Execute cluster is somewhat mixed and associates with GIS procedures with selection type commands, but also graphics related commands. The mixture of functionality in this cluster is consistent with the elevated U-matrix values within it. Both of these clusters are consistent with the 2nd and, to a lesser degree, the 3rd principal components. The Linear Analysis-Execute cluster includes GIS procedures that also invoke data base management system types of commands, the 5th principal component could be weakly associated (this component is itself weakly associated with database management through the 'ASEXECUTE' command).

Cluster 5, named "Cursor," a little larger than the previous two clusters and includes more neurons that are farther from the peak of dissimilarity. The cluster is tightly associated with the 'CURSOR' GIS command. Based on the associated GIS procedures, this cluster is the amalgamation of the Cursor and the Cursor-Graphics cluster in the default SOM. The Cursor cluster matches the 3rd principal component, which loaded 89% onto the 'CURSOR' command

Cluster 6, named "Raster Processing," encompasses GIS procedures that favor a number of raster processing functionality for deriving new zone features or surfaces, as well as data management

functionality. Towards the region of high uncertainty, some other types of commands become prevalent. These include mathematical operators and graphics generation commands, tracking with the 4th principal component (which loads most heavily on data management and raster creation types of GIS commands). This cluster is an amalgamation of the Raster Create and Raster Kill clusters in the default SOM.

Cluster 7, named "Graphics," straddles the vertical edges of the SOM, features GIS procedures that use graphics-generation types of GIS commands. Some neurons favor selection type GIS commands, as well. These GIS commands are consistent with those that load heavily in the 2nd principal component of the original explicit keyword matrix. .

Cluster 8, named "Uncertain," is like Linear Analysis-Execute cluster because it is well within the region of high dissimilarity and exhibits low internal stability. The command that comes closest to dominating is 'CALC'. As with the default SOM cluster of the same name, the unifying purpose or type of the GIS procedures associated with this cluster was not apparent to the author. The Uncertain cluster could be associated with the 1st principal component because of its usage of the 'CALC' command. The mix of other GIS commands in this cluster and higher U-matrix values could indicate the influence of one or more additional components, such as the 4th principal component.

Cluster 9, named "Calculation," again recreates a cluster seen in the default SOM. The GIS procedures heavily associates with tabular calculation of new information and some sort of selection process, usually based on attribute information as opposed to a spatial query (such as "what is near to this point?"). Again, there is a mix of raster processing oriented GIS procedures. As with the default SOM, this cluster appears to have the second lowest average dissimilarity (or second highest stability). This new cluster is also proximal to the Mixed cluster and shows relatively low U-matrix values at the shared boundaries. The Calculation cluster, which relies heavily on the 'ARC' command and various forms of selection, tracks with the most important GIS commands in the 5th principal component.

The PCA projection of PCA-driven SOM is shown in Figure 4-15. Although this PCA analysis of PCA coordinates is in fact redundant, it is presented here to create a consistent set of displays across all experiments to aid comparison. The first two principal components from the post-SOM training PCA were found to increase the percentage of variance explained in the input data set by approximately ten percent to 52%. There does appear to be a substantially greater degree of spread of the data points when they are expressed using PCA coordinates (as opposed to the explicit keyword coordinates used in the default and optimized SOMs); the data are still bunched in the corner or along the two axes of the display. The fit of the neurons to the outlying data points is relatively poor, which is a function of the number of neurons in the PCA-driven SOM (140), set as a function of the number of PCA dimensions used, 15), relative to the optimized SOM. The separation of the colors (corresponding to the breakouts of the K-means clusters in Figure 4-12a) is poor. By interactively zooming on the figure within MATLAB, a minor degree of separation was more obvious, but a high degree of overlap between data points associated with different clusters was always found.

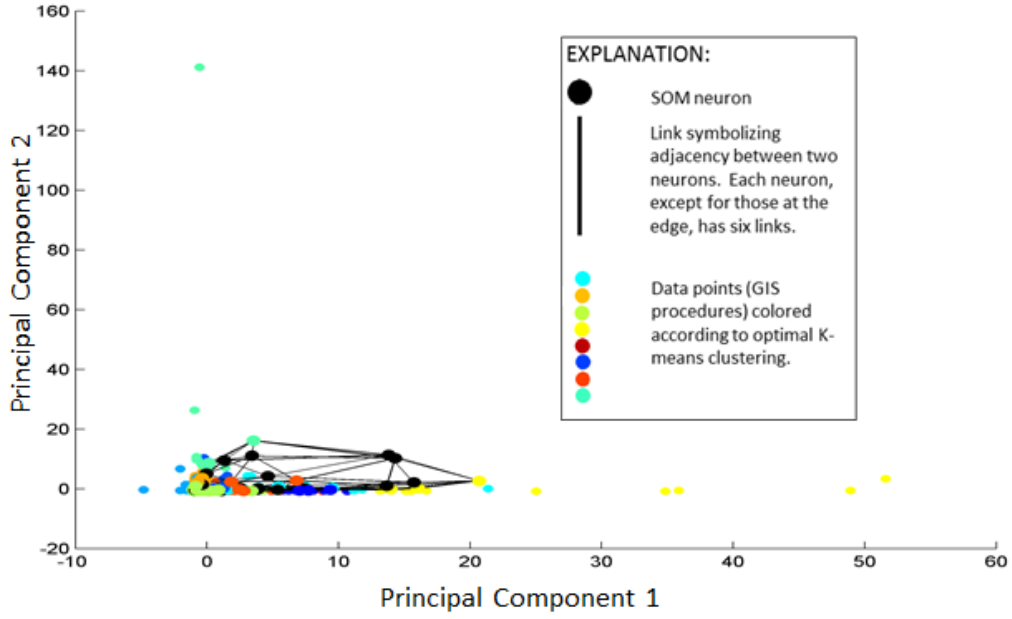


Figure 4-15 Data points and SOM neurons plotted using coordinates of the first two PCA components for the SOM trained with PCA coordinates and parameters derived from Wendel and Buttenfield (2010). This explained 52.06 percent of the variance in the data points.

The quality of the PCA-driven SOM is much better than the optimized SOM and marginally better than the default SOM. The maximum U-matrix value was 13.3, better than the default SOM, and the mean and median were 2.97 and 2.0, respectively, both of which are slightly higher than for the default SOM (but much less than for the optimized SOM). The standard deviation is 2.74, which improves on the default SOM (2.85) slightly. The spatial pattern of U-matrix values is noticeably better in the PCA-driven SOM relative to the two previous efforts. The U-matrix for the default SOM did not exhibit much structure, other than a smooth trend of deteriorating similarity from the upper right corner towards the bottom of the SOM. Extremely dissimilar neurons were bunched at the far (lower) edge of the SOM, as if the training algorithm was unable to deal with particular GIS procedures. The PCA-driven SOM shows a high uncertainty feature towards the SOM center and forms a boundary or buffer region around which the clustering methods defined a number of clusters. The automated cluster algorithms successfully detected and used a number of less obvious boundaries to define clusters. None of the previous SOMs have done this.

Although the PCA-driven SOM shows the best result of the explicit keyword experiments, it still suffers some of the same problems as its predecessors. A very large, single cluster is consistently found by the automated clustering algorithms. The cluster has a number of hot spot neurons with high hit frequencies, but some of these hot spots appear very far away from others. To a degree, this indicates that the SOM spatial structure did not have enough information to allow the automated clustering algorithms to differentiate between these hotspots. More distant points should be more dissimilar, according to Tobler's Law. The PCA-driven SOM, despite substantial analysis and modification of the procedure matrix, is not able to extract enough information to reflect this property. In general, the dissimilarity within clusters (used as a stability indicator) of this SOM is still high outside of the very large cluster.

4.7 Summary

This chapter described the details of the explicit keywords experiments, including the development of the procedure matrix from the set and the training of three different SOMs. One was developed using the default training parameters derived by the tools distributed within the SOM Toolbox. As evidenced from the SOM and both conventional methods of clustering, groups are formed largely on the basis of frequency with which a keyword is used, rather than semantic similarity. A second SOM was developed using recommendations of Wendel and Buttenfield (2010) to optimize the training parameters. Results of this experiment indicate so much fragmentation as to obstruct cluster interpretation. A third SOM was developed by deriving a new set of keywords from a principal component analysis that were used to establish a new procedure matrix. The results of this experiment showed that even when a dimension-reduction technique was used to simplify the expression of the procedure matrix, only marginal improvements were achieved.

This set of SOMs is referred to as the explicit keyword experiment because it demonstrates the limited effectiveness of using the SOM technique to differentiate GIS procedures using only the GIS commands as explicit keywords. The visualizations of each of the SOMs were presented and discussed, as were error figures (described in Table 4-5). The sections below give additional details comparing the three explicit keyword experiments.

4.7.1 Overview of SOM Analyses

In addition to these SOM-wide metrics of quantization and topographic error, a U-matrix was derived and visualized for each SOM. These were informative because they helped communicate the similarity or dissimilarity between individual neurons. Perhaps more importantly, these displays present quantitative information that helped the reader recognize larger groups of similar neurons and the arrangement of these groupings across a SOM. From this information, the reader is able to make

judgments about where boundaries around a grouping might exist. From these boundaries, a cluster might be designated which could then allow evaluation of the "type" of GIS procedures associated with it.

Each of the SOMs was evaluated using two different automated clustering techniques, K-means clustering and Ward's Linkage clustering. These techniques both use the same metric to define an "optimal" number of clusters, the Davies-Bouldin Index. While the results of these clustering techniques were not expected to be truly "correct" or semantically meaningful, they did provide valuable information about the arrangement of neurons across a given SOM. The cluster boundaries were superimposed on displays of the U-matrices and hit histograms in order to gain an understanding of the homogeneity across neurons within a cluster and how crisply clusters are separated. These characteristics are used to assess the quality of the SOM.

Principal Components Analysis (PCA) provided estimates of power of the dimensions (i.e., the GIS commands) to explain the spread of the input data points (i.e., the GIS procedures). The first two principal components of the explicit keyword procedure matrix (which refer to transformation through direct calculation and to selection) explained 42.23 percent of the variation in the GIS procedures as originally expressed by the GIS commands. PCA was also used to analyze each of the three SOMs by visualizing the projection of input data points and the neurons of each SOM into PCA-space. These displays were informative because, like the U-matrices, they presented the separation between neurons. Rather than visualizing neurons as a regularly spaced set of cells and visualizing separation by a color code, the actual plotting coordinates of the neurons derived from PCA provided this separation information. In addition, the points were colored according to the automated clusters from the K-means analysis. The two-component plots for the three explicit keyword SOMs indicate very tight clustering of GIS procedures when characterized by the set of explicit keywords. It's difficult to interpret regions in the two-component plots in a semantically meaningful manner. Although not presented here, the

author interactively zoomed in on various regions of the PCA projections to look for separation of data points associated with different clusters. Regardless of magnification, no clean separation of sets of data points (i.e., GIS procedures) was found.

The lack of the SOM separation of the GIS procedures might be based on the fact that the magnitude of the command frequencies, regardless of the command, was limited in range from 0 to less than 10. This would have the effect of the GIS procedures being heavily concentrated around the origin even in the high-dimensional information space. The lack of separation could also be caused by many GIS procedures containing the same sets of GIS commands, although this was not in evidence within the set of GIS procedures.

4.7.2 Review of SOM Metrics

Table 4-5 recaps the error figures for all three SOMs. There are important differences between the original procedure matrix used for the default and optimized SOM trainings, and the PCA-derived procedure matrix (that used principal components in lieu of the explicit keywords). Although the quantization error (average separation between signature of a GIS procedure and its best matching neuron) did not improve from the default to the optimized SOM, the topographic error (proportion of GIS procedures whose second BMU is *not* adjacent to first BMU) is cut in half. The PCA-driven SOM, which replaced the explicit keywords dimensions with PCA coordinates, greatly reduced the quantization error to 1/3 of the magnitude for the previous two SOMs, but increased the topographic error to more than for the default matrix. This somewhat surprising result implies that creating a new set of dimensions which are composites of the original characterizing variables (i.e. 15 principal components in place of the 148 GIS commands) made it easier for the BMU to mimic the signature of the GIS procedures. At the same time, it implies that the overall structure of the SOM deteriorated.

Table 4-5 Errors and U-matrix statistics of the three explicit keyword SOMs.

	Error		U-Matrix Statistics			
SOM	Quantization	Topographic	Max	Mean	Median	Stan Dev.
Default	4.285	0.047	18.1341	2.6233	1.7444	2.8526
Optimized	4.363	0.024	85.6366	3.262	1.8202	5.3543
PCA-driven	1.413	0.06	13.2991	2.969	1.995	2.7418

In comparing the topographic errors, the default SOM shows some degree of coherence, meaning that high frequencies of GIS procedures are clustered together. The topographic error (0.047) reflects this coherence, at least at a local scale. However, the quantization error of the default SOM is relatively high, indicating that the fit of the data points to the best-matching neurons in the SOM was not precise. This might be interpreted to be caused by a matrix that undersized and resulted higher numbers of GIS procedures competing to adjust the signature of a best-matching neuron, resulting in a compromise solution that did not necessarily fit any of the procedures well. Oddly, the quantization error for the optimized SOM is of a similar value, counter to expectations.

The U-matrix for the optimized SOM shows a large increase in the maximum value, which increases the mean and standard deviation. The median shows little difference for any of the explicit keyword SOMs. This indicates that although a few extremely high values do occur, the overall distribution did not shift substantially. This SOM showed a high degree of fragmentation, evidenced not only by the ringed patterns in the U-matrix, but also by the failure of both cluster analyses. The large number of neurons in this SOM served to isolate individual commands rather than cluster or regionalize the solution space. It is possible that such a solution could form the initial basis for a hierarchical clustering, but that direction lies beyond the scope of the present dissertation. It is interesting to note that despite the relatively vast number of neurons in the optimized SOM and the low average frequency of BMU hits, the quantization error did not decrease. One would expect that as a neuron's signature was trained by fewer GIS procedures, it would more closely match the signature of dimensions of those procedures. The topographic error of the optimized SOM was reduced by about half of the magnitude

of the topographic error for the default SOM, which implies that because the optimized SOM matrix is so large and individual data points so sparsely spread in general, there was relatively little competition to influence neighboring neurons (i.e., that each data point not only had its own neuron, but its own neighborhood).

When the explicit keyword procedure matrix and the neurons of both of the default and optimized SOMs were projected into PCA space, neither showed very good separation of the clusters established by the K-means analyses. In fact, there was relatively little separation of the GIS procedures regardless of cluster designations. This tends to indicate that both the SOM and the PCA techniques had difficulty either with dealing with so many dimensions (i.e. the 148 GIS commands/explicit keywords), that the values assigned to these dimensions failed to indicate substantial differences between the GIS procedures, or that the dimensions themselves (i.e., the GIS commands) were insufficient to fully distinguish among the GIS procedures. The latter two are more likely given that both these techniques are generally successful with high-dimensional data.

The PCA-driven SOM improved upon the previous two solutions in several ways. By reducing the number of dimensions to the first 15 principal components, the resultant SOM matrix was much smaller. This resulted in a less sparse hit histogram and more concentrated clusters of neurons than seen with the optimized SOM. The quantization error was reduced to approximately 1/3 of the magnitude seen with the default and optimized SOMs. Interestingly, the topographic error increased substantially, implying that the overall organization of the SOM was poorer than for either of the previous two SOMs. While the topographic error figure is useful as a general synoptic statistic, this result tends to indicate more detailed examination of the overall SOM structure is warranted. The automated cluster analyses are helpful in this regard.

In the interpretation of cluster meanings for the PCA-driven SOM, one-to-one correspondence between the PCA-driven and default clusters was very clear. In two cases, a PCA-driven SOM cluster

represented an amalgamation of the types of GIS procedures found in two cluster in the default SOM. Specifically, the PCA-driven Raster Processing cluster was associated with almost every GIS procedure found in the Raster Create and Raster Kill clusters found in the default SOM; the same was true for the PCA-driven Cursor cluster, which corresponded with the default SOM Cursor and Cursor-Graphics clusters. Although a minority of GIS procedures ended up in clusters of differing type as a result of the PCA pre-processing, the results indicate that there was no substantial change to the organization of the SOMs.

When the coordinates of the first two principal components of the procedure matrix used in the PCA-driven SOM (populated with the first fifteen principal components of the explicit keyword matrix) were examined, 52.07 percent of the variation was explained. This was an improvement of approximately ten percent over the original explicit keyword procedure matrix used in the default and optimized SOMs. In addition, the plotting of both the PCA coordinates of the data and the neurons in the PCA-driven SOM revealed an improvement in the overall separation of information. While the clusters designated by the K-means clusters did seem to show a slight degree of separation in this display, overall quality of separation of clusters was still poor.

One might conclude from these three experiments that the training of a SOM based on explicit keywords was improved to the maximum degree possible. Two major enhancements were applied, the optimization of training parameters and the dimension-reduction of the input data, resulting in minimal/conflicting improvements in the error figures and the PCA projections. Overall quality of the clustering by all the explicit keyword SOMs could be improved. The PCA projections produced real, but limited, improvement by deriving an alternative to the explicit keyword matrix using information that tried to encapsulate the most important information within the original matrix. This demonstrated that improvement in the quality of a SOM by modification of the explicit keyword matrix is possible. If one accepts the arguments discussed in Chapter 1 regarding implicit keywords, another way to improve

upon the explicit keyword procedure matrix is to augment it with semantically meaningful information. It seems likely that the heuristic selection of implicit keywords can be designed to be more sensitive to the perspective of a particular user or community, the content of the set of GIS procedures, or both, than even automated dimension-reduction techniques represented by PCA.

For these reasons the next chapter will present two SOM experiments using implicit keywords. The following chapter will describe the derivation of sets of implicit keywords, their use in deriving new forms of the procedure matrix, and the SOMs produced using the new versions of the procedure matrix. The visualizations and error analyses will be consistent with those presented here and will rely on the methodological explanations presented in this chapter.

5 The Implicit Keyword Experiments

This chapter describes the construction of implicit keywords that are associated with GIS commands and used to augment the explicit keyword procedure matrix. To reiterate, the purpose is to improve the description of the set of GIS procedures and to improve the resultant SOMs developed to organize that set. Two sets of implicit keywords are developed. The first is developed based on Albrecht's (1999) Ph.D. dissertation. This set of implicit keywords is intended to represent generally held views, as established by Albrecht, about types of GIS functionality. The second set is constructed to reflect a specialized domain of application, namely environmental modeling. This set of keywords is based on the expertise of the author and the data set described by Wendel and others (2008a; 2008b). Each set of implicit keywords is used to train a SOM. The results are presented and compared, as are error statistics, in a manner consistent with those shown for the explicit keyword experiments in the previous chapter. Following this exploration of implicit keywords, Chapter 6 will compare all the explicit and implicit approaches studied in this research.

5.1 Albrecht's Typology of Universal GIS Commands

Albrecht's (1999) typology of GIS commands was based on his surveys of new GIS students and conversations with experts at professional meetings over several years. He found that because so many users conceptualized their tasks based on data and data models, up to 80 percent of all GIS commands were not part of what he considered an "analytic task" (p. 32, Albrecht, 1999). Albrecht's typology emphasizes differentiation of analytical GIS commands and gives the types shown in Table 5-1. He also identifies several other types of GIS commands, but does not use them in the later part of his dissertation. This is not because these types of commands are irrelevant to other users or researchers, but because that author's stated goal was to design a GIS that included only what he identified as "universal" commands that are all capable of carrying out spatial analysis of a "virtual geodata model"

(p. 32, Albrecht, 1999). Examples of what he considered data-centered types of GIS commands, referred to here as “non-analytical,” are listed in Table 5-2.

Table 5-1 Albrecht's (p. 62,1999) analytical types of universal GIS commands.

Analytical types of commands	Examples of type
Search	attribute search, spatial search, (re-)classification
Location Analysis	corridors, buffer, Thiessen polygons, overlay
Terrain Analysis	slope, aspect, flow direction, stream extraction
Distribution and Neighborhood Analysis	cost/diffusion/spread, proximity, nearest neighbor
Spatial Analysis	multivariate analysis, pattern, dispersion, centrality, connectedness and other topological measures
Measurements	frequency, distance, direction, statistics (mean, min, max, etc), shape, similarity, adjacency, contiguity

Albrecht provides a relatively terse definition of his types of analytical commands, relying on the specification of exemplars of each type. In some cases, he supplies a rationale for the assertion that a command is of a given type. Note that his examples of GIS commands, specified in Table 5-1, are generic. In some cases, there may not be a corresponding ArcInfo command. The following paragraphs attempt to capture his definitions and are based on interpretation of his written text and the data he presented. Instances of overlap among types will be identified; these overlaps indicate that devising any scheme to make crisp classifications is at best extremely difficult.

Search: This type of command includes searches based on attributes or on spatial features. The search operation usually applies some kind of selection criteria and possibly an analysis of the results. Reclassification is an example of an analysis preceded by an attribute-based selection process. Albrecht (p.62, 1999) states, albeit without support from published literature, that “the whole concept of Map

Algebra can be regarded as a form of reclassification". Therefore Map Algebra functions are associated here with this type in the implicit keyword experiment. Commands that set the GIS environment parameters or tolerances for editing or selection are also associated with this type.

Locational Analysis: This type of command includes the process of location-allocation, which is defined as "the process of finding the best locations for one or more facilities that will service a set of points and then assigning those points to the facilities, taking into account factors such as … impedance from a facility to a point" (p. 125, Wade and Sommer, 2006). Albrecht's example of the type (Table 5-1), "corridors," is an example of what are commonly referred to as cost-surface commands, which rely on the concept of impedance to carry out the analysis. The buffer command is a special type of cost-surface function that uses Euclidean distance from a feature (facility) as an indicator of impedance. This also implies a threshold is applied—a reclassification of the impedance surface. Reclassification is covered in the Search command type. Delineation of Thiessen polygons around a set of points is much the same as buffering in that it determines the distance by Euclidean or other metric to the members of the set from all positions and allocates each position belonging to the nearest point.

Examples Albrecht (1999) gives of overlay commands include "clip," "split," "identity," "union," and "intersect." Overlay commands can be thought of as allocating not just entire features but also possibly fractions of features. For example, the "clip" command could be used to apply a geographic "cookie-cutter" in the shape of a watershed to a set of polygons representing geology. The result would be subset of the geology polygons. Where the original geology polygons straddled the watershed boundary, the geology polygons were clipped to the watershed boundary. This analysis determines which positions in the geology map belong to the watershed, a form of allocation. A reclassification has been applied because those that do not belong are discarded.

The interpretation Albrecht gives to the "Location Analysis" type of GIS command assumes a relatively high degree of semantic overlap with the "Search" type of GIS command because commands

that set GIS environment parameters to control things like search area are considered to directly impact the analysis of position and relative position.

Terrain Analysis: This group of commands deals with analysis of position in three dimensions (i.e. XY and elevation). The input data could be raster (a Digital Elevation Model), vector contours, or a Triangulated Irregular Network (TIN). Height, as well as relative position, is usually integral to this type of command. Even if one of the input data sets does not explicitly include height, one may imply it (e.g. a flow direction gives the steepest downslope direction). Some of this type of command could be interpreted as also being associated with the "Spatial Analysis" type of commands that deal with "connectedness." This type also refers to viewshed analysis.

Distribution/Neighborhood: Albrecht describes this type as the "most geographic of all" (p. 64, Albrecht, 1999). These commands make spatial queries about the relationship between geographic features. His "cost/diffusion/spread" commands assign attributes to neighboring features based on a relative distance. Because the spread of a characteristic is sensitive to barriers or cost, these commands can also be considered as belonging to the Locational Analysis type. The definition provided for this type is relatively weak. This type is also interpreted to include overlay functions, as these deal with the spatial relationship between features.

Spatial Analysis: Albrecht refers to some of these commands as "secondary," arguing that many are composites of other types of commands, some of which are not necessarily based in a GIS at all. For example, his "multivariate statistics" could refer to functionality that can be found in an external statistics package or could be a composition of commands from the "Measurement" type. He includes several other commands that seem to encompass at least some fundamental kinds of GIS analyses, such as analysis of connectedness, dispersion or separation, and analysis of shape. These three examples seem to have characteristics that are strongly shared with other types.

Measurement: This type of command includes mathematical and statistical operations. Simple geometric calculations, such as distance, direction, area, are included. Albrecht (1999) also includes some topological measures, listing adjacency, and what he calls "doughnuts/holes." The inclusion of topology in this type associates it with several other groups. These include the "Distribution/Neighborhood" type, which deals with proximity, the "Spatial Analysis" type, which includes connectedness and shape types of commands, as possibly the "Locational Analysis" type, which deals with allocation.

Albrecht points out that several commands could easily be extracted from his types and assembled into a new one. More specifically, he notes that commands that deal with topology could be grouped into a type called "Network/Flow" (p. 63). He suggests that commands relating to connectivity, nearest neighbor, shortest path, Thiessen polygons, and flow between regions could populate such a group. This thought, along with his frank discussion of overlaps between his type groupings, indicate that even with his sizable empirical data sets, the nonexclusive and subjective nature of the definition of and assignment to type groups still has a large impact on what he ultimately designates as "universal" GIS functions.

Although Albrecht essentially excludes what he refers to as "data-centered" commands from his later analyses, he does provide the types listed in Table 5-2. He does not define them explicitly, stating that their meaning is self-evident. The distinction between "Make a map" and "Display a map" is potentially unclear. GIS commands relating to symbol scaling (graduated circles, color ramps, thickness of lines), placement of text in the map body (along roads or streams, e.g.), are understood by most cartographers as map making. Map display commands are understood as those used to create the layout of a map, for example controlling the placement of a scale bar and legend, create a map title, and other graphical elements onto the map. Because of Albrecht's focus on analysis, these two types of GIS functionality are merged within this experiment into a single type and referred to as "Visualization."

Table 5-2 Albrecht's (1999) non-analytical ("data-centered") types of GIS commands.

Non-analytical types of commands
Make a map
Enter data
Select items
Display a map
Classify attribute data

This author disagrees with Albrecht's contention that "Classify attribute data" is not an analytical type of functionality within GIS. While classification could be thought of as non-spatial, it is not necessarily non-analytical. While Longley and others (p. 253, 2001) do point out that this task is often carried out in order to control visualization (i.e., "Map Making"), it entails the development of rules to abstract data content in semantically appropriate ways. In addition, Longley's assertion seems to be in direct conflict with Albrecht's statement (p. 62, 1999) that argues for the inclusion of reclassification in the Search command type because although "(re-)classification is basically a database operation…in most cases, …the filter that is used for reclassification has a spatial determinant." As a result, this type of functionality is treated as part of the Search command type and the "Classify attribute data" is eliminated from the experiment. The "Select items" type is redundant to the selection concept he uses to define the "Search" type of analytical command and is also excluded. This reduces the set of implicit keywords for non-analytical GIS commands to "Visualization" and "Enter Data."

Once implicit keywords were developed for analytical types of GIS commands (i.e., not including the non-analytical types), a vector of values was created for each GIS command, according to the procedure described for explicit keywords in Section 3.3.1 and Table 3-2. Some GIS commands were associated with none of Albrecht's analytical types, while some were associated with as many as five. Table 5-3 the variety of types (i.e. the "variety") that were associated with a GIS command and how many GIS commands showed that variety. Different combinations of the types might yield the same

variety. For instance, if GIS command “A” was considered to be an example of the command types “Search,” “Location Analysis,” and “Terrain Analysis,” it would have a variety of three. If GIS command “B” was considered to be an example of command types “Distribution and Neighborhood Analysis,” “Spatial Analysis,” and “Measurements,” then it would also have a variety of three. The existence of GIS commands “A” and “B” would be added to the tally for GIS commands having a variety of three (i.e., the fourth row in Table 5-3).

Table 5-3 The number of GIS commands showing various levels of variety in Albrecht's analytical types associations.

Variety of analytical Albrecht types	Number of GIS commands
0	91
1	26
2	15
3	3
4	11
5	2

Because 91 of 148 GIS commands found in the set were not associated with any of the analytical GIS command types, it is clear that additional implicit keywords are likely needed to build a general organization of the GIS commands in the set. Most of the GIS commands with no non-zero values for the analytical implicit keywords pertain to rendering a map or visualization. Most of the GIS commands with a single association pertain to selection or constraining the geographic extent of the analysis environment.

After adding the two new implicit keywords (“Visualization” and “Enter Data”) for non-analytical (“data-centered”) GIS commands, only a single GIS command did not associate with at least one of Albrecht’s types (Table 5-4). This fuller set of implicit keywords, based on both analytical and non-analytical types of GIS commands, was used for this experiment.

Table 5-4 The number of GIS commands associated with numbers of Albrecht's types, including non-analytical types.

Variety of all Albrecht types	Number of GIS commands
0	1
1	99
2	24
3	11
4	11
5	2

5.1.1 Validating the Transfer of Implicit Keyword Information into the Procedure Matrix

As described in 3.3.1, software was written to derive a procedure matrix augmented with the information contained in the Albrecht set of implicit keywords. This was accomplished by setting a vector of values for each GIS command (each element in the vector corresponds to an Albrecht implicit keyword) and assembling the vectors into a matrix, determining the frequency of each GIS command within each GIS procedure, relating the respective command to the matrix of implicit keyword vectors, multiplying the command frequency by the associated vector of implicit keyword values, and summing the products for each GIS procedure's GIS commands into a vector of values with as many values as there are implicit keywords. This section demonstrates how the author manually validated the performance of this software. This is important because the software is at the heart of the implementation of the main conceptual contribution of this dissertation, which is the addition of implicit understanding to the description of a GIS procedure in order to better understand the procedure and to better organize large sets of GIS procedures. A detailed description was given in Section 3.3.1 to describe how the implicit keyword information is used to augment the procedure matrix.

The example check was done using the GIS procedure, "addedit." Table 5-5 lists the first step in the evaluation process, the implicit keyword matrix for GIS commands. The rows show a subset of GIS commands found throughout the set of GIS procedures. The columns correspond to the Albrecht

implicit keywords. The cells within the table show how each GIS command was evaluated for the cross-referenced implicit keyword. Once the set of implicit keywords were defined, the cell values were assigned by the author for all GIS commands. This table is a subset of the full Albrecht implicit keyword matrix. The full Albrecht implicit keyword matrix is provided in Table A-1 in Appendix A.

Table 5-5 Subset of the Albrecht implicit keyword matrix showing evaluation of implicit keywords for selected GIS commands. ("Distrib / Nbrhd" is the implicit keyword for the distribution and neighborhood analysis type of GIS command).

GIS commands appearing in addedit	Frequencies per implicit keyword							
	Search	Location Analysis	Terrain Analysis	Distrib / Nbrhd	Spatial Analysis	Measure-ment	Visual-ization	Enter Data
'ASELECT'	1	0	0	0	0	0	0	0
'CLEARSELECT'	1	0	0	0	0	0	0	0
'CURSOR'	1	0	0	0	0	0	0	1
'MAPEXTENT'	1	0	0	0	0	0	1	0
'MARKER'	0	0	0	0	0	0	1	0
'MARKERSYMBOL'	0	0	0	0	0	0	1	0
'MOVE'	1	1	0	1	0	0	0	0
'POLYGONSHADES'	0	0	0	0	0	0	1	0
'READSELECT'	1	0	0	0	0	0	0	1
'RESELECT'	1	0	0	0	0	0	0	0
'SEARCHTOLERANCE'	1	1	0	0	1	0	0	0
'WRITESELECT'	1	0	0	0	0	0	0	1

The explicit keyword procedure matrix, which lists the GIS procedures in the rows and the GIS commands in the columns, revealed that 12 GIS commands appeared in the "addedit" procedure. (Table 4-1 showed a subset of the explicit keyword matrix and Section 4.1 explained the construction of the explicit keyword procedure matrix in detail.) Table 5-6 lists all of the GIS commands that appeared in the "addedit" procedure and how many times they appeared.

Table 5-6 Subset of procedure matrix, showing row for "addedit" and the GIS commands with non-zero frequencies.

GIS Procedure	GIS Command 'ASELECT'	'CLEARSELECT'	'CURSOR'	'MAPEXTENT'	'MARKER'	'MARKER-SYMBOL'	'MOVE'	'POLYGON-SHADES'	'READSELECT'	'RESELECT'	'SEARCH-TOLERANCE'	'WRITESELECT'
addedit	2	11	8	5	3	9	5	2	14	11	8	11

Table 5-7 shows the result of multiplying the frequency associated with each explicit keyword (Table 5-6) by the eight implicit keywords associated with it (Table 5-5). The 'MOVE' GIS command (the seventh row in Table 5-7) is used here an example to illustrate this process. The vector of Albrecht implicit keyword values for the 'MOVE' command, determined from the row labeled 'MOVE' in Table 5-5, is (1,1,0,1,0,0,0,0). According to Table 5-6, the 'MOVE' command appeared five times. The product of multiplying each member of the vector by the frequency is (5,5,0,5,0,0,0,0,), as shown in Table 5-7.

Table 5-7 Tabulation of Albrecht implicit keyword frequencies for each GIS command in a sample GIS procedure.

GIS commands appearing in addedit	Frequencies per implicit keyword							
	Search	Loc Anlys	Terrain Analysis	Distrib / Nbrhd	Spatial Analysis	Meas	Viz	Enter Data
'ASELECT'	2	0	0	0	0	0	0	0
'CLEARSELECT'	11	0	0	0	0	0	0	0
'CURSOR'	8	0	0	0	0	0	0	8
'MAPEXTENT'	5	0	0	0	0	0	5	0
'MARKER'	0	0	0	0	0	0	3	0
'MARKERSYMBOL'	0	0	0	0	0	0	9	0
'MOVE'	5	5	0	5	0	0	0	0
'POLYGONSHADES'	0	0	0	0	0	0	2	0
'READSELECT'	14	0	0	0	0	0	0	14
'RESELECT'	11	0	0	0	0	0	0	0
'SEARCHTOLERANCE'	8	8	0	0	8	0	0	0
'WRITESELECT'	11	0	0	0	0	0	0	11
Total frequencies	75	13	0	5	8	0	19	33

The bottom row in Table 5-7, labeled "Total frequencies," shows the sums of the vectors of implicit keyword values that appear in the Albrecht-derived procedure matrix. This sum was determined for each GIS procedure and was stored in the Albrecht implicit keyword-augmented procedure matrix, which was used to train the Albrecht SOM. The "Total Frequencies" value in Table 5-7 was derived by both the software and the manual calculation. A number of additional spot checks were made. Across the entire set of 738 GIS procedures in the set, there were 441 different total frequency vectors (i.e., unique combinations of Albrecht implicit keyword valuations associated with GIS procedures). The total frequency vectors for each GIS procedure were assembled into a matrix, with a single GIS procedure appearing on each row and the columns defined by the Albrecht implicit keywords. The fully populated Albrecht implicit keyword procedure matrix and is presented in full in Table A-2 within Appendix A.

5.1.2 SOM trained with Albrecht's Types of Universal GIS Commands

The procedure matrix augmented with the Albrecht implicit keywords produced the SOM shown Figure 5-1. This SOM features 400 neurons, as determined by the recommendations of Wendel and Buttenfield (2010), arranged into a matrix with 16 neurons on the x axis and 25 neurons on the y axis. The radius of the neighborhood around a neuron over which adjustments were made during the training process was set as a function of the matrix size, again as per Wendel and Buttenfield (2010), starting out as the length of the long side of the matrix and decreasing to half that size through the training iterations. The quantization error for the SOM was 2.6084 and the topographic error was 0.1111. The quantization error is less than for the default (4.285) and optimized (4.363) SOMs, but more than for the PCA-driven SOM (1.413). The topographic error for this SOM was higher than error values for any explicit keyword experiment (0.047, 0.024, and 0.060, respectively).

The hit histogram for this SOM, shown in Figure 5-1a), reveals a concentration of very high frequencies in the upper right corner of the SOM, which displays a region of low dissimilarity in the U-

matrix. There is a relatively even spread of low frequencies through the rest of the map. Many small regions, usually comprised of fewer than five neurons, are spread relatively evenly across the SOM. Table C-5 in Appendix C lists the neuron identification numbers of the Best Matching Units and the associated GIS procedures.

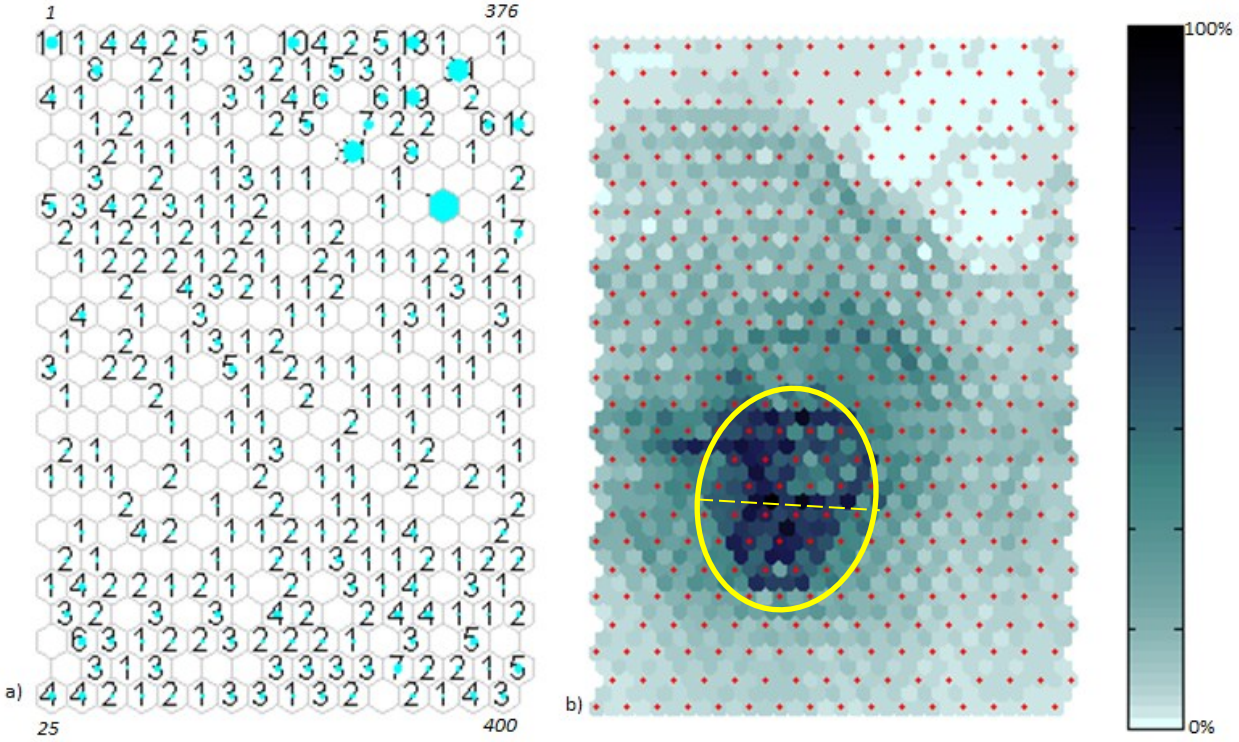


Figure 5-1 SOM trained with the procedure matrix modified with Albrecht keywords and parameters derived from Wendel and Buttenfield (2010). (a) Neurons labeled with frequency of best-matching GIS procedures. The size of the blue patch also indicates match frequency. Corner neuron identification numbers posted in italics. Neurons are numbered sequentially from top to bottom, left to right. (b) U-matrix for the SOM, showing dissimilarity as a darker color. Color indicates percent of range in U-value within SOM. Actual values range from 0.08 to 35.20. Regions of lighter colors indicate clusters of similar neurons and darker values indicate separation between clusters. The red dots indicate locations of neurons.

There are only 6 neurons with a hit frequency in excess of 10. The default SOM and the PCA-driven SOM had 16 and 19, respectively. The low number of neurons with extremely high hit frequencies in the Albrecht SOM indicates that there was enough information in the Albrecht implicit keyword augmented procedure matrix to allow the SOM training process to differentiate GIS procedures and spread them across the landscape of the SOM. By contrast, the optimized SOM had only 8 neurons with

a hit frequency that exceeded 5 because of its massive matrix size; it had more than 75% of all neurons with no hits at all, compared with 33% for the Albrecht SOM, 23% for the PCA-driven SOM, and 17% for the default SOM (notes that the PCA-driven and default SOMs used smaller matrices). The Albrecht SOM appears to have reduced the stove-pipe effect seen in the explicit keyword SOMs (particularly the optimized SOM).

The visual pattern seen in the U-matrix (Figure 5-1b) is dominated by an especially dark region in the lower left of the image (circled in yellow). While it does appear that there is some similarity among neurons within the region (more so for the upper part, above the dashed line), there does not appear to be a consistent degree of similarity within the region. If one interprets the image as an elevation surface, the rings seem to form a mountain top (like a volcano). The neurons around the edges of the SOM have a consistently low U-value and seem to form flats around the mountain region. These flat-area neurons may be similar enough to form a single region.

The optimal K-means and Ward's Linkage clusterings of the Albrecht SOM are shown in Figure 5-2. Names are given to the K-means clusters in the table in Figure 5-2a) and used in the interpretations described in the remainder of this section. Prior to examining the meaning of the individual clusters in detail, some general observations are presented. The optimal K-means clusters (Figure 5-2a) and the optimal Ward's Linkage clusters (Figure 5-2b) are similar. A significant difference is that the optimal level of clustering is much lower (6 clusters) for the Ward's Linkage solution than for the K-means (15 clusters). The Ward's Linkage cluster 1 is an agglomeration of many of the clusters designated within the K-means analysis. Clusters 6 and 7 (at the top of the SOM) were merged. At the bottom of the figure, clusters 2, 5, and some of 15 from the K-means analysis are also merged with cluster 1. Clusters 8 and 9 (from the K-means analysis) merged into a single unit. The Ward's cluster labeled 3 is an amalgam of several of the K-means clusters.

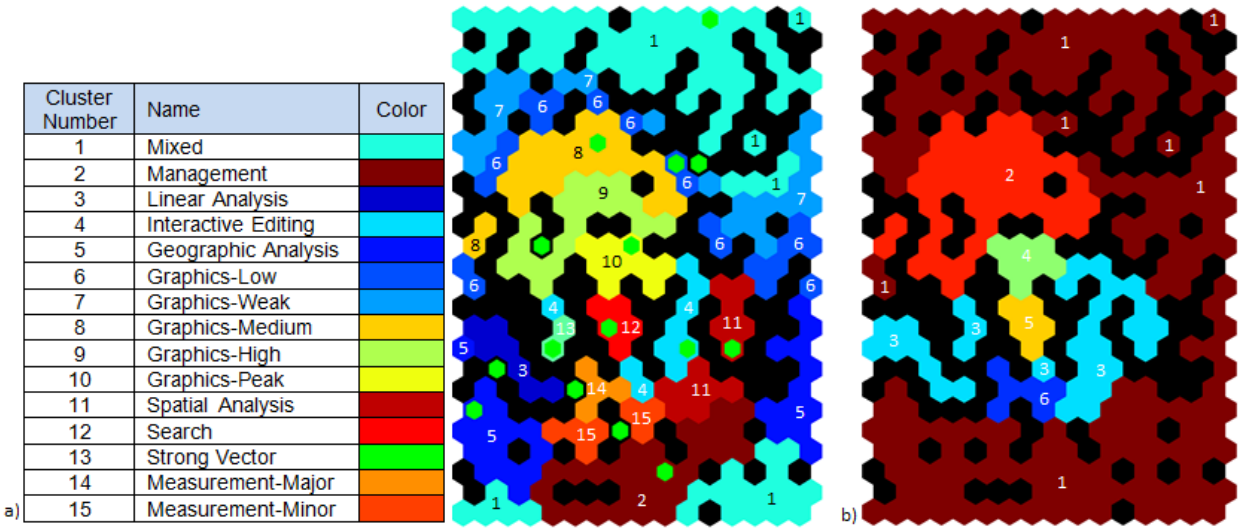


Figure 5-2 Optimal clustering of SOM trained with the procedure matrix modified with Albrecht keywords and parameters derived from Wendel and Buttenfield (2010). (a) K-means created 15 clusters. Green neurons are cluster centroids. Clusters are named in the table at left. (b) Ward's Linkage created 6 clusters. The number of clusters was optimized using the Davies-Bouldin Index in both sub-figures. Black neurons have no GIS procedures associated with them.

The overall arrangement of clusters in the K-means results (Figure 5-2a) are very different from the cluster maps from the explicit keyword experiment. As is expected, a large number of smaller clusters appear to be arranged around the volcano feature. Although fragmented, many of these clusters appear in a concentric pattern around the volcano. The PCA-driven SOM revealed clusters looked more like pie slices, with the surrounding clusters abutting or encompassing the region of high dissimilarity. It should be noted in the current result that a cluster may be comprised of multiple non-adjacent sets of neurons. For example, there are multiple groups comprising the cluster numbered 1 in the "flat" areas at the top and bottom margins of the SOM. Stepping in towards the center of the SOM, from the upper edge of the cluster numbered 7 is first to be found. Beyond that is the cluster numbered 6. Both clusters are fragmented but are arranged in a pattern that wraps around the volcano feature. After these, cluster 8 also appears to wrap around cluster 9 which, in turn, encapsulates the cluster numbered 10.

Cluster 1, named “Mixed,” is composed of neurons that are weakly weighted for all of the Albrecht implicit keywords when compared to the rest of the SOM. There is a slightly stronger influence for “Enter Data” types of GIS commands at the top edge of the SOM, towards the center. The range of GIS commands emphasized by the GIS procedures in this cluster is fairly large, but there is definitely a unifying theme of raster processing and data management. There is a shift towards a chaotic mix of GIS commands in the portion of the Mixed in the lower right corner of the SOM, with data base, relate, and editing operations appearing more commonly. The GIS procedures included in this cluster can be determined by examining C-5.

The portion of the Mixed cluster at the top edge of the SOM has the lowest dissimilarity values (Figure 5-3a). There is a smooth trend of degradation of U-matrix values away from the upper right corner. The cluster does not show major internal instability and has an even spread of hits of GIS procedures. There does appear to be some dissimilarity within the cluster that is created with a single neuron that is not attached to the main mass of the cluster (down about 7 rows from the upper edge, towards the right edge). By cross-referencing with the hit histogram (Figure 5-4), it can be determined the segregation of this island is the result of neighboring neurons with no hits. These empty neurons effectively isolate other neurons and lead to patterns in the U-matrix. The isolated neuron has the highest hit frequency (71, more than double the next nearest value) in the entire SOM. During the training process this neuron became an increasingly attractive BMU that collected more GIS procedures with continued iteration. This process appears to have drained BMU hits from the surrounding area, leaving neighboring neurons with zero hit frequencies. In addition, the process created an island of similarity around the neuron.

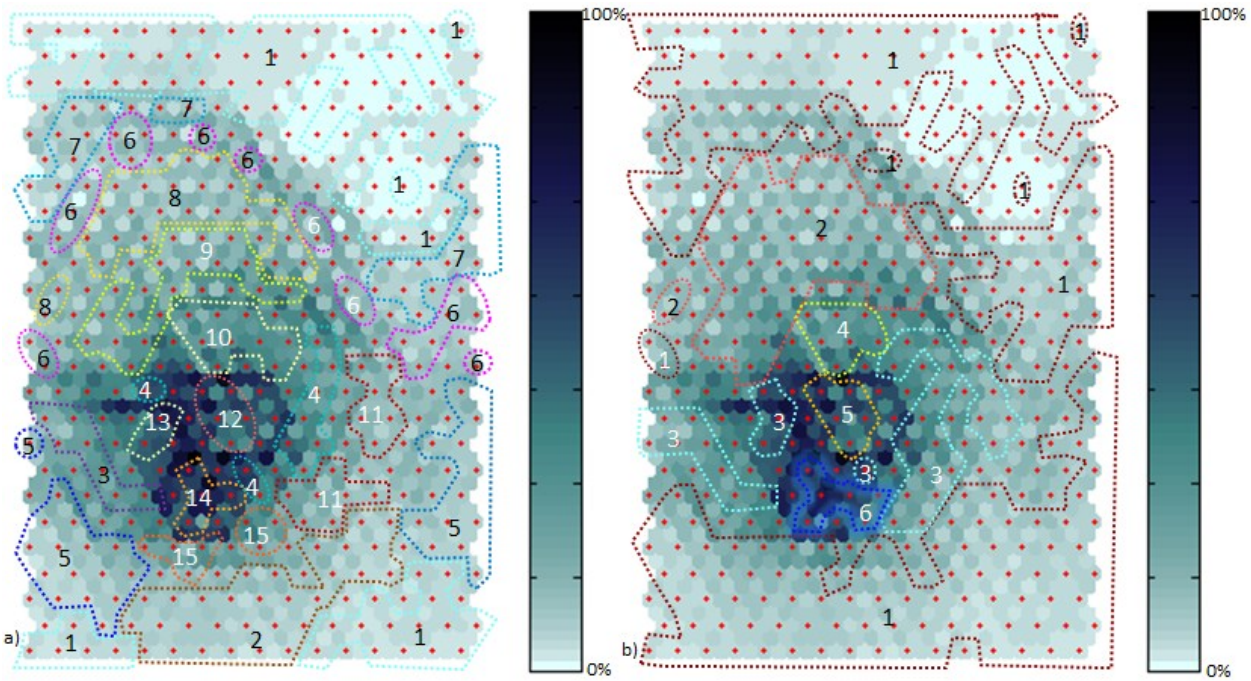


Figure 5-3 Boundaries of optimized clusters superimposed on the U-matrix. Clusters derived from SOM trained with the procedure matrix modified with Albrecht keywords and parameters derived from Wendel and Buttenfield (2010) using (a) K-means and (b) Ward's Linkage clustering. The U-matrix was derived from the same SOM. Boundary colors correspond to those in Figure 5-2.

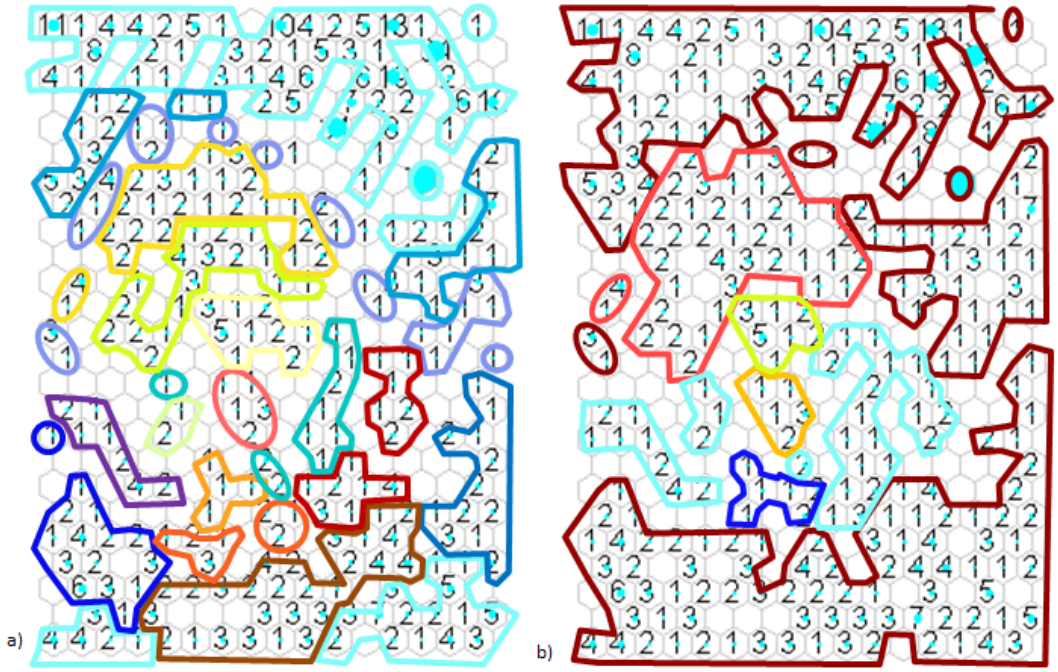


Figure 5-4 Boundaries of optimized clusters superimposed on the hit histograms. Clusters derived from SOM trained with the procedure matrix modified with Albrecht keywords and parameters derived from Wendel and Buttenfield (2010) using (a) K-means and (b) Ward's Linkage clustering. The hit histogram was derived from the same SOM. Boundary colors correspond to those in Figure 5-2.

The Mixed cluster is the largest in the SOM. It occupies 25% of the neurons in the SOM with non-zero hit frequencies (17% of all neurons). This is much smaller than the largest cluster in the default SOM (46% of non-zero hit frequency neurons, the smallest of the explicit keyword SOMs). In all of the experiments, the Mixed cluster is interpreted as failure in the explanatory power of the procedure matrix because the cluster groups what are interpreted as a wide range of types of GIS procedures. Therefore, the substantial reduction in the size of the Mixed cluster in the Albrecht SOM is interpreted as an important improvement in SOM quality compared to the explicit keyword results.

Stepping in from the lower edges of the SOM, clusters 2 and 5 form the outer layer of clustering around the region of high dissimilarity (see Figure 5-3). There are a number of smaller clusters also present in the area. These clusters continue to exhibit the concentric pattern of organization (rather than the relatively noisy organization seen around the region of high dissimilarity in the explicit keyword SOMs). This concentric pattern breaks down with at the center of the region.

Cluster 2, named “Management,” has relatively weak implicit keyword weightings compared to rest of SOM. “Enter Data” is dominant, as is the case for adjacent neurons in the Mixed cluster. “Search” is stronger in upper right portion of the cluster and “Measurement” becomes more important in the center-left of cluster, with “Distribution and Neighborhood Analysis”’ becoming prominent in neurons at far left edge, with boundary to cluster 5. In terms of the actual types of GIS procedures that are associated with the cluster, there is a mixture. The dominant type is raster processing, which includes the derivation of raster zones, surfaces, and tabular analysis of raster inputs. The neurons at the upper edges in the middle and extreme right of the cluster extent also include more vector-oriented functionality, emphasizing interactive editing at the left and spatial analysis at the right end of the cluster. This cluster does not have a particularly strong affinity for any single GIS command. The unifying characteristic of this cluster is a data management. Dissimilarity within this cluster is relatively low, as might be expected given the relatively low implicit keyword weightings (these weightings are the neuron

coordinates and Euclidean distances between pairs of these neurons are used in the U-matrix as an indicator of separation/dissimilarity).

Cluster 3, named “Linear Analysis,” has relatively strong values for all the analysis types of Albrecht implicit keywords (“Search,” “Location Analysis,” “Distribution and Neighborhood Analysis,” and “Spatial Analysis”) except for “Terrain Analysis,” which really did not appear anywhere in the set and so has no real influence in any of the SOM neurons. “Distribution and Neighborhood Analysis” is the strongest influence in all of the cluster neurons. To the lower right of the cluster, the “Measurements” and “Enter Data” dimensions becomes increasingly important, while the “Location Analysis” and “Spatial Analysis” dimensions become less important. The GIS procedures in this cluster deal with linear features. What is interesting here is that this groups both vector and raster based procedures. The linear analysis clusters in the explicit keyword SOMs were vector-based only. This cluster forms a contour around the region of high dissimilarity and exhibits relatively uniform internal stability.

Cluster 4, named “Interactive Editing,” rings cluster 12. A single neuron at the left extreme of the cluster is surprisingly different from the rest. Its dimensions really make it more similar to cluster 13. The remaining neurons in this cluster are dominated by the “Search” implicit keyword. Most neurons also show a strong secondary influence from the “Visualization” keyword, although the two neurons that form an island to the lower left of the main part of the cluster are very strong with the “Enter Data” dimension instead.

Cluster 5, named “Geographic Analysis,” shows relatively constant emphasis on the “Dimension and Neighborhood Analysis” dimensions, with other dimensions showing trends of change the top to the bottom of the cluster extent. At the top of the cluster, the “Location Analysis” and “Spatial Analysis” dimensions are important but reduces to a near-zero strength at the bottom. This cluster shows relatively high stability, with only the Mixed and Management clusters appearing more so. The GIS procedures in this cluster vary in that both raster and vector data types are handled, but generally share

a focus on search and measurement of geographic features. This cluster includes a mix of different kinds of geographic analyses that causes it to have similarity with several other clusters, notably the Linear Analysis and Spatial Analysis clusters.

The K-means analysis realizes a sequence of clusters that represent a continuum of types of GIS procedures that varying in the how much graphics-related work they carry out. For the most part, these clusters combine with slightly different additional types of GIS commands. In some cases, it is difficult to infer what the other focus is. Cluster 6, named "Graphics-Low," shows fairly low values for all dimensions. The importance of the "Visualization" implicit keyword is the strongest of the keywords for neurons in this cluster, closely followed by the "Search" keyword. There is a slight increase in the strength of influence of the "Location Analysis," "Distribution and Neighborhood Analysis," and "Spatial Analysis" keywords towards the lower neurons of the cluster. This cluster is composed of nine patches arranged in a broad crescent that goes from the left edge of the SOM to the right. It essentially forms a contour around layers of clusters that are progressively closer to the region of high dissimilarity. The fragmentation of this cluster could indicate internal instability, but the voids in the chain of patches are neurons with zero hit frequencies—making it difficult to come to a conclusion. In general, dissimilarity looks to be relatively low and constant through the region where this cluster spread. This cluster includes some editing-oriented procedures, functionality for displaying datasets in external databases, and analyses for developing visualization (e.g., set up of attribute-driven symbology).

Cluster 7, named "Graphics-Weak," shows fairly strong similarity to the Graphics-Low cluster because it too features weak values for all implicit keyword weightings in general with a slightly elevated value for the "Visualization" keyword. The weightings for this cluster are all slightly lower than for the Graphics-Low cluster, which makes intuitive sense because the Graphics-Weak cluster is further from the region of high dissimilarity than the Graphics-Low cluster. The Graphics-Weak cluster forms another graphics-related contour interval within the SOM. The fact that the signatures for the neurons in this

cluster have all been reduced so severely makes sense because the outside of this cluster (i.e., away from the region of high dissimilarity) is adjacent to the Mixed cluster, which is largely undifferentiated. Cluster stability is good. This cluster has a secondary focus on GIS commands that help to set the environment within which graphics are created, setting and managing symbology and other default values.

Cluster 8, named “Graphics-Medium,” is located on the other side of the Graphics-Low cluster, closer towards the central region of high dissimilarity. This cluster is again similar to the Graphics-Low cluster with generally low influence of all implicit keywords and a stronger weighting on the Visualization keyword. This cluster shows a substantial increase in the strength of the “Visualization” component over the Graphics-Low cluster. The U-matrix shows internal dissimilarity to be roughly equivalent to the two previous graphics-related clusters. The degree to which the GIS procedures in this cluster rely on graphics type GIS commands exclusively is perhaps the strongest of all the clusters in the SOM. This cluster is interpreted as being the “purest” concentration of graphics type GIS procedures.

Cluster 9, named “Graphics-High,” continues the trend of increasing the weighting of the “Visualization” implicit keyword in the signature of cluster neurons as position towards the central region of dissimilarity decreases. At the right side of this cluster, there is a slight increase in the emphasis on the “Search” and “Enter Data” dimensions. The lower left of this cluster shows an increase in “Terrain Analysis.” This variability is mirrored by a slight increase in the average U-matrix values within the cluster, particularly towards the central region.

Cluster 10, named “Graphics-Peak,” shows a stronger “Visualization” component still, coupled with a major increase in “Terrain Analysis” and a smaller increase in “Search” functionality. In addition to GIS procedures focused on visualization, there terrain analysis procedures for viewshed analysis. This cluster is immediately adjacent to the region of high dissimilarity and suffers a degradation of internal stability.

Cluster 11, named “Spatial Analysis,” is physically distinct from the sequence of graphics clusters, appearing to the right of the region of high dissimilarity. There is still a strong influence of visualization, but the strongest dimension in this cluster is related to “Search” algorithms. This cluster is parallel to the Interactive Editing cluster, sitting further away from the region of high dissimilarity. The signatures of the two clusters are relatively similar, with the Spatial Analysis cluster showing relatively lower weights for the implicit keywords. The GIS procedures in this cluster are almost exclusively related to spatial analysis of vector data. The few routines that are not pertain to table manipulation and graphics. The complete absence of any raster processing types of procedures is of interest.

Cluster 12, named “Search,” is the end in the sequence of increasingly graphics-driven clusters, exhibiting a very slightly reduced emphasis on visualization of the set relative to the Graphics-Peak cluster. It is interesting to note that through the sequence of graphics clusters, the non-“Visualization” keywords are very slowly increasing in weight but not as rapidly as “Visualization.” By the Graphics-Peak cluster, the secondary keywords start to show increased growth in their weights. In the Search cluster, “Visualization” is waning slightly while the others are still increasing. This cluster also shows the most mixed range of strong influences from several other dimensions, “Search” and “Enter Data” being the most prominent. This cluster is similar to the cluster 13 and, to a lesser degree, the cluster 14 that member neurons both show some of the strongest influence across the most dimensions of any neurons across the entire SOM. All of these clusters that show very high weights in their neuron signatures also show greater measures of dissimilarity. There is a mix of GIS procedures associated with this cluster, ranging from terrain analysis for visualization, interactive editing, and linear analyses. No clear definition of a unified purpose for this cluster is obvious. This is also true of the remaining clusters.

Cluster 13, named “Strong Vector,” is the smallest cluster in the SOM and is associated with only two neurons and three GIS procedures. It, like the Search cluster, shows some extremely high weightings for most dimensions, although it places especially high emphasis on “Location Analysis,”

"Distribution and Neighborhood Analysis," and "Spatial Analysis." Dissimilarity is very high within and around this cluster because it sits in the middle of the region of high dissimilarity around which then entire SOM is organized. The GIS procedures in this cluster all pertain to vector data. Two of the three procedures determine spatial relationships between two different vector dataset and the third manipulates individual arc features in single data set.

Cluster 14, named "Measurement-Major," is also relatively sparsely populated (4 GIS procedures), but all have a strong focus on "Measurement" and "Enter Data" types of functionality. It is proximal to the Strong Vector cluster and the region of highest dissimilarity. The meaning of this cluster was not clear based on the 4 GIS procedures. The 'CALC' GIS command is used heavily in the GIS procedures.

Cluster 15, named "Measurement-Minor," also favors the "Measurement" and "Enter Data" dimensions, but with less than half the intensity of the Measurement-Major cluster. These two clusters seem to be gradations of the same type of GIS procedure, much like the sequence of clusters related to visualization types of GIS procedures. The Measurement-Minor cluster is outside of the Measurement-Major cluster relative to the region of high dissimilarity. The GIS procedures in this cluster carry out a variety of types of tasks, including database access, data management, and editing.

Direct correlation between the K-means clusters and the Albrecht implicit keywords is difficult because there are more clusters than keywords (especially in the area of the volcano where cluster fragmentation is severe). It is possible to visualize the weight of a single dimension (i.e., a keyword in the procedure matrix) for each neuron in the SOM with a color-coded presentation of the SOM. This has not been presented for the explicit keyword experiments because there were far too many dimensions to handle effectively or meaningfully. Figure 5-5 shows each of the eight dimension (i.e., Albrecht implicit keyword) values across all neurons, with a different sub-figure for each keyword. The display is a gray-scale with low values shown as dark blue and high values as white. The lowest values are not shown in

black in order to maintain visibility of the K-means cluster boundaries. All the sub-figures feature a maximum towards the lower left of the SOM. Although the exact locations of the maximum values for each Albrecht implicit keyword vary, there is overlap between regions surrounding the maxima across keywords.

These weightings are interesting because the headings for each sub-figure reveal the maximum value for that dimension. Because all dimensions are in the same units, the magnitude of these numbers indicates the relative importance of each dimension. “Search”, “Viz”, and “Data Entry” are the most important. “Terrain Analysis” is almost a non-factor. “Loc Analysis”, “Dist+Nbrhd”, and “Spat Analysis” are also relatively unimportant drivers of the information space. The lower maxima for “Terrain Analysis”, “Loc Analysis”, “Dist+Nbrhd”, and “Spat Analysis” corresponds with the fact that GIS commands associated with these Albrecht implicit keywords were not widely used in the GIS procedures. This does not necessarily mean that these types of commands are not important in general. For instance, GIS procedures associated with “Terrain Analysis” are likely to be located within a relatively small region of the SOM (see Figure 5-5c).

Although the procedure matrix used to create this SOM was not built using the GIS commands (i.e., it used Albrecht implicit keywords), it is still informative to assess how the K-means clusters correspond to the GIS commands as well as the Albrecht implicit keywords. The author examined the most commonly occurring GIS commands for all GIS procedures in all neurons. In general, individual clusters were not dominated by the small number of GIS commands as with the explicit keyword experiments. Additionally, multiple clusters associated with similar sets of the GIS commands, offering more subtle shifts in cluster focus.

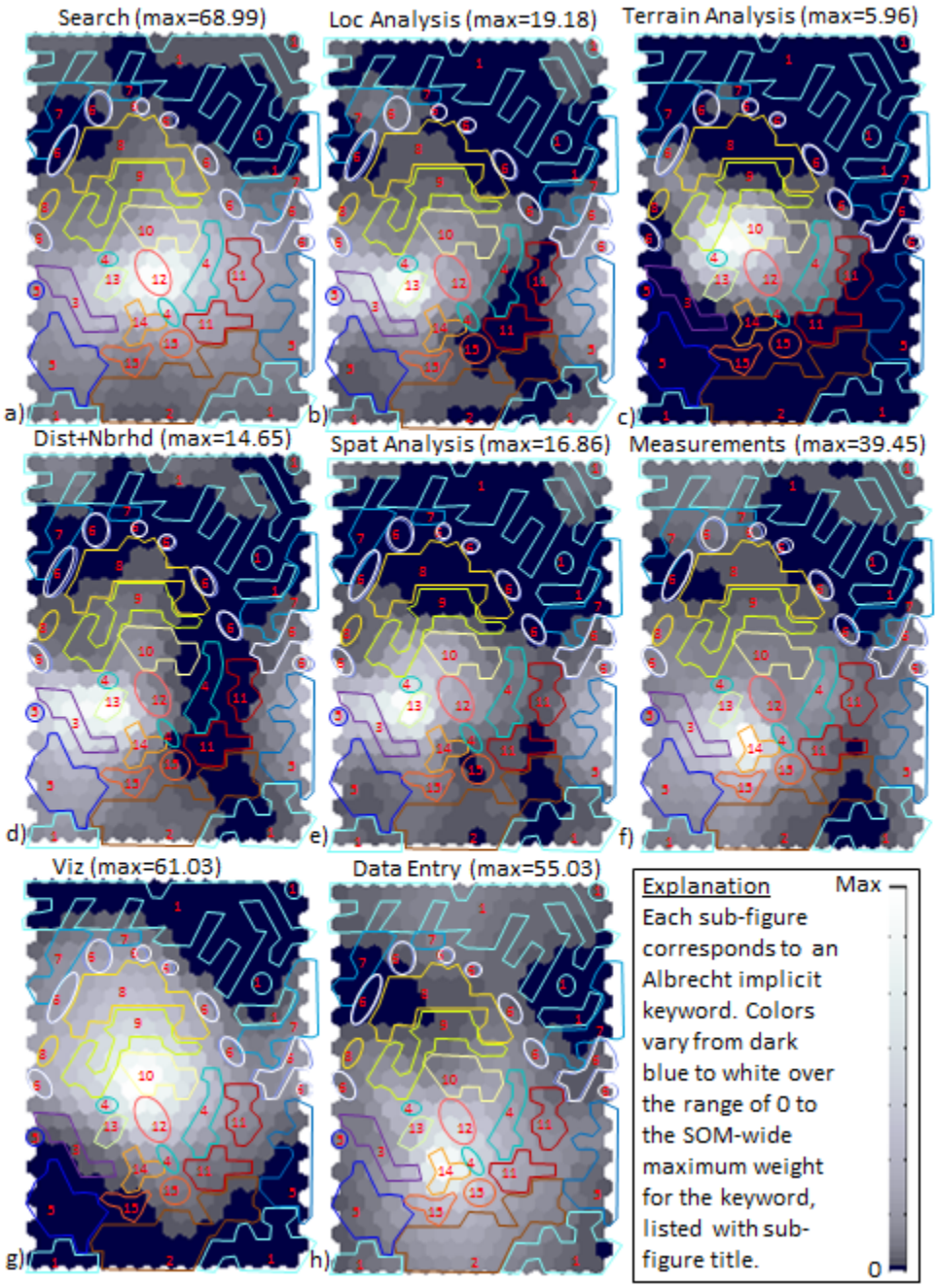


Figure 5-5 Display showing the strength of Albrecht implicit keywords for each neuron in the SOM trained with the procedure matrix modified with Albrecht keywords and parameters derived from Wendel and Buttenfield (2010), with K-means clusters boundaries superimposed.

As with Figure 4-10a) and b) and Figure 4-15, Figure 5-6 was generated by deriving PCA coordinates from the Albrecht implicit keyword-augmented procedure matrix and from the

corresponding SOM. The plot, which uses the first two principal components for the procedure matrix and the corresponding SOM, shows major improvements in both the percentage of the data variance explained (more than a 25 percent increase to 78 percent) and the spreads and patterns of both the data and the SOM neurons (black dots), relative to the PCA of the explicit keyword experiments. Although there is still overlap in the data, the visualization no longer shows all the data bunched in linear patterns parallel to each axis. In addition, the neurons, colored to reflect the K-means groupings, has begun to clearly separate. Though some instances of overlapping clusters persist, the PCA projection shows clear separation of clusters, with maximum dispersion for the red, orange, and yellow clusters (which map to cluster numbers 12, 14, and 10, respectively).

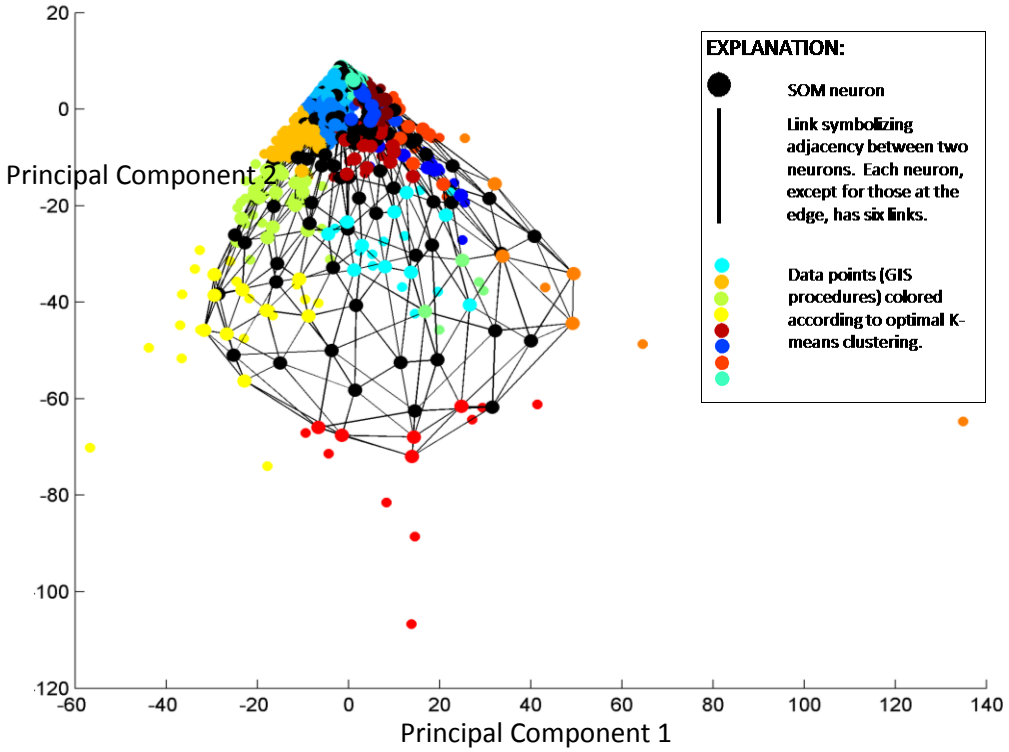


Figure 5-6 Data points and SOM neurons plotted using coordinates of the first two PCA components for the SOM trained with the procedure matrix modified with Albrecht keywords and parameters derived from Wendel and Buttenfield (2010). This explained 78.00 percent of the variance in data points.

The Albrecht SOM exhibits a number of improved characteristics when compared to the explicit keyword SOMs. There was a relatively even spread of hits across the entire SOM, with relatively few "super" neurons showing very large hit frequencies. While neurons with large numbers of hits are not necessarily inappropriate, the cause for these should be verified because they tend to distort the region around them during the training process. This can lead to extra clusters being created or creating instability within a larger cluster. The spread of hits in this SOM is interpreted to mean that the Albrecht augmented procedure matrix provided additional information with which the SOM training process was able to differentiate the GIS procedures more effectively than with any of the explicit keyword procedure matrices.

The K-means clusters derived from the Albrecht SOM show a new, desirable pattern of concentric clusters surrounding the region of high uncertainty. The concentric clusters imply decreasing certainty or confidence in the cluster as its position moves closer to the region of high uncertainty. In addition, this set of clusters represented a shifting set of specialization in graphics-oriented GIS procedures. This was consistent with visual interpretation of the display of the dimensions of the SOM in Figure 5-5. This pattern persists relatively far into the region of dissimilarity before it breaks down into smaller clusters that were less credible. Put another way, the impact of the region of high uncertainty is limited to a relatively small area of the SOM.

Although this relationship is somewhat loose, the graphics clusters are reminiscent of contours. Within each of these clusters, a relatively consistent level of internal stability is exhibited. The hit histogram (Figure 5-4a) shows that these clusters all have a relatively even spread of hits, which is correlated with smoothness in the U-matrix and implies that individual neurons were trained based on interaction with a number of GIS procedures (which avoids the creation of isolated types of neurons).

The clusters that surround the lower edges of the region of high dissimilarity show a lower stability, although individual clusters, such as Spatial Analysis or Measurement-Minor, have consistent

U-matrix values within themselves. The clarity of type or purpose of the Search, Strong Vector, and Measurement-Major clusters is poor because of the high dissimilarity even within the cluster (small) populations of neurons. This breakdown is typical of all the explicit keyword SOMs. The current SOM shows improved structure because it successfully isolates the impact of this region to a few clusters and builds clusters that represent clearer types of GIS procedures around the area.

The Ward's Linkage clusters are similar in general pattern to the K-means results but, because of its substantially smaller number of clusters, lack some of the desirable patterns seen in the K-means due to over-aggregation. For example, the K-means clusters, Graphics-Weak and Graphics-Low, are part of cluster 1 (essentially an even larger "Mixed" cluster) in Figure 5-3b), creating a new cluster boundary that crosses a number of patches that exhibit elevated U-matrix values resulting in a cluster with a higher degree of instability. Ward's Linkage cluster numbered 2 encompasses what were two separate clusters in the K-means analysis, Graphics-Medium and Graphics-High, which also results in a greater range in U-matrix values within the cluster. The expansion of cluster 1 from the lower edge of the SOM has a similar effect (for instance, consuming K-means Measurement-Minor cluster).

The PCA projection of both the GIS procedures and SOM neurons expressed using the Albrecht implicit keyword-enhanced procedure matrix also shows a marked improvement. The first two components of the PCA of the GIS procedures explains 78% of the variance in the procedure matrix, an improvement of over 25% from the best of the explicit keyword experiments (the PCA-driven SOM). This margin is large enough to indicate that the Albrecht procedure matrix provides a better description of the GIS procedures than the explicit keywords or principal components derived from those explicit keywords.

All of these improvements in the organization of the SOM are related to the enriched information contained in the augmented procedure matrix. The Albrecht implicit keywords enable the SOM training process to more readily differentiate GIS procedures, which results in the more uniform

spread of hits throughout the SOM. This distribution results in a better spatial pattern in the signatures of the SOM neurons, and helps automated algorithms to extract individual clusters that are generally more consistent and meaningful than those found in the explicit keyword experiments. In addition, the spatial arrangement of clusters across the SOM as a whole is more meaningful. The PCA analysis of the results is consistent with this interpretation.

5.2 Implicit Keywords based on Environmental Modeling

The author developed an alternative set of keywords based on Wendel and others (2008a; 2008b) that reflects concerns typical of work done to manipulate spatial data for use in environmental modeling. This differs from the Albrecht typology in a conceptual sense that it is intended to represent a subset of characteristics specific to an applied domain instead of general or universal characteristics. For simplicity, this will be referred to as the "Enviro Modeling" implicit keywords. The Enviro Modeling set of implicit keywords is listed in Table 5-8. The first type includes GIS commands that are used to generate graphics or some kind of visualization. The second type breaks out GIS commands that are often used in managing whole datasets ('COPY', 'RENAME', 'KILL') or identifying subsets of content from within a dataset. The GIS commands set in the third group do not actually produce new spatial data or even touch the data. Instead, these commands make settings that inform the GIS how to execute subsequently issued commands. For instance, the 'SETMASK' command specifies to the raster processing of ArcInfo the spatial silhouette by which to constrain the derivation of new data sets (regardless of the command used to derive the new data set). The "Raster" type is associated with GIS commands that either use or produce a raster dataset. The "Vector" type indicates whether a command uses or produces a vector type of dataset. Note that the "Raster" and "Vector" types are not mutually exclusive. For example, the author has associated the "Additem" command, which adds a new field to the attribute table of the spatial dataset, to both types because it can be applied to examples of both

types of GIS data. The “Derive” type indicates whether the GIS command actually produces a new GIS dataset. This type can be associated with both “Raster” and “Vector” GIS commands. It is also useful for finding or avoiding GIS command that only produce graphics, manage the datasets, or make changes to the processing environment of the GIS. Table B-1 in Appendix B lists the evaluation of the GIS commands according to the Enviro Modeling implicit keywords.

Table 5-8 Types of GIS commands from the environmental modeling perspective.

Analytical types of commands	Examples of type
Graphics	Gridshades, Linesymbol
Selection/ Data Management	Copy, Kill, Clearselect
Environment	Drawenvironment, Editfeature, Searchtolorance, Setmask
Raster	Zonalmajority, Gridshades, Additem
Vector	Additem,Intersect
Derive	Zonalmajority, Save, Unload

The purpose of both sets of implicit keyword experiments is to demonstrate the utility of implicit keywords to gain better understanding of GIS procedures and to improve organization of sets of GIS procedures. The Albrecht implicit keywords explored these topics using a general view of GIS analysis functionality and were modified to incorporate commands in the set which the Albrecht set did not recognize, such as visualization commands. The Enviro Modeling set of implicit keywords was initially designed to highlight functions that were especially important to environmental modeling; this design was also modified as a result of the analysis of the set. The variety of GIS commands found in the set did not support additional environmental modeling-specific GIS commands as originally envisioned. Therefore, the realization of this set of implicit keywords was expanded and tailored in response to the distribution of GIS commands discovered within the set used for this experiment. It should also be noted that the Enviro Modeling set was not necessarily engineered to reflect the “best” classification of GIS functionality, but rather to help differentiate GIS procedures.

The procedure matrix augmented with the Enviro Modeling implicit keywords is listed in Table B-2 in Appendix B. This version of the procedure matrix, in conjunction with the recommendations of Wendel and Buttenfield (2010), produced the SOM shown Figure 5-7. This SOM features 345 neurons that are arranged into a matrix with 15 neurons on the x axis and 23 neurons on the y axis. The radius of the neighborhood around a neuron over which adjustments were made during the training process was set as a function of the matrix size, again as per Wendel and Buttenfield (2010), starting out as the length of the long side of the matrix and decreasing to half that size through the training iterations. The quantization error for the SOM was 3.1773 and the topographic error was 0.1165, both of which are slightly elevated relative to those for the Albrecht SOM. The quantization error is less than for the default and optimized SOMs, but more than for the PCA-driven SOM. The topographic error was almost double for this SOM than for any from the explicit keyword experiment. The hit histogram for this SOM, shown in Figure 5-7a), shows a similar distribution as that seen for the Albrecht SOM. There are a small handful of neurons with high frequencies that are located relatively close to each other with the remainder of the map showing an even spread of low or zero hit frequency neurons. Table C-6 in Appendix C lists the neuron identification numbers and the GIS procedures that found the respective neuron to be the best matching.

The U-matrix from this SOM, shown in Figure 5-7b), also shows patterns similar to the Albrecht SOM. Towards the top of the figure, in the region circled in yellow there is a single region of very high dissimilarities surrounded / flanked by a relatively steep slope that leads to a region of relatively high similarity (i.e. low U-matrix values). In the case of this SOM, the ring(s) is located at the north of the image and the “flats” are in the center. The maximum U-matrix value (41.38), representing the highest dissimilarity, is slightly higher than that for the Albrecht SOM (35.20), both of which are higher than for both the default and PCA-driven SOMs (18.13 and 13.30, respectively).

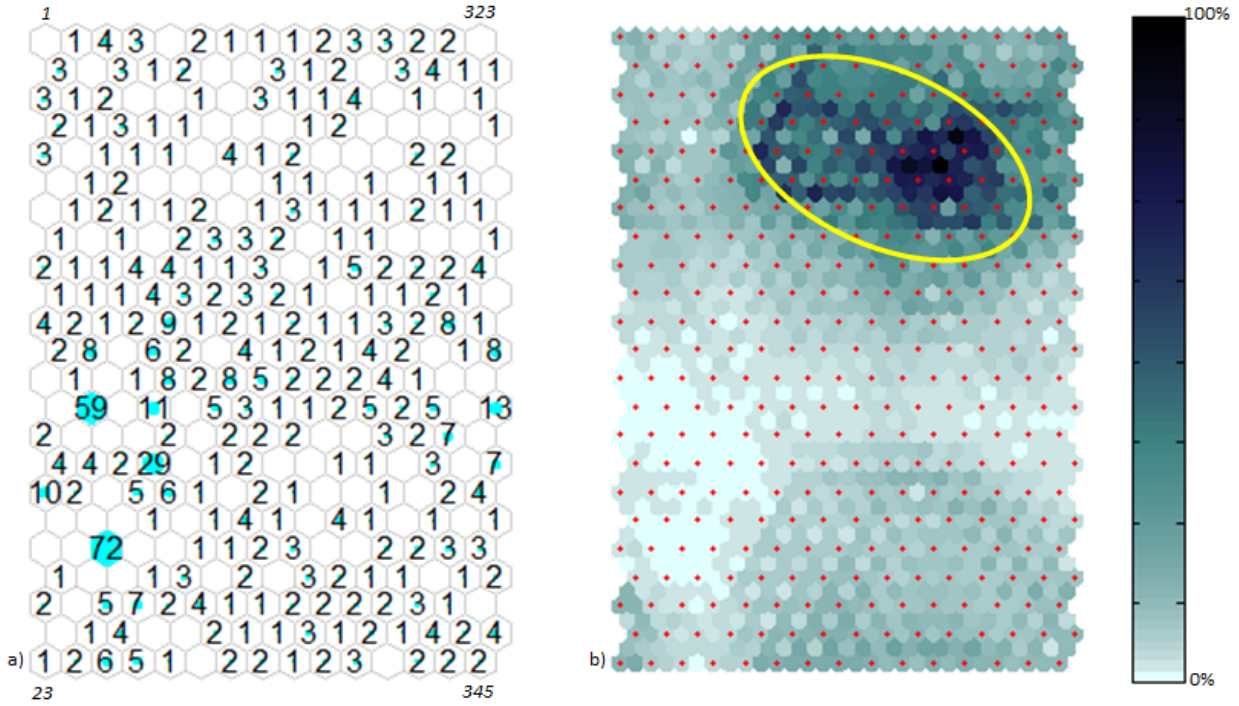


Figure 5-7 SOM trained with the procedure matrix modified with the Enviro Modeling keywords and parameters derived from Wendel and Buttenfield (2010). (a) Neurons labeled with frequency of best-matching GIS procedures. The size of the blue patch also indicates match frequency. Corner neuron identification numbers posted in italics. Neurons are numbered sequentially from top to bottom, left to right. (b) U-matrix for the SOM, showing dissimilarity as a darker color. Color indicates percent of range in U-value within SOM. Actual values range from 0.15 to 41.38. Regions of lighter colors indicate clusters of similar neurons and darker values indicate separation between clusters. The red dots indicate locations of neurons.

At first glance, the pattern of the optimal K-means clustering shown in Figure 5-8a) appears chaotic. This due at least in part to the large number of clusters found (14), especially given that there are only six organizing dimensions. The fact that the region of high uncertainty is close to the perimeter of the SOM tends to force some clusters to wrap around the edges, as seen with the cluster number 4. In general, the arrangement of clusters resembles that developed for the Albrecht SOM, which features a central region with relatively small clusters that are more uncertain that is surrounded by concentric layers of clusters. For instance, at the bottom of the SOM, the cluster numbered 3 surrounds cluster 2, which surrounds cluster 5, which abuts the region of high uncertainty. At the left of the SOM, cluster 4 is (to some degree) behind the neurons of cluster 6, which is behind (the relatively small) cluster 12. The

"flat" region in the U-matrix (Figure 5-7b), seen in the region of relatively low U values that run horizontally through the center of the SOM (shown in cyan), is largely encapsulated as a single cluster (numbered 1). The Ward's Linkage clustering results (Figure 5-8b) exhibit similar patterns to those shown with the K-means clustering, except that the Ward's solution further homogenizes clusters in the middle and bottom of the SOM. The optimal results for Ward's Linkage analysis yield 10 instead of 14 clusters. Ward's cluster 1 encompasses the neurons of K-means clusters 1,2,4,13, and 14. K-means clusters 2 and 5 are represented by a single cluster, numbered 2, in the Ward's Linkage results. The fragmentation around the region of high dissimilarity is roughly constant across the two clusterings.

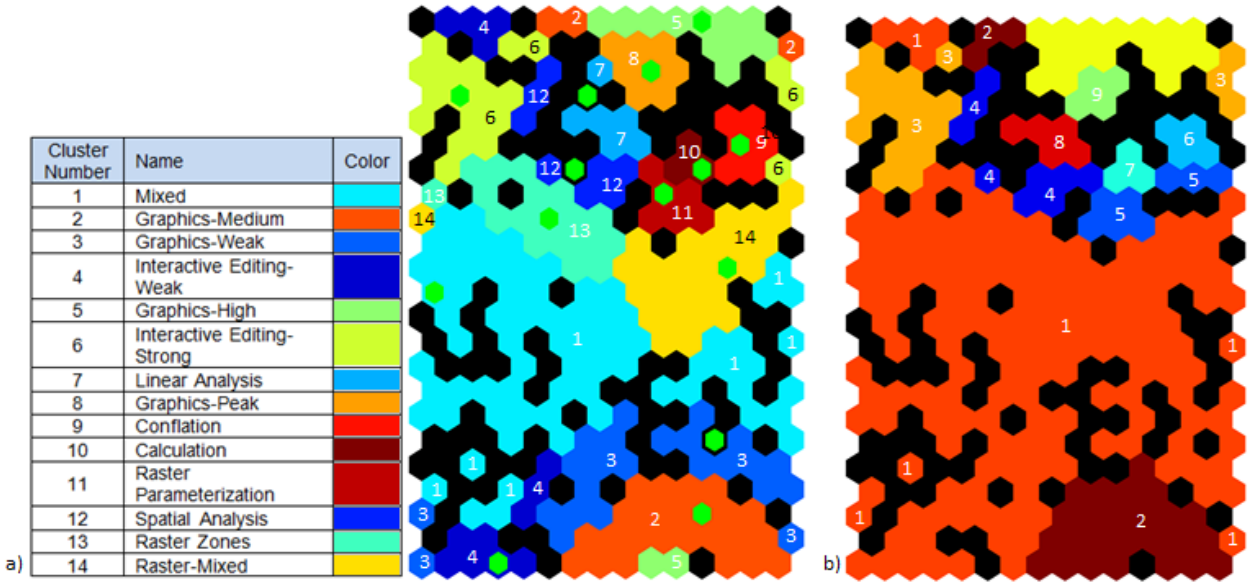


Figure 5-8 Optimal clustering of SOM trained with the procedure matrix modified with the Enviro Modeling keywords and parameters derived from Wendel and Buttenfield (2010) using (a) K-means (14 clusters) and (b) Ward's Linkage (9 clusters). The number of clusters was optimized using the Davies-Bouldin Index in both sub-figures. Black neurons have no GIS procedures associated with them. Green neurons are cluster centroids in sub-figure (a).

As with all the other SOMs, the Ward's Linkage results feature fewer clusters than those of the K-means. Generally, clusters are composed of the same neurons in both sets of results or whole clusters are grouped together to form an aggregate cluster in the Ward's Linkage results. There are on the order of a dozen neurons that crossed boundaries in the entire SOM. This is interpreted to indicate that statistical evaluation of the SOM content is stable across clustering algorithms. Clusters that persist in

both sets of results are considered to be relatively strongly distinct from neighboring clusters, while those that are collapsed into others might be considered to be specializations or sub-types of another cluster.

The table in Figure 5-8a) lists the names assigned to each of the K-means clusters. The interpretations used to define these names will be described in the remainder of the section. Cluster 1, named “Mixed,” is the largest cluster in the SOM and occupies horizontal band across the middle of the SOM. It occupies 28% of the neurons with non-zero hit frequencies (see Figure 5-9a), a comparable size to the largest cluster in the Albrecht SOM. This cluster includes neurons with the three highest hit frequencies (72, 59,29).

The average uncertainty associated with this cluster is the lowest for any of the K-means clusters, as indicated by Figure 5-10a). By examining Figure 5-9 and Figure 5-10 together, one can see that although the three highest hit frequencies are near to each other and are separated by low U-matrix values, each these neurons is adjacent to several no-hit neurons. As seen with the Albrecht SOM, this indicates that the SOM training process isolated individual neurons, resulting in relatively strong differences across the cluster. This is interpreted to mean that the cohesiveness of this cluster is relatively weak.

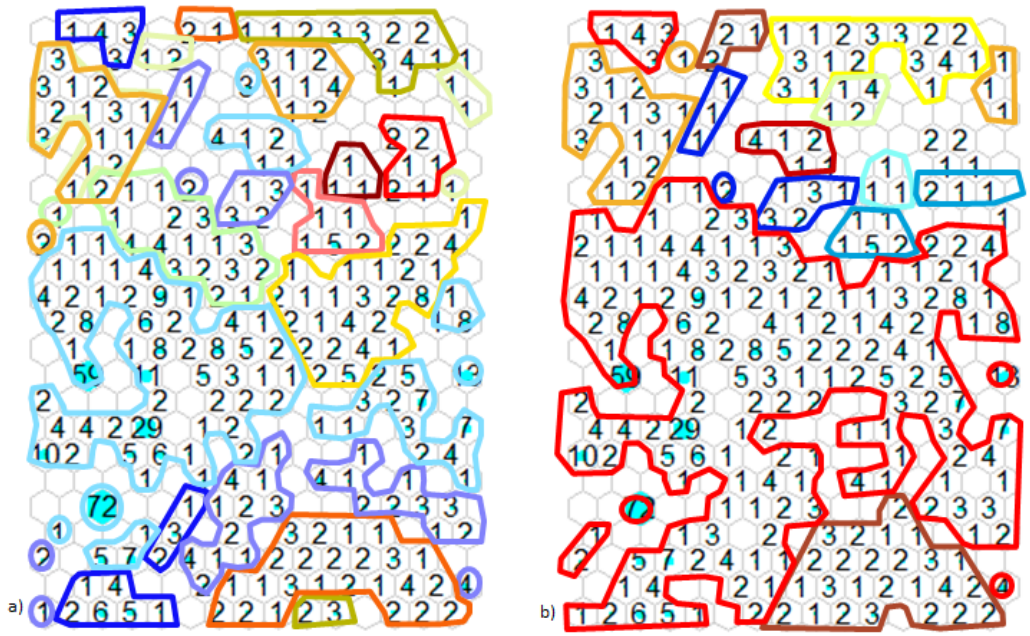


Figure 5-9 Boundaries of clusters derived from the SOM trained with the procedure matrix modified with the Enviro Modeling keywords and parameters derived from Wendel and Buttenfield (2010) using (a) K-means and (b) Ward's Linkage clustering superimposed on the hit histogram for that SOM. Boundary colors correspond to those in Figure 5-8.

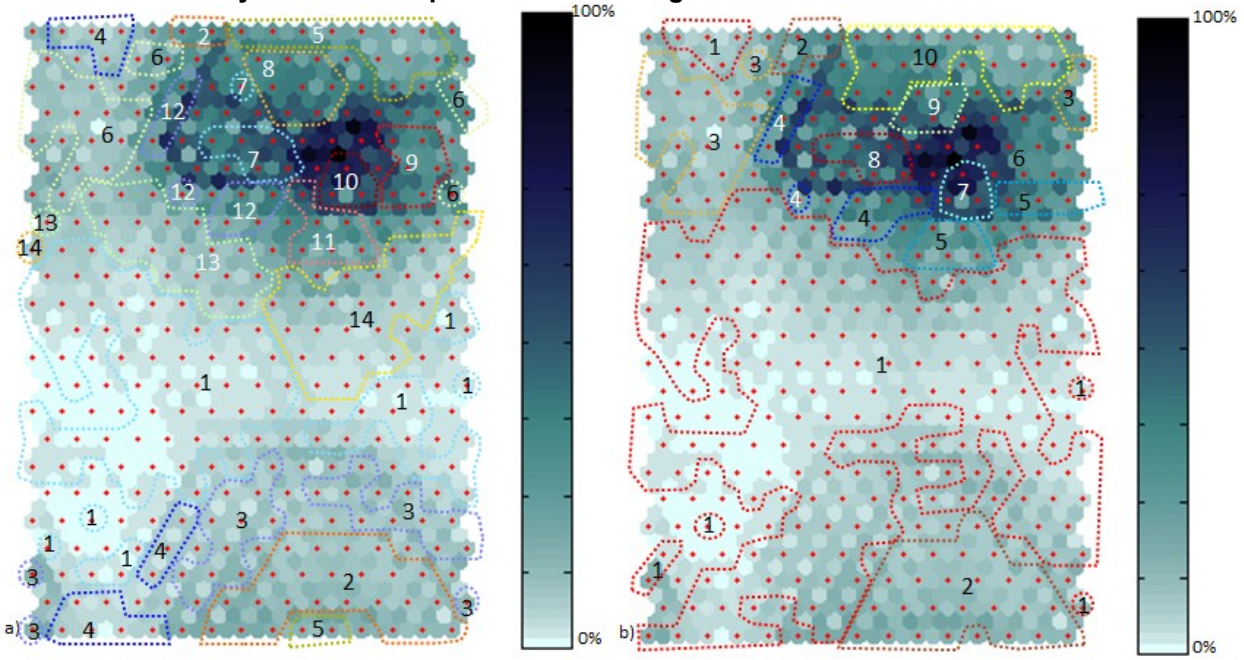


Figure 5-10 Boundaries of clusters derived from the SOM trained with the procedure matrix modified with the Enviro Modeling keywords and parameters derived from Wendel and Buttenfield (2010) using (a) K-means and (b) Ward's Linkage clustering superimposed on the U-matrix. Boundary colors correspond to those in Figure 5-8.

Before continuing with the interpretation of each cluster, it is worth noting several general patterns in the clusters relative to the U-matrix, shown in Figure 5-10. The arrangement of clusters immediately adjacent to region of highest dissimilarity has reverted to the style seen with the PCA-driven SOM, where all clusters have at least part of their boundary in direct contact with high U-matrix values. While this is not inherently bad, it does tend to result in the inclusion of relatively dissimilar neurons within individual clusters, reducing their stability and possibly the distinctiveness of each cluster. Rather than having one or a few clusters with high uncertainty, this results in many clusters with a slightly higher uncertainty.

There are patterns of values in the U-matrix that correlate with the cluster boundaries. The high U-matrix values at the lower edge of cluster 9 track with the separation of this cluster from clusters 14 and 6. There also appears to be a strong separation between clusters 7 and 12, and also between clusters 7 and 8. (Note that hand-drawn cluster boundaries group neurons, shown as red dots, and that the patches between neurons, indicating separation between neurons, are not actually part of any cluster. Overlap of cluster boundaries onto patches between neurons is an unintended by-product of manually drawn boundaries.) There is a definite boundary ring of higher U-matrix values that goes all the way around cluster 7, which indicates a good degree of spatial ordering in the SOM.

The display of weightings of the individual keywords, shown in the sub-figures of Figure 5-11, indicates that the Mixed cluster neurons have low weights for all keywords (as indicated by the fact that the neurons in the Mixed cluster are darker gray or blue for all keywords). This is similar to the Mixed cluster for the Albrecht SOM, and is again interpreted to mean that there is no single unifying type associated with GIS procedures in this cluster. The Mixed cluster in the Enviro Modeling SOM features weights that are noticeably stronger. Put another way, the neurons and associated GIS procedures in the Enviro Modeling SOM Mixed cluster show better differentiation from each other and from other clusters in the SOM than for the corresponding units in the Albrecht SOM.

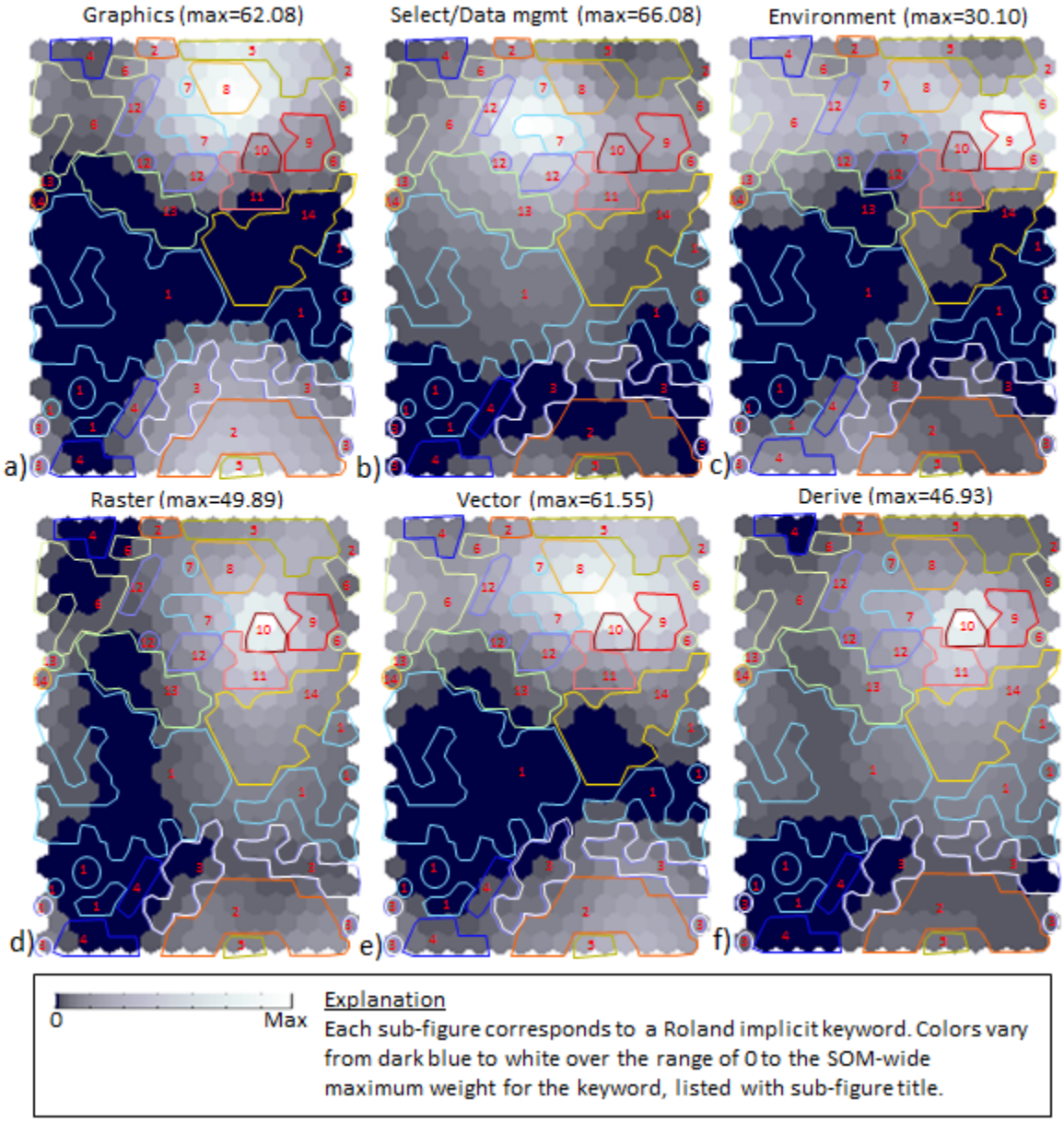


Figure 5-11 Display showing the strength of the Enviro Modeling implicit keywords for each neuron in the SOM trained with the procedure matrix modified with the Enviro Modeling keywords and parameters derived from Wendel and Buttenfield (2010), with K-means cluster boundaries superimposed.

By interpreting the weightings of the individual keywords (Figure 5-11), three regions are detected within the Mixed cluster. At the upper left of the cluster, there is an emphasis on GIS procedures that carry out derivation of raster products (Figure 5-11d) and data management (Figure 5-11b) associated with these data. At the right edge of the cluster, there is an emphasis on vector data

(Figure 5-11e) and either selection or data management. The third region, in the lower left of the cluster shows the lowest values for all dimensions in the entire SOM. There is an inclusion of a small group of neurons that are oriented towards database access. There is also an isolated neuron that has a relatively large hit frequency that is associated with GIS procedures for format conversion (although these procedures concentrated in all other SOMs, as well).

Cluster 2, named “Graphics-Medium,” features GIS procedures that are predominantly focused on generation of graphics (Figure 5-11a) such as graticules and labels. This cluster is very consistent in character because it shows medium intensity usage of GIS functionality for visualizing vector-based data (Figure 5-11e). There is a slight increase in the usage of other types of GIS functionality at the lower edge of the SOM. At the left side, this pertains to setting of parameters that control the analysis (Figure 5-11f) or visualization environment (Figure 5-11c). At the right side, there is instead an emphasis on raster functionality (Figure 5-11d).

As with the Albrecht SOM, there is a sequence of clusters that deal with variations on graphics generation. Cluster 3,”Graphics-Weak,” has a similar signature as the Graphics-Medium cluster that it surrounds, although the magnitude of usage of visualization functionality is less. Most of the GIS procedures in this cluster have a heavy focus on managing symbology. Some GIS procedures are involved in analyses used to support visualization, such as viewshed analysis. The usage of vector-specific functionality is greatly reduced in this cluster relative to the Graphics-Medium cluster. As with the Graphics-Medium cluster, there is an increase in usage of other types of functionality at its lower edges. Specifically, adjustment and control of the analysis or visualization environment increases at the lower left.

Cluster 4, named “Interactive Editing-Weak,” shows generally weak usage of all types of GIS functionality, except for a slightly elevated intensity for adjusting parameters of the analysis or visualization environment (Figure 5-11c) and handling of vector information (Figure 5-11e). Examination

of the GIS procedures in this cluster indicate that the purpose of most pertain to interactive editing within the Workstation ArcInfo module ArcEdit.

Cluster 5, named “Graphics-High,” is wrapped by the Graphics-Medium cluster at the lower edge of the SOM and flows to the top edge of the SOM. This cluster contains a mix of raster functionality and the use of annotation. There is a trend of increasing usage of graphics functionality from the Graphics-Weak cluster through the Graphics-Medium cluster and into this one. This cluster shows strength higher usage of graphics and vector handling functionality, but also shows an increase in all the remaining types of functionality.

Cluster 6, named “Interactive Editing-Strong,” is similar to the Interactive Editing-Weak cluster, except that individual GIS procedures do even more of this kind of work. There is an increase in the handling of vector data in this cluster relative to the Environment Control-Weak cluster. There is also an increase in the usage of graphics-related functionality.

Cluster 7, named “Linear Analysis,” includes GIS procedures that specialize in the selection and management of vector data features (Figure 5-11b,e). A number of the associated GIS procedures carry out some form of linear analysis, including location-allocation and address matching. At its right edge, there is an increase in procedures for graphics creation. This makes sense because this cluster is adjacent to cluster 8 (named “Graphics-peak”). The higher U-matrix values along the boundary of this cluster (Figure 5-10a) indicate that adjacent clusters are relatively different, although there is a separate single neuron patch of this cluster that seems to be more similar to the Graphics-Peak cluster (numbered 8). In general, Figure 5-10a indicates that clusters 7-12 exhibit elevated average internal uncertainty.

Cluster 8, named “Graphics-Peak,” associates with GIS procedures that the Enviro Modeling implicit keywords indicate carry out the most graphics-intensive work in the set. The procedures in this cluster also use all of the other types of GIS functionality described by the Enviro Modeling implicit

keywords, but not to the same level of intensity as graphics-related functionality. This is interpreted to indicate that these procedures are some of the most complex in the set. The procedures in this cluster feature a relatively strong usage of the 'CURSOR' GIS command.

Cluster 9, named "Conflation," uses selection types of GIS functions, but is dominated by interactive editing for manipulating vector data. It has a strong similarity with the Interactive Editing-Strong cluster and could really be considered to be a variation on it. The most commonly occurring types of procedures in this cluster are for conflation. Editing for annotation is slightly less prevalent.

Cluster 10, named "Calculation," is only three neurons, each of which associated with a single GIS procedure. The apparent common type of functionality is use of the 'CALC' command, although the three neurons show a high dissimilarity to each other. This cluster is located in the middle of the region of highest dissimilarity in the SOM. Stability of this cluster is the lowest in the SOM.

Cluster 11, named "Raster Parameterization," shows minor presence of selection commands and is dominated by raster processing (Figure 5-11d) and data management (Figure 5-11b) commands. There are relatively high dissimilarities between member neurons. Almost half of the procedures in this cluster are for generating tabular information (parameters) from raster data sources (Figure 5-11 d,f).

Cluster 12, named "Spatial Analysis," is dominated by GIS procedures that carry out selection and data management, and provide database access. Although not in the majority, the single most common type of GIS procedure in this cluster is for spatial analysis. This cluster shows an intermediate level of internal dissimilarity and is relatively clearly separated from clusters around it (judging by the U-matrix).

Cluster 13, named "Raster Zones," is relatively large. The GIS procedures in this cluster carry out selection and data management (Figure 5-11b). Although the purpose of this cluster is somewhat mixed, there is a concentration of GIS procedures that derive new raster feature (zones) datasets towards the right side of the cluster. It shows good internal stability and low dissimilarity.

Cluster 14, named “Raster-Mixed,” is very similar to the Raster Zones cluster, but generally shows higher values for the implicit keyword dimensions. There are a relatively large number of GIS procedures in the cluster that carry out either the generation of new raster feature datasets or the derivation of tabular information (parameters) of raster datasets (both of which are indicated by Figure 5-11f). This cluster shows good internal stability and low dissimilarity. The ‘CALC’ GIS command is used heavily by procedures in this cluster.

The PCA projection, visualized in Figure 5-12, further distributes the neurons within the two-component space, relative to the displays generated for explicit keyword or the Albrecht SOMs. The data points and the neurons are much more spread out relative to the three SOMs generated with the explicit keyword procedure matrix (i.e. the default SOM, the optimized SOM, and the PCA-driven SOM). The clusters developed by K-means clustering map relatively cleanly to distinct regions of the PCA projection space. Neuron coloring indicates distinct clusters, showing some mixing among the red and blue clusters but good distinction among other clusters. The clusters designations are taken from the K-means analysis, although the color assignments used in Figure 5-12 are different. The first two principal components derived from the set of implicit keywords for this SOM explained approximately 76 percent of the variance in the data points. Clusters 8 (yellow orange at lower left) and 10 (burgundy at far right) show the best spread. Cluster 7 is the set of dusty blue points in lower center. Although the overall spread of the clusters in this display appears to be reduced from that of the Albrecht SOM, the individual clusters appear to be more tightly grouped.

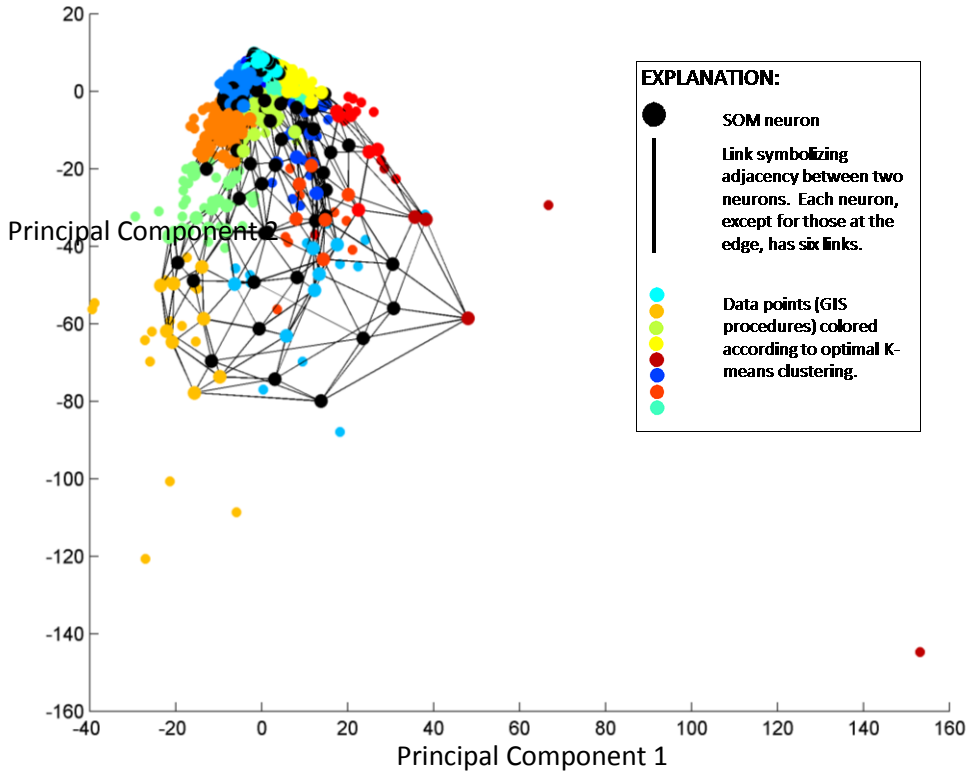


Figure 5-12 Data points and SOM neurons plotted using coordinates of the first two PCA components for the SOM trained with the procedure matrix modified with the Enviro Modeling keywords and parameters derived from Wendel and Buttenfield (2010). This explained 75.63 percent of the variance in the original data.

The Enviro Modeling SOM performed similarly to the Albrecht SOM, although the Albrecht SOM is judged to have more effectively separated and organized GIS procedures because the spatial arrangement of clusters in concentric rings around the region of high uncertainty shows clearer organization. In addition, the arrangement of clusters immediately adjacent to the region of high uncertainty of the Albrecht SOM tends to create a more stable composition of clusters by reducing the within-cluster U-matrix values more effectively. In most other metrics, the Enviro Modeling SOM is slightly inferior (Table 5-9). For example, quantization and topographic error are a little higher (3.12 and 0.12 vs. 2.61 and 0.11) for the Enviro Modeling SOM than for the Albrecht SOM. In addition, the maximum U-matrix values are higher (41.38 vs. 35.20). In all of these statistics, both of the implicit keyword SOMs were substantially poorer than for the explicit keyword SOMs. Despite this, the implicit

keyword SOMs generated clusters that exhibited logical spatial patterns and good within-cluster stability. The coordinates of the implicit keyword procedure matrix, when projected into PCA space, resulted in much improved separation of clusters.

Table 5-9 Errors and U-matrix statistics of the three explicit keyword and two implicit keyword SOMs.

SOM	Error		U-Matrix Statistics			
	Quantization	Topographic	Max	Mean	Median	Stan Dev.
Default	4.285	0.047	18.1341	2.6233	1.7444	2.8526
Optimized	4.363	0.024	85.6366	3.262	1.8202	5.3543
PCA-driven	1.413	0.06	13.2991	2.969	1.995	2.7418
Albrecht	2.6084	0.1111	35.1995	5.1799	3.1615	5.5509
Enviro Modeling	3.1773	0.1165	41.3801	5.9071	3.4117	6.5501

5.3 Summary

This chapter presents details of two variations of the implicit keyword experiment, one based on a set of implicit keywords derived from Albrecht (1999) and the other, termed the Enviro Modeling implicit keywords, based on the author's assessment of types of GIS commands within the set specifically important to environmental modeling. When the standard analyses of the SOM results, including error statistics, hit histograms, and U-matrices are compared with those for the explicit keyword experiments, it is difficult to determine whether using implicit keywords improves the process. The topographic error figures for both implicit keyword SOMs are in fact slightly poorer than for any of the explicit keyword SOMS. Although the quantization error for both the implicit SOMs is better than for the default and optimized SOMs, these figures are poorer than for those of the PCA-driven SOM.

The automated clustering of both the implicit keyword-augmented SOMs, particularly the K-means, shows an interesting shift in the pattern of clusters. Individual clusters are, in some cases, constituted from multiple, spatially-disparate patches of neurons. This spreading of the neuron patches within a cluster could be simply due to the fact that the matrices for both of these SOMs are larger than those for the default or PCA-driven SOM, which allows the data to spread far enough apart to break a

cluster. This could also reflect the detection of sub-types within the cluster. Resolving the meaning of the fragmentation of individual clusters is beyond the scope of this dissertation, but would be interesting to investigate in future.

In all SOM experiments, the Ward's Linkage has resulted in fewer clusters than the K-means algorithm, except in the PCA-driven SOM (where it produced the same number, 9). The impacts of this seem especially strong for the implicit keyword SOMs, where the K-means is producing noticeably more clusters (Table 5-10). One possible explanation for this is that the spatial patterns of neuron signatures in the explicit keyword SOMs is sufficiently constrained in some manner that the two clustering algorithms were not able to differentiate themselves. Within the implicit keyword experiments, the Ward's Linkage results appear to be much less specialized than those of the K-means. Although the selection of clustering method could impact the delineation of clusters, similar patterns are likely to emerge across methods as seen with the K-means and Ward's Linkage analyses. This enables interpretations of cluster semantics that are relatively stable regardless of the clustering method used.

Table 5-10 Numbers of K-means and Ward's Linkage clusters for each SOM.

SOM	K-means	Ward's Linkage
Default	11	9
Optimized	3	1
PCA-driven	9	9
Albrecht	15	6
Enviro Modeling	14	9

Both sets of implicit keywords resulted in increased explanatory power when evaluated through manual analysis of the BMU information and the displays of per-neuron weightings for each implicit keyword (Figure 5-5 and Figure 5-11) relative to the explicit keyword experiments. The clarity of the clusters and the SOM-wide organization of the clusters are more apparent when implicit keywords were used to augment the procedure matrix. The implicit keyword experiments improve on the PCA-driven SOM (judged to be the best of the explicit keyword experiments) by 25% or more for a total explanatory power exceeding 75%. The plots of the coordinates from the PCA in both cases produce a radically

improved spread of both the data points (i.e., the GIS procedures) and of the neurons in the respective SOMs. The separation of the K-means clusters, which were symbolized as colors in the PCA plots, was also greatly improved, although there was definitely still some overlap between data points associated with different clusters.

The next chapter will expand on the meaning of the results from this Chapter and from Chapter 4. In addition, it will account for the impacts of choices made in the execution of both the explicit keyword and the implicit keyword experiments on the quality of the resultant SOMs. It will identify ways in which the experimental design could be improved, including improved ways to analyze the results.

6 Discussion

This chapter is divided into three sections. The first provides a discussion on the effectiveness of each SOM's spatial arrangement of GIS procedures and how well clusters might be inferred from them. The second section will develop a critique of the experimental design and outline possible enhancements that could be made in future. The concluding section will summarize the findings of the dissertation.

6.1 Analysis of Results

This dissertation described a pair of experiments, both of which used the Self-Organizing Map (SOM) neural network technique to organize a large set of GIS procedures using either explicit or implicit keywords into information spaces. More than simply a visualization, this experiment uses SOMs to organize seemingly disparate items such as software modules in a coherent system has important ramifications for discovering, evaluating, and sharing them. The use of SOMs has not been previously reported for the direct analysis of software source code, largely because of the relatively low explicit semantic content of the tokens/words of programming languages. This dissertation examined whether the tokens of a GIS programming language, the individual GIS commands, are in fact meaningful enough to effectively evaluate and organize a set of GIS procedures. In the first explicit keyword experiment, the GIS commands explicitly found within the source code of the GIS procedures were used as keywords by which to carry this out. Three variations of the SOM training process were based on the explicit keywords. The results showed that explicit keywords imposed organizational structures that were generally informative of differences between GIS procedures, especially when those procedures contained frequently invoked GIS commands (e.g., 'calc', 'KILL', 'SELECT'). The three explicit keyword SOMs suffered however from a lack of separation between types of GIS procedures at potential cluster boundaries. Automatically-extracted clusters cannot easily process the mixture of commands in these

areas. The lack of clarity in the results makes it difficult to interpret semantic differences between clusters.

In the second explicit keyword experiment, optimization of the SOM training parameters resulted in a very large matrix. The sparseness of data points across the matrix seriously hampered the use of automated algorithms to define clusters of GIS procedures. One benefit was that individual neurons were no longer matched to sets of GIS procedures that were of apparently different or conflicting types, as was seen in the default SOM. Although this SOM is not suitable for use with K-means, there may be other kinds of analysis (such as hierarchical clustering) that are able to work effectively with the overall distribution of types of GIS procedures across the SOM, which generally made sense. Individual types of neurons tended to be proximate if interceding gaps of empty neurons are discounted. Whether these gaps are important is open to interpretation. The PCA-driven SOM, which reduced the number of dimensions from 148 to 15, was only marginally superior to the comparably-sized default SOM. The result of these experiments is the understanding that although SOMs derived from explicit keywords can produce some meaningful organization, better results are obtainable.

The second part of the experimental pair utilized implicit keyword descriptors not explicitly contained within the GIS procedures. The motivation for this second experiment was to define and demonstrate a relatively simple method for adding new knowledge to the process of creating an information space, and to produce a semantically meaningful view on a set of GIS procedures. In order to demonstrate the utility of this new method, two sets of implicit keywords were developed and used to develop train SOMs. The results were found to be superior to those of the explicit keyword-only (explicit keyword) SOMs. The objective here was not necessarily to establish what is the “best” or “optimal” set of implicit keywords, but rather to demonstrate that differing implicit keywords will elicit

particular and differing semantic patterns in a set of GIS commands. The results for two independently selected implicit keyword sets were found to be superior to those of the explicit keyword SOMs.

The Albrecht and Enviro Modeling SOMs exhibited very similar overall organizations. This was expected to a degree because both sets of implicit keywords ended up being relatively general. It is notable that without the addition of the "non-analytical" types of GIS commands to the Albrecht scheme, the resultant SOM quality would likely have suffered significantly because so many of the GIS procedures in the set were in fact characteristic of such types of GIS tasks (particularly for visualization). In general, the Enviro Modeling SOM seemed to exhibit a crisper separation of types, especially for smaller clusters, but the Albrecht SOM produced a more intuitive arrangement of clusters. Both yielded clusters that showed variations of functionality that was much more subtle than those seen with the explicit keyword SOMs. These clusters showed more consistent internal stability and a more meaningful spatial organization across the SOM than was seen in the explicit keyword experiments.

This method for augmenting the procedure matrix with implicit keyword information is important because it allows the definition of alternate sets of keywords about a set that can be easily integrated into the SOM process (or any other statistical analysis, really). Each set of implicit keywords can be thought of as a characterization of "what's important" according to a different disciplinary or community perspective. The implicit keyword method allows the integration of that community's worldview into clustering processes with minimal effort to create a product that is more meaningful to that community. Accomplishing this by automated clustering methods could produce indices of information archives which are more usable or more cognate for specific communities, which in turn can empower and facilitate software sharing and exchange. The implicit keyword set projects the items to be classified into differentiable frameworks (i.e., SOM realizing different organizations), thus highlighting or prioritizing various aspects of the framework (in comparison with other frameworks).

6.2 Critique of the Experiment

A number of features of both experiments could be improved. These pertain mostly to the analysis of the set to generate the frequencies that were used to populate the explicit keyword procedure matrix. The implicit keyword experiments were, of course, highly sensitive to the definition and evaluation of the implicit keywords, in counterpoint to the sensitivity in the explicit keyword experiments to command frequencies. All three are discussed in the following sections.

6.2.1 Improved Analysis of the Set

The frequency tabulation process could be enhanced with more sophisticated rules. The most valuable enhancement would be tabulating not only the GIS commands within a given GIS procedure's source code, but also associating the frequencies of GIS commands found in all the sub-routines (i.e. other GIS procedures that are invoked by the one being analyzed). This could have a major impact, especially on the characterization of procedures taken from the GIS Weasel (Viger and Leavesley, 2007; Viger, 2008). This package was designed as a cohesive system in an object-oriented fashion in order to promote the reuse of code. The result of this design principle is that all procedures that need to use a command, such as "flow direction," call the GIS procedure that invokes it rather than invoking the GIS command directly, to ensure consistent data management throughout the system). The result is that although many GIS procedures used the result of this GIS command, relatively few actually invoked it and therefore were not weighted to reflect this reality.

In addition, alternate spellings of words should have been but were not summed. For example, the GIS command "CALCULATE" was frequently and legitimately abbreviated as "CALC," and the two frequencies were independently tracked. Although it is difficult to estimate the impact of summing these frequencies on the SOM results, it is known that this combined frequency would be of an increased magnitude that would cause the SOM training process to emphasize the influence of this GIS

command in defining neurons when using explicit keywords. This would likely result in a local peak for this explicit keyword in a relatively small region of the SOM. When training using implicit keywords, there would likely be no impact, because the assignment of implicit keyword values to "CALCULATE" and "CALC" should have resulted in identical vectors (if this heuristic process was applied consistently to the two variants). This means that the combined effect of these two GIS commands on the implicit keyword-augmented procedure matrices would be the same as treating the two forms as independent commands (and the SOM organization would not change).

6.2.2 Defining Implicit Keywords and Assigning Values

The process of setting values for implicit keywords for the GIS commands in the set, a manual task, makes apparent how well the selected implicit keywords correspond to the range of GIS procedures actually found in the set. For instance, while Albrecht's definition of universal analytical GIS functions were designed for all possible commands, the frequency analysis of the set revealed a very large number of GIS commands that pertained to non-analytical functions. As a result, it became apparent that without the addition of the two non-analytical keywords to the Albrecht set, a large number of GIS commands would have had a zero value assigned for all implicit keywords. The expected result would be that all the non-analytical GIS procedures would have likely been associated with a single undifferentiated cluster or region of the SOM, with the organization of the rest of the SOM focusing on differentiating the remaining (analytical) GIS procedures. It could be argued that this actually would have been desirable if the real purpose of the set of implicit keywords was to differentiate analytical functions from each other.

Future efforts could carry out the frequency analysis first and then focus on the definition of implicit keywords. The benefit of this would be that the researcher can avoid defining and evaluating implicit keywords that have no correspondence with the GIS commands actually found in the set being

analyzed. Defining unused or underused implicit keywords is not expected to deteriorate the quality of the results; they just cost effort to set up. For example, “Terrain Analysis” was associated with only four GIS commands but was maintained as an implicit keyword in the SOM training process. Ultimately, this is merely an efficiency tactic. A researcher or user can define their implicit keywords to reflect their world view, without ever looking at the set in advance.

The exercise of setting values (i.e. assigning a 1 or a 0 to a GIS command for a given implicit keyword) also revealed the difficulty in discriminating between Albrecht’s types. This is not really an indictment of those types, but an observation that might help define implicit keywords that are easier to evaluate in future. In the case of the Albrecht keywords, because concepts like connectedness from the “Spatial Analysis” type and the adjacency from the “Measurements” type are so similar to the proximity and nearest-neighbor characteristics of the “Distribution/Neighborhood” type, there was a high degree of overlap between the three. While this is not necessarily a negative factor and, in fact, could serve to emphasize semantically important characteristics, it will tend to create a smaller number of larger clusters in the resultant SOM. “Locational Analysis” also showed a relatively high degree of overlap because the buffer type of command very much relies on the concept of proximity.

The Enviro Modeling implicit keywords were originally envisioned as focusing on organizing GIS procedures for specific uses in environmental modeling. This strategy was adjusted when it became apparent that the keyword set would provide inadequate differentiation of commands. Instead, the Enviro Modeling implicit keyword set was engineered to maximize differentiation of the GIS procedures in the set, and with knowledge of the set gained in the execution of the other four SOMs. This produced what is judged to be the best spatial arrangement of neuron types among the SOMs. To a degree, the Enviro Modeling SOM “cheated” because this *a priori* knowledge is analogous to training a data set used in a supervised classification scheme. One could argue that this fore-knowledge creates an unfair bias in the result. The counter argument is that applying *a priori* knowledge (i.e., training the classification)

demonstrates an upper cap or ceiling on how far the implicit keyword method can go in providing semantically meaningful organization to a set. The argument about unfair bias also misses the point that doing so contrasts with the Albrecht keywords (derived without any benefits of fore-knowledge). Taken together, these two implicit keyword experiments demonstrate the approach defined in this dissertation, namely that using an inductive process of learning about and organizing a body of GIS procedure is sensitive to the inclusion of knowledge in the form of implicit keywords. The Albrecht and Enviro Modeling keyword sets elicit different patterns in the underlying semantic structure as well.

Future research could be carried out to evaluate the sensitivity of the resultant SOM to the definition and evaluation of keywords. For instance, two "non-analytical" keywords were added to the Albrecht set for completeness. It would be interesting to examine how much loss of explanatory power would be exhibited if these were omitted from the set. Further, it would be interesting to carry out a series of experiments to determine which of a set's keywords are the most important for creating a meaningful SOM.

6.2.3 Understanding the Role of Empty Neurons

Empty neurons play an important role in defining boundaries between clusters. In the case of the optimized SOM, there were simply too many empty neurons which essentially caused the clustering algorithms to fail. Across all the SOMs, the region of high uncertainty is centered on one the regions where there are empty cells. It is interesting to note that another region with a relatively large number of empty neurons is at the lower left edge of the Enviro Modeling SOM, around the neuron with a hit frequency of 72 in Figure 5-9. The extremely high frequency has given this single neuron enough weight to influence the neighborhood, making adjacent neuron signatures more closely resemble itself and thereby reducing the U-matrix values. There is a positive-feedback cycle in these areas, in that a neuron with a very strong signature is likely to attract other GIS procedures, which further strengthens the

neuron and starves neighbors of hits. Although authors (such as Wendell and Buttenfield, 2010) have attempted to devise “rules-of-thumb” for determining the appropriate size of the SOM matrix, there is continued potential for helpful research on how best to define this very important SOM training parameter to avoid these issues.

An additional research topic in this area is the appropriate treatment of a procedure matrix with varying numerical magnitudes. While the numbers used in this dissertation could easily have been normalized, the semantic meaning of this action was unclear and was therefore avoided. It appears that relative differences were still handled well by the SOM training process, but that varying magnitudes could have had an impact on clustering algorithms.

6.3 Conclusion

GIS commands that compose GIS procedures, termed *explicit keywords*, were used as input to an unsupervised exploratory data analysis technique (the Self-Organizing Map neural network) to create two-dimensional maps of sets of GIS procedures. Clusters that were automatically derived from these maps were, to a degree, successful in identifying groupings of similar types of GIS procedures. This approach is a useful way to quickly organize a large set of GIS procedures into a generic information space with no input from any user.

The author then designed and implemented an approach that exploits semantically important information that is not ordinarily included in traditional keyword-driven information retrieval approaches. This was done by creating what are termed here as *implicit keywords*, descriptors designed to recognize characteristics not explicitly recorded within the GIS procedure source code. The implicit keyword information was then used to augment the descriptions of GIS procedures, which were then used as input to the same exploratory data analysis technique.

The quality of the resultant SOM maps using two different sets of implicit keywords was

markedly better than the permutations derived using only explicit keywords. This quality was exhibited not only through quantitative metrics, such as the explanatory power of the first two principal components derived from the enhanced descriptions of the GIS procedures (in the *augmented procedure matrix*), but also in the tightness of individual clusters and the separation of adjacent clusters. Further, subjective interpretations found the implicit keyword clusters to be more meaningful to the human observer.

The experiments described in this dissertation answers the original research question as to whether the use of implicit keywords leads to a substantially different and possibly richer organization of GIS procedures in the affirmative. Again, the purpose of this research was not to define the "best" set of implicit keywords for a set of GIS procedures or for all user communities. Rather, the demonstrated approach provides an important mechanism through which individual users or user communities can insert their views on different GIS functionality into the process of organizing large and complex sets of GIS procedures. Depending on the quality of the keywords used to describe the GIS procedures, the result does indeed have the potential to be substantially richer.

The experiments also successfully addressed the stated goals of the research design. A method was devised to automatically create an information space based on a set of GIS procedures, to characterize the similarity between individual GIS procedures and clusters of GIS procedures. Both explicit and implicit keywords were evaluated as input to this process. Both were found to be suitable for organizing GIS procedures, to varying degrees. The implicit keyword experiments compared the results of the organization process to confirm that it (i.e., SOM training) was sensitive to the choice of implicit keywords. This last accomplishment is important because it indicates that users can impact the resultant organization of GIS procedures through these choices. This means that the approach is sufficiently dynamic or responsive to produce information spaces that represent the views or opinions of a user at least in part. The subjective interpretations of the resultant clusters indicate that the

approach does a good job of indicating the similarity between GIS procedures based on the choice of keywords.

With regard to utility of the approach in practice, the final goal of the research design, individuals may choose to develop sets of implicit keywords and use them to characterize GIS procedures. Once the SOM is trained, associating new GIS procedures with a neuron in the map (i.e., a Best Matching Unit) is a quick process of classification that requires no new input from the user. While the number of individuals with sufficient GIS expertise to support this might be limited, whole communities focused on a particular domain might come together to define implicit keywords through a consensus-driven process to better describe their common ideas about GIS. Examples of communities that may have substantially divergent views are soil scientists, hydrologists, social scientists, computer programmers, graphic artists, and land managers. Each could develop their own set of implicit keywords to create alternate organizations of the same set of GIS procedures. A community could release a trained SOM as a reclassification tool as described for the individual user, or could be presented through a web-based tool into which new GIS procedures could be submitted for classification.

In terms of query and retrieval, users might prefer to view a SOM map of the information space and browse for a type of GIS procedure. Tools could be easily implemented to query a neuron in a trained SOM display to report the both the values for the keywords of the neuron and the GIS procedures associated with that neuron. Further, if the SOM had been clustered, then information about the whole cluster could be presented. If the designation of clusters was held relatively static over time, then the community presenting the SOM could attach subjective interpretations to the each of the clusters (as was reported in this dissertation). Alternate visualizations of a SOM could be built to highlight certain characteristics, such as trends in the strength of a certain keyword across the neurons in the map. All of these examples serve to illustrate that there are many different ways to allow humans to interact with a set of GIS procedures, and that different users may prefer different or context-

sensitive displays or tools.

The experiments of this dissertation demonstrated that the use of GIS commands as explicit keywords can produce helpful organizations of GIS procedures. The experiments further demonstrate that implicit keywords can be used to moderate, improve, and specialize the results of the explicit keyword process. The different experiments not only show the differing impacts of applying different keyword schemes, but bear witness also to the fact that GIS functionality can be organized in potentially very different ways with consistent methodological rigor but using different ways to reprioritize specific types of functionality. By facilitating mechanisms for improved software sharing and exchange, the methods described here may in the future enable researchers in the selection of more appropriate procedures for a given task.

References

AAAI, 2008, Natural Language, Association for the Advancement of Artificial Intelligence, web version http://www.aaai.org/AITopics/pmwiki/pmwiki.php/AITopics/NaturalLanguage, November 2008.

Agarwal, P., and Skupin, A., 2008, Self-organising maps : applications in geographic information science: Chichester, England; Hoboken, NJ, Wiley.

Ahalt, S.C., Krishnamurthy, A.K., Chen, P.K., and Melton, D.E., 1990, Competitive learning algorithms for vector quantization: Neural Networks, v. 3, no. 3, p. 277–290.

Ahlberg, C., and Shneiderman, B., 1994, Visual information seeking: tight coupling of dynamic query filters with starfield displays, in Proceedings of ACM Conference on Human Factors in Computer Systems, 24-28 April 1994, Boston, MA, USA, CHI '94 Conference Proceedings. Human Factors in Computing Systems 'Celebrating Interdependence', ACM, p. 313–317.

Albrecht, J., 1994, Universal elementary GIS tasks: beyond low level commands: Advances in GIS Research. Proceedings of the Sixth International Symposium on Spatial Data Handling. Taylor & Francis, Bristol, p. 57–66

Albrecht, J., 1995, Semantic net of universal elementary GIS functions, in AUTO-CARTO 12, Bethesda, Maryland, American Congress on Surveying and Mapping, p. 235–244.

Albrecht, J., 1999, Universal Analytical GIS Operations: a task-oriented systematisation of data structure-independent GIS functionality leading towards a geographic modelling language, *in* Craglia, M., and Onsrud, H., eds., Geographic Information Research: Transatlantic Perspectives London, Taylor and Francis, p. 577–591.

Aldenderfer, M.S., and Blashfield, R.K., 1984, Cluster Analysis: Beverly Hills, California, Sage Publications, Inc., v. 07-044, 87 p.

Ampazis, N., and Perantonis, S.J., 2004, LSISOM - A latent semantic indexing approach to Self-Organizing Maps of document collections: Neural Processing Letters, v. 19, no. 2, p. 157–173.

Arabie, P., 1991, Was Euclid an unnecessarily sophisticated psychologist: Psychometrika, v. 56, no. 4, p. 567–587.

Azcarraga, A.P., Hsieh, M.H., Pan, S.L., and Setiono, R., 2005, Extracting salient dimensions for automatic SOM labeling: Ieee Transactions on Systems Man and Cybernetics Part C-Applications and Reviews, v. 35, no. 4, p. 595–600.

Azcarraga, A.P., Yap, T.N., Tan, J., and Chua, T.S., 2004, Evaluating keyword selection methods for WEBSOM text archives: Ieee Transactions on Knowledge and Data Engineering, v. 16, no. 3, p. 380–383.

Bartell, B.T., Cottrell, G.W., and Belew, R.K., 1992, Latent Semantic Indexing is an optimal special case of multidimensional scaling, in Proceedings of the 15th Annual International ACM SIGIR Conference

on Research and Development in Information Retrieval, Jun 21-24 1992, Copenhagen, Den, SIGIR Forum (ACM Special Interest Group on Information Retrieval), ACM, p. 161–167.

Belkin, N.J., and Croft, W.B., 1992, Information filtering and information retrieval: two sides of the same coin: Communications of the Acm, v. 35, no. 12, p. 29–38.

Berry, J.K., 1989, Fundamental operations in computer-assisted map analysis, Fundamentals of geographic information systems: a compendium: Bethesda, MD, USA, American Society for Photogrammetry and Remote Sensing, p. 81–98.

Bittner, T., Donnelly, M., and Smith, B., 2009, A spatio-temporal ontology for geographic information integration: International Journal of Geographical Information Science, v. 23, no. 6, p. 765-798.

Burrough, P.A., 1992, Development of intelligent geographical information systems: International Journal of Geographical Information Science, v. 6, no. 1, p. 1–11.

Buttenfield, B.P., Viger, R.J., Wendel, J., and Smith, J.M., Characterizing GIS commands using a SOM spatialization (manuscript in preparation).

Callan, J.P., Croft, W.B., and Harding, S.M., 1992, The INQUERY retrieval system, Wien, Austria, DEXA 92. Database and Expert Systems Applications. Proceedings of the International Conference, Springer-Verlag, p. 78–83.

Card, S.K., Robertson, G.G., and Mackinlay, J.D., 1991, The information visualizer, an information workspace, in Human Factors in Computing Systems. Reaching Through Technology. CHI '91. Conference Proceedings, 27 April-2 May 1991, New Orleans, LA, USA, Human Factors in Computing Systems. Reaching Through Technology. CHI '91. Conference Proceedings, ACM, p. 181–188.

Chan, L.M., 1981, Cataloging and classification : an introduction: New York, McGraw-Hill.

Chen, C., and Czerwinski, M., eds., 2000a, Special Issue on Empirical Evaluation of Information Visualisations, International Journal of Human Computer Studies, v. 53 (5).

Chen, C.M., and Czerwinski, M.P., 2000b, Empirical evaluation of information visualizations: an introduction: International Journal of Human-Computer Studies, v. 53, no. 5, p. 631–635.

Chen, M.S., Han, J.W., and Yu, P.S., 1996, Data mining: An overview from a database perspective: Ieee Transactions on Knowledge and Data Engineering, v. 8, no. 6, p. 866–883.

Chomsky, N., 1965, Aspects of the theory of syntax: Cambridge, M.I.T. Press, 261 p.

Chrisman, N., 1999, A transformational approach to GIS operations: International Journal of Geographical Information Science, v. 13, no. 7, p. 617–637.

Cohen, J., and Cohen, P., 1983, Applied multiple regression correlation analysis for the behavioral sciences: Hillsdale, NJ, Erlbaum, 703 p.

Comber, A.J., Fisher, P.F., and Wadsworth, R.A., 2008, Semantics, Metadata, Geographical Information and Users, Transactions in GIS, Wiley-Blackwell, v. 12, p. 287-291.

Cottrell, M., Fort, J.C., and Pages, G., 1998, Theoretical aspects of the SOM algorithm: Neurocomputing, v. 21, no. 1–3, p. 119–138.

Couclelis, H., 1998, Worlds of information: the geographic metaphor in the visualization of complex information: Cartography and Geographic Information System, v. 25, no. 4, p. 209–220.

Dangermond, J., 1982, A classification of software components commonly used in Geographic Information Systems, *in* Peuquet, D., and O'Callaghan, J.F., The Design and Implementation of Computer-Based Geographic Information Systems: Honolulu, Hawaii, U.S. National Science Foundation, v. 1, p. 70–91.

Davies, D.L., and Bouldin, D.W., 1979, Cluster separation measure: Ieee Transactions on Pattern Analysis and Machine Intelligence, v. 1, no. 2, p. 224-227.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., and Harshman, R., 1990, Indexing by Latent Semantic Analysis: Journal of the American Society for Information Science, v. 41, no. 6, p. 391–407.

Dervin, B., 1983, An overview of sense-making research: Concepts, methods, and results to date., in International Communication Association Annual Meeting, Dallas, TX.

Dodge, M., and Kitchin, R., 2000, Mapping cyberspace: London; New York, Routledge.

Douglass, J., 2004, Self-Organizing Maps: A Tourist's Guide to Neural Network (re)Presentation(s), University of California - Santa Barabara, v. 2004, no. December.

ESRI, 2001, ArcInfo Workstation (8.1 ed.).

Fabrikant, S.I., 2000, Spatial metaphors for browsing large data archives: Boulder, University of Colorado, 142 p.

Fabrikant, S.I., 2003, Abstraction and scale in spatialisation, Accessible New Zealand; Capitalising on Contemporary Technologies: Wairakei Resort, Taupo, New Zealand, New Zealand Cartographic Society, p. 35–43.

Fabrikant, S.I., and Buttenfield, B.P., 2001, Formalizing semantic spaces for information access: Annals of the Association of American Geographers, v. 91, no. 2, p. 263–280.

Fabrikant, S.I., and Montello, D.R., 2008, The effect of instructions on distance and similarity judgements in information spatializations: International Journal of Geographical Information Science, v. 22, no. 4, p. 463–478.

Fabrikant, S.I., Montello, D.R., and Mark, D.M., 2006, The distance-similarity metaphor in region-display spatializations: Ieee Computer Graphics and Applications, v. 26, no. 4, p. 34–44.

Fabrikant, S.I., Rebich-Hespanha, S., Andrienko, N., Andrienko, G., and Montello, D.R., 2008, Novel method to measure inference affordance in static small-multiple map displays representing dynamic processes: Cartographic Journal, v. 45, no. 3, p. 201–215.

Fabrikant, S.I., and Skupin, A., 2005, Cognitively plausible information visualization, *in* Dykes, J., MacEacheran, A.M., and Kraak, J.-M., eds., Exploring geovisualization: Amsterdam, Elsevier.

Farley, B.G., and Clark, W.A., 1954, Simulation of self-organizing systems by digital computer: Ire Transactions on Information Theory, no. 4, p. 76–84.

Feiner, S., and Beshers, C., 1990, Worlds within worlds: metaphors for exploring n-dimensional virtual worlds, in UIST. Third Annual Symposium on User Interface Software and Technology. Proceedings of the ACM SIGGRAPH Symposium, 3-5 Oct. 1990, Snowbird, UT, USA, UIST. Third Annual Symposium on User Interface Software and Technology. Proceedings of the ACM SIGGRAPH Symposium, ACM, p. 76–83.

Foody, G.M., and Arora, M.K., 1997, An evaluation of some factors affecting the accuracy of classification by an artificial neural network: International Journal of Remote Sensing, v. 18, no. 4, p. 799–810.

Furnas, G.W., Landauer, T.K., Gomez, L.M., and Dumais, S.T., 1987, The vocabulary problem in human-system communications: Communications of the ACM, v. 30, p. 964–971.

Gahegan, M., 1996, Specifying the transformations within and between geographic data models: Transactions in GIS, v. 1, no. 2, p. 137–152.

Gahegan, M., 1999, Guest Editorial: What is Geocomputation?: Transactions in GIS, v. 3, no. 3, p. 203–206.

Gahegan, M., 2000, On the application of inductive machine learning tools to geographical analysis: Geographical Analysis, v. 32, no. 2, p. 113–139.

Gahegan, M., 2003, Is inductive machine learning just another wild goose (or might it lay the golden egg)?: International Journal of Geographical Information Science, v. 17, no. 1, p. 69–92.

Gahegan, M., Takatsuka, M., Wheeler, M., and Hardisty, F., 2002, Introducing GeoVISTA Studio: an integrated suite of visualization and computational methods for exploration and knowledge construction in geography: Computers, Environment, and Urban Systems, v. 26, p. 267–292.

Gan, G., Ma, C., and Wu, J., 2007, Data clustering : theory, algorithms, and applications: Philadelphia, Pa.; Alexandria, Va., SIAM, Society for Industrial and Applied Mathematics ; American Statistical Association, 466 p.

Gershon, N., and Eick, S.G., 1998, Guest editors' introduction: Information visualization. The next frontier: Journal of Intelligent Information Systems, v. 11, no. 3, p. 199–204.

Giordano, A., 1984, A conceptual model of GIS-based spatial analysis: Cartographica, v. 31, no. 4, p. 44–57.

Goldstone, R.L., 1994, Similarity, interactive activation, and mapping: Journal of Experimental Psychology-Learning Memory and Cognition, v. 20, no. 1, p. 3–27.

Goodchild, M.F., 1987, Towards an Enumeration and Classification of GIS Functions, Proceedings of the International GIS Symposium: Crystal City, VA, v. II, p. 67–79.

Goodchild, M.F., and Brusegard, 1989, Spatial analysis using GIS, in AM/FM International Conference XII, New Orleans, Louisiana, National Center for Geographic Information and Analysis.

Gould, P., 1970, Is Statistix Inferens the Geographical Name for a Wild Goose?: Economic Geography, v. 46, p. 439–448.

Guptill, S.C., 1988, A process for evalutating geographic information systems: U.S. Geological Survey 88-105 [Open File].

Hadzilacos, T., 1996, On layer-based systems for undetermined boundaries, *in* Burrough, P.A., and Frank, A.U., eds., Geographic Objects with Indeterminate Boundaries: London, Taylor & Francis, p. 237–255.

Hartley, A.A., 1977, Mental measurement in magnitude estimation of length: Journal of Experimental Psychology-Human Perception and Performance, v. 3, no. 4, p. 622–628.

Havre, S., Hetzler, B., and Nowell, L., 2000, ThemeRiver: visualizing theme changes over time, in Proceedings of IEEE Visualization 2000, 9-10 Oct. 2000, Salt Lake City, UT, USA, IEEE Symposium on Information Visualization 2000. INFOVIS 2000. Proceedings, IEEE, p. 115–123.

Hill, H.H., 2008, Formal Language, *Glossary of Terms for Advanced Symbolic Logic*, web version http://cstl-cla.semo.edu/hill/pl330/asl%20glossary.htm#f, November, 2008

Hotelling, H., 1933, Analysis of a complex of statistical variables into principal components: Journal of Educational Psychology, v. 24, p. 417–441.

Hotho, A., Jaschke, R., Schmitz, C., and Stumme, C., 2006, Information retrieval in folksonomies: Search and ranking, *in* Sure, Y., and Domingue, J., eds., Semantic Web: Research and Applications, Proceedings: Berlin, Springer-Verlag Berlin, p. 411-426.

Howe, D., undated, Procedural Language, *The Free On-line Dictionary of Computing*, web version http://dictionary.reference.com/browse/procedural language, January 2009.

Huxhold, W.E., 1991, An introduction to urban geographic information systems: New York, Oxford University Press.

Jackendoff, R., 1992, Languages of the Mind: Essays on Mental Representation: Cambridge, MA, MIT Press.

Jardine, N., and van Rijsbergen, C.J., 1971, Use of hierarchic clustering in information retrieval: Information Storage and Retrieval, v. 7, no. 5, p. 217–240.

Johnson, B., and Shneiderman, B., 1991, Tree-maps: a space-filling approach to the visualization of hierarchical information structures, in Proceedings Visualization '91, 22-25 Oct. 1991, San Diego, CA, USA, Proceedings Visualization '91 (Cat. No.91CH3046-0), IEEE, p. 284–291.

Kaski, S., Honkela, T., Lagus, K., and Kohonen, T., 1998a, WEBSOM - Self-organizing maps of document collections: Neurocomputing, v. 21, no. 1–3, p. 101–117.

Kaski, S., Kangas, J., and Kohonen, T., 1998b, Bibliography of self-organizing map (SOM) papers: 1981-1997: Neural Computing Surveys, v. 1, no. 3&4, p. 1–176.

Kohonen, T., 1988, An introduction to neural computing: Neural Networks, v. 1, no. 1, p. 3–16.

Kohonen, T., 1993, Things you haven't heard about the self-organizing map, in 1993 IEEE International Conference on Neural Networks, Piscataway, NJ, USA, IEEE, p. 1147–1156.

Kohonen, T., 1998, The self-organizing map: Neurocomputing, v. 21, no. 1–3, p. 1–6.

Kohonen, T., 2001, Self-organizing maps (3rd ed ed.): Berlin; New York, Springer, v. 30, 501 p.

Kotz, S., Johnson, N.L., and Read, C.B., 1988, Encyclopedia of Statistical Sciences.

Kruskal, J.B., Wish, M., and NetLibrary Inc., 1978, Multidimensional scaling: Beverly Hills, Calif., Sage Publications, 93 p.

Kuhn, W., and Blumenthal, B., 1996, Spatialization: spatial metaphors for user interfaces, Conference companion on Human factors in computing systems: common ground: Vancouver, British Columbia, Canada, ACM.

Lagus, K., and Kaski, S., 1999, Keyword selection method for characterizing text document maps, London, UK, ICANN99. Ninth International Conference on Artificial Neural Networks (IEE Conf. Publ. No.470), IEE, p. 371–376.

Lagus, K., Kaski, S., and Kohonen, T., 2004, Mining massive document collections by the WEBSOM method: Information Sciences, v. 163, no. 1–3, p. 135–156.

Longley, P., Goodchild, M., Maguire, D.J., and Rhind, D.W., 2001, Geographic Information Systems and Science: Chichester, Wiley & Sons, 454 p.

Lyman, P., and Varian, H.R., 2003, How much information (http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/), University of California at Berkeley, accessed December 27, 2008.

Mackinlay, J.D., Robertson, G.G., and Card, S.K., 1991, The Perspective Wall: detail and context smoothly integrated, in Human Factors in Computing Systems. Reaching Through Technology. CHI '91. Conference Proceedings, 27 April-2 May 1991, New Orleans, LA, USA, Human Factors in Computing Systems. Reaching Through Technology. CHI '91. Conference Proceedings, ACM, p. 173–179.

Marchionini, G., 1995, Information-seeking in electronic environments: Cambridge, UK, Cambridge University Press, v. 9.

Maudlin, M.L., 1991, Conceptual Information Retrieval: Dordrecht, The Netherlands, Kluwer Academic Publishers.

McCulloch, W.S., and Pitts, W., 1943, A logical calculus of the ideas immanent in nervous activity: Bulletin of Mathematical Biophysics, v. 5, p. 115–133.

Merkl, D., 1998, Text classification with self-organizing maps: Some lessons learned: Neurocomputing, v. 21, no. 1–3, p. 61–77.

Miller, H.J., and Han, J., 2001, Geographic Data Mining and Knowledge Discovery - An overview, Geographic Data Mining and Knowledge Discovery: CRC Press, p. 3–32.

Miller, N.E., Wong, P.C., Brewster, M., and Foote, H., 1998, TOPIC ISLANDS - a wavelet-based text visualization system, in Proceedings of the 1998 IEEE Visualization Conference, Oct 18-23 1998, Research Triangle Park, NC, USA, Proceedings of the IEEE Visualization Conference, IEEE, p. 189–196.

Mitev, N., Venner, G., and Walker, S.E., 1985, Designing an Online Public Access Catalogue: Okapi, a Catalogue on a Local Area Network: British Library 39 [Library and Information Research Report].

Montello, D.R., Fabrikant, S.I., Ruocco, M., and Middleton, R.S., 2003, Testing the first law of cognitive geography on point-display spatializations, *in* Kuhn, W., Worboys, M., and Timpf, S., eds., Spatial Information Theory, Proceedings - Foundations of Geographic Information Science: Berlin, Springer-Verlag Berlin, p. 316–331.

Morse, E., Lewis, M., and Olsen, K.A., 2000, Evaluating visualizations: using a taxonomic guide: International Journal of Human-Computer Studies, v. 53, no. 5, p. 637–662.

Nguyen, N.T., Jo, G.S., Howlett, R.J., and Jain, L.C., 2008, Agent and Multi-Agent Systems: Technologies and Applications, Berlin, Heidelberg, Springer Berlin Heidelberg.

Oja, M., Kaski, S., and Kohonen, T., 2003, Bibliography of self-organizing map (SOM) papers: 1998-2001 addendum: Neural Computing Surveys, v. 3, no. 1, p. 1–156.

Openshaw, S., and Openshaw, C., 1997, Artificial intelligence in geography: Chichester; New York, Wiley.

Pascoe, R.T., and Penny, J.P., 1995, Constructing Interfaces between (and within) Geographical Information-Systems: International Journal of Geographical Information Systems, v. 9, no. 3, p. 275–291.

Pearl, J., 1988, Probabilistic reasoning in intelligent systems: networks of plausible inference: San Mateo, Calif., Morgan Kaufmann Publishers.

Pearson, K., 1901, On lines and planes of closest fit to systems of points in space: Philosophical Magazine, v. 2, no. 6, p. 559–572.

Peng, Z.R., 2005, A proposed framework for feature-level geospatial data sharing: a case study for transportation network data: International Journal of Geographical Information Science, v. 19, no. 4, p. 459-481.

Pike, W., and Gahegan, M., 2003, Constructing semantically scalable cognitive spaces, Spatial Information Theory, Proceedings: Berlin, Springer-Verlag, p. 332–348.

Reitsma, F., Laxton, J., Ballard, S., Kuhn, W., and Abdelmoty, A., 2009, Semantics, ontologies and eScience for the geosciences: Computers & Geosciences, v. 35, no. 4, p. 706-709.

Rhind, D.W., and Green, N.P.A., 1988, Design of a geographical information system for a heterogeneous scientific community: International Journal of Geographical Information Science, v. 2, no. 2, p. 171–189.

Robertson, G.G., and Mackinlay, J.D., 1993, Document lens, in Proceedings of the 6th Annual Symposium on User Interface Software and Technology, Nov 3-5 1993, Atlanta, GA, USA, Proceedings of the ACM Symposium on User Interface Software and Technology, Publ by ACM, New York, NY, USA, p. 101.

Robertson, G.G., Mackinlay, J.D., and Card, S.K., 1991, Cone Trees: animated 3D visualizations of hierarchical information, in Human Factors in Computing Systems. Reaching Through Technology. CHI '91. Conference Proceedings, 27 April-2 May 1991, New Orleans, LA, USA, Human Factors in Computing Systems. Reaching Through Technology. CHI '91. Conference Proceedings, ACM, p. 189–194.

Robertson, S.E., 1977, The probability ranking principle in IR: Journal of Documentation, v. 33, no. 4, p. 294–304.

Robertson, S.E., 1997, Overview of the Okapi projects: Journal of Documentation, v. 53, no. 1, p. 3–7.

Rolleke, T., 1999, POOL: Probabilistics object-oriented logical representation and retrieval of complex objects; a model for hypermedia retrieval: Dortmund, Univeristy of Dortmund, PhD.

Rosenblatt, F., 1958, The perceptron - a probabilistic model for information storage and organization in the brain: Psychological Review, v. 65, no. 6, p. 386–408.

Sakharov, A., 2008, Formal Language, *MathWorld--A Wolfram Web Resource*, web version http://mathworld.wolfram.com/FormalLanguage.html, November 2008.

Salton, G., 1971, The SMART retrieval system: experiments in automatic document processing: Englewood Cliffs, N.J., Prentice-Hall, xix, 556 p.

Salton, G., 1989, Automatic text processing : the transformation, analysis, and retrieval of information by computer: Reading, Mass., Addison-Wesley, xiii, 530 p.

Salton, G., and Buckley, C., 1988, Term-weighting approaches in automatic text retrieval: Information Processing & Management, v. 24, no. 5, p. 513–523.

Salton, G., and McGill, M.J., 1983, Introduction to modern information retrieval: New York, McGraw-Hill, xv, 448 p.

Schwering, A., 2008, Approaches to Semantic Similarity Measurement for Geo-Spatial Data: A Survey: Transactions in GIS, v. 12, no. 1, p. 5-29.

Sharma, R., Poole, D., and Smyth, C., 2010, A framework for ontologically-grounded probabilistic matching: International Journal of Approximate Reasoning, v. 51, no. 2, p. 240-262.

Sheth, A.P., and Larson, J.A., 1990, Federated database systems for managing distributed, heterogeneous, and autonomous databases: Computing Surveys, v. 22, no. 3, p. 183–236.

Shneiderman, B., 1998, Desiging user interfaces: Strategies for effective human-computer interaction: Reading, MA, Addison-Wesley.

Skinner, H.A., 1978, Differentiating the contribution of elevation, scatter and shape in profile similarity: Educational and Psychological Measurement, v. 38, no. 2, p. 297–308.

Skupin, A., 1998, Organizing and visualizing hypermedia information spaces: Buffalo, SUNY-Buffalo, 155 p.

Skupin, A., 2002a, A cartographic approach to visualizing conference abstracts: IEEE Computer Graphics and Applications, v. 22, no. 1, p. 50–58.

Skupin, A., 2002b, On geometry and transformation in map-like information visualization, Visual Interfaces to Digital Libraries: Berlin, Springer-Verlag Berlin, p. 161–170.

Skupin, A., 2008, Spatialization, *in* Kemp, K.K., ed., Encyclopedia of Geographic Information Sciences: Thousand Oaks, CA, Sage Publications, p. 418-422.

Skupin, A., and Buttenfield, B.P., 1996, Spatial Metaphors For Visualizing Very Large Data Archives, GIS/LIS '96 Annual Conference and Exposition Proceedings: Denver, Colorado, American Society for Photogrammetry and Remote Sensing, p. 607–617.

Skupin, A., and Fabrikant, S.I., 2003a, Spatialization methods: A cartographic research agenda for non-geographic information visualization, in Cartography and Geographic Information Science.

Skupin, A., and Fabrikant, S.I., 2003b, Spatialization methods: A cartographic research agenda for non-geographic information visualization: Cartography and Geographic Information Science, v. 30, no. 2, p. 99–119.

Soboroff, I., 2002, Information Retrieval (online class materials), University of Maryland - Baltimore, accessed Jan 3, 2009.

Sparck-Jones, K., 1972, A statistical interpretation of term specificity and its application in retrieval: Journal of Documentation, v. 28, no. 1, p. 11–21.

Talmy, L., 1983, How language structures space, *in* Pick, H.L., and Acredolo., L.P., eds., Spatial Orientation: Theory, Research and Application: New York, Plenum, p. 225–282.

Tanasescu, V., 2007, Spatial semantics in difference spaces, *in* Winter, S., Duckham, M., Kulik, L., and Kuipers, B., COSIT '07 Proceedings of the 8th international conference on Spatial information theory Melbourne, Australia, Springer-Verlag Berlin, Heidelberg, v. LNCS 4736, p. 96-115.

Tangsripairoj, S., and Samadzadeh, M.H., 2006, Organizing and visualizing software repositories using the growing hierarchical self-organizing map: Journal of Information Science and Engineering, v. 22, no. 2, p. 283–295.

Tobler, W., 1979, A transformational view of cartography: The American Cartographer, v. 6, p. 101–106.

Tobler, W.R., 1970, A computer movie simulating urban growth in the Detroit region: Economic Geography, v. 46, no. 2, p. 234–240.

Tomlin, C.D., 1983, Digital Cartographic Modeling Techniques in Environmental Planning (unpublished), Yale University.

Tomlinson, R.F., and Boyle, R., 1981, The State of Development of Systems for Handling Natural Resources Inventory Data: Cartographica, v. 18, no. 4, p. 65–95.

Torgerson, W.S., 1952, Multidimensional scaling .1. Theory and method: Psychometrika, v. 17, no. 4, p. 401–419.

Tsichritzis, D., and Klug, A., 1978, The ANSI/X3/SARC DBMS framwork: Information Systems, v. 3, no. 4.

Tukey, J.W., 1977, Exploratory data analysis: Reading, Mass., Addison-Wesley Pub. Co., xvi, 688 p.

Tversky, A., 1977, Features of similarity: Psychological Review, v. 84, no. 4, p. 327–352.

Tversky, B., 1981, Distortions in memory maps: Cognitive Psychology, v. 13, p. 407–433.

Tversky, B., 1993, Cognitive maps, cognitive collages, and spatial mental models, *in* Frank , A., and Campari, I., eds., Spatial Information Theory: A Theoretical Basis for GIS. Lecture Notes in Computer Science.: New York, Springer-Verlag, p. 14–24.

Tversky, B., and Hemenway, K., 1983, Categories of environmental scenes: Cognitive Psychology, v. 15, p. 121–149.

Tversky, B., and Hemenway, K., 1984, Objects, parts, and categories: Journal of Experimental Psychology, v. 113, no. 2, p. 169–193.

Tversky, B., and Lee, P.U., 1998, How space structures language, *in* C. Freksa, C.H.a.K.F.W., ed., Spatial Cognition: An Interdisciplinary Approach to Representing and Processing Spatial Knowledge: Berlin, Springer-Verlag, p. 157–117.

Tversky, B., and Teiffer, E., 1976, Development of strategies for recall and recognition: Developmental Psychology, v. 12, no. 5, p. 406–410.

Ultsch, A., and Siemon, H.P., 1990, Kohonen's self organizing feature maps for exploratory data analysis, Proceedings of INNC'90, International Neural Network Conference: Dordrecht, Netherlands, Kluwer, p. 305-308.

van Rijsbergen, C.J., 2000, Getting into Information Retrieval, *in* Agosti, M., Crestani, F., and Pasi, G., eds., Lecture notes in Computer Science: Heidelberg, Springer-Verlag, p. 21–50.

Viger, R.J., 2004, GEOLEM: Improving the Integration of Geographic Information in Environmental Modeling Through Semantic Interoperability: Boulder, University of Colorado, Master's, 135 p.

Viger, R.J., 2008, The GIS Weasel: an interface for the treatment of geographic information in modeling: Computers & Geosciences, v. 34, no. 8, p. 891–901.

Viger, R.J., and Leavesley, G.H., 2007, The GIS Weasel user's manual: U.S. Geological Survey Book 6, chap. B4 [Techniques and Methods].

Wade, T., and Sommer, S.E., eds., 2006, A to Z GIS: An Illustrated Dictionary of Geographic Information Systems: Redlands, California, ESRI Press, 288 p.

Wendel, J., and Buttenfield, B.P., 2010, Formalizing Guidelines for Building Meaningful Self-Organizing Maps in GIScience 2010—Sixth International Conference on Geographic Information Science, Zurich, Switzerland, p. 6.

Wendel, J., Buttenfield, B.P., Viger, R.J., and Smith, J.M., 2008a, Characterizing GIS hydrology commands: spatialization with SOM Mapping, in GIScience 2008—Fifth International Conference on Geographic Information Science, Park City, Utah.

Wendel, J., Smith, J.M., Buttenfield, B.P., and Viger, R.J., 2008b, Characterizing GIS commands: spatialization with Self-Organizing Maps (SOMs), in American Association of Geographers 2008 Annual Meeting, Boston, Massachussetts, American Association of Geographers.

Wikimedia Foundation, 2008a, Library classification, *Wikipedia The Free Encyclopedia*, web version http://en.wikipedia.org/wiki/Library_classification, November 2008.

Wikimedia Foundation, 2008b, Natural Language, *Wikipedia The Free Encyclopedia*, web version http://en.wikipedia.org/wiki/Natural_language, November 2008.

Wise, J.A., 1999, The ecological approach to text visualization: Journal of the American Society for Information Science, v. 50, no. 13, p. 1224–1233.

Ye, H., and Lo, B.W.N., 2001, Towards a self-structuring software library: IEE Proceedings-Software, v. 148, no. 2, p. 45–55.

Young, F.W., 1983, Multidimensional scaling, Encyclopedia of Statistical Sciences, John Wiley & Sons, Inc, v. 5.

Young, F.W., and Lewyckyj, R., 1987, ALSCAL: User's Guide: L.L. Thurstone Psychometric Laboratory, University of North Carolina.

Zahedi, F., 1991, An introduction to neural networks and a comparison with artificial intelligence and expert systems: Interfaces, v. 21, no. 2, p. 25–38.

Appendix A. Albrecht Implicit Keyword Matrix and Modified Procedure Matrix

This appendix contains two tables. The first, Table A-1, lists the values given for the Albrecht implicit keywords for each GIS command. The second, Table A-2, shows the result of augmenting the default procedure matrix with the Albrecht implicit keywords. Within Table A-1, each GIS command is shown on a row. The GIS commands are ordered alphabetically. After the first column, the columns correspond to Albrecht implicit keywords as labeled. Section 5.1 provided the rationale for defining the Albrecht implicit keywords. Table A-1 shows the values that the author heuristically assigned for these keywords to each GIS command.

Table A-1 Matrix showing the assignment of values to GIS commands for the Albrecht set of implicit keywords.

GIS Command	Search	Location Analysis	Terrain Analysis	Distribution and Neighborhood	Spatial Analysis	Measurements	Viz	Enter Data
^	1	0	0	0	0	1	0	0
+	1	0	0	0	0	1	0	0
=	1	0	0	0	0	1	0	1
ADD	0	0	0	0	0	0	0	1
ADDITEM	0	0	0	0	0	0	0	1
ANNOSIZE	0	0	0	0	0	0	1	0
AP	0	0	0	0	0	0	1	0
ARC	0	0	0	0	0	0	0	0
ARCS	0	0	0	0	0	0	1	0
ARCSNAP	0	1	0	1	1	1	0	0
ARROWSIZE	0	0	0	0	0	0	1	0
ARROWTYPE	0	0	0	0	0	0	1	0
ASCONNECT	0	0	0	0	0	0	0	1
ASELECT	1	0	0	0	0	0	0	0
ASEXECUTE	0	0	0	0	0	0	0	1
ASEXECUTE	0	0	0	0	0	0	0	1
AXIS	0	0	0	0	0	0	1	0
BACK-ENVIRONMENT	0	0	0	0	0	0	1	0
BACK-SYMBOLITEM	0	0	0	0	0	0	1	0
BOX	0	0	0	0	0	0	1	0
CALC	0	0	0	0	0	1	0	1
CALCULATE	0	0	0	0	0	1	0	1
CIRCLE	0	0	0	0	0	0	1	0
CLASS	1	0	0	0	0	0	0	1
CLEARSELECT	1	0	0	0	0	0	0	0
COMBINE(	1	1	0	1	1	0	0	0
CON(	1	0	0	1	0	1	0	0
COORDINATE	0	0	0	0	0	0	0	1
COPY	0	0	0	0	0	0	0	1

CREATE	0	0	0	0	0	0	0	1
CURSOR	1	0	0	0	0	0	0	1
DBMSCURSOR	1	0	0	0	0	0	0	1
DBMSEXECUTE	1	0	0	0	0	0	0	1
DEFINE	0	0	0	0	0	0	0	1
DESCRIBE	1	0	0	0	0	0	0	0
DISPLAY	0	0	0	0	0	0	1	0
DRAW-ENVIRONMENT	0	0	0	0	0	0	1	0
EDIT	0	0	0	0	0	0	0	1
EDITDISTANCE	1	1	0	1	1	1	0	0
EDITFEATURE	1	0	0	0	0	0	0	0
EVENT-SOURCE	1	0	0	0	0	0	0	1
GRAIN	1	1	0	1	1	1	0	0
GRAPHICS	0	0	0	0	0	0	1	0
GRIDSHADES	0	0	0	0	0	0	1	0
INFOFILE	0	0	0	0	0	0	0	1
INT(	1	0	0	0	0	1	0	0
INTERSECTARCS	1	1	0	1	1	0	0	0
ISNULL(	1	0	0	0	0	1	0	0
ITEMS	0	0	0	0	0	0	0	1
KEYAREA	0	0	0	0	0	0	1	0
KILL	0	0	0	0	0	0	0	1
LAYER	1	0	0	0	0	0	1	1
LAYERDRAW	0	0	0	0	0	0	1	0
LEADERARROWS	0	0	0	0	0	0	1	0
LEADERS	0	0	0	0	0	0	1	0
LEADERSYMBOL	0	0	0	0	0	0	1	0
LEADER-TOLERANCE	0	0	0	0	0	0	1	0
LIBRARY	1	0	0	0	0	0	0	1
LINE	0	0	0	0	0	0	1	0
LINECOLOR	0	0	0	0	0	0	1	0
LINECOPY	0	0	0	0	0	0	1	0
LINEDELETE	0	0	0	0	0	0	1	0
LINEDELETE-LAYER	0	0	0	0	0	0	1	0
LINEPEN	0	0	0	0	0	0	1	0
LINESET	0	0	0	0	0	0	1	0
LINESIZE	0	0	0	0	0	0	1	0
LINESYMBOL	0	0	0	0	0	0	1	0
LINETYPE	0	0	0	0	0	0	1	0
LIST	0	0	0	0	0	0	1	1
LISTOUTPUT	0	0	0	0	0	0	1	1
MAPEXTENT	1	0	0	0	0	0	1	0
MAPLIMITS	1	0	0	0	0	0	1	0
MAPPOSITION	0	0	0	0	0	0	1	0
MAP-PROJECTION	0	1	0	0	0	1	1	0
MAPSCALE	0	1	0	0	0	1	1	0
MARKER	0	0	0	0	0	0	1	0
MARKER-COLOR	0	0	0	0	0	0	1	0
MARKERCOPY	0	0	0	0	0	0	1	0
MARKER-DELETE	0	0	0	0	0	0	1	0
MARKERMASK	0	0	0	0	0	0	1	0
MARKERSET	0	0	0	0	0	0	1	0
MARKERSIZE	0	0	0	0	0	0	1	0
MARKER-SYMBOL	0	0	0	0	0	0	1	0
MOVE	1	1	0	1	0	0	0	0
NODECOLOR	0	0	0	0	0	0	1	0

NODESNAP	1	1	0	1	1	0	0	0
NSELECT	1	0	0	0	0	0	0	0
OVERFLOW	0	0	0	0	0	0	1	0
OVERPOST	0	0	0	0	0	0	1	0
PAGEEXTENT	0	0	0	0	0	0	1	0
PAGESIZE	0	0	0	0	0	0	1	0
PAGEUNITS	0	0	0	0	0	0	1	0
PATCH	0	0	0	0	0	0	1	0
POLYGON-SHADES	0	0	0	0	0	0	1	0
READSELECT	1	0	0	0	0	0	0	1
RECLASS(	1	0	0	0	0	0	0	1
RELATE	1	0	0	0	0	0	0	0
REMOVEEDIT	0	0	0	0	0	0	0	1
RENAME	0	0	0	0	0	0	0	1
RESELECT	1	0	0	0	0	0	0	0
SAVE	0	0	0	0	0	0	0	1
SCALAR(	1	0	0	0	0	1	0	0
SEARCH-TOLERANCE	1	1	0	0	1	0	0	0
SELECT	1	0	0	0	0	0	0	0
SETCELL	1	0	0	0	0	1	0	0
SETMASK	1	0	0	0	0	0	0	0
SETWINDOW	1	0	0	0	0	0	0	0
SHADECOLOR	0	0	0	0	0	0	1	0
SHADECOPY	0	0	0	0	0	0	1	0
SHADEDELETE	0	0	0	0	0	0	1	0
SHADEOFFSET	0	0	0	0	0	0	1	0
SHADESET	0	0	0	0	0	0	1	0
SHADE-SYMBOL	0	0	0	0	0	0	1	0
SHADETYPE	0	0	0	0	0	0	1	0
SIN(	1	0	0	0	0	1	0	0
SNAP-COVERAGE	1	1	0	0	1	1	0	0
SNAPPING	1	1	0	0	1	1	0	0
SORT	1	0	0	0	0	0	0	1
STATISTICS	0	0	0	0	1	1	0	0
SURFACE-DRAPE	0	0	1	0	1	0	1	0
SURFACEOBSERVER	0	0	1	0	1	0	1	0
SURFACETARGET	0	0	1	0	1	0	1	0
SURFACEVIEW-FIELD	0	0	1	0	1	0	1	0
SYMBOLITEM	0	0	0	0	0	0	1	0
TEXT	0	0	0	0	0	0	1	0
TEXTANGLE	0	0	0	0	0	0	1	0
TEXTCOLOR	0	0	0	0	0	0	1	0
TEXTCOPY	0	0	0	0	0	0	1	0
TEXT-DIRECTION	0	0	0	0	0	0	1	0
TEXTFONT	0	0	0	0	0	0	1	0
TEXT-JUSTIFICATION	0	0	0	0	0	0	1	0
TEXTMASK	0	0	0	0	0	0	1	0
TEXTOFFSET	0	0	0	0	0	0	1	0
TEXTQUALITY	0	0	0	0	0	0	1	0
TEXTSET	0	0	0	0	0	0	1	0
TEXTSIZE	0	0	0	0	0	0	1	0
TEXTSTYLE	0	0	0	0	0	0	1	0
TEXTSYMBOL	0	0	0	0	0	0	1	0
TRANSACTION	0	0	0	0	0	0	0	1
UNLOAD	0	0	0	0	0	0	0	1
UNSELECT	1	0	0	0	0	0	0	0

WEEDTOLERANCE	1	1	0	1	1	0	0	0
WINDOWS	0	0	0	0	0	0	1	0
WRITESELECT	1	0	0	0	0	0	0	1
ZONALMAJORITY(	1	1	0	0	1	1	0	0
ZONALMAX(	1	1	0	0	1	1	0	0
ZONALMEAN(	1	1	0	0	1	1	0	0
ZONALSTATS(	1	1	0	0	1	1	0	0

Table A-2 shows the procedure matrix after it was modified according to the Albrecht implicit keyword matrix (Table A-1). The process by which this modification was carried out was specified generically in Section 3.3.1. A specific example was also given in Section 5.1.1. Briefly, the number of times each explicit keyword (i.e. GIS command) was found within each GIS procedure was determined. The vector of Albrecht implicit keyword values for that explicit keyword (given in Table A-1) is multiplied by this frequency. The vectors for all explicit keywords within a GIS procedure were then summed to produce a row within Table A-2, which describes the GIS procedures using the Albrecht implicit keywords.

Table A-2 Procedure matrix after modification according to the Albrecht implicit keyword matrix (Table A-1). This table was used to generate the Albrecht SOM presented in Chapter 5.

GIS Procedure	Search	Location Analysis	Terrain Analysis	Distribution and Nbrhood	Spatial Analysis	Measure-ments	Viz	Enter Data
addalias	3	3	0	1	2	0	0	2
addedit	75	13	0	5	8	0	19	33
addgenerate	4	1	0	1	1	5	0	6
addinput	2	0	0	0	0	2	0	0
additem	0	0	0	0	0	0	0	2
addmeasure	6	6	0	0	6	0	4	0
addresscreate	3	0	0	0	0	3	0	0
addressenv	0	0	0	0	0	0	0	0
addressfile	4	0	0	0	0	1	0	3
address_select	89	6	0	0	6	0	47	32
add_select	1	0	0	0	0	0	0	1
adrggrid	0	0	0	0	0	0	0	0
aeclean	4	4	0	4	4	4	0	0
aedefaults	4	5	0	4	4	3	11	0
aedriver	4	0	0	0	0	2	3	12
aesymset	0	0	0	0	0	0	4	0
ae_library	4	0	0	0	0	0	0	4
ae_project	4	3	0	0	0	4	6	1
aggregate	23	0	0	0	1	9	0	17
allocation	20	0	0	0	0	4	15	0
annoadd	4	0	0	0	0	8	0	10
annoaddenv	4	0	0	0	0	4	5	4
annoedit	1	0	0	0	0	28	0	28
annoposline	0	0	0	0	0	0	0	0
annopospoint	0	0	0	0	0	0	0	0
aoi_delineation-auto	1	0	0	1	0	1	1	3
aoi_delineation-pre	5	0	0	4	0	5	0	2
aoi_delineation-seed	2	0	0	1	0	1	1	3
append	1	0	0	0	0	1	0	1
ap_delin	0	0	0	0	0	0	19	0
ap_graph	1	1	0	1	0	0	12	0
ap_mru-num	2	0	0	0	0	0	15	0

ap_mru-select	2	0	0	0	0	0	15	0
arcdfad	0	0	0	0	0	0	0	0
arcdime	0	0	0	0	0	0	0	0
arcdlg	0	0	0	0	0	0	0	0
arcdxf	0	0	0	0	0	0	0	0
arciges	0	0	0	0	0	0	0	0
arcmodeltools	1	0	0	0	0	1	0	1
arcshape	3	0	0	0	0	3	0	3
ascheckout	25	0	0	0	0	0	7	75
asciigrid	0	0	0	0	0	0	0	0
as_append	5	0	0	0	0	1	0	2
as_drawenv	1	0	0	0	0	5	0	5
as_open	4	0	0	0	0	0	0	1
as_routines	16	0	0	0	0	4	1	18
as_set	7	0	0	0	0	0	1	8
as_transbrowse	0	0	0	0	0	0	0	6
as_trans_mngr	11	0	0	0	0	1	0	18
autocontour	10	4	0	4	4	5	0	9
backcover	5	0	0	0	0	1	17	1
backenv	1	0	0	0	0	1	9	1
backitem	1	0	0	0	0	0	2	0
barriers	10	0	0	0	0	0	8	4
bas_end	0	0	0	0	0	0	0	6
batch-delin-tp-tree	9	1	0	5	1	6	0	6
batch-flint	2	0	0	0	0	1	0	0
batch-lauren-delin-absolute-slice	2	0	0	0	0	1	0	1
batch-lauren-delin-absolute	2	0	0	0	0	1	0	1
batch-lauren-delin-range	2	0	0	0	0	1	0	1
batch-lauren-delin-standard	2	0	0	0	0	1	0	1
bifur	4	2	0	1	3	5	23	7
browsecover	1	0	0	0	0	0	2	5
browsedb	3	0	0	0	0	0	0	5
buf	13	0	0	6	0	7	0	11
buffer	0	0	0	0	0	0	0	0
calculate_ap	0	0	0	0	0	1	2	4
calc_item	0	0	0	0	0	1	1	3
camera	15	8	45	8	45	0	92	0
center_edit	33	0	1	0	1	5	20	13
centroid-albers-meadesranch	0	0	0	0	0	0	0	0
centroid-xy	0	0	0	0	0	2	0	2
cf_batchmatch	3	2	0	0	2	2	0	1
cf_batchshape	14	0	0	0	0	4	6	16
cf_driver	14	2	0	0	2	4	7	13
cf_manedit	3	1	0	1	1	1	8	2
cf_mantran	8	1	0	1	1	1	5	0
cf_nodetrans	29	8	0	0	8	19	8	26
cf_rpt	14	0	0	0	0	4	10	16
cf_setup	5	0	0	0	0	2	7	11
check_oracle_table	5	0	0	0	0	0	0	4
class_anno	10	0	0	0	0	4	36	2
class_barrier	12	0	0	0	0	0	0	12
class_box	3	1	0	1	0	1	30	1
class_boxfill	3	1	0	1	0	1	33	1
class_center	27	0	0	0	1	1	45	8
class_circle	3	1	0	1	0	1	22	1

class_circlefill	3	1	0	1	0	1	30	1
class_cover	24	0	0	0	0	4	55	9
class_event	21	0	0	0	0	0	39	16
class_graph	8	1	0	1	0	6	48	3
class_graticule	4	0	0	0	0	3	16	3
class_grat_grid	3	0	0	0	0	2	14	3
class_grat_hatch	2	0	0	0	0	1	13	2
class_grat_label	2	0	0	0	0	1	22	2
class_grat_mrkr	2	0	0	0	0	1	13	2
class_grid	6	5	0	0	0	5	24	4
class_gridcomp	4	5	0	0	0	5	14	3
class_hillshade	1	0	0	0	0	0	4	1
class_image	7	5	0	0	0	6	14	3
class_key	3	1	0	1	0	1	51	1
class_keyfile	5	1	0	1	0	3	61	3
class_line_primitive	3	1	0	1	0	1	20	1
class_link	15	0	0	0	0	0	48	6
class_mapview	17	8	0	1	0	8	69	1
class_marker	3	1	0	1	0	1	17	1
class_mesh	2	0	5	0	5	1	20	2
class_neatline	5	1	0	1	0	1	32	1
class_northarrow	4	1	0	1	0	2	35	2
class_piechart	5	1	0	1	0	3	17	3
class_plotfile	3	1	0	1	0	1	18	1
class_polyfill	3	1	0	1	0	1	30	1
class_region	11	0	0	0	0	4	65	3
class_route	15	0	0	0	0	4	62	5
class_scalebar	10	8	0	8	0	1	73	1
class_section	4	0	0	0	0	0	38	1
class_stop	29	0	0	0	1	1	45	8
class_text	6	2	0	2	0	3	11	3
class_textfile	6	2	0	2	0	3	11	3
class_tin_edge	5	0	0	0	0	1	11	2
class_tin_node	4	0	0	0	0	0	18	1
class_tin_triangle	5	0	0	0	0	1	22	2
clip	0	0	0	0	0	0	0	0
cluster	3	0	0	0	1	2	2	3
cogo	0	0	0	0	0	0	0	0
colorpick	0	0	0	0	0	0	9	0
columns	13	0	0	0	0	0	4	12
combines_stats	1	1	0	0	1	1	0	0
command_tools	2	0	0	0	0	2	0	2
composite	10	0	0	3	0	7	0	0
con-simple	7	0	0	0	0	7	0	1
connect	0	0	0	0	0	0	0	0
contour	1	0	0	0	0	1	0	4
convertitem	8	1	0	1	0	0	0	7
coord	0	0	0	0	0	0	0	14
copystack	2	0	0	0	0	2	0	2
costpath	6	0	0	3	0	4	32	0
cover_mngr	2	0	0	0	0	0	3	9
cov_text	4	0	0	0	0	4	29	2
create_center	5	0	0	0	0	0	0	4
create_cover	6	0	0	0	0	0	0	6
create_stop	6	0	0	0	0	0	0	5
cut_fill	0	0	0	0	0	0	0	0
data_assign-mono	8	2	0	1	2	1	0	6
data_assign	12	1	0	1	1	1	0	8

data_bin-cut	4	0	0	0	0	0	0	1
data_bin-project	0	0	0	0	0	0	0	12
dbi_routines	2	0	0	0	0	0	0	1
dbmspull	4	0	0	0	0	0	0	3
dbmspush	4	0	0	0	0	0	0	5
define_sde	1	0	0	0	0	0	3	3
deflayer_mngr	4	0	0	0	0	0	6	4
demlattice	0	0	0	0	0	0	0	0
desymbol	0	0	0	0	0	0	3	0
dfadarc	0	0	0	0	0	0	0	0
dig_simplemenu	1	0	0	0	0	0	5	0
dimearc	0	0	0	0	0	0	0	2
dimension_map_generator	2	0	0	0	0	0	0	2
disp	1	1	0	1	0	0	33	0
disp_delin	0	0	0	0	0	0	6	0
disp_dem	5	0	0	0	0	0	0	1
disp_image	0	0	0	0	0	0	0	0
disp_legend	4	0	0	0	0	0	26	4
dissolve	0	0	0	0	0	0	0	0
dlgarc	0	0	0	0	0	0	0	0
dot_density	7	0	0	0	1	4	11	1
drain	10	0	0	5	0	8	0	5
drawcover	4	0	0	0	0	4	15	4
drawenv	3	0	0	0	0	3	16	3
dropfeatures	9	0	0	0	0	9	0	0
dtedgrid	0	0	0	0	0	0	0	0
dump_delta	6	0	0	3	0	3	0	6
dxfarc	1	0	0	0	0	0	3	0
edarc	19	7	0	7	6	2	4	3
edarcenv	19	21	0	17	18	6	0	0
edarc_more	0	0	0	0	0	0	0	0
edboundary	7	1	0	1	1	1	3	1
edcontrol	10	0	0	0	0	0	4	2
edgematch	11	7	0	0	7	7	1	0
editfclass	3	0	0	0	0	0	3	0
edit_annogen	13	0	0	0	0	16	8	32
edit_annopar	17	0	0	0	0	16	10	32
edit_fat	22	0	0	0	0	8	1	27
edit_fat_calc	0	0	0	0	0	5	0	5
edit_land_prop	20	0	0	0	0	0	3	8
edit_parcel	30	0	0	0	0	2	3	16
edit_poly	11	0	0	0	0	0	3	8
edit_table	23	0	0	0	0	7	2	28
edit_table_calc	0	0	0	0	0	5	0	5
edit_table_sort	3	0	0	0	0	0	0	2
edit_tools	3	0	0	0	0	1	5	2
edlab	1	0	0	0	0	2	0	2
edlabenv	0	0	0	0	0	0	0	0
edregion	1	0	0	0	0	0	1	1
edroute	11	0	0	0	0	0	6	0
edrteenv	11	16	0	16	11	5	0	0
ed_backgr	8	8	0	8	6	4	0	9
ed_nocogo	16	8	0	8	7	3	1	4
erase	0	0	0	0	0	0	0	0
etakarc	0	0	0	0	0	0	0	0
eventsource_mngr	7	0	0	0	0	0	0	7
event_dissolve	4	0	0	0	0	0	0	4

event_overlay	4	0	0	0	0	0	0	4
event_pullitems	1	0	0	0	0	0	0	1
event_transform	3	0	0	0	0	0	0	3
export	3	0	0	0	0	3	0	0
extended	52	34	0	20	30	22	5	10
extract_manual-update	6	1	0	2	1	4	4	3
fdr-four	2	0	0	1	0	1	0	2
featover	11	0	0	0	0	6	2	9
featureprox	18	0	0	0	0	0	3	13
fill	0	0	0	0	0	0	0	4
fillet_bndry	2	0	0	0	0	0	0	10
fill_dem_depressions	0	0	0	0	0	0	0	7
floatgrid	0	0	0	0	0	0	0	0
flyby	65	1	5	1	6	12	73	67
fly_around	31	2	5	2	6	7	73	33
formgen	8	0	0	0	0	0	0	7
forms	7	0	0	0	0	0	1	7
formsinfo	11	0	0	0	0	0	1	9
form_maker	16	0	0	0	0	0	1	13
fullpath	1	0	0	1	0	1	0	3
geaddlinks	0	0	0	0	0	0	0	3
gedrawarcs	1	0	0	0	0	0	1	3
gegraphic	0	0	0	0	0	0	2	0
gen_model	3	0	0	0	0	2	2	4
gen_snaps	20	0	12	0	12	2	24	23
gesnapopts	4	4	0	0	4	4	2	0
getdeflayer	2	0	0	0	0	0	5	0
geteventsource	6	0	0	0	0	0	0	6
getext	1	0	0	0	0	0	9	5
getextprop	0	0	0	0	0	2	4	2
getsymset	0	0	0	0	0	0	22	0
getsymsetae	0	0	0	0	0	0	4	0
gewarp	4	0	0	0	0	0	2	12
girasarc	0	0	0	0	0	0	0	2
gradsym	4	0	0	0	1	1	7	1
graph	0	0	0	0	0	0	26	0
graphics_output	7	3	0	2	1	4	28	7
graph_theme	5	0	0	0	0	4	40	3
grassgrid	0	0	0	0	0	0	0	0
gridascii	0	0	0	0	0	0	0	0
gridexpressiontools	0	0	0	0	0	0	0	0
gridfloat	0	0	0	0	0	0	0	0
gridimage	0	0	0	0	0	0	0	0
gridline	0	0	0	0	0	0	0	0
gridpoint	0	0	0	0	0	0	0	0
gridpoly	0	0	0	0	0	0	0	0
grid_anal_env	31	0	0	0	0	16	18	8
grid_expr_build	26	0	0	0	0	25	0	5
grid_mngr	6	2	0	0	0	3	30	9
grid_mngr2	2	0	0	0	0	0	3	7
grid_modeler	1	0	0	0	0	1	8	1
group	1	0	0	0	0	0	0	0
grp_edit	3	0	0	0	0	0	0	3
hist	6	1	0	4	0	3	37	2
hist1back	4	0	0	0	0	0	0	4
histdrill	10	0	0	0	0	0	1	5
histfeat	21	0	0	0	0	0	2	12
hview_gen	2	0	0	0	0	0	0	2

hypso	8	1	0	2	2	5	45	8
identity	0	0	0	0	0	0	0	0
igdsarcc	0	0	0	0	0	0	0	0
imagegrid	0	0	0	0	0	0	0	0
import	0	0	0	0	0	0	0	0
inflow-ranking-ofpl	2	1	0	0	2	6	0	5
inflow-ranking	3	1	0	0	2	5	0	5
infofile_mngr	0	0	0	0	0	0	2	5
inforeport	2	0	0	0	0	0	0	3
info_point	1	0	0	0	0	1	0	3
integrate	44	24	0	16	15	12	21	25
intersect	0	0	0	0	0	0	0	0
itemaccum	7	0	0	0	1	3	2	8
joinitem	2	0	0	0	0	0	0	1
join_bndry	6	5	0	5	5	3	0	5
junkrowcol	3	0	0	0	0	1	0	0
kriging-s	11	0	0	0	0	0	21	5
kriging-su	7	0	0	0	0	0	13	5
la	92	0	0	0	0	4	44	13
lacandidate	31	0	0	0	0	6	23	17
laconfig1	16	0	0	0	0	1	5	5
laconfig2	16	0	0	0	0	1	5	5
lademand	1	0	0	0	0	0	24	0
lanetwork	1	0	0	0	0	0	1	0
lasolve	9	0	0	0	0	0	1	0
lat	8	1	0	1	1	4	0	3
lat2	6	1	0	1	1	3	0	2
latticedem	0	0	0	0	0	0	0	0
latticetin	0	0	0	0	0	0	0	0
lat_driver	2	0	0	0	0	1	0	0
lat_gen	1	0	0	0	0	0	1	4
lat_reg	3	3	0	3	2	1	0	0
layers	0	4	0	0	0	4	9	0
layout_bndry	5	3	0	3	3	3	0	11
layout_tie	4	3	0	3	3	3	0	4
license	0	0	0	0	0	0	0	0
line-slope	1	0	0	0	0	0	0	2
linesymbol	0	0	0	0	0	0	15	0
lineupdate	10	6	0	6	4	3	3	9
linkopts	6	6	0	0	6	6	0	0
loadmap	0	0	0	0	0	0	5	0
logicalae	0	0	0	0	0	0	0	0
logicalap	32	0	0	0	0	2	37	9
logicalap2	6	0	0	0	0	0	24	0
logicalsde	2	0	0	0	0	1	17	1
los	11	0	0	0	0	1	21	10
main_set	3	0	0	0	0	0	3	0
mapdriver	6	1	0	0	0	1	28	0
mapprops	0	0	0	0	0	0	12	0
map_library	6	0	0	0	0	0	5	13
map_object_mngr	2	0	0	0	0	1	19	1
map_prefs	0	0	0	0	0	0	6	0
map_tools	1	0	0	0	0	1	8	1
markersymbol	0	0	0	0	0	0	12	0
mask-con	3	0	0	1	0	3	0	0
mask-random	3	0	0	1	0	2	0	1
mask-select	4	0	0	0	0	4	0	0
mask-xy	6	0	0	0	0	5	0	0

measure	7	0	1	0	1	1	13	5
module_chk	1	0	0	0	0	1	4	0
mossarc	0	0	0	0	0	0	0	0
mru-combo	8	1	0	4	1	6	0	1
mru-slice	3	1	0	3	1	2	0	0
mru_dissolve	2	0	0	2	0	2	0	2
mru_gen_pre_reg	3	3	0	3	2	2	0	1
mru_id-change-assign-display_atts	0	0	0	0	0	0	5	0
mru_id-change-reclass	1	0	0	0	0	4	0	6
mru_id-change-update	0	0	0	0	0	9	0	9
mru_id-change	1	0	0	1	0	5	0	4
mru_numbers	0	0	0	0	0	0	4	0
nchan_rasterize-highlight	3	0	0	0	0	0	6	0
nclim-list	10	0	0	0	0	3	0	6
nclim	7	1	0	1	1	7	28	9
near	0	0	0	0	0	0	0	0
nearstream_reroute	4	0	0	1	0	2	0	3
network_edit	21	0	0	0	0	4	10	6
newcover	8	0	0	0	0	2	6	9
ngrid	2	0	0	0	0	1	0	0
node	2	0	0	0	0	1	0	1
nodeprop	0	0	0	0	0	0	7	0
nofpl	6	0	0	5	0	6	0	11
north_arrow	3	4	0	3	0	1	7	0
opencover	2	0	0	0	0	0	2	0
outdirs_exist	0	0	0	0	0	0	0	0
outlet	5	1	0	1	1	4	0	2
output-2d	4	0	0	0	0	0	0	4
output	0	0	0	0	0	0	6	6
pagesetup	0	0	0	0	0	0	10	0
panzoom	7	0	0	0	0	3	13	3
parallel_bndry	2	0	0	0	0	0	0	9
param_2nd_dimension	1	1	0	0	1	1	0	1
param_area-1st_order-smallest	2	0	0	0	0	1	0	3
param_area-acres	0	0	0	0	0	1	0	1
param_area-hectare	0	0	0	0	0	0	0	2
param_area-km	0	0	0	0	0	0	0	2
param_area-miles-accumulate	2	1	0	1	1	2	0	3
param_area-miles	0	0	0	0	0	1	0	1
param_area-smallest	1	0	0	1	0	1	0	2
param_area-total-nhru-acres	0	0	0	0	0	1	0	1
param_area-total-nhru-km	2	1	0	1	1	2	0	2
param_aspect-arctan2	3	1	0	1	1	3	0	1
param_chan-width	2	0	0	0	0	1	0	3
param_cov-den-summer-dominant	5	1	0	3	1	5	0	4
param_cov-den-summer	8	5	0	3	5	12	0	20
param_cov-den-winter	8	5	0	3	5	13	0	21
param_cov-den-winter2	6	1	0	4	1	5	0	3
param_cov-den-winter3	6	1	0	4	1	5	0	2
param_cov-type-klinefelter	1	1	0	0	1	1	0	2
param_cov-type-prms	1	1	0	0	1	1	0	2

param_cov-type-prms2	19	6	0	12	6	19	0	6
param_cov-type-prms3	21	7	0	13	7	21	0	5
param_cov-type	1	1	0	0	1	1	0	2
param_daf_pct_area	3	1	0	1	1	1	0	5
param_dajunction-down	2	1	0	1	1	2	0	2
param_dist2headwater-miles	0	0	0	0	0	0	0	2
param_dist2headwater	0	0	0	0	0	0	0	2
param_elevation-max-meters	1	1	0	0	1	1	0	2
param_elevation-mean-feet	2	1	0	0	1	2	0	2
param_elevation-mean-meters	1	1	0	0	1	1	0	2
param_elevation-min-meters	0	0	0	0	0	0	0	2
param_elevation-range-feet	1	0	0	0	0	1	0	2
param_elevation-range-meters	0	0	0	0	0	0	0	2
param_elevation-std-meters	1	0	0	0	0	1	0	2
param_gen-imperv-binary	1	0	0	1	0	2	0	2
param_gen-ov-colrowmap	0	0	0	0	0	0	0	1
param_inflow-primary	3	1	0	0	1	3	0	2
param_inflow-secondary	3	1	0	0	1	3	0	2
param_inflow-tertiary	3	1	0	0	1	3	0	2
param_intcp-mean-snow	1	1	0	0	1	1	0	2
param_intcp-mean-srain	1	1	0	0	1	1	0	2
param_intcp-mean-wrain	1	1	0	0	1	1	0	2
param_intcp-snow	2	0	0	1	0	1	0	4
param_intcp-snow2	5	1	0	3	1	4	0	2
param_intcp-srain	2	0	0	1	0	1	0	4
param_intcp-srain2	5	1	0	3	1	4	0	2
param_intcp-wrain	2	0	0	1	0	1	0	4
param_intcp-wrain2	5	1	0	3	1	4	0	2
param_intersect-gwcell-col_id	3	1	0	0	1	1	0	2
param_intersect-gwcell-row_id	3	1	0	0	1	1	0	2
param_line-slope	0	0	0	0	0	3	0	3
param_loni-bin-st-ac	4	0	0	0	0	0	0	5
param_loni-mean	1	1	0	0	1	1	0	2
param_loni-nbins	3	1	0	1	1	4	0	11
param_loni-nbins2	3	1	0	1	1	4	0	11
param_loni-nbins3	3	1	0	1	1	4	0	11
param_ndanode-local	4	2	0	1	2	2	0	6
param_nnny-nnnx-id	4	0	0	0	0	0	0	4
param_nnny-nnnx-nssr	6	2	0	1	2	3	0	8
param_num-chan	0	0	0	0	0	6	0	6
param_ofpl-inflow-primary	1	1	0	0	1	1	0	2
param_ofpl-inflow-secondary	1	1	0	0	1	1	0	2
param_ofpl-inflow-	2	1	0	1	1	2	0	2

secondary3								
param_ofpl-length	0	0	0	0	0	3	0	3
param_one-plane_area	0	0	0	0	0	0	0	2
param_one-plane_ellmaj	1	0	0	0	0	0	0	2
param_one-plane_ellmin	1	0	0	0	0	0	0	2
param_one-plane_ndabranch	1	1	0	0	1	1	0	2
param_one-plane_perimeter	1	0	0	0	0	0	0	2
param_order-shreve	6	1	0	3	1	6	0	2
param_oregon-calibration-assign-display_atts	5	0	0	0	0	0	14	0
param_oregon-calibration-assign	2	0	0	0	0	142	0	142
param_ov-area-pct	4	1	0	1	1	1	0	5
param_ov-area-pct2	4	1	0	1	1	2	0	6
param_ov-area	2	1	0	1	1	1	0	5
param_perimeter	0	0	0	0	0	0	0	2
param_poly2point	1	0	0	1	0	1	0	5
param_rock-depth-mean-meters	1	1	0	0	1	1	0	2
param_root-depth-mean-meters	1	1	0	0	1	1	0	2
param_root-depth	1	1	0	0	1	1	0	2
param_slope-10-85	22	7	0	9	7	15	3	27
param_slope-degrees-mean	1	1	0	0	1	1	0	2
param_slope-mean	1	1	0	0	1	1	0	2
param_snow-threshold	2	0	0	0	0	1	0	3
param_snowdepletion-curve	0	0	0	0	0	2	0	2
param_soil-awc	1	1	0	0	1	1	0	2
param_soil-bulk-density	1	1	0	0	1	1	0	2
param_soil-depth	1	1	0	0	1	1	0	2
param_soil-field-capacity-mean	1	1	0	0	1	1	0	2
param_soil-moist-meters	1	1	0	0	1	1	0	2
param_soil-organic-matter	1	1	0	0	1	1	0	2
param_soil-pct_clay-mean	1	1	0	0	1	1	0	2
param_soil-pct_sand-mean	1	1	0	0	1	1	0	2
param_soil-pct_silt-mean	1	1	0	0	1	1	0	2
param_soil-perm-mean-meters	1	1	0	0	1	1	0	2
param_soil-perm	1	1	0	0	1	1	0	2
param_soil-porosity-mean	1	1	0	0	1	1	0	2
param_soil-szm	1	1	0	0	1	1	0	2
param_soil-wilt-point-mean	1	1	0	0	1	1	0	2
param_stream-shreve	0	0	0	0	0	0	0	2
param_stream-strahler	1	1	0	0	1	1	0	1

param_temp-adj-max	2	1	0	0	1	3	0	1
param_temp-adj-min	2	1	0	0	1	3	0	1
param_topmodel-ach-d	6	1	0	1	1	1	0	10
param_topmodel-ach-d2	3	0	0	0	0	4	0	8
param_topmodel-ach	8	2	0	4	2	6	0	2
param_topmodel-d	2	1	0	1	1	2	0	1
param_tree-dom	1	1	0	0	1	1	0	2
param_velocity-coefficient	0	0	0	0	0	4	0	4
param_wcov-trans-density	6	1	0	4	1	5	0	3
param_wcov-trans	1	1	0	0	1	1	0	2
param_wcov-trans2	2	1	0	1	1	2	0	1
parcel_prefs	1	1	0	1	1	1	3	2
parcel_storm	3	0	0	0	0	0	6	22
partition	32	0	1	0	3	8	13	16
par_addlist	1	0	0	0	0	2	0	2
par_ap-dump	2	0	0	0	0	0	0	3
par_batch-output	3	1	0	1	1	5	0	4
par_combine-zone-param	5	3	0	3	3	6	0	8
par_oui-relate	5	1	0	2	1	2	0	3
par_overlay-chk	5	2	0	2	2	0	0	3
par_rename-item	0	0	0	0	0	1	0	1
par_unload	0	0	0	0	0	3	0	5
place_subdiv	19	12	0	9	10	6	8	3
plotcopies	0	0	0	0	0	0	5	0
plotdivide	0	0	0	0	0	0	29	0
plotmulti	0	0	0	0	0	0	8	0
point2zone	2	0	0	0	0	0	0	2
pointdistance	0	0	0	0	0	0	0	0
polygon_event	1	0	0	0	0	0	0	1
precedence	6	0	0	0	1	1	15	3
prim_edtr	0	0	0	0	0	0	0	0
profile	0	0	0	0	0	0	20	0
propertydriver	2	0	0	0	0	0	16	5
property_tools	2	0	0	0	0	2	4	2
prop_panzoom	4	0	0	0	0	0	9	0
q	5	0	0	3	0	5	0	2
quickdraw	15	8	0	0	0	8	50	7
quickplot	0	0	0	0	0	0	6	0
reclass	40	0	0	0	0	0	16	34
rectify	0	0	0	0	0	0	0	0
redefine	3	0	0	0	0	2	0	6
regionclass	0	0	0	0	0	0	0	0
regiondissolve	0	0	0	0	0	0	0	0
regionselect	13	0	0	0	0	0	29	0
register	0	0	0	0	0	0	0	0
relate	3	0	0	0	0	1	0	1
relate_mngr	6	0	0	0	0	0	0	0
remeassec	0	0	0	0	0	12	0	12
remeasure	2	0	0	0	0	12	0	12
remove_obj	0	0	0	0	0	0	0	12
reudl	0	3	0	0	0	3	6	0
rg-cat	3	0	0	1	0	1	1	6
rotate_arcs	10	5	0	2	5	3	2	6
route_font	0	0	0	0	0	0	13	0
route_hatch	4	0	0	0	0	0	0	4
route_hatch_font	0	0	0	0	0	0	9	0

route_offset	6	0	0	0	0	2	0	4
route_text	6	0	0	0	0	2	0	4
routing	81	7	2	1	8	5	21	25
routing_property	0	0	0	0	0	0	6	0
rule_submit	3	0	0	0	0	0	3	2
save_object_as	3	0	0	0	0	3	0	5
scalebar	2	2	0	2	0	0	20	0
scratch_kill	0	0	0	0	0	0	0	2
sde_edit_calc	0	0	0	0	0	0	0	0
sdtsexport	0	0	0	0	0	0	0	0
seed	7	0	0	4	0	7	0	1
select-attr	4	0	0	0	0	4	0	0
select_sde	0	0	0	0	0	0	7	0
selprefs	4	4	0	4	4	4	0	0
sel_statasc	4	0	0	0	0	0	0	8
setmaplibenv	7	0	0	0	0	0	2	5
setnull	2	0	0	0	0	2	0	0
setsdeenv	7	0	0	0	0	0	7	7
setstormenv	8	0	0	0	0	0	0	8
setup_composite	6	0	0	2	0	5	0	1
set_analysis_window	6	0	0	0	0	0	0	0
set_dimension_point	15	1	0	1	1	9	0	9
se_dbmsexists	0	0	0	0	0	0	0	0
se_featclass	3	0	0	0	0	3	0	3
se_loaddbms	2	0	0	0	0	0	0	2
se_loadinfo	0	0	0	0	0	0	0	0
sfc_aspect	0	0	0	0	0	0	0	3
sfc_cov-type-prms	2	0	0	1	0	1	0	5
sfc_cov-type-wt	1	0	0	0	0	0	0	5
sfc_cov-type	2	0	0	1	0	1	0	5
sfc_downcell-id	2	1	0	1	1	2	0	4
sfc_elv-focalmean	0	0	0	0	0	0	0	2
sfc_enns-resrv	0	0	0	0	0	0	0	2
sfc_enns-topvar	0	0	0	0	0	0	0	2
sfc_flow-accumulation	3	0	0	2	0	2	0	3
sfc_flow-direction	0	0	0	0	0	0	0	3
sfc_flowlength-down-3d	2	0	0	0	0	0	0	2
sfc_flowlength-down	6	0	0	2	0	4	0	2
sfc_flowlength-up	2	0	0	0	0	0	0	2
sfc_focalvariety-data	0	0	0	0	0	0	0	2
sfc_focalvariety-nodata	0	0	0	0	0	0	0	2
sfc_imperv	1	0	0	0	0	0	0	5
sfc_intcp-snow	2	0	0	1	0	1	0	5
sfc_intcp-srain	2	0	0	1	0	1	0	5
sfc_intcp-wrain	2	0	0	1	0	1	0	5
sfc_jh-coef	0	0	0	0	0	0	0	2
sfc_jh-coef2	0	0	0	0	0	0	0	2
sfc_leaf-loss	1	0	0	0	0	0	0	3
sfc_loni-contour-width	1	0	0	1	0	1	0	2
sfc_loni-delta-elv	2	0	0	2	0	2	0	7
sfc_loni-distance	1	0	0	1	0	1	0	2
sfc_loni-fac	4	0	0	2	0	4	0	2
sfc_loni	0	0	0	0	0	0	0	2
sfc_radpl	3	1	0	2	1	2	0	8
sfc_rechr-depth	1	0	0	1	0	1	0	2
sfc_reclass-interactive	1	0	0	0	0	0	0	3
sfc_reclass	1	0	0	0	0	0	0	4
sfc_rock-depth-max	1	0	0	0	0	0	0	2

sfc_root-depth	3	0	0	2	0	2	0	6
sfc_sinks	1	0	0	1	0	1	0	5
sfc_slope-degrees	0	0	0	0	0	0	0	3
sfc_slope	0	0	0	0	0	0	0	3
sfc_soil-awc	2	0	0	0	0	0	0	2
sfc_soil-bulk-density	2	0	0	0	0	0	0	2
sfc_soil-depth-meters	1	0	0	0	0	0	0	2
sfc_soil-depth	1	0	0	0	0	0	0	2
sfc_soil-ne	1	0	0	1	0	1	0	2
sfc_soil-organic-matter	2	0	0	0	0	0	0	2
sfc_soil-percent-clay	2	0	0	0	0	0	0	2
sfc_soil-percent-sand	2	0	0	0	0	0	0	2
sfc_soil-percent-silt	2	0	0	0	0	0	0	2
sfc_soil-perm	2	0	0	0	0	0	0	2
sfc_soil-texture-prms	3	0	0	2	0	2	0	2
sfc_soil-wilt-point	1	0	0	0	0	1	0	2
sfc_temp-adj-max	1	0	0	0	0	0	0	4
sfc_temp-adj-min	1	0	0	0	0	0	0	4
sfc_wcov-trans	1	0	0	1	0	1	0	3
sfc_wcov-trans2-density	3	0	0	2	0	2	0	3
sfc_wcov-trans2	0	0	0	0	0	0	0	2
shadesymbol	1	0	0	0	0	1	17	1
shapearc	0	0	0	0	0	0	0	0
showtable	6	0	0	0	0	0	0	5
shutoff	97	2	0	2	0	4	5	24
slfarc	0	0	0	0	0	0	0	0
slice	40	0	0	0	0	0	23	33
snap2grid	0	0	0	0	0	0	12	0
snapenv	1	1	0	0	1	1	0	0
snapopts	6	6	0	0	6	6	0	0
snaprotate	49	34	0	20	30	22	5	10
snaprotate2	10	5	0	2	5	3	2	6
soils_convert	6	1	0	1	3	14	0	25
solrad	54	0	0	29	0	54	0	45
solution_edit	14	0	0	0	0	0	8	0
spatial	6	0	2	0	2	1	54	1
spatialsde	0	0	0	0	0	0	6	0
spatialsel	54	0	2	0	2	1	54	17
spatial_event	17	0	0	0	1	3	11	4
splitbuffer	1	0	0	0	0	0	0	3
split_bndry	2	0	0	0	0	0	0	9
split_parcel	5	0	0	0	0	0	0	0
sql_builder	5	0	0	0	0	0	35	0
sql_event	6	0	0	0	0	0	0	6
ssmodel	5	0	0	0	0	0	0	4
stack_mngr	7	2	0	0	0	3	28	8
statistics_ap	2	0	0	0	1	1	1	2
stats_tour	5	0	0	2	0	4	0	5
stop_edit	40	0	1	0	1	6	20	20
stormselect	2	0	0	0	0	0	10	2
stream	1	0	0	1	0	1	0	1
streamshed	7	0	1	0	1	1	35	32
stream_edit	4	1	0	3	1	6	1	9
stream_extract	1	0	0	1	0	1	0	1
strm_beef	0	0	0	0	0	1	0	1
subselect	0	0	0	0	0	0	0	0
subselprefs	3	3	0	3	3	3	0	0
supervised	5	0	0	0	0	1	4	0

surface	0	0	0	0	0	0	0	0
surfacelocator	1	0	24	0	24	1	35	1
textitem	1	0	0	0	0	0	1	0
textsymbol	3	0	0	0	0	3	18	3
themeclasses	0	0	0	0	0	0	1	0
theme_mngr	1	1	1	0	1	2	10	1
thiessen	0	0	0	0	0	0	0	0
tinarc	0	0	0	0	0	0	0	0
tinvrml	0	0	0	0	0	0	0	0
tools	24	1	4	0	5	1	20	10
tool_cogo_adjust	10	0	0	0	0	0	5	8
tool_qa	6	0	0	0	0	1	6	4
topology	11	0	0	0	0	11	0	11
traceover	10	0	0	0	0	0	7	9
transfer	0	0	0	0	0	0	0	0
transform	0	0	0	0	0	0	0	0
turn_edit	34	2	0	0	2	0	35	12
udlayers	0	4	0	0	0	4	7	0
ungenerate	0	0	0	0	0	0	0	0
union	0	0	0	0	0	0	0	0
unsupervised	0	0	0	0	0	0	0	0
vector_display-classify	4	1	0	1	0	0	10	3
vector_display-identify	0	2	0	0	0	2	18	0
vector_display-measure	0	0	0	0	0	0	16	0
vector_overlay-classify	4	1	0	1	0	0	6	3
vector_overlay-identify	0	2	0	0	0	2	4	0
vector_overlay-measure	0	0	0	0	0	0	13	0
veriplot	2	2	0	1	0	1	13	0
version	5	0	0	0	0	2	0	15
view	0	0	4	0	4	0	10	0
viewdriver	45	11	0	0	0	14	59	27
viewplot	3	0	0	0	0	0	23	0
view_prefs	1	2	0	0	0	3	13	1
view_select	15	0	0	0	0	0	6	8
view_zoom	22	7	0	0	0	10	24	5
volume	0	0	0	0	0	0	1	1
v_dclassify	10	0	0	0	3	3	0	10
v_dlegend	11	0	0	0	3	4	29	17
workspace	0	0	0	0	0	0	0	0
workspace_mngr	0	0	0	0	0	0	0	0
zonalstat_factory	2	1	0	0	1	3	0	1
zone_accumulation	0	0	0	0	0	0	0	2
zone_area-firstorder	2	0	0	1	0	3	0	4
zone_areas-internal	3	0	0	2	0	3	0	2
zone_centroid-point	0	0	0	0	0	0	0	2
zone_centroid	4	2	0	1	2	3	3	8
zone_chan-segs-local	2	0	0	2	0	2	0	2
zone_chan-segs	2	0	0	0	0	8	0	15
zone_distance-euclidean	0	0	0	0	0	0	0	2
zone_distance-flowlength	4	0	0	2	0	4	0	2
zone_down-id	4	0	0	4	0	4	0	2
zone_fac-min-pt	1	0	0	1	0	1	0	2
zone_flow-accumulation	3	0	0	2	0	3	0	2
zone_flow-accumulation2	2	0	0	1	0	2	0	2
zone_fullpath	25	2	0	11	2	19	0	9
zone_fullpath2	23	2	0	10	2	17	0	9

zone_headwater-area	0	0	0	0	0	0	0	2
zone_headwater-pts	2	0	0	2	0	2	0	2
zone_headwaters-internal	1	0	0	1	0	1	0	2
zone_internal-cells	2	0	0	1	0	2	0	2
zone_loni-bin	5	0	0	1	0	3	0	15
zone_loni-bin2	5	0	0	1	0	3	0	15
zone_loni-bin3	5	0	0	1	0	3	0	15
zone_loni	0	0	0	0	0	0	0	2
zone_main-link-tops	1	0	0	1	0	1	0	2
zone_main-link	27	3	0	15	3	23	0	10
zone_nchan-id	2	0	0	0	0	1	0	4
zone_ndajunction	6	1	0	3	1	7	0	8
zone_ndanode	8	2	0	4	2	7	0	8
zone_ntopchan-headwater-small	4	0	0	2	0	2	0	4
zone_ntopchan-local	11	4	0	9	4	9	0	4
zone_ntopchan-local2	3	1	0	3	1	2	0	2
zone_ntopchan-mainlink	2	0	0	1	0	1	0	3
zone_ntopchan-segs	1	0	0	1	0	2	0	8
zone_ntopchan	2	1	0	1	1	3	0	7
zone_offset-pp-elevation	1	0	0	1	0	1	0	2
zone_offset-pp-flowlength	1	0	0	1	0	1	0	2
zone_offset-pp2pp-elevation	2	1	0	1	1	1	0	5
zone_offset-pp2pp-flowlength	2	1	0	1	1	1	0	5
zone_one-plane	0	0	0	0	0	0	0	2
zone_out-dsheds	1	0	0	1	0	1	0	2
zone_out-fdr-cmb	1	1	0	1	1	0	0	2
zone_out-fdr	1	0	0	1	0	1	0	2
zone_out-maxfac-flag	1	0	0	1	0	1	0	2
zone_out-maxfac	2	0	0	1	0	2	0	2
zone_outlet-downstream2	1	0	0	1	0	1	0	3
zone_outlets-downstream2	10	0	0	10	0	10	0	2
zone_perimeter-all	3	0	0	2	0	3	0	2
zone_perimeter-dsheds	1	0	0	1	0	1	0	2
zone_perimeter-external	2	0	0	1	0	2	0	2
zone_perimeter-headwater-external	1	0	0	1	0	1	0	2
zone_perimeter-headwater	0	0	0	0	0	0	0	2
zone_radpl	4	1	0	1	1	6	0	5
zone_range-absolute-slice-x	1	0	0	0	0	1	0	2
zone_range-absolute-slice-y	1	0	0	0	0	1	0	2
zone_range-absolute-slice-z	1	0	0	0	0	1	0	2
zone_range-absolute-x	1	0	0	0	0	0	0	3
zone_range-absolute-y	1	0	0	0	0	0	0	3
zone_range-absolute-z	1	0	0	0	0	0	0	3
zone_range-relative-x	1	0	0	0	0	2	0	3
zone_range-relative-y	1	0	0	0	0	2	0	3
zone_range-relative-z	1	0	0	0	0	2	0	3

zone_route-non-ordered	2	1	0	2	1	1	0	2
zone_route-ordered	1	0	0	0	0	2	0	6
zone_shape-ratio	5	2	0	1	2	5	3	7
zone_slice	2	0	0	1	0	4	0	8
zone_strink	0	0	0	0	0	0	0	2
zone_three-plane	2	0	0	1	0	2	0	2
zone_tops	1	0	0	1	0	1	0	2
zone_two-plane-network	5	0	0	3	0	5	0	3
zone_two-plane	5	0	0	3	0	5	0	3
zone_two-plane2	7	2	0	4	2	7	0	3
zone_watershed	0	0	0	0	0	0	0	2
zone_x	1	0	0	1	0	1	0	2
zone_y	1	0	0	1	0	1	0	2
zone_z	1	0	0	1	0	1	0	2

Appendix B. Enviro Modeling Implicit Keyword Matrix and Modified Procedure Matrix

This appendix contains two tables. The first, Table B-1, lists the values given for the Enviro Modeling implicit keywords for each GIS command. The second, Table B-2, shows the result of augmenting the default procedure matrix with the Enviro Modeling implicit keywords. Within Table B-1, each GIS command is shown on a row. The GIS commands are ordered alphabetically. After the first column, the columns correspond to Albrecht implicit keywords as labeled. Section 5.1 provided the rationale for defining the Enviro Modeling implicit keywords. Table B-1 shows the values that the author heuristically assigned for these keywords to each GIS command.

Table B-1 Matrix showing the assignment of values to GIS commands for the Enviro Modeling set of implicit keywords.

Command	Graphics	Selection/ Data Mgmt	Environment	Raster	Vector	Derive
+	0	0	0	1	0	1
^	0	0	0	1	0	1
=	0	0	0	1	0	1
ADD	0	1	0	0	1	1
ADDITEM	0	1	0	1	1	1
ANNOSIZE	1	0	0	0	1	0
AP	1	0	0	0	0	0
ARC	0	0	0	0	0	0
ARCS	1	0	0	0	1	0
ARCSNAP	0	0	1	0	1	0
ARROWSIZE	1	0	0	0	1	0
ARROWTYPE	1	0	0	0	1	0
ASCONNECT	0	1	0	0	0	0
ASELECT	0	1	0	0	0	0
ASEXECUTE	0	1	0	0	0	0
ASEXECUTE	0	1	0	0	0	0
AXIS	1	0	0	0	0	0
BACKENVIRONMENT	1	0	1	0	1	0
BACKSYMBOLITEM	1	0	0	0	1	0
BOX	1	0	0	0	0	0
CALC	0	0	0	1	1	1
CALCULATE	0	0	0	1	1	1
CIRCLE	1	0	0	0	0	0
CLASS	1	1	0	0	0	0
CLEARSELECT	0	1	0	0	0	0
COMBINE(	0	0	0	1	0	1
CON(	0	0	0	1	0	1
COORDINATE	0	0	1	0	0	0

COPY	0	1	0	0	0	1
CREATE	0	1	0	0	0	1
CURSOR	0	1	0	0	0	0
DBMSCURSOR	0	1	0	0	0	0
DBMSEXECUTE	0	1	0	0	0	0
DEFINE	0	1	0	0	0	1
DESCRIBE	0	1	0	0	0	0
DISPLAY	1	0	0	0	0	0
DRAWENVIRONMENT	1	0	1	0	1	0
EDIT	0	0	1	0	1	1
EDITDISTANCE	0	1	1	0	1	0
EDITFEATURE	0	0	1	0	1	0
EVENTSOURCE	0	1	1	0	1	0
GRAIN	0	0	1	0	1	0
GRAPHICS	1	0	0	0	0	0
GRIDSHADES	1	0	0	1	0	0
INFOFILE	0	1	0	0	0	1
INT(	0	0	0	1	0	1
INTERSECTARCS	0	0	1	0	1	0
ISNULL(	0	0	0	1	0	1
ITEMS	0	1	0	0	0	0
KEYAREA	1	0	0	0	0	0
KILL	0	1	0	0	0	0
LAYER	0	1	0	0	0	0
LAYERDRAW	1	1	0	0	0	0
LEADERARROWS	1	0	0	0	1	0
LEADERS	1	0	0	0	1	0
LEADERSYMBOL	1	0	0	0	1	0
LEADERTOLERANCE	1	0	0	0	1	0
LIBRARY	0	1	0	0	0	0
LINE	1	0	0	0	1	0
LINECOLOR	1	0	0	0	1	0
LINECOPY	1	0	0	0	1	0
LINEDELETE	1	0	0	0	1	0
LINEDELETELAYER	1	0	0	0	1	0
LINEPEN	1	0	0	0	1	0
LINESET	1	0	0	0	1	0
LINESIZE	1	0	0	0	1	0
LINESYMBOL	1	0	0	0	1	0
LINETYPE	1	0	0	0	1	0
LIST	0	1	0	0	0	1
LISTOUTPUT	0	1	0	0	0	1
MAPEXTENT	1	0	1	0	0	0
MAPLIMITS	1	0	0	0	0	0
MAPPOSITION	1	0	0	0	0	0
MAPPROJECTION	1	0	1	0	0	0
MAPSCALE	1	0	1	0	0	0
MARKER	1	0	0	0	1	0
MARKERCOLOR	1	0	0	0	1	0
MARKERCOPY	1	0	0	0	1	0
MARKERDELETE	1	0	0	0	1	0
MARKERMASK	1	0	0	0	1	0
MARKERSET	1	0	0	0	1	0
MARKERSIZE	1	0	0	0	1	0
MARKERSYMBOL	1	0	0	0	1	0
MOVE	0	0	0	0	1	1
NODECOLOR	1	0	0	0	1	0
NODESNAP	0	0	1	0	1	0

NSELECT	0	1	0	0	0	0
OVERFLOW	1	0	1	0	1	0
OVERPOST	1	0	1	0	1	0
PAGEEXTENT	1	0	0	0	0	0
PAGESIZE	1	0	0	0	0	0
PAGEUNITS	1	0	0	0	0	0
PATCH	1	0	0	0	0	0
POLYGONSHADES	1	0	0	0	1	0
READSELECT	0	1	0	0	0	0
RECLASS(	0	1	0	1	0	1
RELATE	0	1	0	0	0	0
REMOVEEDIT	0	1	1	0	1	0
RENAME	0	1	0	0	0	0
RESELECT	0	1	0	0	0	0
SAVE	0	1	0	1	0	1
SCALAR(	0	0	0	1	0	1
SEARCHTOLERANCE	0	1	1	0	1	0
SELECT	0	1	0	0	0	0
SETCELL	0	0	1	1	0	0
SETMASK	0	0	1	1	0	0
SETWINDOW	0	0	1	0	0	0
SHADECOLOR	1	0	0	1	1	0
SHADECOPY	1	0	0	1	1	0
SHADEDELETE	1	0	0	1	1	0
SHADEOFFSET	1	0	0	1	1	0
SHADESET	1	0	0	1	1	0
SHADESYMBOL	1	0	0	1	1	0
SHADETYPE	1	0	0	1	1	0
SIN(	0	0	0	1	0	1
SNAPCOVERAGE	0	0	1	0	1	0
SNAPPING	0	0	1	0	1	0
SORT	0	0	0	0	0	0
STATISTICS	0	0	0	1	1	1
SURFACEDRAPE	1	0	0	1	1	0
SURFACEOBSERVER	1	0	1	1	1	0
SURFACETARGET	1	0	1	1	1	0
SURFACEVIEWFIELD	1	0	1	1	1	0
SYMBOLITEM	1	0	0	1	1	0
TEXT	1	0	0	0	0	0
TEXTANGLE	1	0	0	0	0	0
TEXTCOLOR	1	0	0	0	0	0
TEXTCOPY	1	0	0	0	0	0
TEXTDIRECTION	1	0	0	0	0	0
TEXTFONT	1	0	0	0	0	0
TEXTJUSTIFICATION	1	0	0	0	0	0
TEXTMASK	1	0	0	0	0	0
TEXTOFFSET	1	0	0	0	0	0
TEXTQUALITY	1	0	0	0	0	0
TEXTSET	1	0	0	0	0	0
TEXTSIZE	1	0	0	0	0	0
TEXTSTYLE	1	0	0	0	0	0
TEXTSYMBOL	1	0	0	0	0	0
TRANSACTION	0	1	0	0	0	0
UNLOAD	0	1	0	0	0	1
UNSELECT	0	1	0	0	0	0
WEEDTOLERANCE	0	0	1	0	1	0
WINDOWS	1	0	0	0	0	0
WRITESELECT	0	1	0	0	0	1

ZONALMAJORITY(	0	0	0	1	0	1
ZONALMAX(	0	0	0	1	0	1
ZONALMEAN(	0	0	0	1	0	1
ZONALSTATS(	0	0	0	1	0	1

Table B-2 shows the procedure matrix after it was modified according to the Albrecht implicit keyword matrix (Table B-1). The process by which this modification was carried out was specified generically in Section 3.3.1. A specific example was also given in Section 5.1.1. Briefly, the number of times each explicit keyword (i.e. GIS command) was found within each GIS procedure was determined. The vector of Enviro Modeling implicit keyword values for that explicit keyword (given in Table B-1) is multiplied by this frequency. The vectors for all explicit keywords within a GIS procedure were then summed to produce a row within Table B-2, which describes the GIS procedures using the Enviro Modeling implicit keywords.

Table B-2 Procedure matrix after modification according to the Enviro Modeling implicit keyword matrix (Table B-1). This table was used to generate the Enviro Modeling SOM presented in Chapter 5.

GIS Procedure	Graphics	Selection/ Data Mgmt	Environment	Raster	Vector	Derive
addalias	0	2	4	0	3	1
addedit	19	65	13	0	27	16
addgenerate	0	5	0	5	6	8
addinput	0	0	0	2	0	2
additem	0	2	0	2	2	2
addmeasure	4	6	6	0	10	0
addresscreate	0	0	0	3	0	3
addressenv	0	0	0	0	0	0
addressfile	0	4	0	1	1	2
address_select	47	87	8	20	51	14
add_select	0	1	1	0	0	0
adrggrid	0	0	0	0	0	0
aeclean	0	0	4	0	4	0
aedefaults	11	0	16	0	16	0
aedriver	3	11	10	2	9	2
aesymset	4	0	0	1	3	0
ae_library	0	4	0	0	0	0
ae_project	6	0	6	1	0	1
aggregate	0	23	0	9	9	9
allocation	15	14	2	4	4	4
annoadd	0	2	0	8	6	10
annoaddenv	5	0	0	4	3	4
annoedit	0	0	0	28	27	28
annoposline	0	0	0	0	0	0
annopospoint	0	0	0	0	0	0
aoi_delineation-auto	1	3	0	1	1	1
aoi_delineation-pre	0	2	1	5	0	4
aoi_delineation-seed	1	3	1	2	0	1
append	0	0	0	1	0	1
ap_delin	19	0	1	0	16	0
ap_graph	12	0	0	0	4	1
ap_mru-num	15	2	2	0	9	0
ap_mru-select	15	2	0	6	15	0

arcdfad	0	0	0	0	0	0
arcdime	0	0	0	0	0	0
arcdlg	0	0	0	0	0	0
arcdxf	0	0	0	0	0	0
arciges	0	0	0	0	0	0
arcmodeltools	0	1	0	1	1	1
arcshape	0	0	0	3	0	3
ascheckout	7	87	19	0	20	2
asciigrid	0	0	0	0	0	0
as_append	0	4	2	1	3	1
as_drawenv	0	1	0	5	5	5
as_open	0	4	1	0	1	0
as_routines	0	22	13	4	15	9
as_set	0	8	8	0	8	6
as_transbrowse	0	6	0	0	0	0
as_trans_mngr	0	27	2	1	2	1
autocontour	0	12	4	3	8	4
backcover	17	0	6	1	11	1
backenv	9	0	3	1	9	1
backitem	2	0	2	0	1	0
barriers	8	10	0	0	8	2
bas_end	0	6	0	0	0	2
batch-delin-tp-tree	0	6	3	7	0	6
batch-flint	0	0	2	1	0	0
batch-lauren-delin-absolute-slice	0	1	2	1	0	0
batch-lauren-delin-absolute	0	1	2	1	0	0
batch-lauren-delin-range	0	1	2	1	0	0
batch-lauren-delin-standard	0	1	2	1	0	0
bifur	23	6	2	6	8	6
browsecover	0	6	0	0	0	2
browsedb	0	6	0	0	0	0
buf	0	9	4	11	0	7
buffer	0	0	0	0	0	0
calculate_ap	0	3	0	1	1	3
calc_item	0	2	0	1	1	2
camera	92	0	46	49	83	8
center_edit	20	32	0	6	20	15
centroid-albers-meadesranch	0	0	0	0	0	0
centroid-xy	0	0	0	2	2	2
cf_batchmatch	0	0	4	0	4	1
cf_batchshape	6	14	10	10	19	12
cf_driver	7	6	21	5	26	13
cf_manedit	8	1	9	2	13	2
cf_mantran	5	4	5	0	2	0
cf_nodetrans	8	17	29	16	43	23
cf_rpt	0	22	0	4	4	14
cf_setup	7	11	7	8	15	11
check_oracle_table	0	5	0	0	0	0
class_anno	37	1	17	8	28	4
class_barrier	0	12	0	0	0	0
class_box	31	1	0	1	22	2
class_boxfill	34	1	0	13	24	2
class_center	45	24	15	5	33	3
class_circle	23	1	0	1	15	2
class_circlefill	31	1	0	13	24	2

class_cover	56	16	20	8	37	7
class_event	40	19	19	0	30	2
class_graph	49	1	0	13	20	7
class_graticule	17	1	0	11	14	3
class_grat_grid	15	1	0	6	13	2
class_grat_hatch	14	1	0	5	12	1
class_grat_label	23	1	0	5	4	1
class_grat_mrkr	14	1	0	5	12	1
class_grid	23	6	8	14	14	3
class_gridcomp	13	3	8	4	4	2
class_hillshade	5	1	0	4	4	0
class_image	13	5	8	5	4	3
class_key	52	1	0	4	14	2
class_keyfile	62	1	0	18	45	4
class_line_primitive	21	1	0	1	15	2
class_link	48	12	17	4	34	2
class_mapview	70	1	10	4	32	2
class_marker	18	1	0	1	12	2
class_mesh	21	1	0	10	20	1
class_neatline	33	1	0	1	23	2
class_northarrow	36	1	0	2	30	3
class_piechart	18	1	0	8	6	4
class_plotfile	19	1	0	1	8	2
class_polyfill	31	1	0	14	25	2
class_region	65	4	20	18	47	6
class_route	62	8	20	8	43	7
class_scalebar	74	1	0	2	27	9
class_section	39	1	17	4	30	0
class_stop	45	26	15	5	33	3
class_text	12	1	0	3	2	5
class_textfile	12	1	0	3	2	5
class_tin_edge	12	1	3	5	8	1
class_tin_node	19	1	3	4	4	0
class_tin_triangle	23	1	3	7	8	1
clip	0	0	0	0	0	0
cluster	0	5	0	2	2	4
cogo	0	0	0	0	0	0
colorpick	9	0	0	9	9	0
columns	1	16	0	0	0	5
combines_stats	0	0	0	1	0	1
command_tools	0	0	0	2	0	2
composite	0	0	3	10	0	7
con-simple	0	0	0	7	0	7
connect	0	0	0	0	0	0
contour	0	4	0	3	3	4
convertitem	0	8	0	1	2	2
coord	0	0	14	0	0	0
copystack	0	0	0	2	0	2
costpath	32	2	0	14	32	4
cover_mngr	0	11	0	0	0	3
cov_text	29	0	0	4	0	4
create_center	0	6	0	0	0	2
create_cover	0	7	2	0	2	2
create_stop	0	9	0	0	0	4
cut_fill	0	0	0	0	0	0
data_assign-mono	0	4	4	4	0	4
data_assign	0	6	8	4	1	4
data_bin-cut	0	1	4	2	0	0

data_bin-project	0	12	0	0	0	4
dbi_routines	0	3	0	0	0	1
dbmspull	0	4	0	0	0	0
dbmspush	0	6	0	0	0	2
define_sde	0	3	0	0	0	2
deflayer_mngr	2	6	0	0	0	0
demlattice	0	0	0	0	0	0
desymbol	3	0	0	3	3	0
dfadarc	0	0	0	0	0	0
dig_simplemenu	5	0	1	0	1	0
dimearc	0	2	0	0	0	0
dimension_map_generator	0	0	0	0	0	0
disp	33	0	1	4	26	1
disp_delin	6	0	0	0	4	0
disp_dem	0	1	5	2	0	0
disp_image	0	0	0	0	0	0
disp_legend	26	4	0	5	10	0
dissolve	0	0	0	0	0	0
dlgarc	0	0	0	0	0	0
dot_density	12	1	3	4	9	4
drain	0	3	0	8	0	8
drawcover	15	0	4	6	12	4
drawenv	16	0	5	5	13	3
dropfeatures	0	0	0	9	0	9
dtedgrid	0	0	0	0	0	0
dump_delta	0	6	3	4	0	3
dxfarc	3	0	1	0	0	0
edarc	4	9	12	0	13	2
edarcenv	0	0	21	0	21	0
edarc_more	0	0	0	0	0	0
edboundary	3	4	7	0	7	0
edcontrol	4	2	13	1	14	2
edgematch	1	0	12	0	12	0
editfclass	3	0	5	0	4	0
edit_annogen	8	16	15	16	36	21
edit_annopar	10	17	11	16	32	18
edit_fat	0	20	0	8	5	9
edit_fat_calc	0	0	0	5	5	5
edit_land_prop	3	18	12	0	12	2
edit_parcel	3	28	18	2	20	4
edit_poly	3	12	9	0	9	2
edit_table	0	23	0	7	5	9
edit_table_calc	0	0	0	5	5	5
edit_table_sort	0	2	0	0	0	0
edit_tools	5	1	3	1	2	1
edlab	0	1	0	2	2	2
edlabenv	0	0	0	0	0	0
edregion	1	2	1	0	2	1
edroute	6	3	10	0	10	0
edrteenv	0	0	16	0	16	0
ed_backgr	0	6	17	0	17	6
ed_nocogo	1	9	11	0	12	2
erase	0	0	0	0	0	0
etakarc	0	0	0	0	0	0
eventsource_mngr	0	7	7	0	7	0
event_dissolve	0	4	4	0	4	0
event_overlay	0	4	4	0	4	0
event_pullitems	0	1	1	0	1	0

event_transform	0	3	3	0	3	0
export	0	0	0	3	0	3
extended	5	24	46	0	48	2
extract_manual-update	4	4	0	4	3	5
fdr-four	0	2	0	2	0	2
featover	0	13	0	6	5	9
featureprox	2	21	0	0	0	6
fill	0	4	0	0	0	0
fillet_bndry	0	9	3	0	0	0
fill_dem_depressions	0	7	0	0	0	3
floatgrid	0	0	0	0	0	0
flyby	73	64	11	16	72	13
fly_around	73	29	10	11	81	8
formgen	0	8	0	0	0	0
forms	1	7	0	0	1	0
formsinfo	1	11	0	0	1	1
form_maker	1	16	0	0	1	0
fullpath	0	3	0	1	0	2
geaddlinks	0	1	2	0	1	1
gedrawarcs	1	1	5	0	5	2
gegraphic	2	0	0	2	2	0
gen_model	2	3	0	4	2	4
gen_snaps	24	25	15	30	18	12
gesnapopts	2	0	6	0	6	0
getdeflayer	5	3	2	0	0	0
geteventsource	0	6	6	0	6	0
getext	9	2	7	0	6	2
getextprop	4	0	0	2	5	2
getsymset	22	0	0	6	18	0
getsymsetae	4	0	0	1	3	0
gewarp	2	10	10	3	8	5
girasarc	0	2	0	0	0	0
gradsym	8	1	3	5	5	1
graph	26	0	0	2	4	0
graphics_output	28	1	1	4	13	7
graph_theme	41	1	0	11	16	4
grassgrid	0	0	0	0	0	0
gridascii	0	0	0	0	0	0
gridexpressiontools	0	0	0	0	0	0
gridfloat	0	0	0	0	0	0
gridimage	0	0	0	0	0	0
gridline	0	0	0	0	0	0
gridpoint	0	0	0	0	0	0
gridpoly	0	0	0	0	0	0
grid_anal_env	18	0	23	38	16	8
grid_expr_build	1	2	0	25	0	25
grid_mngr	27	9	4	22	19	4
grid_mngr2	0	9	0	0	0	3
grid_modeler	8	0	2	1	2	1
group	0	0	1	0	1	0
grp_edit	0	3	0	0	0	0
hist	37	2	3	6	11	4
hist1back	0	5	0	0	0	1
histdrill	1	11	0	0	0	1
histfeat	2	20	2	0	0	3
hview_gen	0	3	0	0	0	0
hypso	44	5	7	8	19	7
identity	0	0	0	0	0	0

igdsarcc	0	0	0	0	0	0
imagegrid	0	0	0	0	0	0
import	0	0	0	0	0	0
inflow-ranking-ofpl	0	0	0	6	5	6
inflow-ranking	0	0	0	5	4	5
infofile_mngr	0	5	0	0	0	2
inforeport	0	1	0	0	0	1
info_point	0	3	0	2	2	3
integrate	21	26	56	4	52	11
intersect	0	0	0	0	0	0
itemaccum	0	11	0	3	2	8
joinitem	0	1	0	0	0	0
join_bndry	0	9	5	0	5	0
junkrowcol	0	0	3	1	0	0
kriging-s	21	5	0	7	6	1
kriging-su	13	5	0	5	4	1
la	44	88	6	5	33	9
lacandidate	23	34	0	14	29	12
laconfig1	0	20	0	1	0	6
laconfig2	0	20	0	1	0	6
lademand	24	1	0	8	24	0
lanetwork	1	0	1	0	0	0
lasolve	1	8	1	0	0	0
lat	0	3	5	5	2	3
lat2	0	1	5	4	1	2
latticedem	0	0	0	0	0	0
latticetin	0	0	0	0	0	0
lat_driver	0	0	2	2	0	0
lat_gen	1	2	4	0	1	0
lat_reg	0	0	3	0	3	0
layers	9	0	4	1	5	0
layout_bndry	0	13	6	0	3	0
layout_tie	0	8	3	0	3	0
license	0	0	0	0	0	0
line-slope	0	2	1	1	0	0
linesymbol	15	0	0	0	15	0
lineupdate	3	6	19	1	16	6
linkopts	0	0	6	0	6	0
loadmap	5	0	0	0	0	0
logicalae	0	0	0	0	0	0
logicalap	37	32	0	14	35	4
logicalap2	24	6	0	10	24	0
logicalsde	17	2	1	6	14	1
los	21	11	0	1	9	1
main_set	3	0	2	1	0	0
mapdriver	28	6	1	3	9	0
mapprops	12	0	0	0	0	0
map_library	5	13	9	0	11	2
map_object_mngr	19	0	0	5	11	1
map_prefs	6	0	0	2	6	0
map_tools	8	0	2	1	2	1
markersymbol	12	0	0	0	12	0
mask-con	0	0	0	3	0	3
mask-random	0	1	2	2	0	1
mask-select	0	0	0	4	0	4
mask-xy	0	0	2	5	0	4
measure	13	6	0	2	13	3
module_chk	4	0	1	3	4	0

mossarc	0	0	0	0	0	0
mru-combo	0	1	1	8	0	7
mru-slice	0	0	0	3	0	3
mru_dissolve	0	2	0	2	0	2
mru_gen_pre_reg	0	0	3	1	4	1
mru_id-change-assign-display_atts	5	0	0	0	0	0
mru_id-change-reclass	0	2	0	5	4	5
mru_id-change-update	0	0	0	9	9	9
mru_id-change	0	0	0	5	4	5
mru_numbers	4	0	2	0	2	0
nchan_rasterize-highlight	6	3	0	0	6	0
nclim-list	0	13	0	3	3	6
nclim	28	6	3	10	23	9
near	0	0	0	0	0	0
nearstream_reroute	0	3	1	4	0	3
network_edit	10	18	0	5	11	8
newcover	6	4	12	2	6	6
ngrid	0	0	2	1	0	0
node	0	0	1	2	0	1
nodeprop	7	0	0	2	7	0
nofpl	0	11	0	6	0	7
north_arrow	7	0	1	0	9	3
opencover	2	0	3	0	2	0
outdirs_exist	0	0	0	0	0	0
outlet	0	0	2	6	1	4
output-2d	0	2	0	0	0	1
output	0	6	0	0	0	6
pagesetup	10	0	0	0	0	0
panzoom	13	0	4	3	3	3
parallel_bndry	0	8	3	0	0	0
param_2nd_dimension	0	1	0	1	0	1
param_area-1st_order-smallest	0	0	0	1	1	1
param_area-acres	0	0	0	1	1	1
param_area-hectare	0	2	0	0	0	1
param_area-km	0	2	0	0	0	1
param_area-miles-accumulate	0	3	0	2	0	3
param_area-miles	0	0	0	1	1	1
param_area-smallest	0	2	0	1	0	2
param_area-total-nhru-acres	0	0	0	1	1	1
param_area-total-nhru-km	0	2	0	2	0	3
param_aspect-arctan2	0	1	0	3	0	3
param_chan-width	0	0	0	1	1	1
param_cov-den-summer-dominant	0	4	0	5	0	6
param_cov-den-summer	0	9	0	15	10	17
param_cov-den-winter	0	9	0	16	11	18
param_cov-den-winter2	0	3	0	6	0	7
param_cov-den-winter3	0	2	0	6	0	6
param_cov-type-klinefelter	0	2	0	1	0	1
param_cov-type-prms	0	2	0	1	0	2
param_cov-type-prms2	0	4	0	20	1	21
param_cov-type-prms3	0	3	0	22	1	22
param_cov-type	0	2	0	1	0	2
param_daf_pct_area	0	6	0	2	1	5
param_dajunction-down	0	2	0	2	0	3

param_dist2headwater-miles	0	2	0	0	0	1
param_dist2headwater	0	2	0	0	0	1
param_elevation-max-meters	0	2	0	1	0	2
param_elevation-mean-feet	0	2	0	2	0	3
param_elevation-mean-meters	0	2	0	1	0	2
param_elevation-min-meters	0	2	0	0	0	1
param_elevation-range-feet	0	2	0	1	0	2
param_elevation-range-meters	0	2	0	0	0	1
param_elevation-std-meters	0	2	0	1	0	2
param_gen-imperv-binary	0	1	0	2	1	2
param_gen-ov-colrowmap	0	1	0	0	0	1
param_inflow-primary	0	2	0	3	2	3
param_inflow-secondary	0	2	0	3	2	3
param_inflow-tertiary	0	2	0	3	2	3
param_intcp-mean-snow	0	2	0	1	0	2
param_intcp-mean-srain	0	2	0	1	0	2
param_intcp-mean-wrain	0	2	0	1	0	2
param_intcp-snow	0	4	0	2	0	3
param_intcp-snow2	0	2	0	5	0	5
param_intcp-srain	0	4	0	2	0	3
param_intcp-srain2	0	2	0	5	0	5
param_intcp-wrain	0	4	0	2	0	3
param_intcp-wrain2	0	2	0	5	0	5
param_intersect-gwcell-col_id	0	4	0	1	0	2
param_intersect-gwcell-row_id	0	4	0	1	0	2
param_line-slope	0	0	0	3	3	3
param_loni-bin-st-ac	0	1	0	0	0	1
param_loni-mean	0	2	0	1	0	2
param_loni-nbins	0	5	0	5	4	7
param_loni-nbins2	0	5	0	5	4	7
param_loni-nbins3	0	5	0	5	4	7
param_ndanode-local	0	3	0	3	1	4
param_nnny-nnnx-id	0	2	0	0	0	1
param_nnny-nnnx-nssr	0	4	0	4	2	6
param_num-chan	0	0	0	6	6	6
param_ofpl-inflow-primary	0	2	0	1	0	2
param_ofpl-inflow-secondary	0	2	0	1	0	2
param_ofpl-inflow-secondary3	0	2	0	2	0	3
param_ofpl-length	0	0	0	3	3	3
param_one-plane_area	0	2	0	0	0	1
param_one-plane_ellmaj	0	3	0	0	0	1
param_one-plane_ellmin	0	3	0	0	0	1
param_one-plane_ndabranch	0	2	0	1	0	2
param_one-plane_perimeter	0	3	0	0	0	1
param_order-shreve	0	2	0	6	0	6

param_oregon-calibration-assign-display_atts	14	5	2	3	8	0
param_oregon-calibration-assign	0	2	0	142	142	142
param_ov-area-pct	0	5	0	2	1	4
param_ov-area-pct2	0	5	0	3	2	5
param_ov-area	0	3	0	2	1	4
param_perimeter	0	2	0	0	0	1
param_poly2point	0	5	0	1	1	4
param_rock-depth-mean-meters	0	2	0	1	0	2
param_root-depth-mean-meters	0	2	0	1	0	2
param_root-depth	0	2	0	1	0	2
param_slope-10-85	3	21	8	28	10	21
param_slope-degrees-mean	0	2	0	1	0	2
param_slope-mean	0	2	0	1	0	2
param_snow-threshold	0	0	0	1	1	1
param_snowdepletion-curve	0	0	0	2	2	2
param_soil-awc	0	2	0	1	0	2
param_soil-bulk-density	0	2	0	1	0	2
param_soil-depth	0	2	0	1	0	2
param_soil-field-capacity-mean	0	2	0	1	0	2
param_soil-moist-meters	0	2	0	1	0	2
param_soil-organic-matter	0	2	0	1	0	2
param_soil-pct_clay-mean	0	2	0	1	0	2
param_soil-pct_sand-mean	0	2	0	1	0	2
param_soil-pct_silt-mean	0	2	0	1	0	2
param_soil-perm-mean-meters	0	2	0	1	0	2
param_soil-perm	0	2	0	1	0	2
param_soil-porosity-mean	0	2	0	1	0	2
param_soil-szm	0	2	0	1	0	2
param_soil-wilt-point-mean	0	2	0	1	0	2
param_stream-shreve	0	2	0	0	0	1
param_stream-strahler	0	1	0	1	0	1
param_temp-adj-max	0	0	0	3	1	3
param_temp-adj-min	0	0	0	3	1	3
param_topmodel-ach-d	0	6	0	2	1	6
param_topmodel-ach-d2	0	1	0	4	4	5
param_topmodel-ach	0	4	0	6	0	7
param_topmodel-d	0	1	0	2	0	2
param_tree-dom	0	2	0	1	0	2
param_velocity-coefficient	0	0	0	4	4	4
param_wcov-trans-density	0	3	0	6	0	7
param_wcov-trans	0	2	0	1	0	2
param_wcov-trans2	0	1	0	2	0	2
parcel_prefs	3	2	3	2	5	1
parcel_storm	6	24	3	0	7	0
partition	15	32	0	9	19	13
par_addlist	0	0	0	2	1	2
par_ap-dump	0	1	0	0	0	1
par_batch-output	0	1	0	6	3	7
par_combine-zone-param	0	4	0	9	4	10

par_oui-relate	0	1	0	3	0	4
par_overlay-chk	0	5	1	3	0	2
par_rename-item	0	0	0	1	1	1
par_unload	0	2	0	3	3	3
place_subdiv	8	4	21	0	20	1
plotcopies	5	0	0	0	0	0
plotdivide	29	0	0	1	19	0
plotmulti	8	0	0	0	2	0
point2zone	0	4	0	0	0	1
pointdistance	0	0	0	0	0	0
polygon_event	0	1	1	0	1	0
precedence	14	7	0	1	15	3
prim_edtr	0	0	0	0	0	0
profile	20	0	0	0	4	0
propertydriver	16	5	11	6	20	1
property_tools	4	0	0	2	0	2
prop_panzoom	9	0	6	0	5	0
q	0	2	0	5	0	5
quickdraw	47	9	14	5	8	3
quickplot	6	0	0	0	0	0
reclass	15	40	0	13	8	5
rectify	0	0	0	0	0	0
redefine	0	5	0	2	0	2
regionclass	0	0	0	0	0	0
regiondissolve	0	0	0	0	0	0
regionselect	29	12	1	20	28	0
register	0	0	0	0	0	0
relate	0	2	0	1	0	1
relate_mngr	0	6	0	0	0	0
remeassec	0	0	0	12	12	12
remeasure	0	0	2	12	14	12
remove_obj	0	12	9	1	9	1
reudl	6	0	3	0	3	0
rg-cat	0	6	1	3	0	3
rotate_arcs	2	7	9	0	10	1
route_font	13	0	0	0	0	0
route_hatch	0	4	0	0	0	0
route_hatch_font	9	0	0	0	0	0
route_offset	0	4	0	2	0	2
route_text	0	4	0	2	0	2
routing	21	79	8	7	18	13
routing_property	6	0	0	0	6	0
rule_submit	1	3	2	2	0	2
save_object_as	0	2	0	5	0	5
scalebar	20	0	1	1	10	2
scratch_kill	0	2	0	0	0	0
sde_edit_calc	0	0	0	0	0	0
sdtsexport	0	0	0	0	0	0
seed	0	1	0	7	0	7
select-attr	0	0	0	4	0	4
select_sde	7	7	0	0	0	0
selprefs	0	4	4	0	4	0
sel_statasc	0	4	0	0	0	4
setmaplibenv	2	5	2	0	0	0
setnull	0	0	0	2	0	2
setsdeenv	0	7	0	0	0	0
setstormenv	0	8	0	0	0	0
setup_composite	0	1	1	6	0	5

set_analysis_window	0	0	6	0	0	0
set_dimension_point	0	6	9	11	0	6
se_dbmsexists	0	0	0	0	0	0
se_featclass	0	0	0	3	0	3
se_loaddbms	0	2	0	0	0	0
se_loadinfo	0	0	0	0	0	0
sfc_aspect	0	3	0	0	0	1
sfc_cov-type-prms	0	5	0	2	0	3
sfc_cov-type-wt	0	5	0	1	0	2
sfc_cov-type	0	5	0	2	0	3
sfc_downcell-id	0	4	0	2	0	4
sfc_elv-focalmean	0	2	0	0	0	1
sfc_enns-resrv	0	2	0	0	0	1
sfc_enns-topvar	0	2	0	0	0	1
sfc_flow-accumulation	0	3	1	2	0	3
sfc_flow-direction	0	3	0	0	0	1
sfc_flowlength-down-3d	0	2	2	2	0	1
sfc_flowlength-down	0	2	2	6	0	5
sfc_flowlength-up	0	2	2	2	0	1
sfc_focalvariety-data	0	2	0	0	0	1
sfc_focalvariety-nodata	0	2	0	0	0	1
sfc_imperv	0	5	0	1	0	2
sfc_intcp-snow	0	5	0	2	0	3
sfc_intcp-srain	0	5	0	2	0	3
sfc_intcp-wrain	0	5	0	2	0	3
sfc_jh-coef	0	2	0	0	0	1
sfc_jh-coef2	0	2	0	0	0	1
sfc_leaf-loss	0	3	0	1	0	2
sfc_loni-contour-width	0	2	0	1	0	2
sfc_loni-delta-elv	0	7	0	2	0	3
sfc_loni-distance	0	2	0	1	0	2
sfc_loni-fac	0	2	0	4	0	5
sfc_loni	0	2	0	0	0	1
sfc_radpl	0	8	0	3	0	6
sfc_rechr-depth	0	2	0	1	0	2
sfc_reclass-interactive	0	3	0	1	0	2
sfc_reclass	0	4	0	1	0	2
sfc_rock-depth-max	0	3	0	0	0	1
sfc_root-depth	0	6	0	3	0	4
sfc_sinks	0	5	0	1	0	3
sfc_slope-degrees	0	3	0	0	0	1
sfc_slope	0	3	0	0	0	1
sfc_soil-awc	0	4	0	0	0	1
sfc_soil-bulk-density	0	4	0	0	0	1
sfc_soil-depth-meters	0	3	0	0	0	1
sfc_soil-depth	0	3	0	0	0	1
sfc_soil-ne	0	2	0	1	0	2
sfc_soil-organic-matter	0	4	0	0	0	1
sfc_soil-percent-clay	0	4	0	0	0	1
sfc_soil-percent-sand	0	4	0	0	0	1
sfc_soil-percent-silt	0	4	0	0	0	1
sfc_soil-perm	0	4	0	0	0	1
sfc_soil-texture-prms	0	3	0	2	0	3
sfc_soil-wilt-point	0	2	0	1	0	2
sfc_temp-adj-max	0	4	0	1	0	2
sfc_temp-adj-min	0	4	0	1	0	2
sfc_wcov-trans	0	3	0	1	0	2
sfc_wcov-trans2-density	0	3	0	3	0	4

sfc_wcov-trans2	0	2	0	0	0	1
shadesymbol	17	0	0	14	17	1
shapearc	0	0	0	0	0	0
showtable	0	6	0	0	0	0
shutoff	4	96	0	5	7	15
slfarc	0	0	0	0	0	0
slice	22	39	0	15	11	4
snap2grid	12	0	0	10	10	0
snapenv	0	0	1	0	1	0
snapopts	0	0	6	0	6	0
snaprotate	5	22	45	0	47	2
snaprotate2	2	7	9	0	10	1
soils_convert	0	18	2	23	24	27
solrad	0	45	0	54	0	54
solution_edit	8	14	0	0	8	0
spatial	54	5	0	23	47	1
spatialsde	6	0	0	3	3	0
spatialsel	54	53	0	23	47	7
spatial_event	11	15	0	3	12	4
splitbuffer	0	4	0	0	0	0
split_bndry	0	8	3	0	0	0
split_parcel	0	3	2	0	2	0
sql_builder	35	5	0	14	35	0
sql_event	0	8	1	0	1	2
ssmodel	0	5	0	0	0	0
stack_mngr	25	9	4	20	19	4
statistics_ap	0	3	0	1	1	2
stats_tour	0	5	2	5	0	5
stop_edit	20	41	0	7	21	20
stormselect	10	0	12	0	14	2
stream	0	1	0	1	0	1
streamshed	35	32	7	16	32	1
stream_edit	1	6	2	6	5	7
stream_extract	0	1	0	1	0	1
strm_beef	0	0	0	1	1	1
subselect	0	0	0	0	0	0
subselprefs	0	3	3	0	3	0
supervised	4	0	0	1	0	1
surface	0	0	0	0	0	0
surfacelocator	35	0	0	25	35	1
textitem	1	0	2	0	2	0
textsymbol	18	0	0	3	0	3
themeclasses	1	0	0	0	1	0
theme_mngr	10	0	2	6	8	1
thiessen	0	0	0	0	0	0
tinarc	0	0	0	0	0	0
tinvrml	0	0	0	0	0	0
tools	18	25	1	5	19	6
tool_cogo_adjust	5	10	9	2	11	2
tool_qa	6	3	9	1	9	3
topology	0	0	0	11	0	11
traceover	5	12	0	4	5	4
transfer	0	0	0	0	0	0
transform	0	0	0	0	0	0
turn_edit	35	34	2	0	35	6
udlayers	7	0	4	0	3	0
ungenerate	0	0	0	0	0	0
union	0	0	0	0	0	0

unsupervised	0	0	0	0	0	0
vector_display-classify	13	3	1	0	4	1
vector_display-identify	18	0	3	0	4	0
vector_display-measure	16	0	1	0	6	0
vector_overlay-classify	9	3	1	0	4	1
vector_overlay-identify	4	0	2	0	2	0
vector_overlay-measure	13	0	2	0	7	0
veriplot	13	0	2	0	9	1
version	0	14	4	4	2	4
view	10	0	0	5	10	0
viewdriver	55	35	36	9	32	7
viewplot	23	0	2	0	4	0
view_prefs	13	0	7	3	11	1
view_select	6	10	12	0	9	3
view_zoom	24	6	20	3	4	4
volume	0	1	0	0	0	1
v_dclassify	5	10	0	3	3	3
v_dlegend	29	16	0	11	15	10
workspace	0	0	0	0	0	0
workspace_mngr	0	0	0	0	0	0
zonalstat_factory	0	0	0	3	1	3
zone_accumulation	0	2	0	0	0	1
zone_area-firstorder	0	3	0	3	2	4
zone_areas-internal	0	2	0	3	0	4
zone_centroid-point	0	2	0	0	0	1
zone_centroid	0	7	0	3	0	9
zone_chan-segs-local	0	2	0	2	0	3
zone_chan-segs	0	5	0	8	8	9
zone_distance-euclidean	0	2	0	0	0	1
zone_distance-flowlength	0	2	0	4	0	5
zone_down-id	0	2	0	4	0	5
zone_fac-min-pt	0	2	0	1	0	2
zone_flow-accumulation	0	2	0	3	0	4
zone_flow-accumulation2	0	2	0	2	0	3
zone_fullpath	0	12	4	21	2	19
zone_fullpath2	0	12	4	19	2	17
zone_headwater-area	0	2	0	0	0	1
zone_headwater-pts	0	2	0	2	0	3
zone_headwaters-internal	0	2	0	1	0	2
zone_internal-cells	0	2	0	2	0	3
zone_loni-bin	0	14	0	5	4	9
zone_loni-bin2	0	14	0	5	4	9
zone_loni-bin3	0	14	0	5	4	9
zone_loni	0	2	0	0	0	1
zone_main-link-tops	0	2	0	1	0	2
zone_main-link	0	11	2	25	0	24
zone_nchan-id	0	1	0	1	1	2
zone_ndajunction	0	2	0	8	4	9
zone_ndanode	0	3	0	9	3	10
zone_ntopchan-headwater-small	0	2	0	2	0	3
zone_ntopchan-local	0	3	0	12	1	13
zone_ntopchan-local2	0	2	0	3	0	4
zone_ntopchan-mainlink	0	2	0	1	0	2
zone_ntopchan-segs	0	7	0	2	1	4
zone_ntopchan	0	3	0	5	4	6
zone_offset-pp-elevation	0	2	0	1	0	2
zone_offset-pp-flowlength	0	2	0	1	0	2

zone_offset-pp2pp-elevation	0	5	0	2	1	5
zone_offset-pp2pp-flowlength	0	5	0	2	1	5
zone_one-plane	0	2	0	0	0	1
zone_out-dsheds	0	2	0	1	0	2
zone_out-fdr-cmb	0	2	0	1	0	2
zone_out-fdr	0	2	0	1	0	2
zone_out-maxfac-flag	0	2	0	1	0	2
zone_out-maxfac	0	2	0	2	0	3
zone_outlet-downstream2	0	3	0	1	0	2
zone_outlets-downstream2	0	2	0	10	0	11
zone_perimeter-all	0	2	0	3	0	4
zone_perimeter-dsheds	0	2	0	1	0	2
zone_perimeter-external	0	2	0	2	0	3
zone_perimeter-headwater-external	0	2	0	1	0	2
zone_perimeter-headwater	0	2	0	0	0	1
zone_radpl	0	3	0	6	3	7
zone_range-absolute-slice-x	0	2	0	1	0	2
zone_range-absolute-slice-y	0	2	0	1	0	2
zone_range-absolute-slice-z	0	2	0	1	0	2
zone_range-absolute-x	0	3	0	1	0	2
zone_range-absolute-y	0	3	0	1	0	2
zone_range-absolute-z	0	3	0	1	0	2
zone_range-relative-x	0	2	0	2	2	3
zone_range-relative-y	0	2	0	2	2	3
zone_range-relative-z	0	2	0	2	2	3
zone_route-non-ordered	0	2	0	2	0	3
zone_route-ordered	0	5	0	3	3	5
zone_shape-ratio	0	7	0	5	2	9
zone_slice	0	5	0	4	2	5
zone_strink	0	2	0	0	0	1
zone_three-plane	0	2	0	2	0	3
zone_tops	0	2	0	1	0	2
zone_two-plane-network	0	3	0	5	0	6
zone_two-plane	0	3	0	5	0	6
zone_two-plane2	0	3	0	7	0	8
zone_watershed	0	2	0	0	0	1
zone_x	0	2	0	1	0	2
zone_y	0	2	0	1	0	2
zone_z	0	2	0	1	0	2

Appendix C. Tables of Best Matching Units for GIS Procedures

This appendix includes a table for each of the SOMs developed in this dissertation. Each row identifies a neuron in the corresponding SOM. The second column lists the GIS procedures that were found to best match that neuron. The best match was determined for each GIS procedure by the finding the neuron to which the Euclidean distance was the shortest. The Euclidean distance was calculated using the values of the keywords in the version of the procedure matrix used to train the SOM as a coordinate specification. In the case of the default and optimized SOMs, the coordinates are defined by the explicit keywords. For the PCA-driven SOM, the first 15 principal component coordinates are used. For the Albrecht and Enviro Modeling SOMs, the keywords are the implicit Albrecht and Enviro Modeling keywords, respectively. Note that for the data points (i.e. the GIS procedures), values for these keywords are the values within the procedure matrix. For instance, the coordinates of the GIS procedure, zone_z, according to the implicit Enviro Modeling keywords are (0,2,0,1,0,2), as is shown in the last row of the preceding table (Table B-2). In order to make the layout of these tables more compact, identifications numbers associated with the GIS procedures are listed instead of the procedure name. The identification numbers were alphabetically assigned, as given in Table C-1. Refer to the appropriate figure for a display showing the layout of the neuron identification numbers (Figure 4-2, Figure 4-6, Figure 4-11, Figure 5-1, and Figure 5-7 correspond to the default, optimized, PCA-driven, Albrecht implicit, and Enviro Modeling implicit SOMs, respectively).

Table C-1 Identification numbers for GIS procedures.

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
1	addalias	31	ap_graph	61	batch-lauren-delin-standard	91	class_graph
2	addedit	32	ap_mru-num	62	bifur	92	class_graticule
3	addgenerate	33	ap_mru-select	63	browsecover	93	class_grat_grid
4	addinput	34	arcdfad	64	browsedb	94	class_grat_hatch
5	additem	35	arcdime	65	buf	95	class_grat_label
6	addmeasure	36	arcdlg	66	buffer	96	class_grat_mrkr
7	addresscreate	37	arcdxf	67	calculate_ap	97	class_grid
8	addressenv	38	arciges	68	calc_item	98	class_gridcomp
9	addressfile	39	arcmodeltools	69	camera	99	class_hillshade
10	address_select	40	arcshape	70	center_edit	100	class_image
11	add_select	41	ascheckout	71	centroid-albers-meadesranch	101	class_key
12	adrggrid	42	asciigrid	72	centroid-xy	102	class_keyfile
13	aeclean	43	as_append	73	cf_batchmatch	103	class_line_primitive
14	aedefaults	44	as_drawenv	74	cf_batchshape	104	class_link
15	aedriver	45	as_open	75	cf_driver	105	class_mapview
16	aesymset	46	as_routines	76	cf_manedit	106	class_marker
17	ae_library	47	as_set	77	cf_mantran	107	class_mesh
18	ae_project	48	as_transbrowse	78	cf_nodetrans	108	class_neatline
19	aggregate	49	as_trans_mngr	79	cf_rpt	109	class_northarrow
20	allocation	50	autocontour	80	cf_setup	110	class_piechart
21	annoadd	51	backcover	81	check_oracle_table	111	class_plotfile
22	annoaddenv	52	backenv	82	class_anno	112	class_polyfill
23	annoedit	53	backitem	83	class_barrier	113	class_region
24	annoposline	54	barriers	84	class_box	114	class_route
25	annopospoint	55	bas_end	85	class_boxfill	115	class_scalebar
26	aoi_delineation-auto	56	batch-delin-tp-tree	86	class_center	116	class_section
27	aoi_delineation-pre	57	batch-flint	87	class_circle	117	class_stop
28	aoi_delineation-seed	58	batch-lauren-delin-absolute-slice	88	class_circlefill	118	class_text
29	append	59	batch-lauren-delin-absolute	89	class_cover	119	class_textfile
30	ap_delin	60	batch-lauren-delin-range	90	class_event	120	class_tin_edge

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
121	class_tin_node	151	define_sde	181	edit_annogen	211	featureprox
122	class_tin_triangle	152	deflayer_mngr	182	edit_annopar	212	fill
123	clip	153	demlattice	183	edit_fat	213	fillet_bndry
124	cluster	154	desymbol	184	edit_fat_calc	214	fill_dem-_depressions
125	cogo	155	dfadarc	185	edit_land_prop	215	floatgrid
126	colorpick	156	dig_simplemenu	186	edit_parcel	216	flyby
127	columns	157	dimearc	187	edit_poly	217	fly_around
128	combines_stats	158	dimension_map-_generator	188	edit_table	218	formgen
129	command_tools	159	disp	189	edit_table_calc	219	forms
130	composite	160	disp_delin	190	edit_table_sort	220	formsinfo
131	con-simple	161	disp_dem	191	edit_tools	221	form_maker
132	connect	162	disp_image	192	edlab	222	fullpath
133	contour	163	disp_legend	193	edlabenv	223	geaddlinks
134	convertitem	164	dissolve	194	edregion	224	gedrawarcs
135	coord	165	dlgarc	195	edroute	225	gegraphic
136	copystack	166	dot_density	196	edrteenv	226	gen_model
137	costpath	167	drain	197	ed_backgr	227	gen_snaps
138	cover_mngr	168	drawcover	198	ed_nocogo	228	gesnapopts
139	cov_text	169	drawenv	199	erase	229	getdeflayer
140	create_center	170	dropfeatures	200	etakarc	230	geteventsource
141	create_cover	171	dtedgrid	201	eventsource-_mngr	231	getext
142	create_stop	172	dump_delta	202	event_dissolve	232	getextprop
143	cut_fill	173	dxfarc	203	event_overlay	233	getsymset
144	data_assign-mono	174	edarc	204	event_pullitems	234	getsymsetae
145	data_assign	175	edarcenv	205	event_transform	235	gewarp
146	data_bin-cut	176	edarc_more	206	export	236	girasarc
147	data_bin-project	177	edboundary	207	extended	237	gradsym
148	dbi_routines	178	edcontrol	208	extract_manual-update	238	graph
149	dbmspull	179	edgematch	209	fdr-four	239	graphics-_output
150	dbmspush	180	editfclass	210	featover	240	graph_theme

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
241	grassgrid	271	integrate	301	loadmap	331	nchan_raster-ize-highlight
242	gridascii	272	intersect	302	logicalae	332	nclim-list
243	gridexpression-tools	273	itemaccum	303	logicalap	333	nclim
244	gridfloat	274	joinitem	304	logicalap2	334	near
245	gridimage	275	join_bndry	305	logicalsde	335	nearstream-_reroute
246	gridline	276	junkrowcol	306	los	336	network_edit
247	gridpoint	277	kriging-s	307	main_set	337	newcover
248	gridpoly	278	kriging-su	308	mapdriver	338	ngrid
249	grid_anal_env	279	la	309	mapprops	339	node
250	grid_expr_build	280	lacandidate	310	map_library	340	nodeprop
251	grid_mngr	281	laconfig1	311	map_object_mngr	341	nofpl
252	grid_mngr2	282	laconfig2	312	map_prefs	342	north_arrow
253	grid_modeler	283	lademand	313	map_tools	343	opencover
254	group	284	lanetwork	314	markersymbol	344	outdirs_exist
255	grp_edit	285	lasolve	315	mask-con	345	outlet
256	hist	286	lat	316	mask-random	346	output-2d
257	hist1back	287	lat2	317	mask-select	347	output
258	histdrill	288	latticedem	318	mask-xy	348	pagesetup
259	histfeat	289	latticetin	319	measure	349	panzoom
260	hview_gen	290	lat_driver	320	module_chk	350	parallel_bndry
261	hypso	291	lat_gen	321	mossarc	351	param_2nd_dimension
262	identity	292	lat_reg	322	mru-combo	352	param_area-1st_order-smallest
263	igdsarcc	293	layers	323	mru-slice	353	param_area-acres
264	imagegrid	294	layout_bndry	324	mru_dissolve	354	param_area-hectare
265	import	295	layout_tie	325	mru_gen_pre_reg	355	param_area-km
266	inflow-ranking-ofpl	296	license	326	mru_id-change-assign-display_atts	356	param_area-miles-accumulate
267	inflow-ranking	297	line-slope	327	mru_id-change-reclass	357	param_area-miles
268	infofile_mngr	298	linesymbol	328	mru_id-change-update	358	param_area-smallest
269	inforeport	299	lineupdate	329	mru_id-change	359	param_area-total-nhru-acres
270	info_point	300	linkopts	330	mru_numbers	360	param_area-total-nhru-km

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
361	param_aspect-arctan2	391	param_intcp-mean-wrain	421	param_oregon-calibration-assign	451	param_temp-adj-max
362	param_chan-width	392	param_intcp-snow	422	param_ov-area-pct	452	param_temp-adj-min
363	param_cov-den-summer-dominant	393	param_intcp-snow2	423	param_ov-area-pct2	453	param_topmodel-ach-d
364	param_cov-den-summer	394	param_intcp-srain	424	param_ov-area	454	param_topmodel-ach-d2
365	param_cov-den-winter	395	param_intcp-srain2	425	param_perimeter	455	param_topmodel-ach
366	param_cov-den-winter2	396	param_intcp-wrain	426	param_poly2point	456	param_topmodel-d
367	param_cov-den-winter3	397	param_intcp-wrain2	427	param_rock-depth-mean-meters	457	param_tree-dom
368	param_cov-type-klinefelter	398	param_intersect-gwcell-col_id	428	param_root-depth-mean-meters	458	param_velocity-coefficient
369	param_cov-type-prms	399	param_intersect-gwcell-row_id	429	param_root-depth	459	param_wcov-trans-density
370	param_cov-type-prms2	400	param_line-slope	430	param_slope-10-85	460	param_wcov-trans
371	param_cov-type-prms3	401	param_loni-bin-st-ac	431	param_slope-degrees-mean	461	param_wcov-trans2
372	param_cov-type	402	param_loni-mean	432	param_slope-mean	462	parcel_prefs
373	param_daf_pct_area	403	param_loni-nbins	433	param_snow-threshold	463	parcel_storm
374	param_dajunction-down	404	param_loni-nbins2	434	param_snowdepletion-curve	464	partition
375	param_dist2headwater-miles	405	param_loni-nbins3	435	param_soil-awc	465	par_addlist
376	param_dist2headwater	406	param_ndanode-local	436	param_soil-bulk-density	466	par_ap-dump
377	param_elevation-max-meters	407	param_nnny-nnnx-id	437	param_soil-depth	467	par_batch-output
378	param_elevation-mean-feet	408	param_nnny-nnnx-nssr	438	param_soil-field-capacity-mean	468	par_combine-zone-param
379	param_elevation-mean-meters	409	param_num-chan	439	param_soil-moist-meters	469	par_oui-relate
380	param_elevation-min-meters	410	param_ofpl-inflow-primary	440	param_soil-organic-matter	470	par_overlay-chk
381	param_elevation-range-feet	411	param_ofpl-inflow-secondary	441	param_soil-pct_clay-mean	471	par_rename-item
382	param_elevation-range-meters	412	param_ofpl-inflow-secondary3	442	param_soil-pct_sand-mean	472	par_unload
383	param_elevation-std-meters	413	param_ofpl-length	443	param_soil-pct_silt-mean	473	place_subdiv
384	param_gen-imperv-binary	414	param_one-plane_area	444	param_soil-perm-mean-meters	474	plotcopies
385	param_gen-ov-colrowmap	415	param_one-plane_ellmaj	445	param_soil-perm	475	plotdivide
386	param_inflow-primary	416	param_one-plane_ellmin	446	param_soil-porosity-mean	476	plotmulti
387	param_inflow-secondary	417	param_one-plane_ndabranch	447	param_soil-szm	477	point2zone
388	param_inflow-tertiary	418	param_one-plane_perimeter	448	param_soil-wilt-point-mean	478	pointdistance
389	param_intcp-mean-snow	419	param_order-shreve	449	param_stream-shreve	479	polygon_event
390	param_intcp-mean-srain	420	param_oregon-calibration-assign-display_atts	450	param_stream-strahler	480	precedence

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
481	prim_edtr	511	rule_submit	541	sfc_flow-accumulation	571	sfc_soil-depth-meters
482	profile	512	save_object_as	542	sfc_flow-direction	572	sfc_soil-depth
483	propertydriver	513	scalebar	543	sfc_flowlength-down-3d	573	sfc_soil-ne
484	property_tools	514	scratch_kill	544	sfc_flowlength-down	574	sfc_soil-organic-matter
485	prop-_panzoom	515	sde_edit_calc	545	sfc_flowlength-up	575	sfc_soil-percent-clay
486	q	516	sdtsexport	546	sfc_focalvariety-data	576	sfc_soil-percent-sand
487	quickdraw	517	seed	547	sfc_focalvariety-nodata	577	sfc_soil-percent-silt
488	quickplot	518	select-attr	548	sfc_imperv	578	sfc_soil-perm
489	reclass	519	select_sde	549	sfc_intcp-snow	579	sfc_soil-texture-prms
490	rectify	520	selprefs	550	sfc_intcp-srain	580	sfc_soil-wilt-point
491	redefine	521	sel_statasc	551	sfc_intcp-wrain	581	sfc_temp-adj-max
492	regionclass	522	setmaplibenv	552	sfc_jh-coef	582	sfc_temp-adj-min
493	region-dissolve	523	setnull	553	sfc_jh-coef2	583	sfc_wcov-trans
494	Regionselect	524	setsdeenv	554	sfc_leaf-loss	584	sfc_wcov-trans2-density
495	register	525	setstormenv	555	sfc_loni-contour-width	585	sfc_wcov-trans2
496	relate	526	setup-_composite	556	sfc_loni-delta-elv	586	shadesymbol
497	relate-_mngr	527	set_analysis-_window	557	sfc_loni-distance	587	shapearc
498	remeassec	528	set_dimension_point	558	sfc_loni-fac	588	showtable
499	remeasure	529	se_dbmsexists	559	sfc_loni	589	shutoff
500	remove-_obj	530	se_featclass	560	sfc_radpl	590	slfarc
501	reudl	531	se_loaddbms	561	sfc_rechr-depth	591	slice
502	rg-cat	532	se_loadinfo	562	sfc_reclass-interactive	592	snap2grid
503	rotate_arcs	533	sfc_aspect	563	sfc_reclass	593	snapenv
504	route_font	534	sfc_cov-type-prms	564	sfc_rock-depth-max	594	snapopts
505	route-_hatch	535	sfc_cov-type-wt	565	sfc_root-depth	595	snaprotate
506	route_-hatch_font	536	sfc_cov-type	566	sfc_sinks	596	snaprotate2
507	route-_offset	537	sfc_downcell-id	567	sfc_slope-degrees	597	soils_convert
508	route_text	538	sfc_elv-focalmean	568	sfc_slope	598	solrad
509	routing	539	sfc_enns-resrv	569	sfc_soil-awc	599	solution_edit
510	routing-_property	540	sfc_enns-topvar	570	sfc_soil-bulk-density	600	spatial

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
601	spatialsde	631	tinvrml	661	workspace	691	zone_ndanode
602	spatialsel	632	tools	662	workspace_mngr	692	zone_ntopchan-headwater-small
603	spatial_event	633	tool_cogo_adjust	663	zonalstat_factory	693	zone_ntopchan-local
604	splitbuffer	634	tool_qa	664	zone_accumulation	694	zone_ntopchan-local2
605	split_bndry	635	topology	665	zone_area-firstorder	695	zone_ntopchan-mainlink
606	split_parcel	636	traceover	666	zone_areas-internal	696	zone_ntopchan-segs
607	sql_builder	637	transfer	667	zone_centroid-point	697	zone_ntopchan
608	sql_event	638	transform	668	zone_centroid	698	zone_offset-pp-elevation
609	ssmodel	639	turn_edit	669	zone_chan-segs-local	699	zone_offset-pp-flowlength
610	stack_mngr	640	udlayers	670	zone_chan-segs	700	zone_offset-pp2pp-elevation
611	statistics_ap	641	ungenerate	671	zone_distance-euclidean	701	zone_offset-pp2pp-flowlength
612	stats_tour	642	union	672	zone_distance-flowlength	702	zone_one-plane
613	stop_edit	643	unsupervised	673	zone_down-id	703	zone_out-dsheds
614	stormselect	644	vector_display-classify	674	zone_fac-min-pt	704	zone_out-fdr-cmb
615	stream	645	vector_display-identify	675	zone_flow-accumulation	705	zone_out-fdr
616	streamshed	646	vector_display-measure	676	zone_flow-accumulation2	706	zone_out-maxfac-flag
617	stream_edit	647	vector_overlay-classify	677	zone_fullpath	707	zone_out-maxfac
618	stream_extract	648	vector_overlay-identify	678	zone_fullpath2	708	zone_outlet-downstream2
619	strm_beef	649	vector_overlay-measure	679	zone_headwater-area	709	zone_outlets-downstream2
620	subselect	650	veriplot	680	zone_headwater-pts	710	zone_perimeter-all
621	subselprefs	651	version	681	zone_headwaters-internal	711	zone_perimeter-dsheds
622	supervised	652	view	682	zone_internal-cells	712	zone_perimeter-external
623	surface	653	viewdriver	683	zone_loni-bin	713	zone_perimeter-headwater-external
624	surfacelocator	654	viewplot	684	zone_loni-bin2	714	zone_perimeter-headwater
625	textitem	655	view_prefs	685	zone_loni-bin3	715	zone_radpl
626	textsymbol	656	view_select	686	zone_loni	716	zone_range-absolute-slice-x
627	themeclasses	657	view_zoom	687	zone_main-link-tops	717	zone_range-absolute-slice-y
628	theme_mngr	658	volume	688	zone_main-link	718	zone_range-absolute-slice-z
629	thiessen	659	v_dclassify	689	zone_nchan-id	719	zone_range-absolute-x
630	tinarc	660	v_dlegend	690	zone_ndajunction	720	zone_range-absolute-y

Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure	Id	GIS Procedure
721	zone_range-absolute-z						
722	zone_range-relative-x						
723	zone_range-relative-y						
724	zone_range-relative-z						
725	zone_route-non-ordered						
726	zone_route-ordered						
727	zone_shape-ratio						
728	zone_slice						
729	zone_strink						
730	zone_three-plane						
731	zone_tops						
732	zone_two-plane-network						
733	zone_two-plane						
734	zone_two-plane2						
735	zone_watershed						
736	zone_x						
737	zone_y						
738	zone_z						

Table C-2 SOM neurons and associated GIS procedure matches based on the default training procedure.

Best Matching Neuron	GIS Procedures
1	41,86,89,104,117
2	655
3	82,90,113,114,116,261,653
4	105
5	304,494,600,607
6	33,109,283,305,314,319,480,650
7	(none)
8	134,163,182,659
9	83,218,219,220,221,227,306,660
10	(none)
11	70,303
12	280,613,639
13	10,279,509,589,602
14	(none)
15	216,217,489,591
16	69,657
17	32,97,159,420,487,513,649
18	256
19	84,85,87,88,91,101,103,108,112,115,240,251,610
20	102,111
21	96,106,166,233,342
22	(none)
23	50,255
24	609
25	49,54,258,285
26	281,282,599,603,632
27	20,211,336,464
28	2
29	259
30	183,188
31	30,92,93,94,98,100,107,120,121,122,126,237,298,311,482,586,592,624,628,644,646,652,654
32	(none)
33	(none)
34	(none)
35	95,110,249
36	118,119,139,626,635,645
37	498,499
38	232,310,500
39	(none)
40	124,142,463,475
41	79,636
42	(none)
43	(none)
44	137,273
45	19
46	(none)
47	46,197
48	74,75,78,174,178,185,186,187,195,198,207,271,473,503,595,596,633,656
49	337
50	(none)
51	21,22,40,129,136,226,484,491,512,530

52	15
53	6,301
54	474,476
55	4,71,141,344,385
56	9
57	144,287,332,346,408,422,424,526
58	145,146,401,453,668,727
59	327,403,404,405,454,468,617
60	210,308,333,467,472
61	80,213,275,294,295,350,605
62	14,47,76,299,483,614
63	177,179,235,634
64	43,77,180,485,606
65	51,52,168,169,191,349
66	29,339,496
67	345
68	158,206,274,466
69	57,128,143,296,623,658
70	133,325,352,353,357,433,451,452,465,471,619,663,689
71	72,359,407,492
72	3,267,286,726
73	239,423,715
74	328
75	409,670
76	81,127,149,150,309,348,462,505,507,508,588,608
77	175,196,228,300,594
78	73,224,231,343,625
79	18,31,45,53,194,229,254,504,506,647
80	156,173,238,253,313,340,488
81	(none)
82	5,12,34,35,36,37,38,42,66,123,153,155,164,165,171,199,200,215,241,242,244,245,246,247,248,262,264,265,272,288,289,290,321,334,478,490,493,495,516,587,590,629,630,631,638,641,642,661,662
83	(none)
84	469,521
85	384
86	386,387,388,434,722,723,724
87	400,413,458,690,697
88	266,329
89	364,365,597
90	(none)
91	17,63,64,152,201,202,203,205,230,268,519,522,524,525
92	1,135,151,223,291,347
93	7,11,13,48,154,170,260,292,293,501,511,520,593,621,622,640,648
94	8,24,25,39,67,68,99,125,132,140,148,160,162,176,190,192,193,204,243,257,263,269,270,276,284,302,307,338,362,479,481,515,529,531,532,601,611,620,627,637,643
95	16,225,234,312,320,326,330,331,510
96	(none)
97	(none)
98	354,355,457,615
99	157,236,375,376,514,552,553,585,716,717,718
100	222,360,426,583,584,692,694
101	373,406,665,700,701
102	691,728
103	44,184,189
104	181
105	23,421
106	131,250,252,315,317,318,518,523

107	(none)
108	(none)
109	58,59,60,61,161,323,527
110	351,450
111	297,368
112	(none)
113	335
114	374,725
115	208
116	28,419,612
117	56,147,172,528,556,696
118	62,651,683,684,685
119	(none)
120	(none)
121	497
122	369,372,377,378,379,380,381,382,383,389,390,391,398,399,402,410,411,414,415,416,417,418,425,427,428,429,431,432,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,460,477,538,539,540,543,545,546,547,554,559,562,564,569,570,571,572,574,575,576,577,578,580,664,667,671,679,686,702,704,714,719,720,721,729,735
123	(none)
124	358,412,555,557,561,573,674,676,681,682,687,695,698,699,703,705,706,707,711,712,713,730,731,736,737,738
125	209,316,361,456,461,618
126	55,138,356,470,533,537,542,563,567,568,581,582,708
127	26,212,214,392,394,396,502,534,535,536,548,549,550,551,560,566,604
128	277,278,565
129	324,393,395,397,541,544,558,579,666,669,672,675,680,710
130	27,130,322,363,366,367,455,459,486,517,673,732,733,734
131	167,693
132	341,370,371,677,678,688,709
133	65
134	598
135	430,616

Table C-3 SOM neurons and associated GIS procedure matches based on the optimized training. Note only neurons that were best-matching units are included.

Best Matching Neuron	GIS Procedures
1	307
6	12,34,35,36,37,38,42,66,123,153,155,164,165,171,199,200,215,241,242,244,245,246,247,248,262,264,265,272,288,289,321,334,478,490,493,495,516,587,590,629,630,631,638,641,642,661,662
13	352,433,689
16	697
19	458
23	23
27	181
35	50
37	498,499
48	202,203
53	339
60	5
61	133
66	267
68	690,691
80	182
85	218,219,220
91	250
93	131
94	317,318,518
95	315,523
98	201,230
100	205
102	622
104	129,136
105	29
114	325,353,357,451,452,465,471,619,663
116	72
120	266,329,715
123	597
127	421
134	83,221
137	134
154	530
159	142
163	128
183	183,188
187	306
190	659
191	255,609
198	326
201	170
202	7,206
205	40,512
207	226
213	57,143,296,623
218	287
219	286
221	3
222	332

224	409
225	328
227	365
244	647
245	644
247	31
248	513
250	506
255	22
260	6
264	658
273	239
275	327
278	364
279	670
283	217
290	660
292	163
295	420
297	645,646
300	504
306	635
308	484
310	173
314	124
317	141
319	359
322	408
336	216
338	489,591
340	227
345	32,330
347	649
353	482
359	253,313
360	191
361	488
363	474,476
366	9
368	4
369	71,344
370	492
372	346,407
373	401
376	454
378	403,404,405,468
382	333
396	655
400	139,626
403	238
406	118,119
407	110
420	385
424	424
425	453
426	423
430	472
431	467

434	210
436	19
444	82
445	116
453	95
456	106
460	654
466	301
469	375,376,552,553,585
475	422
477	668
478	727
483	308
484	137
493	89,104
497	90
500	62,256
505	240
509	111
512	156
515	309,348
523	360
525	373,726
531	146
535	273
536	259
541	2
546	114
547	113
550	261
555	91
570	716,717,718
573	222,426
575	692
578	526
580	144
582	145
584	617
589	509
595	86,117
604	101,115
610	84,108
615	496
620	700,701
625	584,694
628	419
629	612
634	528
636	651
650	653
658	85,88,112
660	87,103
663	475
666	497
668	398,399
673	374,725
676	384
677	583

678	406
683	683,684,685
691	589
692	279
694	10
695	602
704	69
707	102
713	311
714	298
724	457
727	157,236,514
729	28
731	728
738	430
749	303
756	105
760	159
766	233
770	477,569,570,574,575,576,577,578
774	354,355
786	172,556
791	616
797	639
802	494
803	600
812	107
814	93,94
816	305
827	368
828	297
832	277,278
835	147,696
840	598
845	280
847	613
854	607
857	487
861	249
865	92
866	628
870	415,416,418,564,571,572
877	55
880	212,470,604
885	214
886	560
898	70
901	336
902	632
904	304
910	251,610
915	126,586
916	592
918	96
925	369,372,410,411,417
929	533,542,567,568
932	535,548
936	566

939	65,341
942	688
946	281,282
948	20
953	603
958	652
963	97
968	99
981	563,581,582
985	534,536,549,550,551
987	502,565
989	56
994	677,678
998	599
1002	464
1007	624
1010	33
1012	657
1020	601
1023	378,379,389,390,391,402,427,428,429,431,432,435,436,437,438,439,440,441,442,443,444,445,446,447,448,460
1030	554,562,719,720,721
1034	392,394,396
1044	370,371
1050	285
1052	211
1054	54
1055	319
1065	98,100
1067	121,122
1068	120
1078	380,382,414,425,449,538,539,540,546,547,559,664,667,671,679,686,702,714,729,735
1086	335
1088	26
1090	363
1091	27
1092	167
1095	709
1100	49
1102	258
1104	636
1107	314,480
1109	109
1110	650
1111	30,342
1113	166
1114	349
1118	237
1122	64,491
1135	356,537,708
1138	324
1139	208
1142	734
1145	693
1150	41
1160	283
1165	485
1166	77

1167	18
1172	138,252
1174	67
1175	68
1180	377,580,704
1188	541
1195	366,459,673
1197	455
1199	463
1205	46
1207	15,500
1214	340
1219	293,501,640
1220	648
1224	63,268
1229	543,545
1232	381,383
1236	358,412,555,557,561,573,674,681,687,698,699,703,705,706,711,713,731,736,737,738
1239	579,669,680
1243	732,733
1255	47
1256	197
1259	310
1260	483
1264	51
1265	52
1277	331,510
1281	161
1284	707
1292	666,675,710
1293	558,672
1295	486,517
1298	367,393,395,397
1303	178
1307	299
1309	235
1314	168,169
1318	522,525
1321	229,519
1323	152,524
1325	347
1329	312,320
1332	527
1341	676,682,712,730
1346	322
1352	75
1355	195
1357	656
1360	224
1362	614
1363	14
1369	17
1376	151
1377	511
1380	16,234
1383	276
1385	316
1387	361,456,461

1389	209,615,618
1393	323
1396	544
1397	130
1409	633
1410	634
1412	76
1418	228
1422	260
1423	48
1429	160
1431	225
1434	338
1438	351,450
1442	695
1446	345
1451	80
1452	74
1454	271
1457	473
1458	174
1461	177
1463	180
1465	337
1467	1,223,231,291
1470	179,300,594
1472	73
1474	43,45
1475	257
1477	140,148,232
1486	58,59,60,61
1499	192
1507	207,595
1510	198
1526	606
1528	254,343,625
1531	13,292
1533	593,627
1534	8,24,25,125,132,162,176,193,243,263,302,481,515,529,532,620,637,643
1538	290
1541	269
1542	469
1543	362
1545	665
1547	722,723,724
1548	434
1553	79
1562	503,596
1565	275,295
1568	135
1578	11,39,190,194,270
1581	520,621
1591	158
1593	521
1597	386,387,388
1600	400
1601	21
1602	44

1606	78
1609	186
1615	213,294,350,605
1621	175,196
1624	505,507,508
1625	127,588
1626	81,149,150
1629	531,611
1632	462
1634	53,284
1641	274
1643	466
1650	413
1652	184,189
1661	185
1663	187
1679	608
1682	204,479
1683	154

Table C-4 SOM neurons and associated GIS procedure matches based on the PCA training.

Best Matching Neuron	GIS Procedures
1	(none)
2	252
3	53,63,173,202,203,205,225,250,268,284,320,488,496,524,525,527
4	16,22,52,129,131,136,234,253,313,340,484,512,530,648
5	342,501,640
6	30,106,120,126,233,592,628,652
7	92,107,298,305,311,586
8	159,475
9	(none)
10	97,251,487,610,657
11	655
12	323
13	(none)
14	(none)
15	58,59,60,61,161,291,351,369,372,377,378,379,380,381,382,383,389,390,391,402,410,411,414,415,416,417,418,425,427,428,429,431,432,435,436,437,438,439,440,441,442,443,444,445,446,447,448,449,450,460,538,539,540,543,545,546,547,554,559,562,564,571,572,580,664,667,671,679,686,702,704,714,719,720,721,729,735
16	(none)
17	276,318
18	7,8,13,17,24,25,29,48,125,132,148,151,152,154,162,176,193,204,243,260,263,302,315,317,338,339,347,479,481,505,507,508,515,518,519,523,529,531,532,620,627,637,643
19	99,160,331,510,601
20	31,32,121,420,644,646,647,649
21	93,94,122
22	84,85,87,88,91,101,102,103,108,111,112,115,240
23	105,249
24	277
25	(none)
26	579,666,669,675,680,710
27	209,316,358,361,412,456,461,555,557,561,573,618,674,676,681,682,687,695,698,699,703,705,706,707,711,712,713,730,731,736,737,738
28	(none)
29	398,399,477,569,570,574,575,576,577,578
30	(none)
31	269,362,511
32	1,64,67,68,81,149,150,223,232,292,520,588,593,611,621
33	170,326,330
34	506
35	95,110,118,119,238,482,504,513,626,645,654
36	139
37	256
38	130,322,366,367,455,459,486,517,673
39	208,393,395,397,544,558,672
40	324,541
41	335,356,392,394,396,537,708
42	55,138,297,368,500,533,542,563,567,568,581,582
43	354,355,457
44	5,6,12,34,35,36,37,38,42,66,123,153,155,158,164,165,171,199,200,206,215,241,242,244,245,246,247,248,262,264,265,272,288,289,290,301,321,334,478,490,493,495,516,587,590,629,630,631,638,641,642,661,662
45	(none)
46	11,39,73,135,180,194,196,224,228,231,254,270,300,343,462,594,608,625
47	140,190

48	255,310,609
49	163
50	659
51	(none)
52	167,370,371,693,709
53	27,363,732,733,734
54	560,565,696
55	26,212,214,278,470,534,535,536,548,549,550,551,566,604
56	374,615,725
57	469
58	274,466,521
59	(none)
60	14,43,76,175,275,295,483,606,614
61	498,499
62	50
63	134,218,219,220,306
64	83,183,188,221,227,660
65	(none)
66	677,678,688
67	56,65,341
68	147,172,528,556
69	28,502,728
70	157,222,236,360,384,406,419,426,514,583,584,692,694
71	57,124,128,143,296,375,376,474,476,552,553,585,623,658,716,717,718
72	(none)
73	(none)
74	198,294,503,596
75	633
76	182,656
77	(none)
78	216,217,489,591
79	19
80	(none)
81	430,598,616
82	651,683,684,685
83	62,691
84	144,424,526,612,726
85	4,9,71,141,344,373,385,407,463,492
86	(none)
87	177,179,197,213,350,605,634
88	178,195
89	74,75,78,174,181,185,186,187,207,271,473,595
90	(none)
91	(none)
92	2
93	(none)
94	(none)
95	(none)
96	333,364,365,670
97	145,617
98	137,146,259,273,308,401,422,423,453,668,727
99	287,346,359
100	(none)
101	(none)
102	77,79,80,235,299,337
103	46
104	(none)
105	41,49

106	70,280,613,639
107	10,279,509,589,602
108	(none)
109	23,421
110	328,597
111	210,327,403,404,405,409,454,467,468,472
112	239,332,408,715
113	133,325,352,353,357,433,451,452,465,471,619,663,689
114	40,169,191,226
115	(none)
116	15,51,98,100,349,485
117	47,127
118	54,258,285,480,636
119	20,211,281,282,464,599
120	303,336,603,632
121	(none)
122	86,117
123	(none)
124	(none)
125	266,329
126	3,72,267,286
127	345
128	(none)
129	18,168,201,229,230,237,307,309,348,491,497,522,622,635
130	156,293,312
131	45,142,166,257
132	33,96,109,283,314,650
133	319,624
134	69,304,494,600,607
135	(none)
136	82,89,90,104,113,114,116,653
137	261
138	(none)
139	21,44,184,189,413,458,690,697
140	192,386,387,388,400,434,665,700,701,722,723,724

Table C-5 SOM neurons and associated GIS procedure matches based on the implicit Albrecht keywords.

Best Matching Neuron	GIS Procedures
1	360,374,386,387,388,412,451,452,456,461,663
2	(none)
3	129,136,192,465
4	(none)
5	(none)
6	(none)
7	160,312,488,510,601
8	340,519
9	(none)
10	(none)
11	(none)
12	168
13	98,118,119
14	100
15	(none)
16	174,198
17	197
18	(none)
19	(none)
20	455,734
21	419
22	131,367,526
23	(none)
24	(none)
25	323,325,361,621
26	725
27	324,669,676,680,682,707,712,730
28	384
29	462
30	648
31	52,253,313
32	126,476,506
33	348
34	652
35	(none)
36	92,110,169,626
37	(none)
38	(none)
39	(none)
40	(none)
41	473
42	196
43	(none)
44	693
45	709
46	130,170,322,517
47	366,459
48	27,393,395,397,486,544
49	558,672,673
50	666,675,694,710
51	356,541,579,584
52	(none)
53	(none)

54	232,484
55	501,640
56	(none)
57	31,309,314,592
58	298,646
59	32,33
60	103,107
61	(none)
62	239,610
63	62,97
64	(none)
65	(none)
66	(none)
67	175
68	370,371
69	(none)
70	(none)
71	56,167
72	(none)
73	363,732,733
74	612
75	335,692
76	40,530,537,665
77	400,413
78	124
79	(none)
80	293
81	650,655
82	504,649
83	30
84	482,513
85	(none)
86	163
87	(none)
88	251,333
89	616,660
90	(none)
91	(none)
92	(none)
93	(none)
94	250,677,678,688
95	(none)
96	65,528
97	341,690,691
98	172
99	3,715,727
100	467
101	329,512
102	458
103	226
104	22
105	628
106	(none)
107	305,586,645
108	95,233
109	238,283
110	84,88,112,159
111	(none)

112	261
113	624
114	(none)
115	227
116	271
117	207,595
118	(none)
119	78,430
120	(none)
121	635
122	(none)
123	468,617
124	(none)
125	266,267
126	44,184,189,327,472
127	(none)
128	(none)
129	347
130	(none)
131	483
132	311
133	654
134	475
135	85,109,607
136	116,240,256
137	91,101,600
138	(none)
139	(none)
140	(none)
141	(none)
142	(none)
143	598
144	(none)
145	23
146	364,365
147	498,499,670
148	21,328
149	(none)
150	409
151	726
152	48,55,214
153	63,67,268
154	(none)
155	231
156	93,94,96
157	111
158	87,122
159	139,308
160	108,137
161	(none)
162	102
163	69,105,113,114,115
164	217
165	653
166	216
167	(none)
168	(none)
169	421

170	(none)
171	597
172	(none)
173	403,404,405
174	(none)
175	454,697,728
176	(none)
177	535,548
178	291
179	68,224
180	(none)
181	614
182	51,106
183	121
184	304
185	494
186	(none)
187	104,487
188	89
189	(none)
190	602
191	10,279,509
192	2,589
193	489,591
194	41
195	181,182
196	(none)
197	651,683,684,685
198	135,294
199	147,213,500
200	556,560,696
201	424,426,534,536,549,550,551,566,700,701
202	133
203	212,563,581,582
204	223,533,542,567,568
205	(none)
206	151
207	(none)
208	420
209	(none)
210	277
211	82
212	(none)
213	86,117
214	303,639
215	(none)
216	(none)
217	(none)
218	(none)
219	183,188
220	46
221	49,463
222	80,310
223	15,235
224	138,350,605
225	252
226	392,394,396,689
227	26,222,270,583,708

228	554,562,604,719,720,721
229	(none)
230	(none)
231	(none)
232	(none)
233	485,644
234	349,480
235	278,319
236	306
237	(none)
238	90
239	(none)
240	(none)
241	70
242	613
243	186
244	19
245	74,75,79
246	(none)
247	(none)
248	453,668
249	406,423,521
250	491,502,565
251	28,695
252	722,723,724
253	(none)
254	297,415,416,418,564,571,572
255	5,157,236,354,355,375,376,380,382,414,425,449,514,538,539,540,546,547,552,553,559,585,664,667,671,679,686,702,714,729,735
256	(none)
257	(none)
258	(none)
259	120
260	(none)
261	(none)
262	(none)
263	657
264	632
265	249,280
266	(none)
267	464
268	259
269	211,221
270	83
271	210,337,659
272	47,273
273	408
274	150,401,422
275	64,373
276	269,352,362,433,466
277	209
278	381,383,580,716,717,718
279	72,434
280	(none)
281	385
282	658
283	(none)
284	191

285	(none)
286	524
287	54
288	(none)
289	20
290	(none)
291	336
292	(none)
293	(none)
294	127
295	220
296	525
297	141,230,522,608
298	(none)
299	17,202,203,257,346,407,505
300	(none)
301	158,260,477,531,543,545,569,570,574,575,576,577,578
302	(none)
303	358,555,557,561,573,674,681,687,698,699,703,705,706,711,713,731,736,737,738
304	615,618
305	11,204,353,357,359,471,479,619
306	(none)
307	(none)
308	(none)
309	194
310	511
311	152,634,647
312	(none)
313	(none)
314	599
315	603
316	281,282
317	185,656
318	(none)
319	145,187,633,636
320	258,332
321	134,201,218,219
322	142,507,508,588
323	81,140,609
324	9,149
325	205,255
326	611
327	368,369,372,377,379,389,390,391,402,410,411,417,427,428,429,431,432,435,436,437,438,439,440,441,442,443,444,445,446,447,448,457,460,704
328	(none)
329	(none)
330	(none)
331	(none)
332	8,12,24,25,34,35,36,37,38,42,66,71,123,125,132,143,153,155,162,164,165,171,176,193,199,200,215,241,242,243,244,245,246,247,248,262,263,264,265,272,288,289,296,302,321,334,344,478,481,490,492,493,495,515,516,529,532,587,590,620,623,629,630,631,637,638,641,642,643,661,662
333	(none)
334	284,625
335	180,307,343
336	622
337	177
338	178
339	195
340	(none)

341	(none)
342	(none)
343	(none)
344	(none)
345	144
346	(none)
347	285
348	(none)
349	45,146
350	190
351	(none)
352	(none)
353	351,450
354	29,39,128,148,274,593
355	254
356	(none)
357	(none)
358	627
359	53
360	229
361	(none)
362	18
363	77
364	(none)
365	(none)
366	(none)
367	503,596
368	(none)
369	(none)
370	228,295
371	286,469,470
372	287
373	43,161,497,527,606
374	276
375	1,398,399,496
376	378
377	(none)
378	(none)
379	4,57,58,59,60,61,290,338,339,523
380	(none)
381	154,173
382	225
383	16,156,234,301,326,330,474
384	99,320
385	331
386	76,237,342
387	166
388	(none)
389	14
390	6
391	299
392	50
393	179,275
394	300,594
395	13,520
396	208
397	318,345
398	(none)

399	7,206,315,317,518
400	73,292,316

Table C-6 SOM neurons and associated GIS procedure matches based on the implicit Enviro Modeling keywords.

Best Matching Neuron	GIS Procedures
1	(none)
2	195,337,634
3	14,76,614
4	178,196
5	175,197,299
6	(none)
7	(none)
8	145
9	286,726
10	133
11	270,406,424,665
12	356,579
13	(none)
14	(none)
15	209,324
16	316,543,545,689
17	39,339,352,353,357,359,362,433,471,619
18	(none)
19	(none)
20	627
21	99,320
22	(none)
23	237
24	228
25	(none)
26	179
27	174
28	(none)
29	235
30	47
31	(none)
32	172
33	43
34	67,611
35	222,554,562,583,708,719,720,721
36	68
37	358,369,372,377,379,381,383,389,390,391,402,410,411,417,427,428,429,431,432,435,436,437,438,439,440,441,442,443,444,445,446,447,448,457,460,555,557,561,573,580,674,681,687,695,698,699,703,704,705,706,711,713,716,717,718,731,736,737,738
38	(none)
39	351,450,615,618
40	29,128
41	(none)
42	(none)
43	(none)
44	(none)
45	625
46	180,462
47	135,224,300,594
48	177,201,230
49	503,596
50	15,633,656
51	198

52	187,500
53	50,275
54	295
55	141
56	522
57	9
58	(none)
59	(none)
60	(none)
61	(none)
62	368,496
63	(none)
64	(none)
65	8,12,24,25,34,35,36,37,38,42,66,71,123,125,132,143,153,155,158,162,164,165,171,176,193,199,200,215,241,242,243,244,245,246,247,248,262,263,264,265,272,288,289,296,302,321,334,344,478,481,490,492,493,495,515,516,529,532,587,590,620,623,629,630,631,637,638,641,642,643,661,662
66	(none)
67	11,204,254,479,593
68	146,161,223,276
69	13,73,287,292,325,527
70	202,203,520
71	6
72	(none)
73	185
74	310
75	(none)
76	294
77	(none)
78	213,285,350,605
79	218,219,524,525
80	81,609
81	17,45,149,212,505,604
82	151
83	148,415,416,418,533,542,564,567,568,571,572
84	(none)
85	346,354,355,375,376,380,382,407,414,425,449,538,539,540,546,547,552,553,559,585,664,667,671,679,686,702,714,729,735
86	269,385,401,466,658
87	274
88	(none)
89	284
90	57,58,59,60,61,290,338
91	(none)
92	1,205,291,606,621
93	(none)
94	599,603
95	(none)
96	186
97	46
98	(none)
99	259
100	127,221
101	83,138,220,258
102	134,252,608
103	48,55,63,64,140,150,152,497,588
104	257,268
105	477,569,570,574,575,576,577,578
106	(none)
107	255,260

108	(none)
109	157,190,236,297,514,531
110	(none)
111	(none)
112	307,330,648
113	53,343
114	(none)
115	77
116	20,54
117	(none)
118	632
119	(none)
120	(none)
121	(none)
122	49,463
123	211,281,282
124	147
125	142,659
126	214
127	(none)
128	535,548
129	398,399,563,581,582
130	(none)
131	26
132	194
133	622
134	173
135	(none)
136	18,191,501,640
137	231,485
138	(none)
139	306
140	(none)
141	(none)
142	(none)
143	2,41,509,589
144	(none)
145	(none)
146	79,183,188
147	651
148	273,332,636
149	556,696
150	347,373,502,565
151	491,521,534,536,549,550,551,566
152	392,394,396
153	507,508
154	28,511
155	(none)
156	301,326,474,488
157	156
158	253,313
159	293
160	98
161	100,420
162	660
163	86,90,117
164	303,616,639
165	(none)

166	279
167	613
168	489
169	19,336
170	683,684,685
171	210,341
172	560
173	453
174	124,422,426,700,701
175	537
176	470,541
177	(none)
178	229,484
179	476
180	348,506
181	(none)
182	349
183	277
184	62,657
185	487
186	104
187	653
188	216
189	10,602
190	280
191	70,464,591
192	(none)
193	(none)
194	65
195	668,727
196	56,612
197	423,728
198	144
199	335,584
200	(none)
201	519
202	(none)
203	31,309,504
204	482,626,645
205	110,121
206	122,139,238
207	308
208	101,261
209	105,115
210	89
211	69,217
212	(none)
213	(none)
214	227
215	430
216	678
217	(none)
218	528
219	455
220	363,408
221	732,733
222	(none)
223	208

224	(none)
225	118,119,278,644
226	(none)
227	111,646
228	95,654
229	163
230	239,256
231	91,116,240
232	(none)
233	102,113,114,600
234	(none)
235	(none)
236	421
237	598
238	597
239	250,370,371,677,688
240	693
241	709
242	130,167,322,734
243	366,459
244	393,395,397,486,512
245	(none)
246	226
247	(none)
248	647
249	(none)
250	32
251	311,513
252	87,103
253	84,108,475
254	85,109,159
255	137,607,624
256	(none)
257	(none)
258	(none)
259	(none)
260	23
261	(none)
262	364,365
263	468
264	170,635,691
265	131,517
266	367,419,526,544
267	27,318
268	558,672,673
269	(none)
270	22
271	(none)
272	649,650
273	314
274	51,106
275	30
276	(none)
277	88,112
278	251,333,494,610
279	82
280	(none)
281	78,271

282	181
283	182,249
284	(none)
285	498,499
286	21,690
287	467,715
288	(none)
289	345
290	469,666,675,694,710
291	317,518
292	451,452,663
293	(none)
294	232
295	52,342
296	(none)
297	298,319,480
298	33,93,169,305
299	107,233
300	283,304
301	97
302	(none)
303	(none)
304	207,595
305	75
306	74
307	(none)
308	328,670
309	697
310	44,184,189,266,267,327,329,409
311	458
312	(none)
313	(none)
314	7,40,206,315,323,361,530
315	(none)
316	400,413
317	(none)
318	312,331,510
319	340
320	166
321	94,96
322	92,586
323	(none)
324	655
325	483
326	473
327	(none)
328	(none)
329	80
330	617
331	3,403,404,405
332	(none)
333	454
334	5,386,387,388,472,722,723,724
335	(none)
336	360,374,378,412,669,676,680,682,692,707,712,725,730
337	(none)
338	4,129,136,384,456,461,523
339	72,192,434,465

340	225
341	16,154,234
342	160,601
343	(none)
344	120,126,628,652
345	168,592