An evaluation of organization methods for data types commonly used in the geographic domain

by

Jochen Wendel

Dipl. -Ing. (FH) University of Applied Science Karlsruhe, 2004

M.Sc. University of Applied Science Karlsruhe, 2006

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

2013

This thesis entitled:

An evaluation of organization methods for data types commonly used in the geographic domain

written by Jochen Wendel

has been approved for the Department of Geography

_____

Professor Barbara P. Buttenfield, committee chair

_____

Professor Stefan Leyk, 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

Jochen Wendel (Ph.D., Geography)

An evaluation of organization methods for data types commonly used in the geographic domain

Thesis directed by Professor Barbara P. Buttenfield

Abstract:

This dissertation designed and implemented approaches to assess the suitability of commonly used unsupervised and supervised grouping methods on data types commonly used in the geographic domain. Four different types of data have been indexed for organization: a full-text data set depicting 30 years of cartographic literature, a raster data set consisting of physiographic characteristics of the U.S., a suite of GIS software commands used in hydrologic analysis, and a catalog of cartographic generalization algorithms. Various clustering and classification methods from the field of statistics and machine learning were evaluated for organizing these different data types. By systematically applying all types of data organization to each type of indexed data, this research addresses the question of whether certain indexing strategies influence the effectiveness of the organization methods. Depending on the data set and the indexing method applied, some clustering and classification methods performed better than others.

The experiments of this dissertation demonstrate that by the systematic evaluation and validation of clustering and classification results recommendations for organizing data can be formulated based on the results of cluster and classification indices. Furthermore, through systematic evaluation and application of the six clustering and classification methods it is possible to match indexing strategy and organization methods for each of the four data sets used in this dissertation.

Acknowledgments

Foremost, I would like to express my gratitude to my advisor, Babs Buttenfield for the continuous support through all stages of my doctorate studies, research and dissertation writing. Her guidance helped me in all times of research and writing of this dissertation. Furthermore, I would like to thank her for being flexible and making it possible to defend my dissertation in a timely manner.

Besides my advisor, I would like to thank the rest of my dissertation committee: Elisabeth Root, Stefan Leyk, Seth Spielman and André Skupin for their encouragement, insightful comments, and flexibility during the process of writing this dissertation.

I would like to thank Roland Viger at the USGS for first exposing me to my dissertation topic and SOM. His tremendous help and extensive knowledge in the subject matter and the implementation of different methods in Matlab helped me in all stages of my dissertation.

Furthermore, I would like to thank my fellow colleagues in the Meridian Lab, especially Chris Anderson-Tarver, Eric Wolf, Mike Gleason, Sam Smith, Petra Norlund, Matthew Ruther, Alex Stum and Galen Maclaurin for advice, discussion, chat and countless lunch hours throughout my time in Colorado. I thank my wonderful friends Susanne Benze and Matthias Brakebusch for providing me with the sometimes much needed German humor and Gemütlichkeit.

I would like to thank my parents, Heinz und Inge Wendel for support and encouragement through my whole studies. Without whom I could not have made it here.

Above all, I would like to express my foremost gratitude and love to my wife Simone who always supported and encouraged me on the entire journey of finishing this dissertation. Most importantly I would like to thank my daughter Marie for considerably accelerating the writing process and always smiling and laughing after a long day of writing: a true thesis baby!

CONTENTS

CHAPTER I 1

The problem of accessing information 1

1.1 Current problems in accessing information 1

1.2 A continuum of automatically and manually derived indexing strategies 3

1.3. Relevance to data organization 6

1.4. Information spaces and their relevance to GIScience 7

1.5 Problem statement 9

1.6 Research questions 9

1.7 Problem significance 10

1.8 Dissertation structure 12

CHAPTER II 13

Literature Review - Overview of existing knowledge 13

2.1 A review of concepts from Information Retrieval 13

2.2 Indexing strategies 18

2.2.1 Natural language indexing 19

2.2.1 Current online indexing strategies 21

2.2.2 Indexing on manually derived keywords 26

2.2.3 Metadata as a source for keywords 26

2.2.4 Auxiliary data as a source for deriving keywords 27

2.2.5 Manual methods for keyword generation 29

2.3 Concepts of Ontology and Semantic Web 30

2.4 IR in the field of GIScience 33

2.5 Future research trends 35

2.6 Summary 37

CHAPTER III 38

Methodological Review 38

3.1 Concepts for organization of data 38

3.2 Measures of similarity and difference used in data organization 40
3.2.1 Categorical data 40
3.2.2 Continuous data 41
3.2.3 Hybrid data 42
3.3 Classical statistical methods for data organization 42
3.3.1 Classical unsupervised (Clustering) methods 42
3.3.1.1 Principal Component Analysis (PCA) 43
3.3.1.2 Cluster analysis 44
3.3.1.3 Cluster evaluation 50
3.3.2 Classical Supervised Methods (Classification) 53
3.3.2.1 k-Nearest Neighbor classification 53
3.3.2.2 Classification trees 55
3.3.2.3 Evaluation methods in supervised classification 57
3.4 Modern methods taken from machine learning 58
3.4.1 Unsupervised Machine Learning: Self-Organizing Maps (SOM) 60
3.4.2 Supervised Machine Learning: Support Vector Machines 65
3.6. Summary and current trends in data organization 68

CHAPTER IV 70

Automatic and manual indexing experiments 70

4.1 Research tasks 70
4.2 Research task framework 71
4.3 Data sets used as exemplar data types 73
4.4 Indexing experiment 75
4.4.1 Full-text articles 75
4.4.2 Spatial data 81
4.4.3 Software 85
4.4.4 Algorithms 87
4.5 Summary of the indexing experiment 92

CHAPTER V 93

Clustering and Classification Experiments 93

5.1 Methodological overview and structure of experiments 93

5.1.1 Clustering and classification methods 93

5.1.2 Comparison and evaluation 94

5.1.3 Software environments 94

5.2 Full-text dataset organization 95

5.2.1 Unsupervised methods 95

5.2.1.1 Cluster evaluation and selection 95

5.2.1.2 Clustering results 102

5.3. Spatial data set 109

5.3.1 Unsupervised methods 109

5.3.1.1 Cluster evaluation and selection 109

5.3.1.2 Clustering results 117

5.3.2 Supervised methods 117

5.4 GIS Commands 122

5.4.1 Unsupervised methods 123

5.4.1.1 Cluster evaluation and selection 123

5.4.1.2 Clustering results 126

5.4.2 Supervised methods 128

5.5 Algorithms 131

5.5.1 Unsupervised methods 131

5.5.1.1 Cluster evaluation and results 131

5.5.1.2 Clustering results 135

2.5.2 Supervised methods 137

5.6 Summary of the grouping experiment 141

CHAPTER VI 143

Discussion of results 143

6.1 Results from the organization experiment 143

6.2 Discussion of results and answers to the research questions 146

6.3 Limitation of the experiment 150

6.4 Future work 152

6.5 Conclusion 153

References 154

Appendix A – Full-text data set 168

Appendix B – GIS commands data set 168

Appendix C – Algorithm data set 168

LIST OF TABLES

Table 1.1 A continuum of indexability for a variety of data types. The table shows alternative options for establishing keywords, and for each keyword option, identifies the extent to which keywords may be extracted automatically. 4

Table 3.1 Overview of different linkage criteria in agglomerative hierarchical clustering 47

Table 3.2 Overview of methods used in this dissertation 69

Table 4.1 The data sets used in this experiment span the continuum of indexability that was introduced in Chapter 1 73

Table 4.2 A small section of the full-text index after pre-processing, stemming, and tf-idf term weighting. 80

Table 4.3 Factors used to classify landscape types (Stanislawski et al., 2011) 81

Table 4.4 The initial keyword set and the final set (in bold) after the degrees of freedom are removed (strikeout text) 86

Table 4.5 Sample of the Boolean matrix characterizing the software data set 87

Table 4.6 Overview of cartographic generalization keywords. 90

Table 4.7 A small section of the final generalization data set indexed after preprocessing, stemming, and term weighting. A sample of generalization taxonomy keywords and automatic derived keywords are shown. 91

Table 5.1 Hierarchical cluster membership stability and formation from 5 to 16 clusters. 98

Table 5.2 k-Means cluster membership stability and formation from 5 to 16 clusters. Cluster formation is independent from those shown in Table 5.1. 99

Table 5.3 Cluster membership stability and formation from 5 to 16 clusters. Cluster formation is independent from those shown in Table 5.1 and Table 5.2. 102

Table 5.4 SVM training parameters using 80%, 50%, and 30% training sample. SVM per class indicates how many training vectors are calculated per class 107

Table 5.5 Classification results compared to the optimal Hierarchical clustering results. Misclassified journal papers are shown in red 108

Table 5.6 SVM training parameters using 30%, 15%, and 7% training samples 119

Table 5.7 Cluster membership assignment for GIS commands using Hierarchical clustering 124

Table 5.8 k-Means cluster membership stability and formation from 5 to 11 clusters. Cluster formation is independent from those shown in Table 5.1. 125

Table 5.9 SOM cluster membership stability and formation from 5 to 11 clusters. Cluster formation is independent from those shown in Table 5.7 and 5.8. 126

Table 5.10 SVM parameters and statistics for the two training data sets 130

Table 5.11 Classification results compared to the optimal clustering results for the selected GIS commands summarized for unsupervised grouping. Misclassified commands are shown in red. 131

Table 5.12 Cluster membership assignments for Hierarchical clustering 133

Table 5.13 Cluster membership assignments from k-Means clustering 134

Table 5.14 SOM cluster membership stability and formation from 3 to 16 clusters. 135

Table 5.12 SVM parameters and statistics for the 80% and 50% training data set 139

Table 5.13 Classification results compared to the optimal clustering results 140

Table 5.14 Overview of all data sets and methods used in this experiment 142

Table A-1 Complete full text data set 167

Table A-2 Cluster and class membership for the full-text data set 167

Table B-1 Cluster and class membership for the GIS commands data set 167

Table C-1 Complete algorithm data set 167

Table C-2 Cluster and class membership for the algorithm data set 167

LIST OF FIGURES

Figure 1.1 Euclidian space versus information space. If two points are placed in the Euclidean space model on the left, properties of these points can be measured by x, y and z values. In the information space model on the left these data points would be described by the attributes which are defining axes of this space. 8

Figure 2.1 Milestones in IR systems (redrawn from Akerkar and Lingras, 2008). The solid blue line indicates known amounts of information, the dashed blue line displays estimated numbers of items in each collection, and the dashed red line displays online information indexed by Google. 14

Figure 2.2 Typical architecture of a search engine (redrawn from Zhou and Davis, 2006). The red box highlights the focus of this dissertation 22

Figure 2.3 An example of an automatic detection method of road network extraction from areal imagery. (http://gis.incogna.com/?p=technology#Road, accessed June 2013) 28

Figure 3.1 The three most commonly used linkage distance measures in hierarchical clustering (Redrawn from Everitt, 2001). 46

Figure 3.2 Elements of a dendrogram 48

Figure 3.3 The k-Means clustering process for 3 clusters and 3 iterations. (a) Input data; (b) three seed points selected as cluster centers and initial assignment of the data points to clusters; (c) and (d) iterations and updating of cluster labels and their centers; (e) final clustering obtained by k-means algorithm at convergence (redrawn from Jain, 2009). 49

Figure 3.4 k-NN classification of hypothetical data in a 2-dimensional feature space 54

Figure 3.5 Example of a simple decision tree using hypothetical data. 56

Figure 3.6 The SOM attribute space at initial (left), intermediate (middle), and final (right) iterations in the unfolding process. Black dots represent input data points and green dots represent SOM cells. Taken from http://blog.peltarion.com/2007/04/10/the-self-organized-gene-part-1/. 62

Figure 3.7 Example of how the units are mapped onto a SOM rectangular mesh. 62

Figure 3.8 This figure shows a SOM mapplet representing each dimension (X,Y). Legends along the bottom of the two mapplets show the color ramp for low-to-high values. The example is modified from http://blog.peltarion.com/2007/04/10/the-self-organized-gene-part-1/. 63

Figure 3.9 Example of a U-Matrix for a SOM containing 16x16 cells. Darker values indicate greater distances (more dissimilarity) between characteristics of neighboring cells. Lighter values indicate neighboring cells whose characteristics are similar. 64

Figure 3.10 Linear plane separation shown in a 2-dimensional feature space 66

Figure 3.11 Delineation of support vectors and margin for SVM using linear separation between two classes of data points. 66

Figure 3.12 Transformation of data points from the original input space to a feature space where data is linearly separable. The phi term in the figure describes the transformed data points into higher dimensional space. 68

Figure 4.1 Methodological framework for indexing and grouping the four data sets, and evaluating the organization methods. Chapter 4 covers the indexing of the data sets, while Chapter 5 discusses the organization of the indexed data sets as well the evaluation of the methods. Chapter 6 will make recommendations about indexing and organizing the four types of data. 72

Figure 4.2 Example of a data entry downloaded from the ISI Web of Knowledge in the standard BibTeX format. The red boxes highlight the information relevant for indexing. Only the author's name, the title of the journal paper, keywords, the full abstract, and the year published are used in the analysis. 76

Figure 4.3 Processing steps involved in automatic keyword generation of the full-text document data set. 77

Figure 4.4 Number of keywords per cutoff value showing on a logarithmic scale on the left. The percentage of 0 values in the term-document matrix by cutoff value is presented on the right. 79

Figure 4.5 Environmental factors (attributes) mapped in geographic space. All attributes are normalized to a range of 0 - 1000. All seven attributes are used for indexing. 82

Figure 4.6 Maximum likelihood estimation of seven data sets used to predict landscape types across the continuous United States, and a map estimating possible misclassifications (Stanislawski et al., 2010). 83

Figure 4.7 Seven-dimensional index created from the spatial data set. The section is located in the north eastern part of the U.S. The red box shows the approximate location. 84

Figure 4.8 (a) Abstract sample from Veregin (2000). Words highlighted show cartographic generalization algorithm specific keywords, demonstrating that one or a small number of keywords will suffice in some cases to isolate those parts of the full-text document referring to the algorithm. **(b)** Abstract sample from Ware et al. (1995). Words highlighted show cartographic generalization algorithm specific keywords, demonstrating that for this abstract multiple keywords are needed for indexing. 88

Figure 4.9 Processing steps involved in automatic keyword generation of the cartographic generalization algorithm data set. Text highlighted in red shows the additional steps necessary to filter out generalization algorithm specific publications. 89

Figure 5.1 Overview and structure of the methods used in this experiment 93

Figure 5.2 Evaluation metrics for unsupervised clustering of the full text data set. For these and subsequent data sets, the blue line shows the progression of values for k-Means clustering (k)

and the red line reflects Hierarchical clustering (h) results. The area shaded in grey shows the range of clusters chosen as optimal based on local extrema and leveling off region. This region will define a range of cluster solutions for cluster stability evaluation. 96

Figure 5.3 SOM visualization of the full-text data set. The U-matrix illustrates the overall compactness for all clusters as a whole and each variable of the dataset is represented in its own attribute plane. The red, blue, and green boxes highlight interesting patterns of coincidence, clustering, and scattering within the attribute space. 100

Figure 5.4 Dendrogram of the full-text data set showing class labels in red for the optimal 10 cluster solution. 103

Figure 5.5 k-NN analysis of selecting the optimal number of nearest neighbors for training of the classifier 105

Figure 5.6 Summary of Random Forest analysis using three different training sample sizes to determine the optimal classifier 106

Figure 5.7 Evaluation metrics for Hierarchical and k-Means clustering. The area shaded in grey shows the range of optimal clusters based on local extrema and leveling off region. The orange line shows the seven class solution from prior analysis by Stanislawski (Stanislawski et al., 2010). 110

Figure 5.8 Range of solutions for Hierarchical clustering. The red boxes show areas of interest. 111

Figure 5.9 Range of physiographic cluster solutions for k-Means clustering. 113

Figure 5.10 SOM BMU clustering solutions applied to the spatial data set. The reader is cautioned that the y-axes in the three panels are not scaled uniformly. 114

Figure 5.11 Comparison of linear SOM and SOM BMU clustering 115

Figure 5.12 Optimal k-Means classical clustering solution and SOM clustering. 116

Figure 5.13 k-NN analysis for selecting the optimal number of k neighbors for three training data sets. Due to the increased complexity and computing time the 30% and 15% training samples show cross validation accuracy fore every other nearest neighbors while the 7% sample shows all 60 nearest neighbors for determining the number of nearest neighbors. 117

Figure 5.14 Summary of Random Forest analysis using three different training sample sizes to determine the optimal classifier for the spatial data set. 118

Figure 5.15 Comparison of supervised classification results. Red areas are misclassified. 121

Figure 5.16 Evaluation indices for Hierarchical and k-Means clustering. The area shaded in grey shows the range of optimal clusters based on local minima and leveling off region. 123

Figure 5.17 GIS commands dendrogram, clusters are shown in red 127

Figure 5.18 k-NN analysis of selecting the optimal number of k for multiple training data sets 128

Figure 5.19 Summary of Random Forest analysis using three different training sample sizes 129
Figure 5.20 Evaluation indices for the algorithm data set 132
Figure 5.21 GIS commands dendrogram, clusters and cluster labels are shown in red. 136
Figure 5.22 Cross validation accuracy of the three training samples for the algorithms data set. 137
Figure 5.23 Summary of Random Forest analysis using three different training sample sizes 138

CHAPTER I

The problem of accessing information

Through the adoption of new technology, data accumulates steadily and information gathering continues to be an important task throughout any research project. In order to analyze a data set, it often needs to be characterized, as the raw data either lacks metadata or cannot be used in the form in which it is stored. The key to efficient data retrieval is the creation of indices (Manning et al., 2009). Organization of indices into a relevant schema will facilitate data retrieval. Many different advanced statistical methods can be applied to organize data, but they vary in effectiveness depending on the data types to which they are applied.

This dissertation will explore problems associated with building indices to organize different data types. The research will explore how to organize data which has been indexed using manually and automatically derived keywords. Four different types of data sets will be indexed: a full-text document depicting 30 years of cartographic literature, a suite of GIS software commands used in hydrologic analysis, a catalog of generalization algorithms, and a raster data set consisting of physiographic characteristics of the U.S. Various clustering and classification methods from the field of statistics and machine learning will be evaluated for organizing these different data types using clustering and classification evaluation indices. By systematically applying all types of data organization to each type of indexed data, this research addresses the question of whether certain indexing strategies influence the effectiveness of the organization methods. Systematic recommendations for indexing different types of data, and for subsequent clustering or classification, will be formulated based on formal evaluation as well as discussion of the data set's inherent structure.

1.1 Current problems in accessing information

In today's world, much of the available published knowledge and information is stored in digital form. Since the advent of the World Wide Web (WWW) and the application and

augmentation of Hypertext (Conklin, 1987), the trend has shifted to online storage, which provides wide accessibility of information on the Web (Berners-Lee, 1990). The evolution from localized storage to online storage and cloud computing involves a paradigm shift that changes many aspects of data organization. Data access and transfer speeds are no longer a primary obstacle to data retrieval. This dissertation argues that as more and more data become available, strategies to organize and index the increasing volume of data will become the primary challenge to information retrieval.

Information is stored in varying formats and media. Information accessible on the Internet is commonly stored as full text records. Many online retrieval frameworks such as Google, Yahoo, and Bing gather information and provide a sharing platform. Retrieval frameworks include online search engines, library catalogs and multimedia databases. A problem with using online retrieval frameworks is how to search for items which are similar to a token, and how to search for multiple data formats. Most search engines can only work on one type of data. In order to make data fully searchable, it needs to be characterized by a certain index (Akerkar and Lingras, 2008).

Generating indices depends on the type of data which needs to be indexed. For example, full-text documents are indexed on different indices than non-textual data such as algorithms or satellite image archives. A variety of indexing tools for textual data are currently available. Most search engine indexing tools are based on word count, word stemming or word appearance and structure (Google Search, 2010). Word count is one of the simplest methods whereas word appearance analyzes the words before and after the word of interest and tries to establish relationships between words. More advanced techniques for analysis of full text documents include the vector space model in which word inclusion or absence is stored in a vector along with weights prioritized to distinguish common or generic words (e.g., articles, conjunctions) from words which are salient to the topic under investigation (Salton et al., 1976).

Indices which characterize textual data cannot be applied readily to other data types, for example algorithms. Running a word count or text stemming method on algorithms won't provide meaningful distinctions among similar and non-similar algorithms (Segaran, 2007). What is needed to generate indices for algorithms is a description of what the algorithm does, rather than culling tokens directly from the source code. Manually generated characterizations are essentially a form of metadata, but many types of data (algorithms, digital data sets, etc.) are not currently stored with metadata.

Additional data is often required to form manually generated keywords. Software commands for example need additional information not stored within the code to generate such a characterization. Online help and additional documentation might serve this purpose. Usually these keywords can be derived manually. For example, Ye et al. (2001) described a manually derived keyword set for use in software repository systems. Wendel et al. (2009) implemented manually generated keywords for hydrological GIS commands. Viger (2011) demonstrated that manually generated keywords can be used to moderate, improve, and specialize the results of the traditional keyword process. However, neither standardization nor guidelines exist at present for indexing software commands.

1.2 A continuum of automatically and manually derived indexing strategies

The dissertation will explore a range of techniques to derive indices for organizing different types of data. Table 1.1 shows a continuum spanning the range from automatically to manually derived indices for different types of data. Starting at the top of the continuum, automatically derived keywords have been demonstrated to provide an effective indexing strategy for full text documents (Blanken et al., 2007). Indexing is usually done by frequency word counts or structural analysis of the text document (Segaran, 2007). Furthermore, text can be structured by title, chapters and sections, and text stemming methods can be adjusted to the content of each section.

In contrast with full-text documents, metadata is needed for describing most other data types. Metadata usually follows a pre-defined schema where the data type is described in textual form. For example, instead of searching for an object directly in a multimedia archive, one can search metadata. To take another example, vector or raster spatial data sets usually include attributes. An automatically derived indexing scheme can be developed using the attributes for keywords. In addition to attribute field names, metadata is sometimes presented and may include information such as generation date, description of the generation process and persons involved in the process.

Table 1.1 A continuum of indexability for a variety of data types. The table shows alternative options for establishing keywords, and for each keyword option, identifies the extent to which keywords may be extracted automatically.

Dataset	Keywords drawn from content	Indexing method	Manual intervention
Full-text	Article	Stemming	Low
	Metadata	Stemming	Low
Spatial data	Raw data	None	None
	Metadata	Stemming	Low
Sound files	Media tags	Stemming	Low
	Sound structure	Pattern recognition	Medium
Image files	Media tags	Stemming	Low
	Content-based	Pattern recognition	Medium
Software	Code	Code analysis	Medium
	Auxiliary data	Publications	High
Algorithms	Metadata	Stemming	Low
	Pseudo code	Structural analysis	Medium
	Auxiliary data	Publication taxonomy	High

For documents containing information in other formats, indexing becomes more challenging, indexing strategies become increasingly complex and require a larger amount of human intervention.

Sound files usually have media tags associated with them. Media tags describe properties such as artist, genre, length, and date (Blanken et al., 2007). The content of media tags creates an analogous indexing method to the metadata descriptors mentioned above. Other approaches analyze the sound structure of sound files by frequency or beats per minute so that less descriptive metadata is needed for indexing. This technique is not widely used due to its computational complexity (Blanken et al., 2007).

Image files consist of pixels describing the color values as well the color depth (Blanken et al., 2007). Auxiliary data (metadata) can support indexing for these kinds of data. Image files usually contain metadata tags describing camera type, properties about how the image was taken. For analog images, metadata might include shutter speed, focal length or aperture, and for digital images it might include platform height, resolution, date of collection, radiometric characteristics, and GPS coordinates. This type of auxiliary data is commonly stored in Exchangeable Image File format (EXIF) which has become a broadly adopted standard (http://www.exif.org , accessed August 2011). More advanced methods for characterization of images include pattern recognition techniques which are used for face recognition or content-based image search (e.g., finding orchards within orthoimagery) (Prathiba et al., 2013; www.facebook.com, accessed October 2011; http://picasaweb.google.com, accessed October 2011).

Moving further down the continuum, the data type exemplified by a set of software commands requires an indexing strategy that requires more human intervention. Principles that are applied to sound and image files cannot be used for software or programming code. There is no standardized metadata for software. Methods exist for generating metadata such as Java Docs, but are not standardized and vary by programming language. Approaches which apply

pattern recognition techniques to find structures in programming code have been proposed, but these cannot be applied to all domains (Tangsripairoj and Samadzadeh, 2006).

At the bottom of the continuum, one encounters algorithms, which describe procedures for completing a particular task. These in a way provide the most extreme example requiring manual human intervention during indexing, as the algorithm may be described in equations, in a workflow diagram, in natural language, or in pseudocode, which in and of themselves provide three distinct data types. At this point, the continuum forms a conceptual loop, since algorithms are commonly described in online help files, journal articles or other full-text documents, from which it should be possible to extract a salient set of automatically derived keywords. Full-text documents describing algorithms might act as manually derived metadata, from which keywords could be generated by stemming, in some cases.

1.3. Relevance to data organization

The four data sets in this dissertation were chosen to represent the whole range of data sets ranging from full-text documents which can be index automatically to data sets where manual indexing is required. No systematic evaluation of indexing and organization strategies across multiple data types exists in the literature. This research evaluates the effeteness of clustering and classification for a given set of indexing schemes. It is expected that some organization methods might perform better than others as all of the four data sets have a different inherent underlying ontological structure. For example, a hierarchical organization method might perform better on the cartographic generalization algorithms data set due to the underlying taxometric nature of the data set and domain it describes. Whereas non-hierarchical organization methods might perform better on a data set where no distinct inherent axonometric structure is present. The research presented in this dissertation does not want to implement new methods but rather evaluate existing methods, given a particular indexing scheme and data type. Future multi-data type retrieval systems will rely on a combination of classical and modern indexing strategies to accommodate the variety of different data types

(Blanken et al., 2007). Therefore it is beneficial to have a better understanding of which methods will succeed or fail for a specific data type. Chapter 2 will give a complete review and discussion of the most notable systems that have been developed.

1.4. Information spaces and their relevance to GIScience

One of the basic foundational concepts in the field of Geography is the idea that space is defined through Euclidean space that constructs the physical world we live in (Batty and Miller, 2006). Distance and direction underlie our understanding of place. Theories have been developed based on this fundamental concept and form the core of geographic analysis. A fundamental aspect of geographic space is the continuum of scale, ranging from the footprint of a small area to the footprint of the whole country (Fabrikant and Buttenfield, 2001).

One key step in the process of data organization is the creation of information spaces. Every time data is organized an information space is formed based on the underlying ontological structure of the data set. An information space can be described as the location where the human mind interacts with information, and content is organized by the experience of the human (Manning et. al. 2009). Information spaces facilitate the storage and retrieval of data and information, processing of data into information, communication of information, navigation through structured information and linking different pieces of information. Instead of constructing a space by Euclidian geometry commonly used to represent our physical world, attributes describing information are used to display relationships within information (Figure 1.1). Similarity is often associated with distance between objects in such an information space. Spatial metaphors can be applied to these information spaces (Fabrikant and Buttenfield, 2001). The First Law of Geography “Everything is related to everything else, but near things are more related than distant things" (Tobler, 1970), can be applied to information spaces just as it can be applied to geographic space.

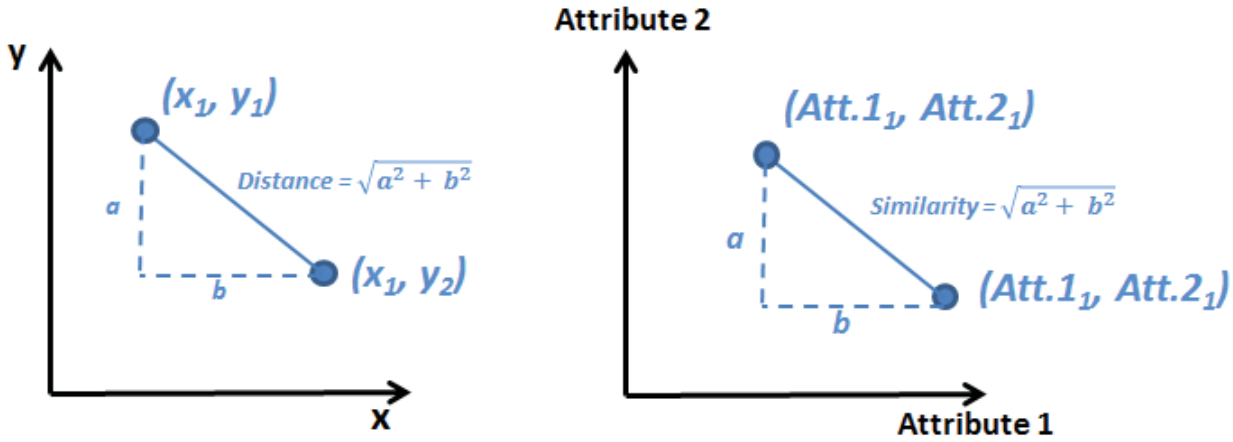


Figure 1.1 Euclidian space versus information space. If two points are placed in the Euclidean space model on the left, properties of these points can be measured by x, y and z values. In the information space model on the left these data points would be described by the attributes which are defining axes of this space. In Euclidean space distance can be measured by the Pythagorean Theorem. This principle can be transferred into the information space model where the measured distanced corresponds to similarity.

Information spaces in GIScience have been studied by geographers over the last 20 years, especially with the notion of using a spatial metaphor for analysis of non-spatial information (Skupin and Buttenfield, 1997; Fabrikant and Buttenfield, 2001; Skupin and Fabrikant 2003). In the field of GIScience this process is called "Spatialization" (Skupin, 2001). Each of the four different indexing strategies used in the dissertation formalizes its own information space. The generated keywords for each data set form the dimensionality of the information space. Therefore each information space can be described as having its own geography. Depending on the indexing and organization method used, each method will place attributes in differing spatial relationships. Chapter 2 will review different concepts of information spaces and semantic reference spaces to more detail.

1.5 Problem statement

This dissertation implements different indexing strategies to different types of data. It discusses the generation of manually generated keywords for data sets where automated keyword generation is not feasible. To demonstrate the feasibility of intelligent data organization, four different data types commonly used in the geographic domain will be first indexed and then organized by supervised and unsupervised methods. The data sets used in this dissertation consist of a full-text document, an inventory of software commands, a catalog of algorithms and a spatial raster data set. The four data sets span the continuum of indexability and ranging from fully automated keyword generation to keyword generation requiring human intervention. Various supervised and unsupervised grouping methods from the field of statistics and machine learning will be used to organize the four data types. Throughout the dissertation, methods drawn from statistics will be referred to as "classical" and methods drawn from machine learning will be referred to as "modern". Grouping results will be compared by common evaluation and validation methods. The discussion of the results is guided by the data set's underlying inherent structure as well as the purpose of the organization of for each of the four different data sets.

1.6 Research questions

During the process of indexing and grouping, two major research questions can be addressed:

1. ***For a given indexing scheme does a particular organization method link clearly to an indexing method and why?***

Automatically generated keywords are derived directly from a document, by word counts or text stemming methods, without requiring further information or knowledge. In contrast, additional data is required for manually indexed keywords. Given the differences in the creation of these two types of indices, certain organizational methods might be better suited

than others and some methods might not be suitable at all. The anticipated finding is that manually indexed data sets might perform well on supervised learning methods as knowledge about this data set has already be gained by prior indexing and therefore link back to the indexing method. An automatically indexed data set might perform poorly on supervised learning methods as indices are derived without human intervention and therefore no prior detailed knowledge about the data set is present. On the other hand, unsupervised clustering should demonstrate better performance with automatic indexing, which should be more objective and more consistent.

2. ***What systematic recommendations can be established for organizing data by unsupervised or supervised methods?***

In the application of unsupervised and supervised organization methods, multiple parameters have to be set beforehand. Some of these parameters are sensitive to the input data set; clustering and classification results are dependent on those parameters. Through implementation and evaluation of the six classical and modern methods from clustering and classification, recommendations can be established. Recommendations have to be formulated differently for clustering and classification. It is anticipated that systematic recommendations can be specifically given for optimal cluster selection as well as for selection of the optimal clustering methods based on the data set and its underlying ontological structure. For supervised classification it is anticipated to establish recommendations for the selection of the training data set size, as well as recommendations for setting model parameters based on the evaluation of the experiment carried out in this dissertation.

1.7 Problem significance

Data retrieval and knowledge exchange is one of the most fundamental challenges in today's research environment. Recommendations are limited on how to formalize manually derived keywords for data when automatic keyword methods fail. No rigorous empirical

comparison of organization methods for different types of data exists. Results from this research will be beneficial to Geographers and to all researchers working on multi-dimensional data sets across different domains.

By evaluating the grouping of different data sets indexed according to different strategies, recommendations can be given on the most effective grouping method for a given indexing scheme, as well which strategy will fail or succeed depending on the data type used. The choice of indexing strategy will dictate the underlying inherent structure of a data set and whether certain organization method will be successful or not. Depending on the domain of the data set, a manually indexed data set is likely to have a more pronounced hierarchical structure than a data set indexed by an automatic indexing strategy. This is due to the human ability of organize and structure data in categorical ways to process information (Everitt et al. 2001). Drawing the linkage between data indexing strategy and data organization method is therefore essential to all researchers working with multiple data types across domain in order to determine the optimal organization methods depending on its inherent structure by indexing and topic domain.

Research in IR shifted from solely developing methods for one single data type such as documents into IR system incorporating multiple data types and complete multimedia retrieval systems (Blanken et al. 2007). Lessons learned from rigorous evaluation of different organizations of the four different data types in this dissertation will help the IR community to better understand the requirements for the implementation of multi-format retrieval systems where the whole range of data types will be present. Results from this research will also support the choice of organization algorithm applied to variously indexed data sets which is an essential prerequisite to effective information retrieval.

The field of Spatialization within the GIScience domain focuses on the transformation of high-dimensional data into lower dimensional geometric representations by applying spatial metaphors to information spaces (Skupin, 2007). This research will be beneficial to GIScientists

as it evaluates indexing strategies for different data types. Each created indexing schema can be interpreted as its own reference system. By developing guidelines about which indexing strategy to use, GIScientist will be able to more efficiently design Spatialization systems and information system visualizations. The four data sets used in this research represent data sets commonly used by GIScientists. Evaluation of indexing and grouping methods on these data sets will be beneficial to the GIScience community as they will provide guidelines to efficient indexing and organization of these data sets.

1.8 Dissertation structure

The following chapters present a literature review split into two chapters. Chapter 2 covers the historical roots of organization data indexing and emphasizes current methods and limitations in deriving a keyword set. Chapter 3 reviews the methodological foundations about organizing data, distinguishing between classical and modern methods for unsupervised and supervised grouping. Chapter 4 introduces the methodological framework of this dissertation and also describes the indexing process for all four data sets. Chapter 5 covers the data clustering and classification, as well as the comparative evaluation of data organization methods to each type of indexed data. Chapter 6 discusses implications of the research and formulates guidelines on indexing and organization for data types where automatic methods cannot be applied. Chapter 6 concludes the dissertation research with a discussion on the limitations of the applied methodology. It also gives an outlook for extending this research.

CHAPTER II

Literature Review - Overview of existing knowledge

This chapter places the dissertation research into the context of current and past research. It focuses on different indexing approaches of various data types using methods from Information Retrieval (IR), data organization, and machine learning. Three sources for generating indices for data types where automatic indexing fails are reviewed, namely metadata or media tags, auxiliary data derived through additional processing, and patterns inherent in the data elicited by semantic or content-based processing. As each data indexing strategy produces its own information space, concept of ontology and current research trends leading to the semantic web are discussed in regards to the data sets used in this dissertation. The final section places this research into the context of current developments in information science and GIScience. The chapter closes with a discussion of gaps in current research and how this dissertation will help to fill those gaps.

2.1 A review of concepts from Information Retrieval

The field of Information Retrieval (IR) is a multi-disciplinary field with research emphases ranging from traditional Library and Information Science (LIS) to advanced multimedia IR systems (MIRS) (Blanken et al., 2007). The problem of how to structure and organize information has had a long chronology in LIS. The literature differentiates between bibliographic cataloging and alphabetical indexing (Chan, 1981; Taylor and Joudrey, 2009).

In bibliographic cataloging, each description of an item usually appears in only one physical place and can be understood as the simplest form of a retrieval tool (Taylor and Joudrey, 2009). Each bibliography has a focus described by subject, author, language, publisher, or form. Traditional bibliographic classification has been developed in conjunction with descriptive cataloging techniques, which assign an index to an item in a catalog in a way that this item can be used as a locator or address to find it again at a later time. Catalogs provide

access to individual items within a collection of information resources, such as physical entities, online resources, or websites (Taylor and Joudrey, 2009). Bibliographic cataloging is not as important in a digital environment, as it was developed for the retrieval of physical information at a single physical location, such as a book on a library shelf.

However, the second approach, alphabetic indexing is applicable to digital data, as multiple terms can be associated with a data item and reorganization of the catalogs can be conducted relatively easily (Taylor and Joudrey, 2009). It is apparent that the size of today's online data libraries are far too large, but also too diverse and dynamic for manually defined catalogs or catalogs which are based on hard coded knowledge (Taylor and Joudrey, 2009). Figure 2.1 gives an overview of the increased size of multiple cataloging systems over time (Akerkar and Lingras, 2008).

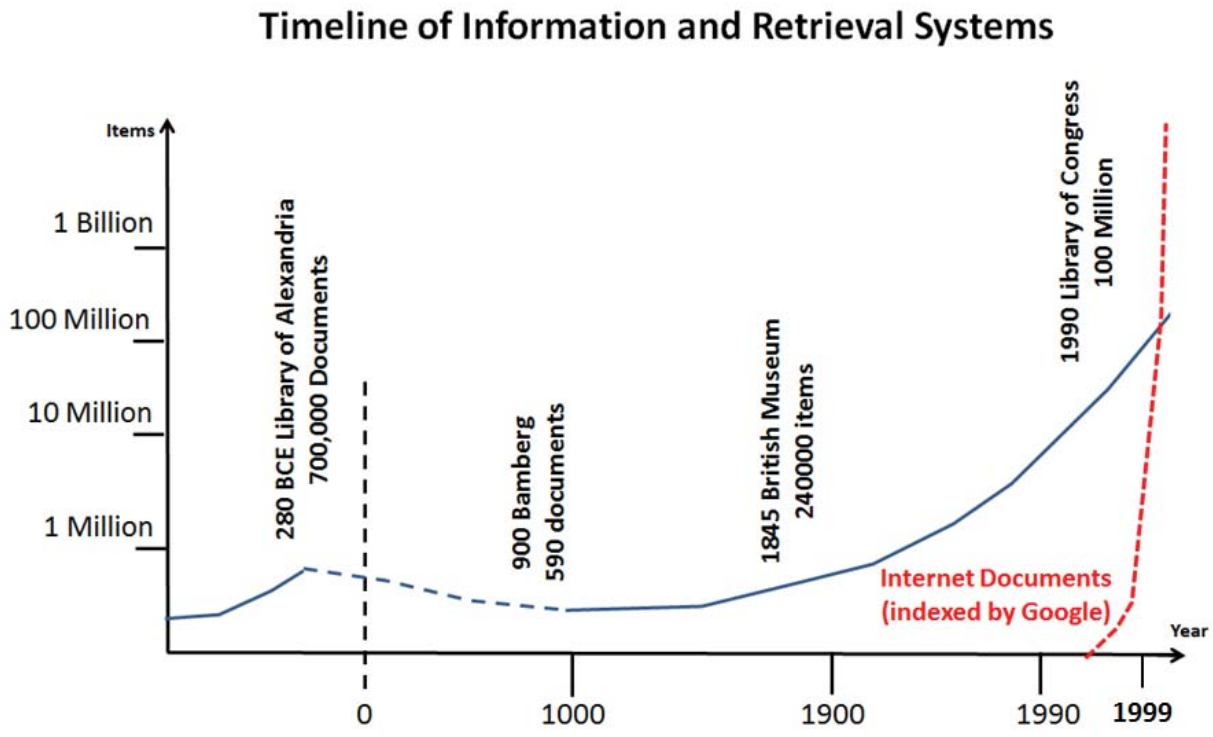


Figure 2.1 Milestones in IR systems (redrawn from Akerkar and Lingras, 2008). The solid blue line indicates known amounts of information, the dashed blue line displays estimated numbers

of items in each collection, and the dashed red line displays online information indexed by Google.

From Figurer 2.1 it is apparent that traditional methods developed in LIS for physical items cannot be applied to the retrieval of very large amounts of data (e.g. Internet), and more advanced methods are necessary.

The necessity to store and retrieve information became increasingly important over centuries. Major inventions such as paper, the printing press, and digital storage technology enabled storage and retrieval of increasingly large amounts of information. With the introduction of personal computers in the 1980's and the emergence of the Internet, data volumes increased and more advanced methods for IR became necessary (Taylor and Joudrey, 2007). With the spread of the Internet alone the amount of catalogued information doubled between 1999 and 2002 (Akerkar and Lingras, 2008). In the field of computer science, methods for data mining and machine learning were developed and applied to IR problems in the mid-1990's (Taylor and Joudrey, 2007; Akerkar and Lingras, 2008; Blanken et al., 2007). The major elements of an IR system are described by Akerkar and Lingras (2008) as including document representation, query representation, ranking and comparison of documents, and evaluation of the quality of retrieval.

Current IR practice incorporates methods from LIS, computer science, linguistics, statistics, and cognitive psychology. Other than LIS, IR research incorporates documents, and also other types of items, such as data, pictures, text, etc. The main idea in all IR systems is to formulate a user specified request and match it against keywords assigned to or found within the text or a collection of data. The idea of a modern IR system was first described by Vannevar Bush's MEMEX (a portmanteau of "memory" and "index") system (Bush, 1945). Bush (1945) envisioned the MEMEX as a device in which individuals would compress and store all of their books, records, and communications, and be able to instantly retrieve all compressed information. The

idea of MEMEX led to the development of early hypertext and is still the foundation of modern IR systems (Segaran, 2007).

Still following the MEMEX concept today, most IR systems are based on the concept that all items in the system can be ranked by estimating the usefulness of a user query. For accommodating large amounts of data, several key developments for summarizing, searching, and indexing include the Vector Space Model (VSM), the probabilistic model, and the inference network model (Segaran, 2007).

Salton's (1971) VSM was the first usable implementation of an IR system. It was first implemented in the System for the Mechanical Analysis and Retrieval of Text (SMART) (Manning et al., 2008). The theoretical foundations of this system are still used in today's modern IR systems. The VSM creates a weight based on the frequency of each token (for example, a word in a full text document), and represents weighted frequencies as vectors in a multidimensional space. The number of generated keywords defines the dimensionality of the vector space. Similarity measurements between vectors are created by subtracting weighted frequencies for corresponding tokens and the entire space is evaluated using the cosine–similarity coefficient. Numerous extensions to VSM have been suggested. Salton and Buckley (1988) introduced a weighting scheme to increase the performance of the model. Weighting schemes are used to offsets the frequency of a word in a text corpus, which helps to control for the fact that some words are more common than other and increase the performance of retrieval (Raghavan and Wong (1986). The VSM model as applied in GIScience incorporates a metaphor in which similarity is measured by distance metrics (Skupin, 1998; Fabrikant, 2000; Viger, 2011). Limitations of the VSM include poor representation of long documents, semantic sensitivity, and the assumption that all terms are statistically independent (Jannach et al., 2010).

The probabilistic IR model, also known as a binary independence retrieval model, was introduced in 1976 (Croft and Harper, 1979; Sparck-Johnes and Willet, 1997). However, the initial concept of probabilistic retrieval systems was published by Maron and Kuhns (1960). The

probabilistic IR model assumes that the IR process can be described as a process in which information needs to be queried and indexed in a probabilistic way. Probabilistic IR systems rank results in decreasing order of probability of their relevance to a user's query. The probabilistic IR model takes into account that there is uncertainty in the representation of the information which states that an IR system is supposed to rank the documents based on their probability of relevance to the query, given all the evidence available (Belkin and Croft, 1992). Other than the VSM which ranks data items by similarity, the probabilistic IR model ranks data items on the probability of relevance (Singhal, 2001). The increased complexity of retrieving information using a probabilistic IR approach makes the model inadequate for web search and "on the fly" indexing, for which no relevant documents are known beforehand and for which queries are typically short. However, the model is helpful in instances such as spam filters. Spam filters accumulate many examples of relevant and irrelevant (spam) documents over time. To decide if an incoming email is spam, the full text of the email can be used instead of just a few query terms (Baeza-Yates and Ribeiro-Neto, 1999).

The Inference Network Model is a newer approach in IR and it is based on a network-based retrieval model. The network based retrieval process is modeled as an inference process in an inference network (Turtle and Croft, 1991). The model encodes probabilistic dependency relationships between variables in the data set. The presentation of probability distributions as directed graphs makes it possible to analyze complex conditional independence assumptions by following a graph theoretic approach (Turtle and Croft, 1991). A benefit of this model is that most models used in IR can be applied within the concept of this model (Singhal, 2001). Inference network models have been widely used in IR systems to implement browsing, document clustering, image retrieval, and video and sound retrieval (Graves and Lalmas, 2002). Inference Network provides better results and performance than VSM and the probabilistic IR model because multiple approaches can be combined into one model (Metzler and Croft, 2004).

This section summarized different fundamental concepts in IR, as well as how information storage and retrieval evolved over time. The research experiment presented in this dissertation makes use of multiple concepts presented in this section. Depending on the data set and on the indexing method used some of the concepts presented here can be applied and others will fail. This dissertation research does not develop new IR models, but rather uses existing models, and shows limitations of the models by using the four data sets. Therefore it is necessary to present and discuss the concepts here.

2.2 Indexing strategies

Indexing is an important step in characterizing data and making it retrievable. In its simplest form, an index can be described as a common method for keeping track of data so that it can be retrieved again (Baeza-Yates and Ribeiro-Neto, 1999). Similar to an index in a book, it is a list in which each entry contains the name of the item and its location (Taylor and Joudrey, 2009). However, computer-based idiocies may point to a physical location on a disk or to a logical location that point elsewhere to the actual location.

An indexing strategy can be described as a method on how to best characterize data. Indexing strategies vary by data type and depend on the data type, as different methods can be applied. Taylor and Joudrey (2009) refer to three types of indexing. Back-of-the-book indexing provides an alphabetical organization; database indexing partitions the resource into categories; and web indexing collects data systematically and continuously. Only database indexing and web indexing are discussed in this chapter, as these are the most relevant indexing strategies for the data sets used in this dissertation research.

The following subsections focus on different approaches for indexing different types of data. The section reviews current and past approaches ranging from automatic document indexing, and indexing data types with metadata, to indexing of data types requiring auxiliary information for creation of an index. This subsection starts with document indexing where

approaches from natural language processing are reviewed. It then gives an overview of current indexing strategies used in online search frameworks. The section ends with a review of indexing strategies of complex data types such as video, sound and software where metadata and auxiliary sources are used for index creation. This dissertation will apply organization methods on four differently indexed data types. Therefore it is necessary to cover previous literature in indexing strategies for different types of data.

2.2.1 Natural language indexing

Document indexing is one of the earliest methods used in IR systems. It has its origin in library science and early implementations of indexing and cataloging documents date back to ancient libraries, such as the Library of Alexandria around 280 B.C (Taylor and Joudrey, 2007). As only digital data sets are used in this dissertation, the focus in this subsection is on modern approaches used in document indexing.

Automatic indexing is used in many different systems ranging from online search engines to natural language processing. Although natural language processing and document indexing in IR are considered separate fields in computer science, both fields rely on the same methods, such as text stemming or text segmentation analysis. Current research and applications conflate the gap between these two fields (Manning et al., 2008). Text stemming in IR also forms the basis for statistical analysis, text clustering, and text classification (Manning et al., 2008).

In order to apply automatic methods for indexing on documents, the text must be broken up into distinct meaningful units, also referred to as tokens. Tokens are formed on simple heuristics, such as identifying white spaces which separate words from each other, or a contiguous string of alphabet characters which form one token (Kaplan and Bresnan, 1982). After a text is broken up into tokens, 'stop word' removal is applied. Stop words are articles, generic pronouns, and conjunctions with little meaning such as "the", "and", or "it" and are usually defined in a collection of frequency. However, over the last decade the trend in IR has

been to not use stop words at all or only use a limited number of stop words for optimization. Such implementations can be found in Apple's Siri, Wolfram Alfa, and Google Search (http://www.apple.com/iphone/features/siri.html, accessed July 2012; http://www.wolframalpha.com/, accessed August 2012; www.google.com, accessed August 2012).

After tokenization and stop word removal, word stemming can be applied. Word stemming refers to the process of word conflation into linguistic stems. For example, word stemming conflates words such as "computer," "compute" and "computation" to the root stem "compute" (Akerkar and Lingras, 2008). The first automated word stemming routines were written by Julie Beth Lovins in 1968 (Lovins, 1968). The literature names the Porter, WordNet database, and Lancaster as the most frequently used stemming algorithms (Hull, 1996; Frakes and Fox, 2003; Bird et al., 2009). The Porter algorithm is the oldest and most frequently used stemmer (Bird et al., 2009). The Porter stemmer is based on context-sensitive suffix removal (Porter, 1980). The algorithm takes a conservative approach of reducing words to stems. As the oldest stemming method, the algorithm performs more slowly than newer stemmers.

A different approach to stemming is implemented in WordNet, a lexical database of English nouns, verbs, and adjectives grouped into synonyms, each expressing a distinct concept and meaning of the word. The WordNet stemmer resembles a thesaurus and is used for grouping words by their meaning (Fellbaum, 2005). The WordNet lexical database also builds the foundation for many newly developed stemmers. A drawback of the WordNet stemmer is that it is based on lexical knowledge for the natural English language and it might not detect computer-specific or algorithm specific words.

The Lancaster algorithm, also referred to as Paice/Husk stemmer, takes a more aggressive approach than the Porter stemmer (Paice, 1990). The algorithm is a conflation based iterative stemmer. It is based on a single set of rules, each specifying the removal or replacement of a word ending in its stem. This method of replacement is used to avoid the problem of spelling

exceptions by replacing word endings, rather than simply removing them (Paice, 1990). The Lancaster is also the fastest algorithm and suitable for large to very large data sets (Bird et al., 2009). The Lancaster stemmer is applied to the full text data set used in this dissertation, as it identifies more alternative word forms than other stemming algorithms, (Bird et al., 2009).

A term-document matrix is constructed after stemming is applied. A term-document matrix is a numerical matrix that describes the frequency of how often a term is present in a document. A term-document matrix without any further processing is usually very sparse. Without setting a cutoff value, every word would appear in the term-document matrix and would create unnecessary noise in the data set. However, setting an appropriate cutoff value depends on the purpose of the term-document matrix. If the purpose is information retrieval or document queering, a low cutoff value should be considered as fine distinctions between documents as necessary. If the purpose is document clustering or classification, a higher cutoff value is beneficial as otherwise too many keywords (dimensions) are kept and complexity increases (Manning et al., 2008). Previous studies suggest setting a cutoff value between 5% and 25% of the number of documents in the term-document matrix (Manning et al., 2008; Bird et al., 2009). However, this number has to be set by empirically accessing the cutoff value based on the purpose of the indexing exercise. Natural language indexing is used for the full-text data set as well for the algorithm data set in a modified form.

2.2.1 Current online indexing strategies

Many different online search services are available. Google, Bing, and Yahoo! are the most commonly used search engines (http://gs.statcounter.com/#search_engine-ww-monthly-201010-201012, accessed November 2011). In addition to general-purpose search engines, other more specific catalogs, for example, online library catalogs, image repositories, such as Getty Images (http://www.gettyimages.com/, accessed November 2011) or spatial data distribution services, such as USGS Explorer (http://www.earthexplorer.usgs.gov/, accessed November 2011) exist. Indexing strategies can be broadly grouped by the type of data which they organize. Each

strategy has advantages and also limitations. The following gives an overview of current online search engines grouped by their media.

Web search engines

Modern Internet search engines, such as Google Search, Bing, or Yahoo! are based on a large scale indexing service. Before a search engine can point to a file or document, the document must first be found by an indexing service. This process of making documents visible in online searches is called web crawling (Kobayashi and Takeda, 2000). The user can only find websites or documents that have been indexed. Indexing is completely done as a back-end process which user interaction such as querying is implemented as a front-end process. Figure 2.2 overviews how online search engines function. The indexing process of modern search engines can be grouped into front-end and back-end processes.

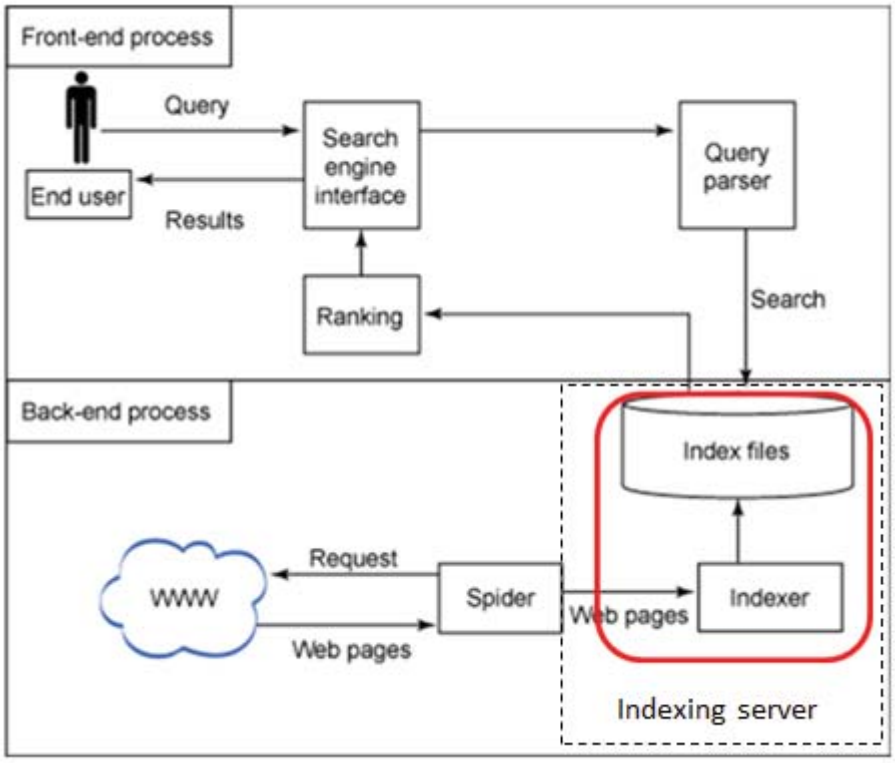


Figure 2.2 Typical architecture of a search engine (redrawn from Zhou and Davis, 2006). The red box highlights the focus of this dissertation

Front-end processes include all elements and steps necessary to interact with a user. That includes the search engine interface, the query parse which translates user requests to be processed by the indexing server. Search results are returned via a ranking service which ranks the retrieved documents according to the user request.

The back-end processes include web crawling and indexing services. A web crawler is a computer program that crawls the web in an orderly manner. A web crawler, also referred to as a spider, acquires a copy of the website while visiting the website. It can be described as an information harvesting system. While the spider is browsing websites and documents, the indexing service keeps track of words within the page and where they can be found on the website or document. An index is built with a ranking system based on word context and frequency (Zhou and Davis, 2006). This process varies by search engine provider. Weights can also be put on certain words depending on their importance. The purpose of the index is to provide searchable results, but more importantly to speed searching for users. Depending on which search provider is used, advertising and paid services offered by the search engine provider can give websites or documents a higher priority in the search result and can also add additional keywords for indexing (Google Search, 2011). The indexing service can be understood as a metadata creation system.

Indexing can be formally defined as the process of describing an information resource in such a way that a user becomes aware of the item's basic characteristics (Taylor and Joudrey, 2009). The examples below highlight five different indexing strategies including library catalogs, multimedia content catalogs, music indexing catalogs, and geospatial data catalogs.

Library catalogs (Library of Congress)

The Library of Congress, with more than 147 million items, is the largest and most comprehensive library in the world in terms of shelf space and number of resources (Cole, 2004). This section will only cover digital searching capabilities and not the physical library catalogs. The Library of Congress offers multiple methods for indexing on relevance, and as

with most other library catalogs, works on descriptive metadata. The Dublin Core elements were developed to standardize the way bibliographic information is stored (http://dublincore.org/, accessed July 2011). Dublin core elements consist of a set of vocabulary terms which act as metadata to describe resources in a library collection. Most library systems index their catalog on this basis. This standard consists of 15 core elements such as contributor, data, format, publisher, or language.

Multimedia content catalogs (Netflix, YouTube)

Multimedia catalogs are a collection of multiple data types specifically targeted to multimedia data such as animation, video, and inveracity content. Netflix is the world's largest online video streaming service (http://www.netflix.com, last accessed August 2011). The Netflix catalog contains a very large collection of TV shows and movies characterized by many different attributes and by a rating system. The rating system consists of a machine learning algorithm (Koren, 2009), which permits a user to rate movies which they have seen and which assumes that users rate older movies differently than movies seen recently (Koren, 2009). The indexing algorithm is considered one of the most advanced and most powerful in multimedia retrieval (Koren, 2009). A limitation of the current indexing system is that it only works on video and is designed solely to establish preference. That is, it is not adapted for analytic searching; for example, one cannot search portions of a movie, nor retrieve all movies which contain storylines about Indonesian culture. Exciting data organization methods have emerged and been promoted by the Netflix Prize (Bennett and Lanning, 2006). Offering a $1 million USD award, Netflix distributed a data set with over 100 million movie ratings, soliciting novel organization methods to improve the current performance of Netflix's organization and retrieval algorithm by at least 10%. The winning research group introduced new matrix factorization techniques for organization and retrieval of the data (Koren and Bell, 2009).

YouTube, the world largest video sharing platform (http://www.youtube.com, accessed September 2011) is another example of an online multimedia search engine. In contrast to

Netflix, YouTube has a much simpler indexing structure based upon predefined metadata where users have to tag their videos schematically. The indexer relies on metadata search, and does not incorporate advanced video analyzing techniques.

Music indexing catalogs (Pandora)

Numerous online music streaming and search engines are available. Pandora is one commonly used online music streaming service (http://www.pandora.com, accessed September 2011). On Pandora, the user can search for music, but addition to giving the user the exact song, similar music is also offered. Pandora is based on the Music Genome Project (Joyce, 2006). Pandora does not index by means of a traditional metadata concept of genre, user connections, or ratings as do other online streaming services. Manual indexing by experts and users is characterized using over 400 musical attributes covering the qualities of melody, harmony, rhythm, and lyrics. Music with similar traits and structures is linked together through this very high dimensional metadata database. Pandora is one of the only large streaming providers in which manually derived keywords are used to index the archive.

Geospatial data catalogs (USGS Earth Explorer)

The USGS Earth Explorer is a comprehensive catalog which allows searching for spatial data sources worldwide. The user is able to search for different types of spatial data sets including aerial photography, raster data sets (e.g. land cover), vector data sets (e.g. National Hydrographic Data set (NHD) stream), or digital imagery (e.g., Shuttle Radar Topography Mission (SRTM) data). The indexing strategy is based upon metadata tags providing a geographic location, a date, or a requested data type.

The major limitation of all the presented exemplars of indexing frameworks, with the exception of Pandora, is that they work on automatically derived indexing schema, either through automatically derived keywords or by requiring that auxiliary data is present in the form of metadata or media tags.

2.2.2 Indexing on manually derived keywords

The two prior sections on natural language processing and online strategies for indexing with the exception of Pandora rely mainly on automatically derived keywords. However, many types of data, such as algorithms, data sets, and imagery, do not contain discriminatory keywords explicitly; as a consequence, a manually derived descriptor set must be established. Indices describe the content of documents. Indices can be created manually or automatically by means of predefined terms, and an algorithm designed to find relationships in the data and translate these relationships into meaningful keywords.

The literature describes three ways of indexing using manually derived keywords: 1) by means of metadata or media tags, 2) by auxiliary data which requires processing beyond simple item frequency counts, or 3) by adopting content-based indexing methods drawn largely from the fields of computer vision, image processing, or signal processing.

Metadata or media tags are customarily separated from the data, imagery, or sound files and include file descriptors, rather than explicit file content. Nonetheless, information for keyword generation may be extracted automatically from metadata or media tags. For auxiliary data that is not metadata, information is processed from the original data, but requires inference or classification to create a keyword set. In contrast to content-based searching which is based on semantic matching and pattern recognition, keywords derived from this type of indexing are purely data-driven.

When all three indexing methods fail, as in the case of indexing software, a manually derived keyword set must be generated. This dissertation will incorporate examples of data types which require automatic and manual keyword generation.

2.2.3 Metadata as a source for keywords

Blanken et al. (2007) describes indexing as the process of deriving metadata from documents, and the storage of the metadata as an index. Metadata also called media tags for

some types of documents. Depending on the data type and on the agency or individual creating those metadata, metadata tags can differ in content and structure. For example, metadata for spatial data usually incorporates keywords for date, spatial footprint, creator, steward, and, on occasion, a short description of the area or object. Media tags for music or video files usually contain information about the date, genre, media format, and artist.

Such a system has been successfully implemented for video on websites such as http://www.youtube.com (accessed October, 2011). The user enters a description of the video based on the provided media tag schema. Media tags may include information about camera type, exposure, focal length, date, or even GPS coordinates added by the capturing device (e. g. camera). Poor metadata updating or inconsistent generation of metadata can lead to an indexing scheme which is initially stable, but which becomes unmanageable over time.

Metadata exist for various data types. But for some data types, such as software or algorithms, no metadata is available, and an implicit keyword set needs to be developed manually and utilized as a surrogate for metadata. Both approaches for metadata indexing will be utilized in this dissertation.

2.2.4 Auxiliary data as a source for deriving keywords

When no metadata is present, keywords cannot be automatically captured as with full text documents; however, additional processing methods can capture auxiliary data and use these to develop keywords. For example, automatic generation of metadata for music has been proposed by Klapuri (1999) where pitch detection, frequency, duration, or periodicity of the audio signal is used to characterize the type of sound. Advanced methods from speech indexing have been proposed by Blanken et al. (2007) where speech recognition is used to generate keywords directly. Using image processing methods, feature recognition can proceed geometrically, for example using size, shape, color, or texture (Jensen, 2005). Another example of auxiliary indexing is given for indexing spatial data in a database environment. Data

architecture, such as R-trees, are used in which objects represented as shapes, lines, and points are grouped using methods, such as overlapping convex hulls or common minimum bounding rectangles (Rigaux et al., 2002).

Content based image retrieval, also referred to as semantic image retrieval, uses the content of the image, rather than metadata, as a basis for indexing. Automated methods from computer vision are used to retrieve content, such as objects or elements in an image file, which is then used for characterization into content. Emphasis in content-based indexing lies on image analysis, where objects or scenes are extracted out of the image (Gonzales et al., 2004). An example of an automatic detection method in a geographic context is shown in Figure 2.3.

Figure 2.3 An example of an automatic detection method of road network extraction from areal imagery. Extracted roads are displayed in red (http://gis.incogna.com/?p=technology#Road, accessed June 2013)

These methods detect the presence of objects or features based on semantic patterns. Automated detection methods are then used to characterize this information for keyword generation (Robert et al., 1973; van der Heijden, 1994). State of the art methods include pattern recognition using correlations (Blanken et al., 2007), face recognition using Eigen objects (Steger, 1998), and recognition of objects using active shape models (Turk and Petland, 1991).

When moving on to data types such as video files, the generation of keywords become even more complex. A problem of video file indexing, when no media tags are provided, is the

added time component. Instead of a static single image, 25 or 30 images per second have to be analyzed. Advanced content-based methods for indexing of video have been implemented by Jain et al. (2000) using statistical pattern recognition. Other methods include implementation of support vector machines (SVM) for semantic video classification (Burges, 1998; Vapnik, 2000).

2.2.5 Manual methods for keyword generation

For certain data types, metadata and media tags may be absent. Auxiliary processing may be insufficient or computationally complex to retrieve relevant content. Automated content-based methods may provide keyword sets which are insufficient for distinguishing among items, or which cannot capture salient, semantically meaningful descriptors. The two specific examples of data types which are characterized by these indexing challenges are software code and algorithm descriptions. This dissertation will explore example datasets of both types, studying manual generation of keywords, and the impact on indexing and data organization.

Automatically derived keywords from software tools can be generated by analyzing the header information, specifically author, chronology of modification, or input and output requirements. But these cannot be generated automatically as no standardized descriptor exists (Viger, 2011). Moreover, the author, chronology, and input/output (I/O) constraints may not be sufficient to identify what task the code actually accomplishes. Automatic documentation of software exists, but also varies depending on which programming language is used (Zanoni, 2011; Wang et al., 2010). An automated index for software might fail because most software, scripting and program command languages are artificial rather than natural. For example, a structural analysis or word count might return the number of 'for' loops, or a list of variable names used, but will not return the meaning of the code. There are automatic processes that can be applied to characterize textual media, but characterization of software or software-code has still to be done manually, as no standardized metadata attribution for algorithm or software code exists (Tangsripairoj and Samadzadeh, 2006). Viger (2011) characterizes software modules in ways that correspond more directly to the abstract concepts chosen by users and to the

environmental models that they seek to use. His work demonstrates that the set of keywords has an effect on the subsequent organization of the data set, whether those keywords are automatically or manually derived. Essentially, the keyword set provides one of many views or perspectives into the data being indexed.

In contrast to software code, algorithms are usually formulated in natural language and their descriptions are customarily embedded in full-text documents, for example in technical papers or scientific journal articles. Algorithms provide a special case in which full-text documents can serve as metadata, and therefore, methods can be modified from document indexing using automatic keyword generation. In the GIScience context, researchers have created taxonomies to characterize algorithms, for example in characterizing cartographic generalization algorithms (McMaster and Shea, 1989; Buttenfield and Mark, 1991; McMaster, 1991; Regnauld and McMaster, 2007). In this dissertation, keyword sets drawn from various taxonomies will be used to index an archive of cartographic generalization algorithms.

2.3 Concepts of Ontology and Semantic Web

The two earlier sections in this chapter reviewed strategies of indexing for data organization. In creating indices, each of these strategies presented earlier construct their own information space as well as creates their own corresponding underlying ontological structure. Furthermore, the four data sets used in this dissertation are organized by multiple organization methods, each generating their own information space. It is therefore important to review concepts and current development of Ontology which are presented in this section. This section will first discuss the definition and concepts of Ontology and then reviews current developments such as the Semantic Web.

Ontology has its origin in the field of Philosophy and describes the study of the nature of being, becoming and existence, as well as the basic categories of being and their relations in the whole context (Oberle et al., 2009). In the literature, the term “Ontology” is often referred to as

Metaphysics which is used in a broader sense to describe the study of what might exist and how existence can be defined (Geisler, 1999). However, from a Computer Science perspective Ontology is focused on establishing fixed, controlled vocabularies where objects can be grouped into (Gruber, 2008). Ultimately, Ontology can be described as a data model that represents a domain. For example most indexing frameworks are Ontology based, such as the Dublin Core or the WordNet database which structure documents by predefined categories (Navigli and Velardi, 2004). A commonality about the definition of Ontology that is shared by the philosophers and computer scientists is how entities can be grouped and related within a hierarchy according to similarities and dissimilarities (Ingarden 1964, Chrisholm, 1996).

Ontology research in the Computer Science field has its roots in the subfield of artificial intelligence (AI). Early research in the mid-1970 focused on the creation of new Ontologies as computational models that could be used for automatic reasoning (Smith and Welty, 2001). Furthermore, researchers argued to use to term ontology to refer to the theory of a modeled word as well as all components that constitute such a system. In the 1980s researchers started drawing concepts and inspirations from philosophical otology research to define ontology research in the field of computer science as computational applied ontology (Gruber, 2008). In 1993, Gruber defined the still valid technical definition of ontology as a mean to specify a conceptualization. Gruber defines Ontology as "… a description (like a formal specification of a program) of the concepts and relationships that can formally exist for an agent or a community of agents. This definition is consistent with the usage of ontology as set of concept definitions, but more general. And it is a different sense of the word than its use in philosophy …" (Gruber, 1993:5).

Current Ontology research can be broadly divided into domain Ontology research and upper Ontology research. Domain Ontology research focuses on a specific domain and methods for generating such systems are specific tailored to model that domain. While being accurate in modeling a specific domain they are often incompatible across domains. Current research

focuses in overcoming the problem of incompatibility by developing generalized techniques for merging multiple domain Ontologies (Zablith, 2008). In contrast to domain specific Ontologies, upper Ontologies are models of common objects that can be applied across a wide range of domains. Multiple standardized upper Ontologies have been developed such as the Dublin Core elements or the WordNet database (Zablith, 2008). In order to generate modern Ontologies multiple Ontology languages have been developed. An Ontology language is a formal language to encode the ontology system. The most common Ontology language is UML (Unified Modeling Language) (http://www.uml.org/ , accessed August 2013). UML offers standard procedures to visualize data schemas and system architectures for systems such as database schemas, software components or business processes.

One of the largest implementation of ontological concepts is the Semantic Web. The Semantic Web, also referred to as Web 3.0 is aiming at converting the current web which consists of unstructured and semi-structured documents into a "web of data" by inclusion of semantic content in web pages (Berners-Lee, 2001). The inclusion of semantics enables searching, sharing and information harvesting and makes it more feasible compared to the non-semantic Web (Chebotko and Lu, 2009). The current structure of webpages is tailored as well as restricted to HTML (Hypertext Markup Language) which allows only to link documents with documents but is not able to semantically linking those pages with content. For example HTML is only able to describe a string of text inside of a HTML tag, but it is not able to describe it as a discrete object which is interlinked on a semantic level with other objects (Chebotko and Lu, 2009). The goal of the Semantic web is to link data to data not only by linking them but also by taking metadata information of each object into account and describing them by semantic understanding (Heery and Wagner, 2002). To support semantic linkage of data objects, many different programming languages have been proposed. Most notable languages are Resource Description Framework (RDF) which is a general method for describing information, Web Ontology Language (OWL) which is a family of knowledge representation languages, and Rule

Interchange Format (RIF) a framework of web rule language dialects supporting rule interchange on the Web.

The adoption of Semantic Web technology has been very slow compared to other technical advances in Computer Science (Butler, 2002). There are many critiques regarding the Semantic Web. Critiques include practical feasibility, privacy concerns and the doubling of output formats. Practical feasibility is limited by personal behavior in generating Metadata for the Semantic Web. Metadata is often misleading or falsified. Gärdenfors (2004) and Honkela et al. (2008) point out that logic-based semantic web technologies cover only a fraction of the relevant phenomena related to semantics. Privacy concerns of the Semantic Web have been raised as methods for generating semantic information by using machine learning and text analysis methods which make it much easier to identity patterns (http://www.policyawareweb.org , accessed August 2013). Generating content for the semantic web is doubling of output formats and the associated time creating that content. However, through the implementation of RDF that allows existing content to be converted for semantic searches this critique has been partially addressed.

Information gained for best usage of indexing and grouping methods for the four different data sets used in this dissertation will contribute to current Ontology and Semantic Web research. The information gained through this experiment will be especially valuable to domain and cross-domain Ontology research as recommendation for best methods usage for data and multi data type systems can be given which will be relevant in developing future Ontology systems.

2.4 IR in the field of GIScience

Over the last decade researchers in the field of GIScience started to apply and adopt methods from IR. IR research in GIScience is mainly focused on the application of geographic principals in knowledge discovery with the use of spatial metaphors. The application of IR

principles in GIScience is also referred to as Spatialization. Spatialization is the transformation of higher-dimension data into lower-dimension representations on the basis of computation methods and spatial metaphors (Skupin, 2007).

The usage of spatial metaphors and the concept of an information space where the distance between items reflects similarity are not unique to GIScience. Major contributions to research can be found within the computer science domain, where researchers for example used information spaces for library collection access (Mayer, 2011) or for accessing music library by genre (Moerchen et al., 2006).

The contribution of GIScience to this field lies mostly in information space handling and in the visualization of information (Kuhn and Blumenthal, 1996; Couclelis, 1998; Dodge and Kitchin, 2000; Fabrikant and Skupin, 2005). Spatialization research can be broadly grouped into computational approaches for visualizing information spaces, and research that concentrates on human-computer interaction, such as cognitive processes of users interpreting the results of an information space. Besides computer science and GIScience research, cartographers apply geographic principles and cartographic practices to the visualization of non-spatial information (Skupin and Fabrikant 2003).

As mentioned above, GIScientists are especially interested in the graphical representation of information spaces. Most graphical displays are limited to two or three spatial dimensions. Graphical variables, such as size, shape, texture, or orientation, can also be used to increase the dimensionality of the system (Fabrikant and Buttenfield, 2001). In order to use an information space, the dimensionality of that space has to be reduced to conform to the limitation of the graphical space. This reduction can be described as a transformation of the information space, in a similar manner to a cartographic projection (Skupin, 2007).

Visualizing information spaces enables users to better understand the complexity of relationships within the data where tabular presentations or statistical summaries might overwhelm the average user (Gahegan, 1999). By applying spatial metaphors to information

spaces, the user is able to explore information spaces in a similar way as a physical landscape. Cartographers have been applying cartographic visualization methods to enable clearer communication about the visualized patterns. A newer research track in Spatialization focuses on the evaluation of cognitive theory and human responses (Tversky and Lee, 1998; Hartley, 1977; Goldstone, 1994; Fabrikant at al., 2006; Fabrikant and Montello, 2008; Fabrikant et al., 2008).

Many researchers in GIScience have developed Spatialization displays and systems. The GeoVISTA Studio from Pennsylvania State University, which includes multiple data visualizations for exploratory data analysis, is perhaps one of the most extensive and interactive implementations. It incorporates multiple displays and analysis methods and links non-spatial and spatial data for analysis. Other spatialization visualization includes the visualization of very high dimensional data sets such as a visualization of the LastFM music library (Skupin, 2012) or the spatialization of AAG conference abstracts (Skupin, 2010). Some IR related research in GIScience also focuses on extending algorithms for spatialization such as the GEO-SOM, which is a geographic extension of the SOM algorithm (Bação et al., 2005).

2.5 Future research trends

The amount of data the average person is dealing with on an everyday basis increases steadily. Developing and adopting methods for accessing information stored in multiple data formats will be essential. New research trends, relevant to this dissertation, can be spotted in natural language processing, multimedia retrieval, and crowdsourced information access.

Besides being one of the oldest research fields in IR, natural language processing methods became increasingly important and new developments emerged, since the introduction of the Internet. Newer research in natural language processing is focusing on optimizing web search engines, junk e-mail filters, and more efficient text indexing methods (Singhal, 2001; Segaran, 2007; Manning et al., 2008). Research is focusing on user centric systems where the user may

enter any natural language, such as words, phrases, or sentences to the system (Blanken et al., 2007). Newer research also focuses on advancing existing methods with term weighting, relevance computing, and probability of usefulness of the results to the user.

Multimedia retrieval research is a relatively new research field. Due to digital recording devices this field became increasingly popular during the last decade. Current research is focusing on combining different data types into one single system. Multiple indexing schemas have to be created to incorporate multiple types of data. Newer research in this area focuses on data types such as digital videos or speech indexing, where automatic indexing becomes complex. Methods from Machine Learning are being applied for generating indices. Newer approaches also incorporate multimodal content-based indexing based on Bayesian networks (Blanken et al., 2007).

Manually created keywords are very expensive to create. Crowdsourced indexing strategies are a newer trend for developing manually created keywords. In a crowdsourced indexing strategy the user is annotating metadata and only a structure of how the metadata has to look like is provided. Many commercial systems have been implemented where otherwise automatic methods would fail for which the general user is tagging and creates an index of the data. Vimeo (http://www.vimeo.com, accessed May 2013) and Picasa (http://picasa.google.com/, accessed August 2012) are the most prominent ones. In both systems, the user is presented with pre-defined media tags, where certain items have to be filled out by the user, such as theme, video type or geographic location. This is in contrast to Pandora in which only expired indexed music by a predefined metadata structure. The national Finnish library introduced a game where users are fixing indexing errors or extending indices of items found in their library system. 93,000 volunteers already indexed documents through their indexing game (http://www.digitalkoot.fi/en/splash, accessed August 2012). An example of modern crowd-sourced geographic data indexing can be found at OpenStreetMaps

(http://www.openstreetmaps.org, accessed August 2012), where users are not only uploading and digitizing geographic objects, but also indexing objects via attribution.

2.6 Summary

The research presented here in this chapter reviews keyword generation, indexing, concepts from the field of Information Science, concepts of Ontology and the Semantic Web, as well as applications in GIScience. The multiple frameworks which have been presented in this chapter are partially applied in this dissertation.

This dissertation research will not implement a complete IR system, but rather evaluate methods for optimal organization of multiple indexed data types. The impact of manually derived keywords on the indexability of data types has not been completely addressed in the literature. Most research focuses on documents and multimedia file types (e.g. video, sound, images). As mentioned by Joyce (2006), creating manually derived indices is time consuming and expensive; thus it is important to establish first that the time and effort is worth the subsequent improvements for indexing these data types.

The focus of this dissertation lies in the evaluation of organization methods on differently indexed data sets. Each data set in this dissertation is indexed by one indexing strategy that is, when possible by automatic indexing. This dissertation does not aim to develop new indexing strategies, but rather use an efficient automatic indexing strategy when possible. Data indexing is the first step in data organization by creating an ontological framework for each of the indexing and organization methods used in this dissertation.

The second context for this research is clustering and classification, which is utilized in many disciplines, for example taxonomy, statistical analysis, data reduction, and machine learning. It focuses on organizing data as well as on methods to assess the quality of classification strategies. This material will be discussed in the next chapter, with regards to the data sets and methods used in this dissertation.

CHAPTER III

Methodological Review

This chapter focuses on the methodological foundation of this dissertation. The chapter starts with an introduction to concepts for organizing data, and discusses past and current developments. Organization utilizes the keywords developed during indexing to group data items in various ways. The major part of this chapter reviews core concepts of organizing data using classical and modern methods from unsupervised clustering to supervised classification. This chapter further discusses optimal cluster selection as well as reviews concepts for cluster evaluation. It would be beyond the scope of this dissertation to discuss all methods. The methods used in this dissertation will be described in the following order:

- Classical unsupervised methods (PCA, Hierarchical and k-Means clustering);
- Classical supervised methods (k-Nearest Neighbor and Classification Trees);
- Modern unsupervised methods (Self-Organizing Maps); and
- Modern supervised methods (Support Vector Machines).

The methods are chosen based on their common usage throughout several disciplines (Estivill-Castro, 2002; Everitt et al., 2001).

3.1 Concepts for organization of data

Organizing objects into groups is a fundamental ability of humans. It goes back to the beginning of human history, when humans tried to distinguish among edible and poisonous foods (Akerkar and Lingras, 2008). Grouping also forms the basis for development of language, which consists of words that help humans to detect and discuss events, objects, and people (Everitt et al., 2001). It is only through grouping that data meanings or relationships which would otherwise be hidden can be derived from large data collections and archives. Indexing and keyword generation forms an essential prerequisite to data organization since it is the keywords, rather than the data or documents themselves, which are organized.

The two types of data organization explored in this dissertation can be referred to as classification and clustering, which are also referred to respectively as supervised and unsupervised organization or learning. Classification, in a scientific sense, has its origins in biology, where it is also known as numerical taxonomy (Sokal and Sneath, 1968). In the field of machine learning, supervised methods are referred to as learning algorithms (Hastie et al., 2001).

Supervised learning is a method in which examples of an expected outcome are provided to improve the system. This whole process is analogous to learning with a teacher. Many pattern recognition algorithms are based on supervised learning, such as the automatic extraction of forested areas from a satellite image by means of presenting typical radiometric characteristics for forested areas. The algorithm is trained with different samples before detecting areas similar to these samples automatically (Bramer, 2007). Classical and modern methods in supervised learning include Bayesian Classification, Nearest Neighbor Algorithms, Decision Trees, and Support Vector Machines. Three of these methods will be utilized in the dissertation.

Unsupervised learning can be seen as the converse of supervised learning. Instead of training an algorithm with sample data, the algorithm sorts input data on the basis of similarities and differences in one or more variables. Taking the forest example from above, an unsupervised learning algorithm cannot distinguish directly between forests and non-forests; rather it can create groups or partitions on the basis of radiometric differences in each spectral band, leaving the interpretation of categories ("forest", "not forest") to the analyst. The best known examples of unsupervised classical methods include cluster analysis and dimensional reduction, often accomplished by some form of factor analysis (Handl et al., 2005). Popular methods in unsupervised learning include hierarchical clustering, k-Means clustering, Principal Component Analysis (PCA), and Self-Organizing Maps (SOMs), all of which will be explored in this dissertation.

3.2 Measures of similarity and difference used in data organization

The most important step in organizing data is to distinguish among differences to find similar objects in a data set, which can form meaningful groups. All methods of clustering and classification organize data on the basis of some kind of similarity measurement. Sokal and Sneath (1963, page 3) define the measurement of similarity as "(…) the ordering of organisms into groups on the basis of their relationship, that is, of their association by contiguities, similarity or both". Many different methods for measuring similarity exist, and are tailored to work with categorical data, continuous data, and hybrid data containing both categorical and continuous data. All three data types are represented in the four data sets used in this dissertation.

3.2.1 Categorical data

Many different methods have been proposed to measure similarity for this kind of data. One common form of categorical data is in binary form (Everitt, 2001), although data is often partitioned into more than two categories for geographic analysis. For example more than two classes are used when working with land use data (e.g., residential, commercial, industrial, vacant) or with ethnicity (e.g., Hispanic, Asian, Black, White).

The simplest dissimilarity measure for categorical data is the Simple Matching Coefficient. This is used when data is in symmetrical format, which means the presence or absence of a characteristic is equally informative. The metric computes pairwise similarities on a binary basis. Either the characteristic is present for both members (positive), absent in both (negative), or present in one, but not the other. The coefficient can be calculated for any number of characteristics by modifying the formula:

$$s_{ij} = \frac{p + r}{t}$$

where p represents the number of positive matches, r represents the number of negative matches, and t represents the number of categories being tested (Everitt, 2001).

A more complex measurement of categorical data is the Jaccard (1908) coefficient, which measures the asymmetric information of binary variables. The Jaccard coefficient is based in set theory, and can be described as the size of the intersection of a number of sets (number of categories which match) divided by the size of the union (the number of categories found in at least one set). It can be applied also to more than two categories:

$$s_{ij} = \frac{p}{p + q + r}$$

where p is the number of positively matching categories, q is the number of matches that are positive for the i^{th} object and negative for the j^{th} object, and r is the number of matches that are negative for the i^{th} object and positive for the j^{th} object. Other categorical similarity measures are summarized in Sokal and Sneath (1963).

3.2.2 Continuous data

When dealing with continuous data, one can measure either similarity (the difference in variable values among observations) or proximity (which can be expressed either as a difference or as a correlation). Two similarity measures applied to continuous data include intra-cluster similarity, describing the distance among items in one group; and inter-cluster similarities, measuring the distance between groups.

Proximity measures for continuous data can be constructed on the basis of distance or correlation (Everitt, 2001). The distance analogy is stated that if two data points in a given data set are quite similar, they will have a small distance between them. If these two are considered close together, a third point will have a similar proximity to each of them. Correlations between data observations have also been proposed, essentially basing similarity on variance (Bansal, 2004).

3.2.3 Hybrid data

Hybrid data includes a mix of both categorical and continuous variables. Different methods are available for measuring similarity for hybrid data. One method dichotomizes all variables and applies binary similarity measurements to each dichotomy (Everitt, 2001). Grower (1971) suggests a more sophisticated similarity measure:

$$S_{ij} = \frac{\sum_{k=1}^{p} w_{ijk} r_{ijk}}{\sum_{k=1}^{p} w_{ijk}}$$

where r_{ijk} is the similarity between the i^{th} and j^{th} individual as measured by the k^{th} variable and w_{ijk} is typically one or zero depending on which comparison is considered for the type of analysis (Grower, 1971).

Grower and Legendre (1969) have characterized these measures by their usage. The established criteria of usage include the nature of the input data, the scale of the data, and the measurement of similarity. However, no clear answer is widely accepted for the best measure to use (Everitt, 2001).

3.3 Classical statistical methods for data organization

This section is divided into classical unsupervised (clustering) and classical supervised (classification) organization methods. Classical methods are based solely on statistical assumptions. Classical methods are distinguished from modern methods which do not require stringent statistical assumptions. Modern methods of organization are drawn from the field of Machine Learning (Bramer, 2007). Modern methods will be presented in section 3.4.

3.3.1 Classical unsupervised (Clustering) methods

Methods are described in chronological order of their earliest appearance in the literature.

3.3.1.1 Principal Component Analysis (PCA)

PCA is one of the earliest implementations for dimensionality reduction (Pearson, 1901; Hotelling, 1933). It is commonly applied to large multivariate data sets, where the dimensions of the data are related. Depending on the field of study, PCA is also referred to as the Karhunen-Loeve transform (KLT), the Hotelling transform, or Proper Orthogonal Decomposition (POD) (Kosambi, 1943). In general, PCA shows the variation within a data set by constructing a set of new uncorrelated variables, which are derived from a linear combination of the original variables. The newly generated variables are calculated in an ordered way, meaning that the first principal component explains the most variation in the original data set. The second principal component then explains the most remaining variance in the data set, subject to being uncorrelated with the first component. The goal in applying PCA is to find out if a reduced number of uncorrelated components can explain the variation given by original variables in the data set (Everitt and Dunn, 2001), eliminating interactions among variables which may confound interpretation of patterns in the data. As with all other methods described throughout this chapter, deciding on the optimal number of clusters, or principal components, can be difficult. The following recommendations for PCA have been developed by Jolliffe (1986):

- A number of principal components should be selected, which explain between 70% and 90% of the total variation of the original variable set.
- Principal components should be discarded whose eigenvalues are smaller than the average eigenvalue, which also describes the variance of the original data set. Therefore eigenvalues smaller than 1.0 should be excluded (Kaiser, 1958).
- The number of components to keep should be determined by plotting eigenvalues against the ordered components in a scree plot. The optimal number of components is located at the "elbow," which is identified where the eigenvalues level off (Cattell, 1966; Jolliffe, 1986).

PCA is used in many fields. PCA analysis forms the basis for many advanced methods of machine learning, which will be discussed later in the chapter (Kohonen, 2001; Hastie et al., 2001). A disadvantage of PCA is that maximizing variance does not always account for maximizing information. While PCA analysis performs well in capturing simple relationships in the data, it may miss complicated relationships. Variants of PCA have been developed, such as Correspondence Analysis (CA) and Multiple Factor Analysis (MFA), which are conceptually similar, and are able to handle qualitative variables (in the case of CA) and heterogeneous sets of variables (in the case of MFA). Factor Analysis is related to, but not equivalent to PCA. In Factor Analysis, the original input variables are defined as a linear combination of all factors; in PCA, the components are derived as linear and non-linear combinations of the factors. Furthermore, Factor Analysis tries to explain the covariance or correlation among the variables, where the goal of PCA is to account for as much of the total variance as possible among variables. PCA is used in this dissertation for dimensionality reduction to visualize the generated groups by k-Means cluster analysis on the first two principal components.

3.3.1.2 Cluster analysis

Classical cluster analysis is described as the process of organizing a set of observations into groups in a way that data observations which fall in the same cluster are more similar to each other than to data observations in another cluster. The definition of a cluster varies between clustering methods. Cormack (1971) and Gordon (1999) define a cluster in terms of internal cohesion (homogeneity) and external isolation (separation). The literature describes a variety of models for clustering. Connectivity models (as in hierarchical clustering) are based on distance measures. Centroid based models, as in k-Means, represent each cluster as a single average point, used for calculating distance to other clusters. Distribution models are based on a statistical distribution, such as a multivariate normal distribution, to calculate distance. Density models define clusters as connected dense regions within a data space. Hierarchical and k-Means clustering are used in this dissertation.

Hierarchical clustering: In hierarchical clustering, a similarity matrix is formed, containing distances computed between each pair of observations. From this, a series of partitions is developed iteratively, based on linkage criteria, which measure pairwise similarity among observations (Figure 3.1). In a divisive approach, which can be described as "top down", an initial single cluster, which spans the whole data set, is incrementally partitioned into smaller individual clusters, which progress to contain only one individual observation. In an agglomerative approach ("bottom up"), each individual observation is first placed into its own cluster. Clusters are merged together as the algorithm moves up the hierarchy (Everitt, 2001). Results of a hierarchical clustering are presented in a dendrogram, which shows the degree of clustering at different levels of similarity. Figure 3.1 shows the three most commonly used linkage distance measures. Single linkage, also referred to as "Minimum Clustering Measures" (MCM) utilizes the shortest distance between groups as the basis for cluster formation. Complete linkage measures cluster distance between the most distant data points. Centroid linkage, also referred to as Unweighted Group Pair Methods Centroid (UPGMC), merges clustered observations with the shortest mean distance. Average linkage, also referred to as Unweighted Paired Group Mean Average (UPGMA), calculates the average of the distance between all data points in each cluster.

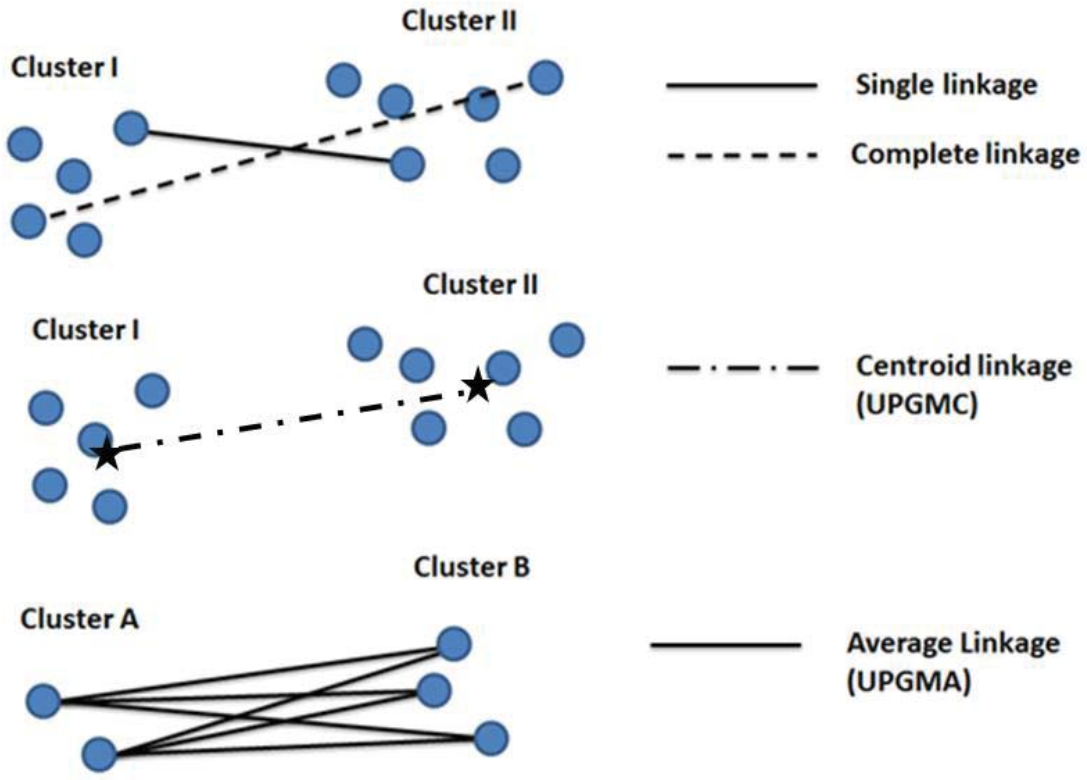


Figure 3.1 The three most commonly used linkage distance measures in hierarchical clustering (Redrawn from Everitt, 2001).

A more advanced hierarchical method uses Ward's (1963) criterion, which is unique because it uses analysis of variance for evaluating the distance between the clusters. This method minimizes the sum of squared distance between any two clusters that are formed at each step, and is therefore, considered very efficient as it tends to form small clusters (Everitt et al., 2001; Ward, 1963). Table 3.1 summarizes the various linkage computations.

Table 3.1 Overview of different linkage criteria in agglomerative hierarchical clustering.

Name	Formula	Reference
Single linkage	$\min\{d(a,b): a \in A, b \in B\}$	Sneath (1957)
Complete linkage	$\max\{d(a,b): a \in A, b \in B\}$	Sorensen (1948)
Average linkage (UPGMA)	$\frac{1}{\lvert A\rvert\lvert B\rvert}\sum_{a\in A}\sum_{b\in B} d(a,b)$	Sokal and Michener (1958)
Centroid linkage	$\frac{1}{n_r}\sum_{i=1}^{n} x_{ri}$	Sokal and Michener (1958)
Median linkage	$\frac{1}{2}(\mathrm{x_p} + \mathrm{x_q})$	Gower (1967)
Ward's method	$\lvert A\rvert\lvert B\rvert \frac{\lVert \bar{a} - \bar{b} \rVert 2^2}{(\lvert A\rvert + \lvert B\rvert)}$	Ward (1963)

A dendrogram is used in hierarchical cluster analysis to visualize formation of clusters at each stage of the clustering process. It uses the similarity matrix as an input. Figure 3.2 shows a dendrogram of a simple hypothetical data set.

The height of a dendrogram, also referred to as distance, describes the similarity between the clustered data objects. From Figure 3.2 it can be seen that a distance (height) of 1.0 would relate to 5 clusters, a distance of 2.0 would generate 3 clusters, and a distance (height) of 3.0 constitute 2 clusters. These units of measure are specifically tied to a data set.

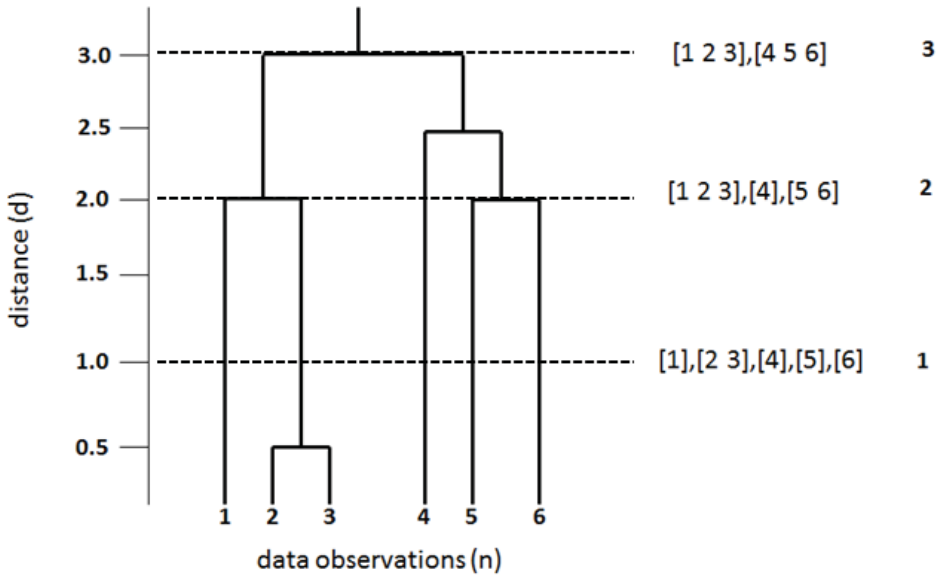


Figure 3.2 Elements of a dendrogram

While hierarchical clustering is frequently used in many statistical applications, there are also some disadvantages of this method. Interpretation of the results is often complex and confusing. All determinations in the clustering process are based on local measurements and are made within one single clustering pass (Augen, 2005).

k-Means Clustering: k-Means remains a popular clustering algorithm, although it was developed decades ago (MacQueen, 1967). The clustering algorithm is easy to implement and performs well on large data sets (Everitt et al., 2001). k-Means clustering is a centroid based clustering method, where each object is assigned only to one cluster (Bramer, 2007). In contrast to hierarchical clustering, the number of clusters (called the k-value) has to be set before the processing takes place. As with hierarchical clustering, a similarity matrix is initially computed. The optimal solution distributes centroids as far apart from each other as possible (Bramer, 2007). Data observations are associated with the nearest centroid. Cluster centroids are recalculated based on mean distance to all observations in the newly generated clusters. Data observations are then assigned to the closest centroid. This process is repeated with cluster

centroids changing locations after each iteration, until the clusters stabilize and the centroids do not move anymore (Figure 3.3). In mathematical terms, K-Means clustering is defined as:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|$$

where $\left\| x_i^{(j)} - c_j \right\|$ describes the distance between data point $x_i^{(j)}$ and the cluster center c_j.

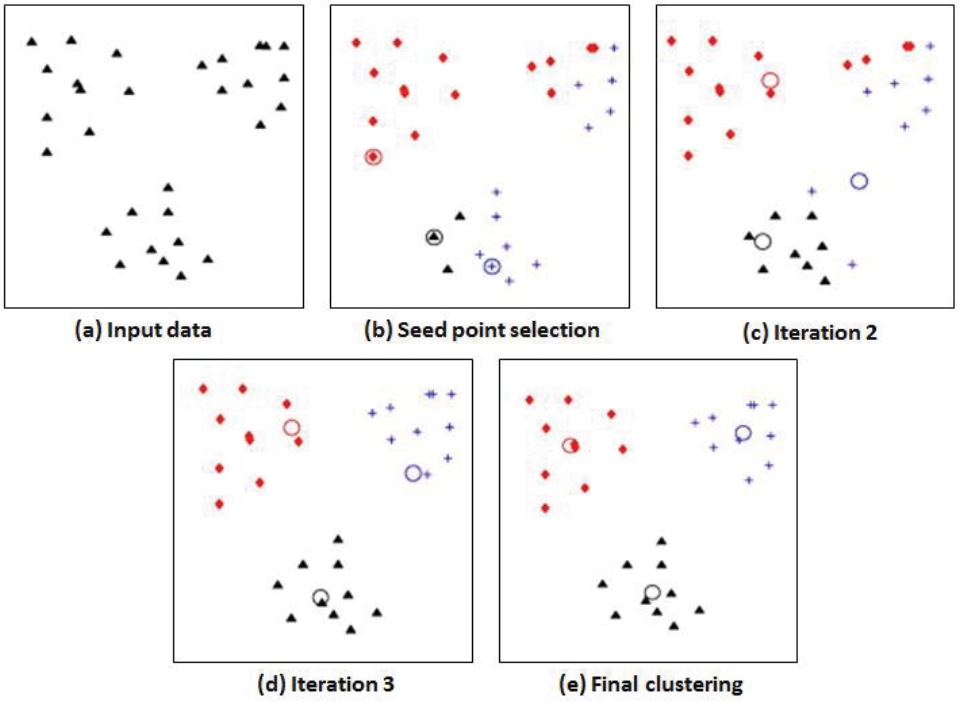


Figure 3.3 The k-Means clustering process for 3 clusters and 3 iterations. (a) Input data; (b) three seed points selected as cluster centers and initial assignment of the data points to clusters; (c) and (d) iterations and updating of cluster labels and their centers; (e) final clustering obtained by k-means algorithm at convergence (redrawn from Jain, 2009).

A disadvantage of k-Means clustering is that the initial placement of the cluster centroid is critical to account for locality and incorrect groupings. Initial placement in the standard k-Means algorithm is done randomly. Therefore, to archive robust clustering results, multiple cluster runs should be performed (Bramer, 2007, Goswami et al., 2007, Everitt et al. 2001). k-Means clustering is often implemented on large data sets (Augen, 2005). k-Means have been

applied in many exploratory data analysis studies, as multiple cluster runs can be accomplished in a short amount of time.

3.3.1.3 Cluster evaluation

As already stated earlier in this chapter, clustering data is about partitioning data points into groups, such that the data points in a cluster are more similar to each other than to points outside the cluster (Guha et. al. 1998). Selecting the optimal number of clusters can be subjective as well as a function of the user's knowledge and expectation of the data set (Everitt et al., 2001). Openshaw et al. (1980: 421) argues that because cluster analysis is an exploratory data analysis technique, "… the results obtained are heavily dependent on the methods used and on a number of arbitrary decisions made during the application of any spatial classification procedure." For this reason, they add "… these decisions must reflect the purpose for which the classification is required." Openshaw (1983: 245) argues that "A classification can only be deemed 'good' or 'poor' when it has been evaluated in terms of the specific purpose for which it is required; there is no magic universal statistical test that can be applied nor is there any possibility of deriving a classification suitable for all purposes." This statement is still considered valid (Jain, 2009).

To overcome the problem of subjectivity, formal measurements and evaluation methods have been suggested. A total of over 30 methods have been developed (Milligan and Cooper, 1985). The number of clusters chosen depends mainly on the clustering method. Studies on cluster evaluation have shown that by applying cluster validation indices it is important not to only rely on one evaluation measure but to take multiple measures as well as the purpose of the clustering into account (McDavid et al., 2011, Rendón et al. 2011, Pacual et al., 2010, Everitt et al., 2001). Beale (1964) developed an exploratory indicator of good clustering that is often used in modern methods, where good clustering is described by well separated clusters or classes (Hardy and Lallemand, 2004; Beale, 1969).

In most clustering processes, the cluster evaluation is a very important step taken at the end of the process (Halkidi et al., 2001). Cluster evaluation tries to find the optimal number of clusters for a given data set by internal and external cluster measurements. A common method is to perform multiple runs with different numbers of classes and then choose the optimal solution based on a quality metric. In general, cluster evaluation methods can be categorized into two classes, internal cluster evaluation and external cluster evaluation.

a) *Internal Cluster evaluation*

Internal cluster evaluation is conducted using internal information from data objects within the dataset. Internal cluster validation measures the fit between the data and the expected clusters and the stability of the cluster solution (Pacual et al., 2010).

Three indices are commonly used to assess the quality of clustering. The Davies-Bouldin index calculates the ratio of the sum of within-cluster distances to between-cluster separation:

$$DB = \frac{1}{n}\sum_{i=1}^{n} \max(i-j)\left\{\frac{S_n(Q_i) + S_n(Q_j)}{S(Q_i, Q_j)}\right\}$$

where n is the number of clusters, S_n is the average distance of all objects to their cluster center, and $S(Q_i, Q_j)$ represents the distance between cluster centers. Small values are associated to the optimal number of clusters (Davies and Bouldin, 1979).

The Dunn (1974) index measures compactness and separation of clusters:

$$Dunn = \min_{l<i<m}\left\{\min_{l<j\leq m\, i=j}\left\{\frac{d(c_i, c_j)}{\max_{l\leq k\leq n}\{d(c_k)\}}\right\}\right\}$$

where $d(c_i, c_j)$ is the distance between clusters measured by intracluster defined by $d(c_k)$ and intercluster defined by c_k. High values account for best clustering.

The Silhouette index (Rousseeuw, 1987) calculates the silhouette width (average within cluster distance) for each data sample, as well as overall silhouette width (minimum average distance of any point to other clusters). The Silhouette index is defined as:

$$S(i) = \frac{(b(i) - a(i))}{\max\{a(i), b(i)\}}$$

where $a(i)$ is described as the average dissimilarity of a data point to other data points in the cluster, and $b(i)$ is described as the minimum average dissimilarity of a data point to all data points in the closest cluster. Higher values account for good clustering. All three cluster validation indices are used for validation of the clustering methods in this dissertation.

b) *External cluster evaluation methods*

External cluster evaluation implies that the evaluation is based on a pre-specified structure, which is imposed on the data set (Rendón et al. 2011). For example, external data, which is not included in the data set is used is used to evaluate the clusters and test the validity of the cluster solution. External cluster evaluation can therefore only be applied if other similar data sets are present and in many cases cannot not be applied for large unlabeled or unique self-complied data sets (McDavid et al. 2011).The literature states the F-Measure, MMI (Normalized Mutual Information) measure and purity index as the most commonly used external validation methods (McDavid et al., 2011, Rendón et al. 2011).

The F-measure, in some cases also referred to as F-1 measure, calculates the harmonic mean of precision and recall which are also applied in supervised methods and discussed into more detail later in this chapter. Precision and Recall is described as:

$$Precision(i,j) = \frac{n_{ij}}{n_j} \qquad Recall(i,j) = \frac{n_{ij}}{n_i}$$

where n_{ij} is the number of object of class i that are in cluster$j$, n_j describes the number of objects in cluster$j$, and n_i, is the number of objects in class i . F measure is then calculated by

$$F(i,j) = \frac{2(Recall(i,j)Precision\ (i,j))}{Precision\ (i,j) + Recall(i,j)}$$

The results from the F Measure range from 0 to 1 with larger values indicated better clustering.

The NMI measures overlapping cluster membership information from the original data set with the external validation data set. MMI is calculated by:

$$NMI(X,Y) = \frac{I(X,Y)}{\sqrt{H(X)H(Y)}}$$

where. $I(X,Y)$ describes the overlapping cluster membership between two random varies X and Y. $H(X)$ denotes the entropy of X where X is the original clustering and Y the clustering derived from the external data set (McDavid et al., 2011, Meila, 2007).

The Purity index calculates the accuracy of a clustering as a fraction of the overall cluster sizes to which the largest class of data objects from an external data set is assigned to (Rendón et al. 2011). The overall purity of a cluster solution is obtained as a weighted sum and can be described as:

$$Purity = \sum_{j=1}^{m} \frac{n_j}{n} P_j$$

where n_j describes the size of cluster j and m refers to the number of cluster. The total number of objects in the data set is denoted as n.

3.3.2 Classical Supervised Methods (Classification)

In supervised classification, a training subset of the data is necessary to guide data organization. As in the previous section, methods are described in chronological order of their first appearance in the literature.

3.3.2.1 k-Nearest Neighbor classification

The objective of k-Nearest Neighbor (k-NN) is to classify objects based on the closest (most similar) training examples in attribute space. The k in k-NN refers to the number of items which will be averaged to compute the final results (Segaran, 2007). While the k value in k-Means describes the number of clusters, the k value in k-NN dictates the neighborhood size in the classification process. The standard k-NN algorithm relies on Euclidian distance as a measure to calculate how close each observation is to the ready-to-be-classified data point (Bramer, 2007). In k-NN the distance is calculated by:

$$D_i = \|x - X_i\|$$

where x is described as the ready-to-be-classified data point and X_i defines each observation surrounding this data point. The observations surrounding the ready-to-be-classified data point are referred to as "nearest neighbors" and are ranked by distance$\{X_{(1)}, X_{(2)}, X_{(3)}, \ldots X_{(n)}\}$. The k[th] nearest neighbor of x is therefore $X_{i(k)}$. Figure 3.4 shows the classification process in a 2-dimensional feature space on a hypothetical data set.

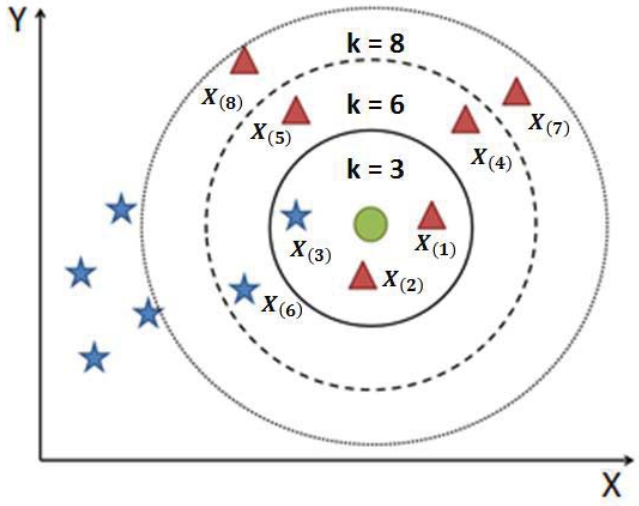


Figure 3.4 k-NN classification of hypothetical data in a 2-dimensional feature space.

The solid line indicates a k value of 3, the dashed circle line indicates a k value of 6, and the fine dotted circle line shows a k value of 8. The ready-to-be-classified observation (indicated in green) should be classified either with the blue stars or the red triangles. For a k value of 3, 6, and 8 the test observation is assigned to the group of red triangles as they are in the majority in every neighborhood. In its simplest form, the k-Nearest Neighbor algorithm can be defined as:

$$R_x = \|X_k - x\| = D_i$$

where R_x is the k[th] order statistic on the Euclidean distance D_i. Before applying a k-Nearest Neighbor's method, it is therefore essential that the elements of X are scaled, so that they are comparable across elements.

The elementary step in setting up k-NN is to select the optimal choice of k. Overall, larger values of k reduce noise (smoother class boundaries) in the data, but also tend to overgeneralize the classification results (Bramer, 2007). Heuristic methods, such as k-fold cross-validation, can be used to calculate optimal values for k, when the model parameters are unknown (Hastie et al., 2007). The general idea of cross validation is to divide the data sample into a number of *k* folds. Folds can be described as randomly drawn, disjointed sub-samples of the whole data set. This random process is successively applied to all possible choices of *k*. At the end of the k folds (iterations), measures of the stability of the model and how well the model classifies data points are given. The above steps are then repeated for various *k* and the value achieving the lowest error or the highest classification accuracy is then selected as the optimal value for k (Witten et al., 2005; Bramer, 2007).

The k-NN algorithm is well suited for continuous data, when Euclidian distance is used to assess similarity. However, there are special cases in which textual data can be used and for these cases, Hamming distance is applied (Hastie et al., 2007). Hamming distance measures the difference between two strings by calculating a metric on the vector space of the word's length (Hamming, 1950).

3.3.2.2 Classification trees

Classification trees classify the value or probable category of a target variable based on multiple input variables. Classification trees are one type of decision trees, which predict a discrete category. The other type of decision trees are regression trees, which predict a numerical value based on probability. In the scope of this dissertation, only classification trees are considered.

Classification trees start at the top most node (the tree root) and test for each observation a set of binary characteristics, which contain information about class membership (Figurer 3.5). This process is repeated until a leaf is reached, which defines the class to which the observation

is assigned. If a classification tree becomes too large for interpretation (which is an indication of overfitting), k-fold cross-validation is used to prune the tree (Bramer, 2007). Figure 3.5 shows an example of a simple decision tree using hypothetical data. At each node in the tree, binary characteristics are tested in order to follow the tree from root to leaves, where a class assignment is predicted. This predication aggregates all the training data points which were followed to reach this leaf (Ripley, 1996). Classification trees are not limited to binary characteristics, and can be applied to continuous, discrete, or categorical data. In this dissertation classification trees are used on continuous and categorical data.

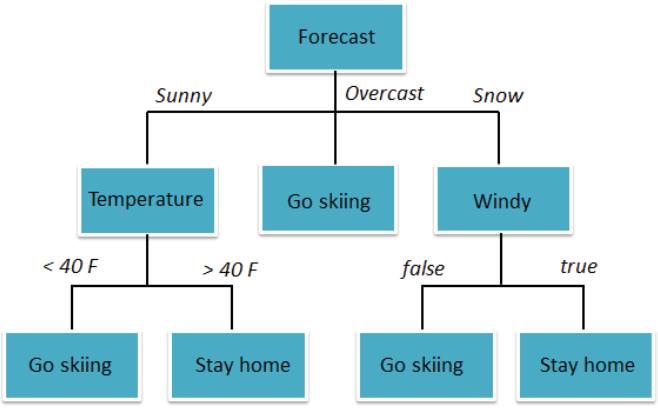


Figure 3.5 Example of a simple decision tree using hypothetical data.

Random Forest classification is a special case of classification trees which will be used in this dissertation experiment. A Random Forest consists of an arbitrary number of classification trees, which are used to determine a set of rules to establish an optimal classification tree which will be used for final classification. Classification is generated on random subsets of the data, using a subset of randomly restricted and selected predictors for each split in each classification tree (Strobl et al., 2008). Due to the creation of many random classification trees, Random Forest classification is able to better assess the classification of each predictor (Strobl et al., 2008). The results of Random Forest have been shown to produce better classification than the results of one classification tree on its own as an optimal classification is established through the generation of many trees (Strobl et al., 2008).

Three different methods to evaluate classification trees measure misclassification rate, average loss, and entropy (Ripley, 1996). Misclassification describes the fraction of cases assigned to the wrong class. Average loss describes the phenomena that some errors are more costly than others. For example, weights can be assigned to classes so that some errors are more strongly weighted than others. Entropy analyzes the conditional probability that misclassification occurred. Entropy is defined as:

$$E = -\sum_{i=1}^{K} p_i \, log_2 \, p_i)$$

where p_i describes the number of occurrences of class i divided by the total number of occurrences, and K denotes all possible classification cases.

3.3.2.3 Evaluation methods in supervised classification

As with unsupervised clustering, three evaluation methods are also applied to supervised classification methods in this dissertation.

In the first method, true and false positives and negatives are based on a confusion matrix. When using true and false positives for validation, two types of classification errors are distinguished: Type 1 and Type 2 error. A Type 1 error is generated when the null hypothesis is true, but is rejected which can also be described as a "false hit." A Type 2 error occurs when an alternative hypothesis (failure to reject the null hypothesis) is rejected when the alternative hypothesis is true.

In the second method, precision and recall are defined in terms of a set of retrieved documents and a set of relevant documents (Baeza-Yates and Ribeiro-Neto, 1999). Precision is a measure of the ability of a system to retrieve only relevant items. Precision can be defined as:

$$precision = \frac{number\ of\ relevant\ items\ retrieved}{total\ number\ of\ items\ retrieved}$$

In classification, a precision score of 1.0 is interpreted as every data item in a defined class has been classified to the correct class. Recall is a measure of the ability of a system to retrieve all relevant items. Recall can be defined as:

$$recall = \frac{number\ of\ relevant\ items\ retrieved}{number\ of\ relevant\ items\ in\ the\ collection}$$

A recall score of 1.0 is interpreted as every data item from a defined class has been correctly assigned, but it does not say anything about how many other data items were incorrectly labeled. In most classifications, application of both measurements is combined into one measure based on the mean from both scores, called an F-measure (Powers, 2011).

The third method is cross-validation, which estimates how well a classification model performs on an untrained data set. The main purpose of cross-validation is the estimation of how a predictive model will perform in practice. There are many different types of cross-validation, such as k-fold, 2-fold, repeated random sub-sampling, and leave-one-out. K-fold cross-validation will be used in this dissertation.

In k-fold cross-validation, the original data set is randomly partitioned into two subsamples. One subsample is set aside for validation of the model and the other subsample is used to train the classification model. Summary statistics are calculated to describe the performance of the classification. The split sampling process is repeated many times, constraining that each observation is used only once as validation data. The cross-validation averages the prediction errors associated with each repetition. The advantage of this method is that all observations are used for both training and validation, and each observation is used for validation exactly once (Geisser, 1993).

3.4 Modern methods taken from machine learning

Methods drawn from the field of machine learning have been developed to overcome limitations from classical statistical methods, mainly the ability to scale up and work with very large multidimensional data sets. Additionally, modern methods do not require strict adherence

to the assumptions which classical methods require. As with classical methods, there are unsupervised and supervised machine learning algorithms.

Unsupervised machine learning algorithms cluster a set of input observations on the basis of similarities in one or more variables. While a training phase may take place, there is no predetermination of correct or incorrect characterization. Similar to classical statistical clustering methods, modern methods do not presuppose a specific number of categories or groups. Rather, they are used to regionalize an attribute space, and often are utilized as a data reduction technique. Two kinds of unsupervised machine learning are reinforcement learning, wherein positive or negative feedback on prior decisions guides current decisions, and clustering, which operates to discover similarities in the data. The assumption is that given sufficient input data, discovered clusters will make intuitive sense (Taylor and Joudrey, 2009).

With very large or complex data sets, a common problem in unsupervised learning is over-fitting the categorization to the training set, rather than learning directly from input data characterizations, and generating clusters accordingly (Hinton and Sejnowski, 1999). The example of Self-Organizing Maps (Kohonen, 1995) will be applied in this dissertation as an example of modern unsupervised learning.

In the field of Machine Learning, as in classical statistics, supervised learning is referred to as classification (Bramer, 2007). Supervised methods place individual data items into groups based on measurements or characteristics of a provided training set in which training instances are attributed as being correct or incorrect. Linear classifiers group items based upon linear combinations of the characteristics (Bramer, 2007). Performance of linear classifiers tends to be very fast, especially when the vector of characteristics is sparse, that is, when values for many characteristics are zero-valued or absent (Hastie et al., 2001). Parameters for a linear classifier can be determined by generative or discriminative models. A generative model specifies a full set of joint probabilities on all variables and is commonly used to simulate (generate) values for any included variable, assuming a specific underlying probability distribution. A discriminative

model works on conditional probabilities to select an unobserved variable, which is conditional on one or more observed variables. Discriminative models offer a good choice when data relationships are simple (Hinton and Sejnowski, 1999). Generative models are generally considered more flexible, but require knowledge about the underlying probability distribution (Segaran, 2007). Support Vector Machines will provide the example of a discriminative model in this dissertation research.

3.4.1 Unsupervised Machine Learning: Self-Organizing Maps (SOM)

Self-Organizing Maps (SOMs) provide a special case of Neural Networks, which are also referred to as Artificial Neural Networks (ANNs). ANNs are based on the idea of biological neural networks (McCulloch and Pitts, 1943) and consist of a mesh or lattice of individual and grouped neurons, in one or multiple dimensions. Data organization proceeds by assigning data points to neurons, which acquire characteristics similar to assigned data points. Neurons, which contain similar characteristics, are linked together. As assignment continues, neuron characteristics change to reflect all of their assigned data points as closely as possible. As neuron characteristics change, changing similarities introduce modifications to the mesh. Organization or clustering is complete when the set of linkages stabilize.

Training of an ANN is considered an elementary step in unsupervised machine learning. Many different variants exist for training, although as stated above, the training step offers no correct or incorrect examples, but rather is undertaken to explore the input data characteristics fully, to initialize a regionalization of the feature space (Hinton and Sejnowski, 1999; Hastie et al., 2001).

SOMs are a type of ANN that use of a neighborhood function to preserve the topology of the input space. SOMs are beneficial for data searching and exploration because they organize data items to clarify relationships and because they can be used for dimension reduction (Agarwal and Skupin, 2007). SOMs differ from other types of ANNs, in that they organize data

by competing for assignment of observations. In the SOM, neurons are called cells or units. Data points are assigned to the Best Matching Unit (BMU) whose characteristics are most similar to the data point characteristics. Similarity is based on Euclidian distance. As a data point is assigned, every BMU has its weight vector (its set of characteristics) adjusted according to the following equation:

$$W(t+1) = W(t) + L(t)\big(V(t) - W(t)\big)$$

where t represents the iteration step, W(t) represents the weight vector, and L represents a fraction of the difference between the old weight vector and the newly assigned data point's weight vector (V). The variable L is called the Learning Rate and controls the amount of adjustment in any unit as each new data point is introduced. L can be user-specified at the outset of classification, and decreases over iterations:

$$L(t) = L_0 \exp\left(-\frac{t}{\tau}\right)$$

where τ represents the rate at which t decreases at each iteration.

Not only do cell characteristics adjust when data points are assigned to them, but cells in the neighborhood of an assigned cell also adjust characteristics to approach the characteristics of the assigned cell (Kohonen, 2001). The size of the neighborhood, within which cell characteristics are adjusted, decreases across iterations:

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\tau}\right)$$

where $\sigma_0$represents the width at each iteration of the SOM mesh at the outset of classification, and $\sigma(t)$ represents the neighborhood size at iteration t.

Another difference between a SOM and other types of ANNs is that the cells are initially ordered in a rectangular or hexagonal mesh. Links between cells are preserved topologically, which constrains their movements. As the mesh organizes, it becomes distorted in a process called unfolding (Figure 3.6).

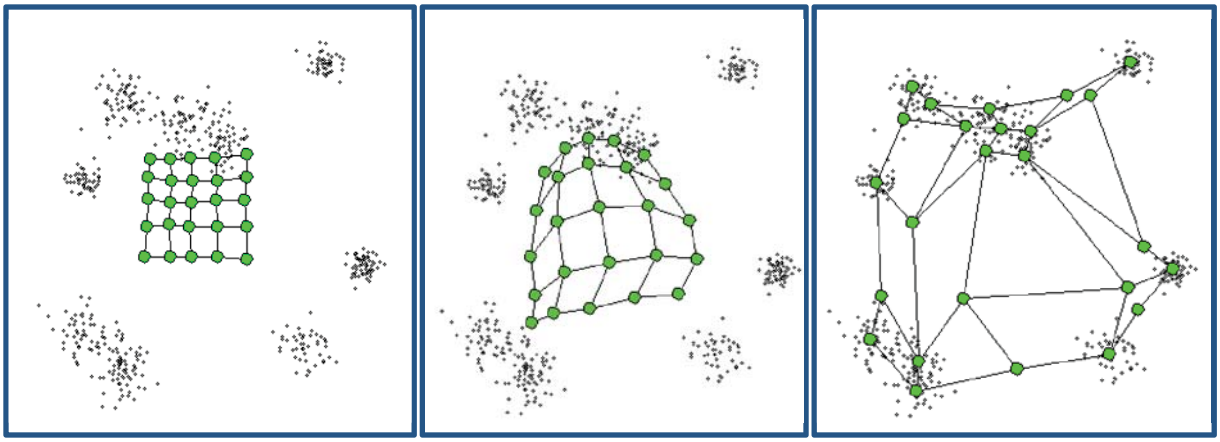


Figure 3.6 The SOM attribute space at initial (left), intermediate (middle), and final (right) iterations in the unfolding process. Black dots represent input data points and green dots represent SOM cells. Taken from http://blog.peltarion.com/2007/04/10/the-self-organized-gene-part-1/.

For visualization purposes, the mesh of cells is mapped back into its regularized pattern (Figure 3.7). The regularized mesh is called a component plane or mapplet.

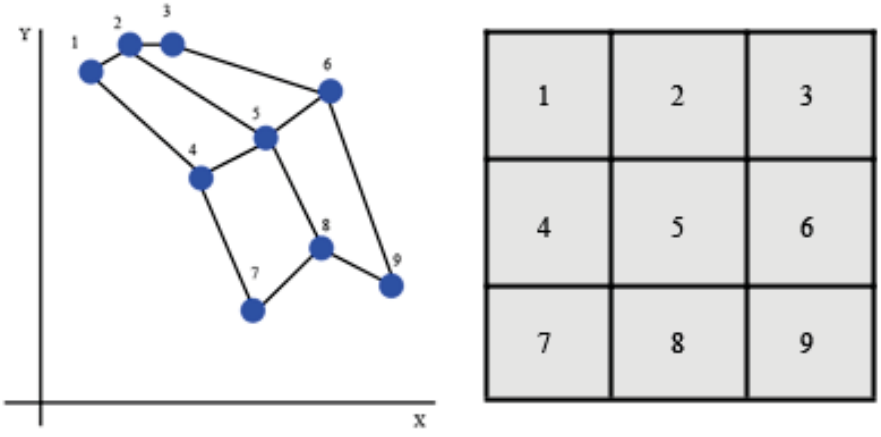


Figure 3.7 Example of how the units are mapped onto a SOM rectangular mesh.

The feature-space that the SOM above occupies has two dimensions: X and Y. A mapplet can be created showing the values from the x and y axis for each dimension individually. Mapplets for these two variables are shown in Figure 3.8, color coding their respective values.

In Figure 3.8, cells 1, 2, and 3 have high Y values (blue) and low X values (white). Likewise, cells 4 and 5 have roughly equivalent X and Y values, and therefore display the similar shades in corresponding cells.

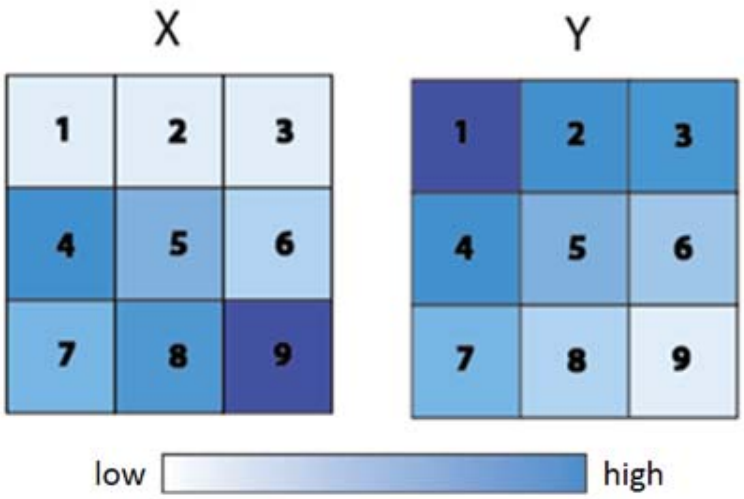


Figure 3.8 This figure shows a SOM mapplet representing each dimension (X, Y). Legends along the bottom of the two mapplets show the color ramp for low-to-high values. The example is modified from http://blog.peltarion.com/2007/04/10/the-self-organized-gene-part-1/.

The typical SOM output also includes a Unified Distance Matrix or U-Matrix, which indicates the distance in unfolded space between adjacent SOM cells (Ultsch and Siemon, 1990). The U-Matrix holds twice the number of cells as the SOM, and each cell contains similarity measurements between each node and its neighboring nodes (Figure 3.9). The U-matrix in effect illustrates actual distances between SOM cells in feature space, with higher values indicating more dissimilarity.

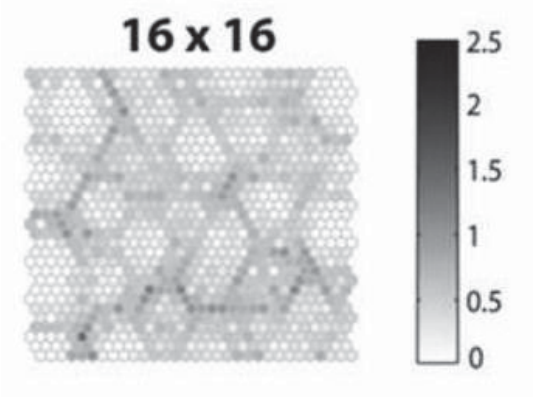


Figure 3.9 Example of a U-Matrix for a SOM containing 16x16 cells. Darker values indicate greater distances (more dissimilarity) between characteristics of neighboring cells. Lighter values indicate neighboring cells whose characteristics are similar.

The quality of the SOM can be evaluated by two measures. The Quantization Error ε_q indicates how well the SOM units match the assigned data points on average, and is defined as:

$$\varepsilon_q = \frac{1}{n}\sum_{i=1}^{n}\left\|x_i - m_{c(x_i)}\right\|$$

where n indicates the number of observations, $x_i$indicates the weight vector (the set of all characteristics) for each input data point, and $m_{c(x_i)}$ indicates the final weight vector for that point's BMU (Kohonen, 1995; Honkela, 1999). Low values for the Quantization Error are best, although values very close to 0.0 indicate model over-fitting (Kohonen, 2001).

The Topographic Error ε_t measures the proportion of the data points whose second-best BMU is adjacent to the BMU to which they are assigned. It is a measure of the coherence of the SOM output, and is defined as:

$$\varepsilon_t = \frac{1}{N}\sum_{k=1}^{N} u(x_k) = \{1, 0\}$$

where $u(x_k)$ equals 1.0 when all best- and second- best matching units are non-adjacent and 0.0 when they are adjacent (Kiviluoto, 1998); as with Quantization Error, the goal is to establish values close to 0 (Kohonen, 2001).

SOMs have become widely used for multivariate data analysis where collapsing of a high dimensional input space to a lower dimensional output space is required. Kaski et al. (1998) provide a bibliography of over three thousand SOM papers published between 1981 and 1997. They state that SOMs have been applied across many fields, and many extensions to the standard SOM algorithm have been developed. In the last decade, SOMs have gained popularity among GIScience researchers who apply spatial metaphors to the SOM output (Skupin, 2002; Douglas, 2004). Bação et al. (2005) developed the Geo-SOM algorithm extension, where geographic location is weighted within the learning process.

3.4.2 Supervised Machine Learning: Support Vector Machines

Support Vector Machines (SVMs) were originally developed by Vapnik and Lerner (1963), and the method has been subsequently refined (Cortes and Vapnik, 1995). SVMs have been applied to many different fields ranging from text recognition to geoscience and climate applications (Bennett and Campbell, 2000; Bo et al., 2009; Mountrakis et al., 2010). Standard SVMs accept a set of input data and predict, for each given input, which of two possible classes to assign membership. For this reason, SVMs are referred to as non-probabilistic binary linear classifiers. SVMs start with a set of training data which characterizes a sample of data points as belonging to one or the other class. SVMs construct a representation of the training data points in attribute space, positioning them in a way that the two classes are as distant as possible. Once training is complete, a set of data points (not used for training) are examined and assigned to one of the two classes. SVMs are based on the concept of decision planes, which separate objects belonging to different classes, such that the distance between the decision plane and any training data point in any class is as large as possible. This distance is called Functional Margin; maximizing the Functional Margin reduces the possibility of classifier error. Figure 3.10 shows that when separating a data set into two classes using two variables, multiple solutions exist. SVMs can operate in higher dimensional spaces as well, in which case the decision planes are referred to as hyperplanes.

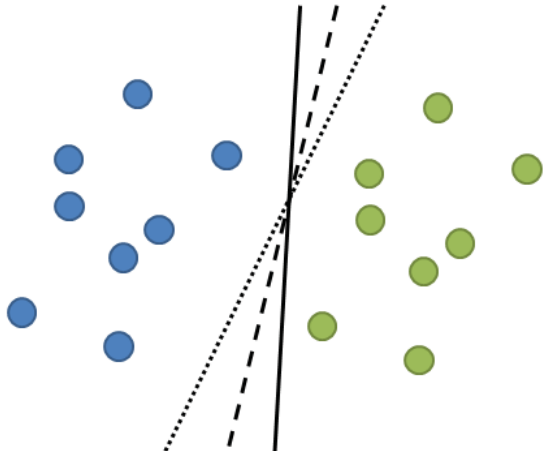


Figure 3.10 Linear plane separation shown in a 2-dimensional feature space

SVMs find the optimal separation by maximizing the margin around the separating (hyper)plane using support vectors (Figure 3.11). The decision function for finding support vectors is fully specified by the training sample (Hastie et al., 2001).

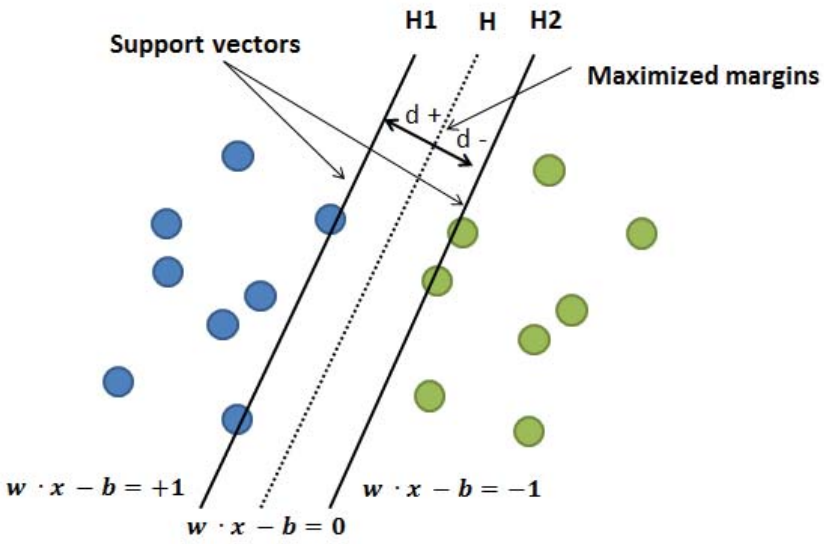


Figure 3.11 Delineation of support vectors and margin for SVM using linear separation between two classes of data points.

Support vectors can be defined as the elements of the training set that would change the position of the dividing hyperplane when removed. A hyperplane can be defined as:

$$W \cdot X - b = 0$$

where $\cdot$ describes the dot product, W represents the normal vector to a (hyper)plane, and X represents a set of points in the data set (Bramer, 2007). Geometrically, the dot product is essentially a buffer surrounding the functional margin in as many dimensions as is needed for characterizing the data set. Support vectors are critical products of the training set. Finding the optimal (hyper) plane dividing two classes in multiple dimensions can be solved by using optimization techniques, such as Lagrange multipliers (Hastie et al., 2001). The algorithm generates the weights in such a way that only the support vectors determine the weights and also define the boundary.

The goal in SVM is to get a classifier with the largest possible Functional Margin. In order to maximize the Margin it is necessary to minimize $||w||$ with the assumption that there are no data points between H1 and H2. SVM uses a quadratic programming paradigm to find the optimal solution. Quadratic programming can be described as a variant of linear programming, in which the objective function is quadratic rather than linear (Bazaraa et al., 2005). Quadratic programming is applied to problems for which linear separation fails and more advanced non-linear classifications are called for. If the input data is not linearly distinguishable, the attribute space can be extended to higher dimensional space by kernel functions using quadratic programming. Various kernel functions are used for transformation; the most commonly used are polynomial, Gaussian or also referred to as Radial Basis Function (RBF), and hyperbolic tangent kernels. Figure 3.12 shows an example of transformed data points.

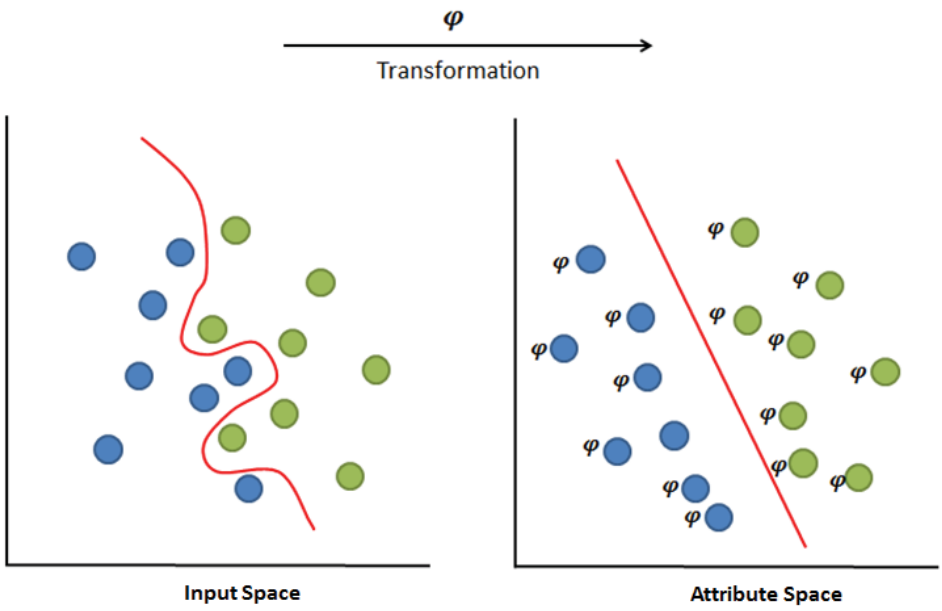


Figure 3.12 Transformation of data points from the original input space to a feature space where data is linearly separable. The phi (φ) term in the figure describes the transformation of data points into higher dimensional space.

φ shows the input features mapped into a linearly separable feature space, after the transformation is applied to the data set. Through kernel function in SVM, an infinite number of dimensions can be processed. For better computational efficiency and to avoid overfitting multiple kernel function can be combined in SVM classification (Dioş et al., 2007)

3.6. Summary and current trends in data organization

This chapter focuses on the methods used in this dissertation specifically for data organization, namely unsupervised clustering and supervised classification methods taken from classical statistics and modern machine learning. From the previous chapter, one can see that indexing clearly plays an important role in supervised and unsupervised data organization, since the selection of keywords will guide the groups which result from organization. Almost every grouping algorithm depends on the characteristics of the data set and on the input parameters used in the clustering process. In order to determine the input parameters that lead

to meaningful clusters reliable guidelines are needed for evaluation and cluster algorithm selection. As described in this chapter, multiple evaluation indices are therefore needed.

In the next chapters, following the indexing of the four different data sets, methods from unsupervised and supervised learning will be used to organize them. Classical and Machine Learning results will be compared to further evaluate automatic and manual derived keywords. The goal of this dissertation is not to extend data organization methods, but rather use existing methods and evaluate them comparatively based on the data sets underlying structure as well as the grouping algorithm used. Most current research comparing different organization methods focuses on one type of data, and analyzes either classification or clustering methods alone. This dissertation extends comparative studies on organization methods (e.g., Mingoti and Lima, 2006; Budayan et al., 2009) by comparing different data sets as well as different organization methods. Such analysis will extend current knowledge by having complete measurements of a broad range of commonly used data sets, which in turn could help other researchers choose optimal methods for the data types they use.

Table 3.2 summarizes the data organization methods used in this dissertation. All methods will be applied to four data sets. Methods are evaluated by common indices for comparison among unsupervised and supervised data organization.

Table 3.2 Overview of methods used in this dissertation

Overview of Methods		
Unsupervised classical		**Validation**
	Hierarchical clustering	Davies-Bouldin, Dunn index, Silhouette index, and Homogeneity and Separation index
	k-Means clustering	
Supervised classical		
	k - Nearest Neighbor	Misclassification and class accuracy
	Classification Trees	
Unsupervised modern		
	Self-Organizing Maps	Quantization and topographic error
Supervised modern		
	Support Vector Machines	Misclassification and class accuracy

CHAPTER IV

Automatic and manual indexing experiments

This chapter outlines the conceptual framework for successful characterization of a variety of data types using automatic and manual strategies for deriving keywords. This chapter also discusses the generation of keywords for indexing data types when automatically generated keywords are not available. Throughout this chapter the term "indexing" refers to the characterization of a data set as well as to the generation of keywords.

This research experiment does not intend to establish new indexing strategies, but rather implement indexing strategies for each of the four data sets used in this dissertation. The data sets were compiled following the continuum of indexability presented in Chapter 1 (Table 1.1). The four data sets include a full-text document, a spatial data set incorporating seven attributes, a list of GIS software commands, and a catalog of generalization algorithms.

The original contribution of this dissertation lies in the evaluation of clustering and classification methods used on the indexed data sets, which will be covered in Chapter 5. Understanding the role served by automatically and manually generated keywords in classical and modern methods for clustering and classification can lead to recommendations for indexing and organization of data types used commonly in geographic analysis and data dissemination. However, indexing is a necessary first step in applying clustering and classification methods to the four data sets.

4.1 Research tasks

Discussion of this experiment spans this chapter and the next, and addresses the following tasks:

- Implementation of automatic or manual indexing strategies on the four different data sets (Chapter 4)
- Organization of indexed data sets by supervised and unsupervised methods (Chapter 5)

- Evaluation and validation of the effectiveness of the grouping results (Chapter 5)

By completing the tasks stated above, this experiment helps answer the research questions asked in this dissertation:

1. *For a given indexing scheme, does a particular organization method link clearly to an indexing method and why?*
2. *What systematic recommendations can be established for organizing data by unsupervised or supervised methods?*

The implementation of different indexing strategies for the four data sets, as well as the systematic data organization by supervised and unsupervised methods, will reveal stabilities and instabilities in the data groups. For example, if nearly identical groups emerge within a particular data type, regardless of the classification method applied, one can conclude that the particular indexing strategy is robust with respect to internal structure of that data set. If groups form which are dissimilar, one must acknowledge that the indexing scheme is sensitive with respect to one or more types of grouping. The systematic comparison of organization methods can help in formulating recommendations for indexing strategies for different types of data, which are needed in today's highly diverse information environments. By showing the benefits and limitations of these indexing methods, it may be possible to propose revised methods for keyword generation and indexing. Formulating guidelines for indexing and organizing different kinds of data may also help to build a foundation for discussing metadata formats for complex data types.

4.2 Research task framework

The dissertation research consists of three major tasks including data characterization (indexing), data organization (supervised and unsupervised learning) and method evaluation (by various metrics dependent on the organization method) as laid out in Figure 4.1.

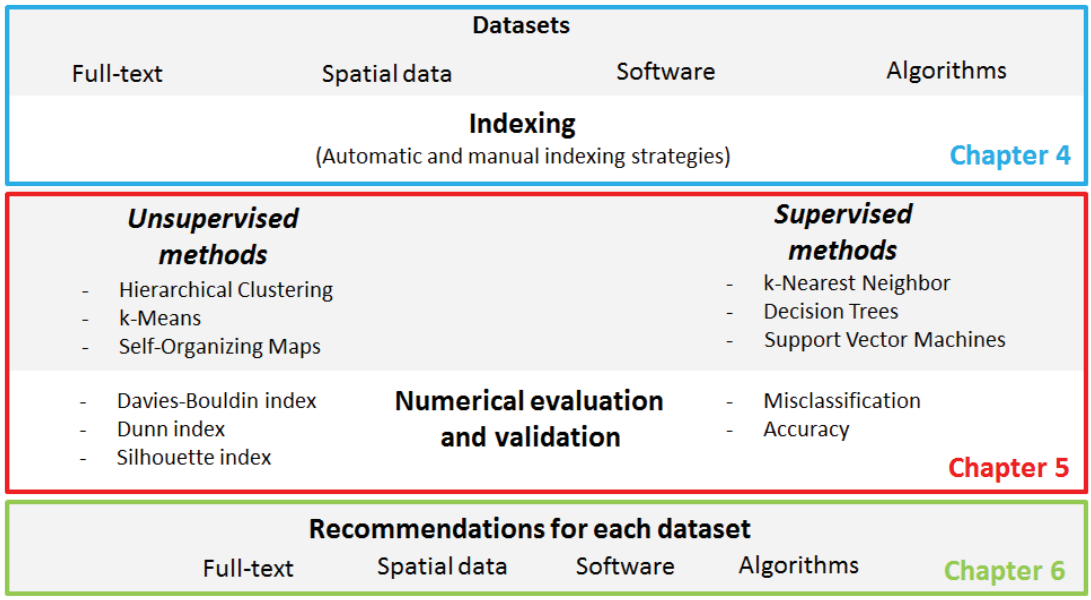


Figure 4.1 Methodological framework for indexing and grouping the four data sets, and evaluating the organization methods. Chapter 4 covers the indexing of the data sets, while Chapter 5 discusses the organization of the indexed data sets as well the evaluation of the methods. Chapter 6 will make recommendations about indexing and organizing the four types of data.

The first task, involves characterization, to build an indexing schema for each of the four data types. An automatic indexing schema is applied if it is both feasible and meaningful. A manual indexing schema will be created if necessary. The assumption throughout this experiment is that an automatic schema is preferable, since it can be derived consistently and is based on actual data content. It is expected that for some data types both indexing types could be generated, but (following the assumption) an automatic method will be utilized. For other data types, only a manual indexing strategy can be applied reasonably. It is important to keep in mind that these experiments will not develop novel indexing strategies, but rather use existing strategies.

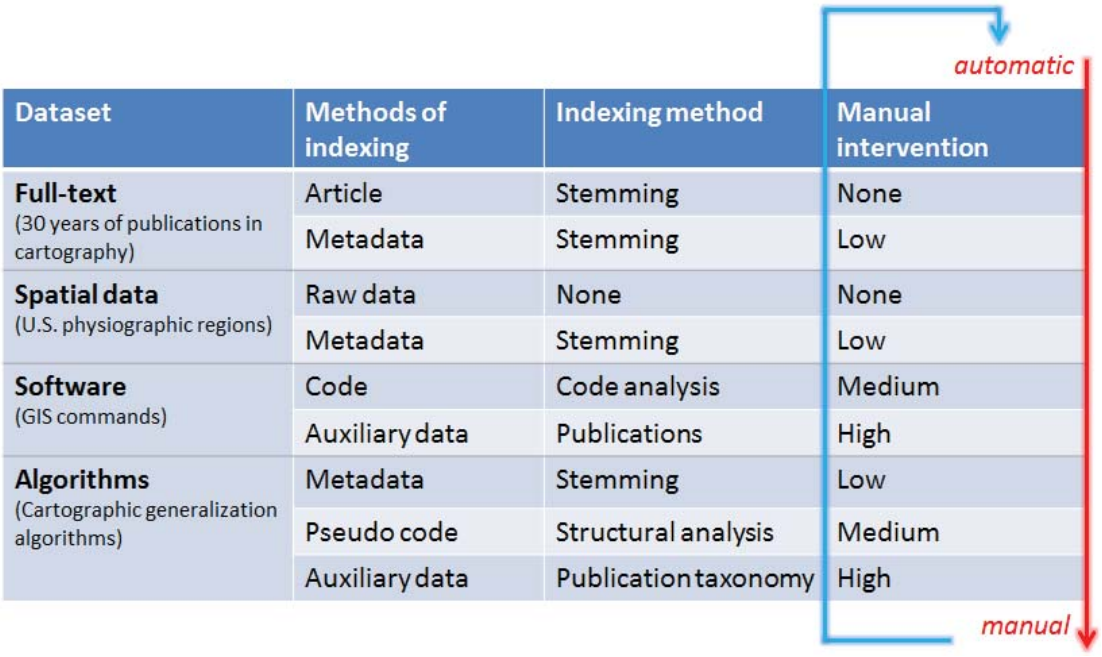
The second task, described in Chapter 5, consists of organizing the four data sets. All six classical and modern approaches will be applied to each data set. The choice of parameter sets

will take into consideration that the results generated by these methods are expected to be comparable. The third task, also covered in Chapter 5, consists of evaluation and validation of the indexing methods. Unsupervised methods are evaluated using cluster separation and cluster compactness measurements, such as the Davies-Bouldin index, the Silhouette index, and the Dunn index. Supervised methods are evaluated using misclassification rates and cross validation methods tailored to the grouping methods.

4.3 Data sets used as exemplar data types

The data sets were chosen based on their relevance and common usage in cartographic analysis, and second, as exemplars to span the indexability continuum. Table 4.1 highlights the specific data sets used in this experiment. The data sets range in size from roughly 108 to 1900 data items, with the exception of the spatial data set, which contains seven co-registered grids each with roughly 500,000 pixels, and the full-text data set, which consists of roughly 4 million words.

Table 4.1 The data sets used in this experiment span the continuum of indexability that was introduced in Chapter 1.



Dataset	Methods of indexing	Indexing method	Manual intervention
Full-text (30 years of publications in cartography)	Article	Stemming	None
	Metadata	Stemming	Low
Spatial data (U.S. physiographic regions)	Raw data	None	None
	Metadata	Stemming	Low
Software (GIS commands)	Code	Code analysis	Medium
	Auxiliary data	Publications	High
Algorithms (Cartographic generalization algorithms)	Metadata	Stemming	Low
	Pseudo code	Structural analysis	Medium
	Auxiliary data	Publication taxonomy	High

As can be observed from Table 4.1, there are alternative ways to establish keywords for each data set. Starting at the top of the continuum, indices for full-text documents can be established by metadata or directly on the raw text data. An example of metadata for full-text documents is metadata tags stored in a bibliographic system, such as keywords describing the article, publication date, authors or editors, and publication series. Indices for full text documents can be derived by stemming and text processing from the field of Natural Language Processing (NLP), which creates keywords directly from the raw text.

Moving on to the spatial data set, no human intervention is required when the raw spatial data set is used. When metadata is used for keyword generation some human intervention is required in selecting and pre-processing of relevant information form the metadata.

Moving further down the continuum, indexing of software requires human intervention, depending on the software, as only sparse auxiliary information exists. Automatic indexing or stemming methods cannot be applied meaningfully to software tools (Wendel et al., 2009; Viger, 2011). Indices can be derived from tool help functions, where keywords can be drawn directly from the content and no or little human intervention is necessary. When no auxiliary data exists, indices can be drawn by structural code analysis or by analysis of input and output parameters of the tool (Tangsripairoj and Samadzdeh, 2006).

Moving to the last row in the table, the algorithm data set, at the end of the continuum, closes the loop of indexability presented in Table 4.1. Algorithms are usually described by full text documents. Modified stemming and query techniques can be applied that only capture the portion of full text documents specifically describing algorithms. Essentially, full-text documents form the metadata of algorithms, and so keywords can be drawn directly from the contents. When no description of the algorithm exists, auxiliary data is needed for indexing. Indices can be derived from additional resources which requires manual indexing or by structural pseudo code analysis, where keywords can be drawn directly from the content with a medium amount of human intervention

4.4 Indexing experiment

The data sets are presented in the order described above, ranging from fully automatic indexing strategies to manual indexing strategies. Each data set is indexed in one unique way, and when possible, without human intervention. Alternative indexing strategies are briefly discussed where applicable.

4.4.1 Full-text articles

The first type of data is exemplified by a full-text document set represented by 30 years of cartographic literature found on the ISI Web of Knowledge (www.webofknowledge.com). The data set consists of 1,432 full text articles. This data set acts as an exemplar for an automatic indexing schema with little human intervention. Approximately 70% of the full-text articles found on the ISI Web of Knowledge are provided with an attached abstract; only the abstract will be used for keyword generation. However, most journal article entries earlier than 1995 are missing an abstract in the ISI database. For this experiment, only the journal articles with an attached abstract are used. The reasoning behind using the abstract only is that a library staff person would organize a collection of articles by reading only the abstract information, rather than by searching within journal articles. The 30 years of full-text articles of cartographic literature were downloaded from the ISI Web of Knowledge webpage using the following three criteria: 1. A cartographically themed article in the field of Geography, Computer Science, Geodesy, or History of Environmental Studies, 2. Published in English language; and 3. A research paper, book review, white paper, proceedings paper or an extended conference abstract.

An example of a journal article download form ISI given below (Figure 4.2).

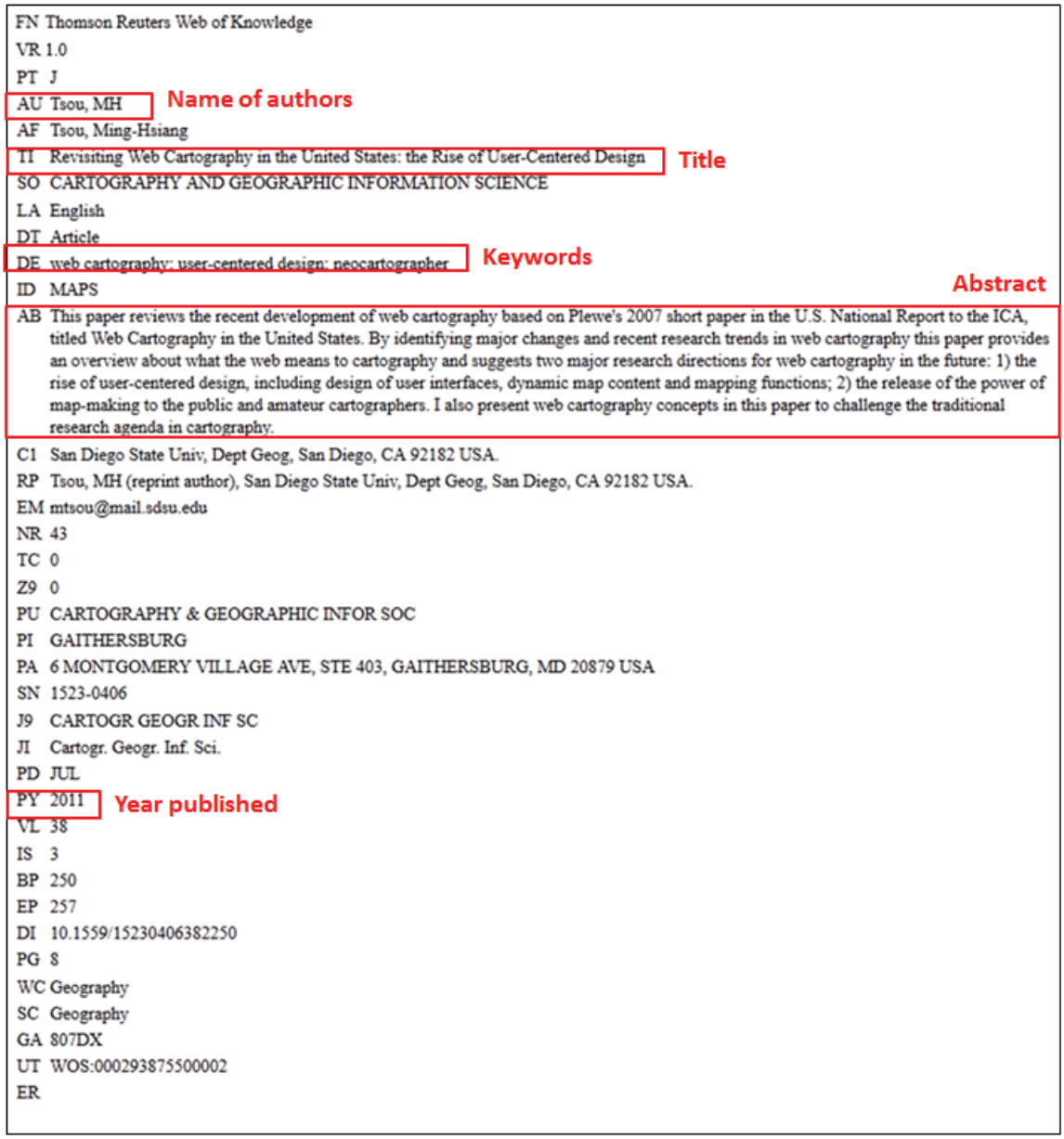


Figure 4.2 Example of a data entry downloaded from the ISI Web of Knowledge in standard BibTeX format. The red boxes highlight the information relevant for indexing. Only the author's name, the title of the journal paper, keywords, the full abstract, and the year published are used in the analysis.

One limitation of the ISI Web of Knowledge website is that it downloads abstracts only in BibTeX, unformatted TXT, and HTML formats. As can be seen from Figure 4.2, pre-processing of the raw data is necessary to remove a large amount of auxiliary data from the indexing

analysis. Only six items (author, journal name, title, keywords, abstract, and published year) are used for this analysis. Programming in Python and R is used to pre-process the data, to filter out unnecessary information, and to convert the abstracts into a useable format (tab-delineated TXT file). Principles from text analysis are then used to automatically index this data set and generate the keyword set. Figure 4.3 gives a detailed overview of the methods used. Each of the methods is described in detail below.

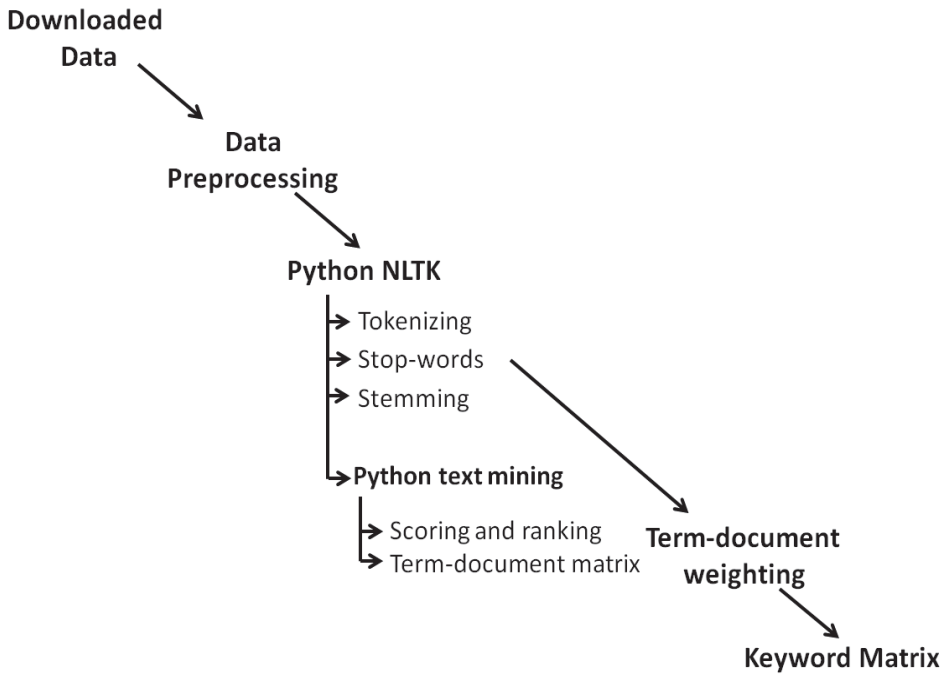


Figure 4.3 Processing steps involved in automatic keyword generation of the full-text document data set

As described in Chapter 2, the first step in text processing is to tokenize the data set. Text stemming, text lemmatization, and word frequency counts are used next to generate the keyword set. The Lancaster stemmer is used for stemming the full text data set used in this dissertation, as it identifies more alternative word forms than other stemming algorithms, such as the Porter stemmer.

Following the preprocessing and text analysis steps outlined above, a term-document matrix is constructed next which is used for formalizing keywords in this data set. For

implementation of the indexing process, the Natural Language Toolkit (NLTK) in Python is used (http://www.nltk.org) (Figure 4.3). NLTK is an open source Python library for natural language processing. NLTK offers implementation of multiple classification, tokenization, stemming, parsing, and semantic reasoning algorithms. The Python library Textmining (http://pypi.python.org/pypi/textmining) is then used for creation of the final term-document matrix.

Drawing from Figure 4.3, the whole process of automatically indexing the cartographic journal data set using the Python NLTK and Textmining library can be partitioned into 7 steps:

1. Tokenizing and stop word removal

The first step in text analysis of the full-text data set is to tokenize the document and to generate a stop word list to remove all words with no substantive meaning, such as prepositions or function words. The generated stop word list, using the Python NLTK package, consists of 544 stop words taken directly from the NLTK package.

2. Application of text stemmer to match tokens in the corpus to get rid of inflections

The Python NLTK package contains a word stemmer algorithm. The NLTK package comes with a list of precompiled text stemming methods, including the Porter and Lancaster stemmer as well the WordNet lexical database. As described earlier, the Lancaster stemmer is implemented and used in this experiment.

3. Construction of a term-document matrix

The Python Textmining library (http://pypi.python.org/pypi/textmining) is used to create the term-document matrix. The constructed matrix consists of 1,800 abstracts analyzed by word frequency and word occurrence in the documents. Every row in the matrix corresponds to an abstract of the data set. Due to the sparse nature of this matrix, further processing using the Python NLTK is necessary.

4. Setting an appropriate cutoff value

Figure 4.4 shows the number of keywords retained in the term-document matrix by cutoff value, as well as how sparse the term-document matrix is by cutoff value. A cutoff value of 1 returns 16,057 keywords, whereas a cutoff value of 1,423 returns only one keyword (namely GIS), meaning that the word "GIS" is present in at least 1,423 documents. As the purpose of this indexing experiment is document clustering and document classification, a more aggressive approach for selecting the cutoff value is being done. By analyzing the sparse nature of the term-document matrix and also considering recommendations from previous studies, as described above, a cutoff value of 300 is chosen. A cutoff value of 300 also follows the recommendation above as it represents around 20% of the number of documents in the term-document matrix.

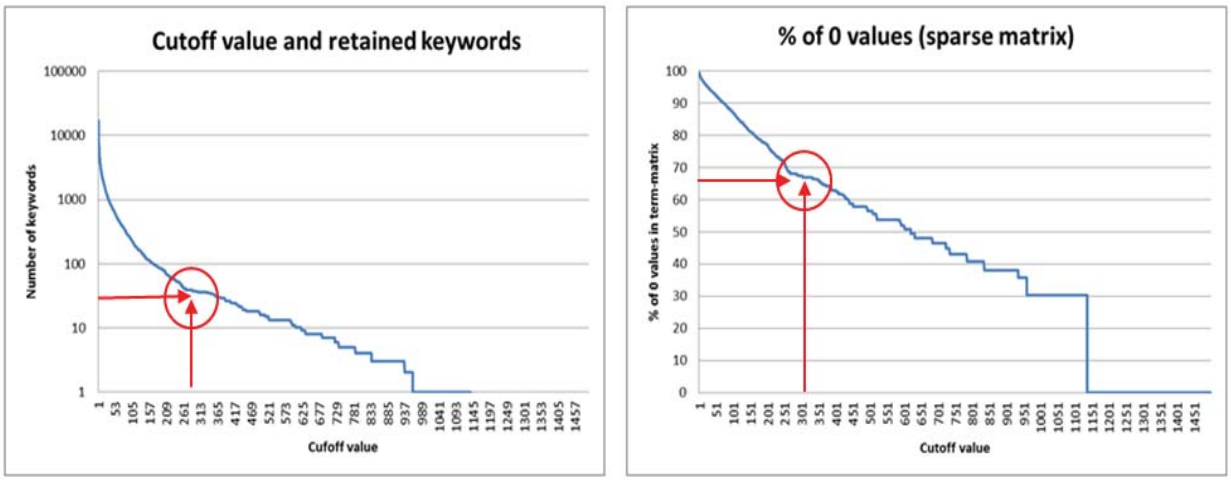


Figure 4.4 Number of keywords per cutoff value showing on a logarithmic scale on the left. The percentage of 0 values in the term-document matrix by cutoff value is presented on the right.

5. Improvement of the term-document matrix (term weighting)

The Python NLTK package as well as the R package "tm" (http://cran.r-project.org/web/packages/tm/index.html) is used for weighting and enhancing the term-document matrix. Term-document weighting schemas are distinguished between local and

global term weighting approaches. Local term weighting only applies weighting to individual documents within one document. Words which appear multiple times in a document are stronger than words that only appear once. In a global term weighting schema, all documents within a data sets are weighted. This weighting schema takes all documents into account, e.g. words that appear in only a few documents are likely to be more significant than words that are distributed across the whole collection of documents within the data set (Manning et al., 2008). As the focus of indexing cartographic publications is on the whole data set, a global weighting schema is used. The term frequency-inverse document frequency (tf-idf) method is applied, using the R "tm" package. Tf-idf weights words by the number of times it appears in documents. The method offsets the weight by the word's frequency in the whole data set.

6. Finalization of the term-document matrix (conversion)

Table 4.2 A small section of the full-text index after pre-processing, stemming, and tf-idf term weighting.

Author	Year	data	spatial	information	geographic	system	study	cartography	system
O'Kelly, ME	2012	0	0.044274	0	0	0	0	0.532321	0
Eisner, WR; et al.	2012	0.211822	0	0.037992	0	0	0	0	0.081759
Moellering, H	2012	0.082375	0.056572	0	0	0	0.120056	0	0
De Coene, K; et al.	2012	0.023168	0	0.033243	0.037109	0.036256	0.067532	0	0
Field, K; Demaj, D	2012	0	0.181839	0.056989	0.08482	0.165743	0	0	0
Hey, A	2012	0.082375	0	0	0	0	0	0	0
Montaner, C; Urteaga, L	2012	0	0	0	0	0	0	0.510141	0
Offen, K	2012	0	0.107189	0	0	0	0.113737	0	0
Andres, AJ; Cia, JC	2012	0	0	0	0	0	0	0	0
Faricic, J; Magas, D; Mirosevic, L	2012	0.169457	0.145471	0	0	0.099446	0	0	0
Carlton, G	2012	0	0.177096	0.046252	0	0	0	0	0.049767
Smallwood, TM	2012	0	0	0.040915	0.091344	0	0.083116	0	0
Ramos, BM; Pastor, IO	2012	0.067398	0	0.024177	0.053976	0.105473	0.147342	0	0
Altic, MS	2012	0.046336	0	0.033243	0	0.072513	0.202595	0	0

In order to generate the final term-document matrix, data processing and conversion using Python and Python NLTK for import into R, MatLab, and SPSS is conducted. Table 4.2 shows a

small section of the generated term-document matrix. Text stemming and stop word removal has already been applied. Term weighting has been applied. Each number in the table refers to the frequency of the token (stemmed words).

As depicted in Table 4.2, this matrix provides a format for the data set containing full text documents to be further processed and organized using clustering and classification. The next sections describe how indexing is applied to the other exemplar data types.

4.4.2 Spatial data

The second data type is exemplified by a spatial data set containing raster information describing the physiographic characteristics of the lower 48 states of the U.S. This data set is provided courtesy of Lawrence V. Stanislawski (personal communication, April 2010), USGS – Rolla, Missouri, and is the product of an ongoing USGS generalization project to typify landscape types in the continental United States.

Table 4.3 Factors used to classify landscape types (Stanislawski et al., 2011)

Environmental factors	Derived from	Literature referenced
Average elevation	USGS 3-arc second DEM	
Standard deviation of average elevation	USGS 3-arc second DEM	Grohmann et al. 2009
Average slope	USGS 3-arc second DEM	Grohmann et al. 2009
Runoff estimated	Watershed mean annual runoff from 1951 to 2000	Wolock and McCabe 1999
Drainage density	High-resolution (HR) NHD catchment areas	Stanislawski et al. 2007
Bedrock density	Bedrock density estimate derived from generalized geologic unit polygons	Reed and Bush 2005; Easson and Robinson 2001
Inland surface water	Water polygons from Medium Resolution (100K) NHD	

The intention is to use these input variables to identify regions with relatively uniform landscape characteristics, where tailored generalization processing sequences can be applied (Buttenfield et al., 2010; Stanislawski et al., 2010). Besides using the raw spatial data as an

automatic indexing schema for generating landscape regions, additional information provided in the data set could be used to create multiple manually derived keyword sets. However, as described earlier in this chapter, automatic indexing schemes are preferred whenever possible.

Figure 4.5 shows seven environmental factors (attributes) mapped to the same scale in geographic space. Darker blue values indicate higher attribute values. All seven variables have been normalized to a common range of 0-1000 (Larry Stanislawski, personal communication, April 2009).

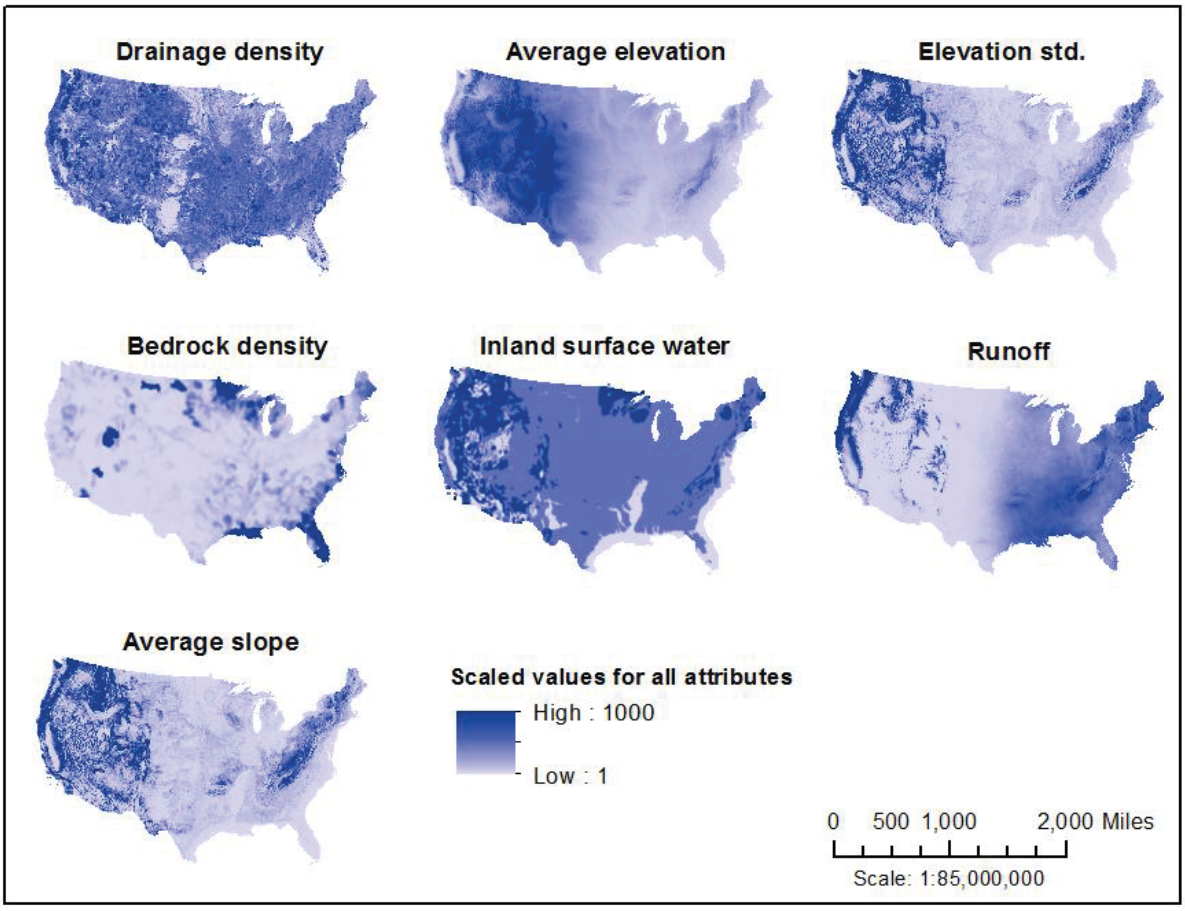


Figure 4.5 Environmental factors (attributes) mapped in geographic space. All attributes are normalized to a range of 0 - 1000. All seven attributes are used for indexing.

Prior analysis of the data set (Stanislawski, 2010) suggests using 7 classes. Figure 4.6 shows the output of a maximum likelihood classification based on the seven variables. As can be seen

from the error estimates in the right panel, the probability of misclassification is much higher in mountainous areas, in swamp / marsh areas, as well as in the arid southwest. A systematic analysis of indexing and organization methods may help to resolve the question of how many landscape types should be categorized, and whether these seven explanatory variables are sufficient to accomplish data organization.

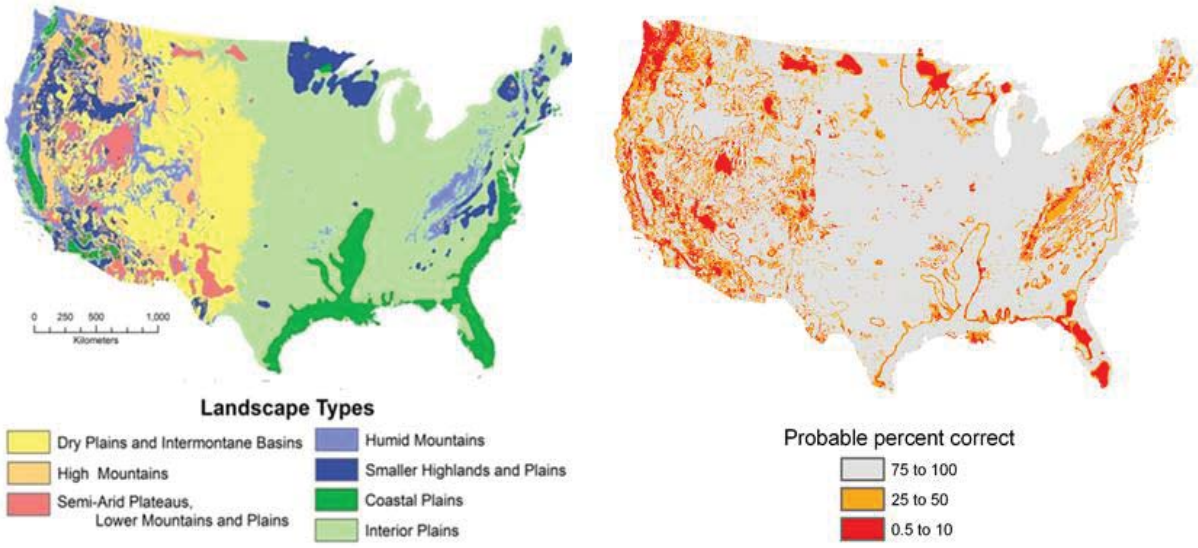


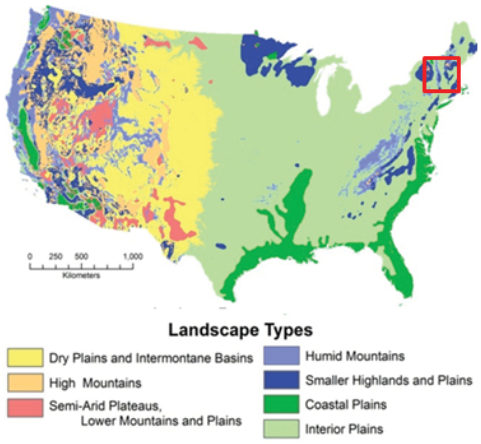
Figure 4.6 Maximum likelihood estimation of seven data sets used to predict landscape types across the contiguous United States, and a map estimating probable misclassifications (Stanislawski et al., 2010).

In the following paragraphs, different indexing strategies are briefly discussed for this spatial data set: automatic indexing on raw data, indexing by metadata, and indexing by manually derived indices. A fully automatic keyword generation is likely feasible, given that the data contains pixels with floating point values and each pixel represents a discrete object.

1. Indexing the raw spatial data

The first way to index this data set is to utilize the seven variables directly. Following the requirement of this experiment, this is the preferred method, as only minor human interaction is required; and therefore it will be used to index this data set. Python programming is used to

transform the seven single raster data sets into a single characterization file where each raster data set is represented as one column in the newly generated file. Figure 4.7 shows a section of the indexed spatial data set showing the 7 raster data sets as attributes in the table.



Drainage density	Average elevation	Elevation std.	Bedrock density	Inland surface water	Runoff	Average slope
74.7378	261.937	549.956	621.622	11.5975	208.329	611.62
100.217	370.285	381.854	614.286	12.2561	226.275	609.083
161.65	363.498	294.482	606.061	12.9609	253.705	469.18
119.424	384.176	500.172	612.903	14.1862	292.807	765.302
48.7029	413.805	465.663	620.69	16.0924	325.343	689.045
75.345	307.07	499.936	611.111	17.7848	270.192	743.44
53.6697	429.014	433.22	520	19.2276	306.522	701.578
42.3113	28.9855	106.972	906.25	7.05103	356.42	102.179
97.3585	56.7875	131.102	911.765	6.85624	368.093	250.882
236.295	32.413	115.989	911.765	6.76697	360.506	106.142
40.7577	31.1062	72.7416	500	2.69351	48.5645	104.52
28.5275	60.6792	322.896	454.636	3.35319	92.3355	230.563
14.1761	22.5064	28.1541	167.333	5.06177	53.6711	32.8539
16.0993	20.2992	16.4434	143.571	5.62388	57.3187	7.78352
46.3706	27.6538	31.1258	172.086	5.78369	67.9696	31.2977

Figure 4.7 Seven-dimensional index created from the spatial data set. The section is located in the north eastern part of the U.S. The red box shows the approximate location.

Each element in this data table is a discrete object and corresponds to one pixel in the data set. Pixels outside the study areas and non-landmass features, such as water bodies, are left out in the indexing table for increased processing performance. However, Python programming is used to merge back the zero pixel values, after the indexing process to the raster data file in order to represent the water bodies within the data set.

2. Alternative indexing strategies

The USGS Landover data sets come with metadata describing; for example, the data capture method or the sampling methods. However, using metadata records will not be helpful to accomplish this task as the provided metadata only contains information about the whole data set (e.g. capturing method, resolution, stewardship), but not pixel level information which is needed for this indexing experiment. A second alternative possibility would involve a pattern

recognition analysis of the data, but this type of advanced processing lies beyond the scope of the dissertation and will not be undertaken. However, it can be argued that manually derived keywords for spatial raster data can be established by aggregation and reclassification. By reclassifying the raster files to a coarser resolution a new manually derived data set can be established at every reclassification step. By transforming the raster data set to a coarser resolution, new semantics are established as information is aggregated. At each resampling set, a new indexing schema is created.

4.4.3 Software

The third type of data is a software data set compiling a list of 108 commands providing a census of all hydrological geoprocessing commands in ArcGIS, Arc Workstation, and Arc Toolbox (Wendel et al., 2009). The software commands. Characterization of this data set must be manual, since the only available information which can be derived automatically is the command names. By definition, these must be unique and specific to each command, thus word stemming methods or word frequency counts used for automatic keyword generation will not identify any commonalities. Metadata is not available, nor is it possible to access the command code, which is proprietary, to perform pattern analysis. Even if pattern analysis were applied, it would likely be unproductive, since the patterns in artificial language, such as software code, by themselves, would not carry sufficient meaning to distinguish one command from another. For this data type, keywords must be generated manually, based on how the commands perform their tasks, and on what type of data they operate.

Table 4.4 indicates an initial set of manually derived keywords, which were derived for the GIS command set, with advice from GIScientists and research hydrologists from the University of Colorado and USGS (Wendel et al., 2009; Viger, 2011). Source materials, such as online help from commercially available GIS products, GIS Manuals, and GIS textbooks, were used to refine and extend the list. The keywords describe characteristics relating to the type of data (raster or vector) accepted by the command; whether the command stands alone (i.e., atomic) or requires

prerequisite commands (i.e., molecular, as in the case of Flow Accumulation); whether the command processes terrain or water flow, etc. A small subset of data management commands (e.g. copy and paste) were inserted into the command set as controls.

Table 4.4 The initial keyword set and the final set (in bold) after the degrees of freedom are removed (strikeout text).

1. Data Management	**9. Local**
2. Raster Only	~~10. Regional~~
~~3. Vector Only~~	**11. Global**
4. Vector and Raster	**12. Geometric Only**
5. Atom	~~13. Attribute Only~~
~~6. Prerequisites~~	**14. Geometric and Attribute**
~~7. Flow Only~~	**15. Changes Spatial Relations**
8. Terrain and Flow	~~16. Does Not Change Spatial Relations~~

Degrees of freedom, refers to the redundancy in the data set. That is, the initial set of 16 keywords includes several pairs and triads which are mutually exclusive; and one needs to describe only one of the pair (or two of the triad) to determine the value of the third. Commands can operate on raster data only, vector data only, or both. One needs to record only two of the alternatives to capture all information given by the triad as a whole. . For example, knowing the number of commands which modify spatial relationships (keyword 15) and the total number of commands in the data set, one can determine by subtraction the number of commands which do not do so (keyword 16). Redundancies can be treated essentially as degrees of freedom, and thus one member is removed from the final set (strikeouts), leaving ten keywords (boldface). A Boolean matrix is created assigning a value of 1 when a GIS command applies, and a value of 0, if it does not apply to that characterization. A sample section of the final index of the GIS commands is shown in Table 4.5, coded for each of the ten variables.

Table 4.5 Sample of the Boolean matrix characterizing the software data set

	Data Management	Raster Only	Raster and Vector	Atom	Terrain and Flow	Local	Global	Geometric	Geometric and Attribute	Changes Spatial Relation
aggregate	0	0	1	1	0	0	0	1	0	0
area weighting	0	0	1	0	0	0	0	0	1	0
aspect	0	1	0	0	1	0	0	1	0	1
combine	0	1	0	1	0	1	0	0	0	1
conditional operation	0	1	0	1	0	1	0	0	0	0
copy	1	0	1	1	0	0	1	0	0	1
cost allocation	0	1	0	0	1	1	0	1	1	1
cost backlink	0	1	0	0	1	1	0	1	1	1
cost distance	0	1	0	0	1	1	0	1	0	0

4.4.4 Algorithms

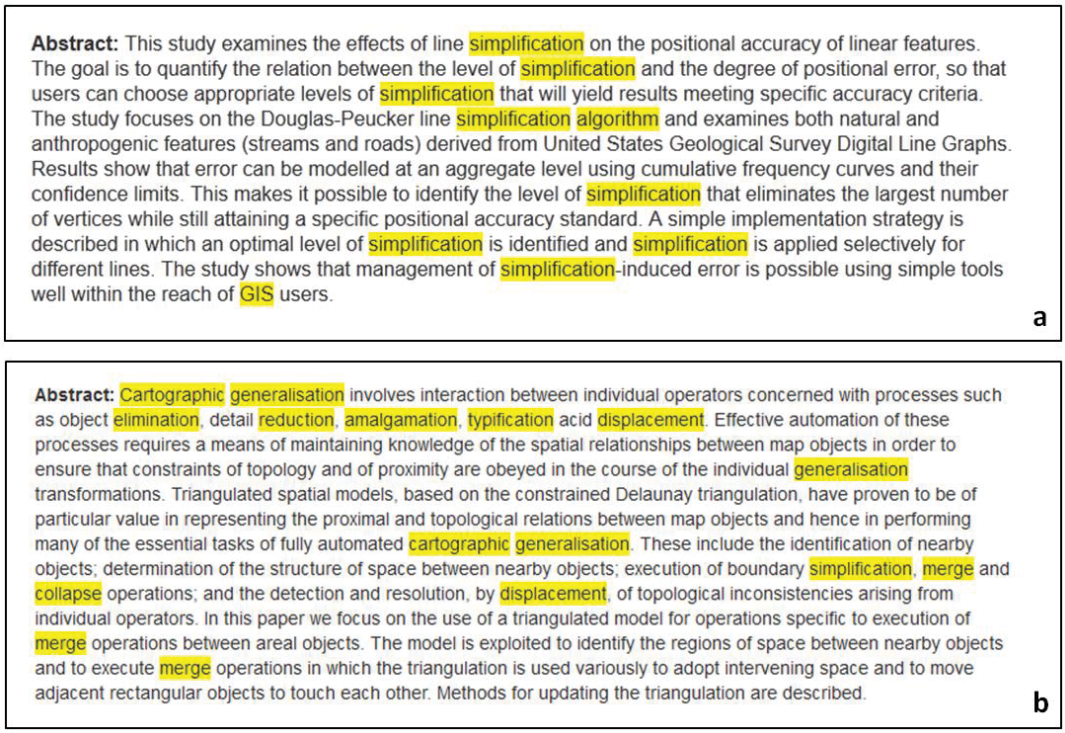
The final exemplar data set lies at the furthest extreme of the continuum and consists of algorithms from the field of cartographic generalization. This data set also closes the loop of indexability presented in Chapter 1. There are multiple possibilities to index this data set ranging from semi-automatic to manual strategies. However, as with the other data sets in the experiment, an automatic strategy is preferred.

Three different indexing strategies for the algorithm data set are available: 1. A combined approach of manual and automatic indexing, including text stemming and manual compilation of cartographic generalization taxonometric keywords, 2. Manual indexing by creating keywords, and 3. Evaluation of algorithm pseudocode. The following subsections will provide a description of how these strategies can be used for indexing, and will also provide the reason why the first method is preferred over the other methods.

a) Combined approach of indexing by full-text documents with the inclusion of taxonometric keywords

The first indexing strategy, also implemented in this experiment, is to index by full-text documents. Most algorithms are described by full text documents. Automatic generated

keywords can be derived from a full text description of the algorithm, using automatic text analysis tools in a similar fashion as the cartographic journal article data set presented earlier. However, the key to this indexing strategy is to select words from the full-text document which specifically refer to the algorithm. This can be accomplished through modified querying and weighting procedures, which are specifically tailored to terms describing cartographic generalization algorithms by including keywords from existing cartographic generalization taxonomies. For example, an article describing an algorithm is different in the way that more technical and algorithm specific terms are included in the text. Figure 4.8 presents two abstracts from Veregin (2000) (Figure 4.9 a) and Ware et al. (1995) (Figure 4.9 b).The keywords highlighted in yellow are algorithm specific keywords. The first abstract example can be indexed by one algorithm specific keyword, while the second example abstract includes eight different cartographic algorithm specific keywords.



Abstract: This study examines the effects of line simplification on the positional accuracy of linear features. The goal is to quantify the relation between the level of simplification and the degree of positional error, so that users can choose appropriate levels of simplification that will yield results meeting specific accuracy criteria. The study focuses on the Douglas-Peucker line simplification algorithm and examines both natural and anthropogenic features (streams and roads) derived from United States Geological Survey Digital Line Graphs. Results show that error can be modelled at an aggregate level using cumulative frequency curves and their confidence limits. This makes it possible to identify the level of simplification that eliminates the largest number of vertices while still attaining a specific positional accuracy standard. A simple implementation strategy is described in which an optimal level of simplification is identified and simplification is applied selectively for different lines. The study shows that management of simplification-induced error is possible using simple tools well within the reach of GIS users.

a

Abstract: Cartographic generalisation involves interaction between individual operators concerned with processes such as object elimination, detail reduction, amalgamation, typification acid displacement. Effective automation of these processes requires a means of maintaining knowledge of the spatial relationships between map objects in order to ensure that constraints of topology and of proximity are obeyed in the course of the individual generalisation transformations. Triangulated spatial models, based on the constrained Delaunay triangulation, have proven to be of particular value in representing the proximal and topological relations between map objects and hence in performing many of the essential tasks of fully automated cartographic generalisation. These include the identification of nearby objects; determination of the structure of space between nearby objects; execution of boundary simplification, merge and collapse operations; and the detection and resolution, by displacement, of topological inconsistencies arising from individual operators. In this paper we focus on the use of a triangulated model for operations specific to execution of merge operations between areal objects. The model is exploited to identify the regions of space between nearby objects and to execute merge operations in which the triangulation is used variously to adopt intervening space and to move adjacent rectangular objects to touch each other. Methods for updating the triangulation are described.

b

Figure 4.8 (a) Abstract sample from Veregin (2000). Words highlighted show cartographic generalization algorithm specific keywords, demonstrating that one or a small number of

keywords will suffice in some cases to isolate those parts of the full-text document referring to the algorithm. **(b)** Abstract sample from Ware et al. (1995). Words highlighted show cartographic generalization algorithm specific keywords, demonstrating that for this abstract multiple keywords are needed for indexing.

The downloaded data set consists of 2,350 journal articles describing generalization algorithms. As before with the full-text data set, most publications before 1995 do not come with a stored abstract in the ISI Web of Knowledge database (www.webofknowledge.com, accessed May 2013). After filtering out those articles, only 1,632 algorithm specific publications are left in the data set. The filtered data set is indexed in a similar fashion to the full-text data set presented in Section 4.4.1. However, algorithm specific modifications in the indexing steps are made. Figure 4.9 highlights the modification to the full-text indexing strategy from the journal data set presented earlier in this chapter. Each modifed step, highlighted in red, will be explaind following Figure 4.10.

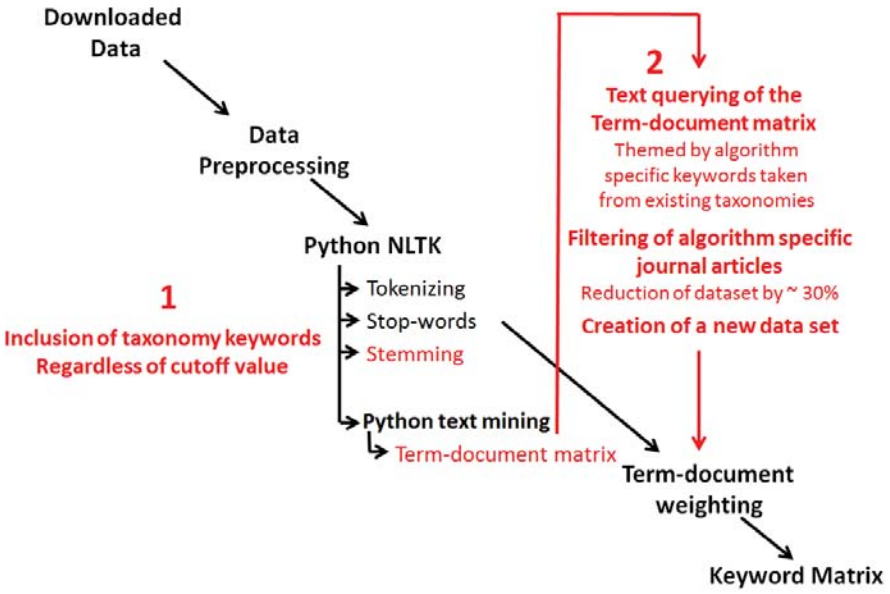


Figure 4.9 Processing steps involved in automatic keyword generation of the cartographic generalization algorithm data set. Text highlighted in red shows the additional steps necessary to filter out generalization algorithm specific publications.

1. Modifications of the stemming procedure

Stemming of the algorithm data set requires multiple extra steps. The downloaded algorithm data set from the ISI Web of Knowledge consists not only of cartographic generalization related articles but also of algorithm related articles from the field of Geosciences, Computer Science, Computer Engineering, and Mathematics. In automatically filtering out those articles, generalization algorithm specific keywords, such as simplification, smoothing, and aggregation from multiple existing taxonomies, have been compiled. The taxonomy from McMaster and Shea (1990) is used in combination with the taxonomy developed by Regnauld and McMaster (2007) in order to capture modern and classic keywords used in cartographic generalization (Table 4.6).

Table 4.6 Overview of cartographic generalization keywords.

Simplification	**Smoothing**
Amalgamation	**Merging**
Refinement	**Typification**
Enhancement	**Displacement**
Aggregation	**Exaggeration**
Collapse	**Classification**
Caricature	**All-in-one generalization**
Enlargement	**Detection**
Network	**Shape**
Scale	**Enlargement**
Interpolation	**Removal**
Categorization	**Resampling**

In addition to the keywords compiled from existing taxonomies, GIS related keywords such as GIS, Cartography, and maps are included in the stemming process.

The Python Natural Language Toolkit (NLTK) package is used for tokenizing, stop-word removal, and text stemming. Python text mining library is then used for generating the term-document matrix.

2. *Filtering of cartographic generalization related publications*

After the initial stemming process, the term-document matrix is taken to filter only cartographic generalization specific articles. This is done by only selecting articles which contain at least one cartographic generalization taxonometric keyword as well as at least one GIScience related keyword (Table 4.7). If an article does not apply to any of the generalization related keywords it is removed from the data set. From the original 1,632 publication, only 979 are retained in the data set. After filtering only cartographic generalization related articles Tf-idf weighting is applied to the document matrix. Table 4.7 shows a small selection of the final term-document matrix. Each number in the table refers to the frequency of the term.

Table 4.7 A small section of the final generalization data set indexed after preprocessing, stemming, and term weighting. A sample of generalization taxonomy keywords and automatic derived keywords are shown.

	Selection of generalization keywords						GIScience keywords			
ID	smoothing	merging	refinement	enhancement	displacement	classification	GIS	map	cartography	generalization
389	0	0	0.1103	0	0	0	0	0.1149	0	0
407	0	0.3860	0	0	0	0.1944	0.1392	0	0	0
1187	0	0	0.1563	0	0	0	0	0.1878	0	0.2696
1195	0	0	0	0	0	1.1668	0.0693	0	0.1298	0
1382	0.4720	0	0	0	0	0	0.1354	0.1144	0	0
568	0	0	0.2558	1.3192	0	0.4321	0	0.3064	0	0.1118
617	0	0	0	0	1.042054	0	0.0870	0.2298	0	0
739	0	0	0	0	0	0.9723	0	0	0	1.2675
770	0	0	0	0	0	1.3890	0.0491	0	0.2498	0
859	0	0	0	0.4240	0	0	0.1248	0	0	0

b) Alternative indexing approaches

Auxiliary information, such as full-text documents, technical reports, and taxonomies, could be used to manually generate a binary index, where cartographic generalization algorithms are characterized by keywords (Wendel and Buttenfield, 2010). This method will not be considered in this experiment, as an automatic indexing strategy is preferred. Another

possibility to index this data set is by evaluation of the algorithm pseudocode. Most algorithms are described in natural language pseudocode. The analysis of pseudocode would be considered semi-automatic, as a pre-compiled pseudocode data set has to be compiled first. Pseudocode is usually a high-level description of algorithms and is formulated in a more structural way than plain full-text documents (Cormen et al., 2009). Input and output features can be analyzed for characterization and building of an index. However, this type of advanced processing lies beyond the scope of the dissertation and will not be undertaken.

4.5 Summary of the indexing experiment

The indexing experiment was established to build an indexing framework for the data sets used in this dissertation. Automatic and manual indexing strategies have been presented for all four data sets, and indexing strategies have been tailored to the specific data sets used. Indexing is a necessary step for applying clustering and classification methods on these data sets. The purpose of this experiment was not to develop and evaluate new indexing strategies, but rather to use one method to index these data sets in a format that can be used for further clustering and classification.

The full-text data set, represented by journal articles, is automatically indexed using methods from natural language processing. The spatial data set, represented by a data set describing the physiographic regions of the U.S., is indexed directly on the unprocessed raw data. The third data set, presented by GIS commands, is indexed manually. The fourth data set, represented by a data set of cartographic generalization algorithms, is indexed by modified stemming of full-text articles describing the algorithms. The second part of this experiment, described in the next chapter, will use the indices created in this chapter, and apply clustering and classification methods to organize the four exemplar data sets.

CHAPTER V

Clustering and Classification Experiments

This chapter outlines the data organization experiment using classical and modern approaches from classification and clustering. The four datasets indexed in Chapter 4 form the inputs for the organization methods. All four datasets are organized using the full set of six clustering and classification methods. Recommendations for organizing each type of indexed data will be given by validation and evaluation of each method.

5.1 Methodological overview and structure of experiments

5.1.1 Clustering and classification methods

The order of the datasets analyzed in this chapter follows the continuum of indexability. Each dataset is organized by classical and modern methods of unsupervised clustering first, followed by classical and modern methods of supervised classification (Figure 5.1). Unsupervised clustering methods include Hierarchical clustering, k-Means clustering, and Self-Organizing Maps (SOMs). Supervised classification is demonstrated by k-Nearest Neighbor (k-NN), Decision Trees, and Support Vector Machines (SVM).

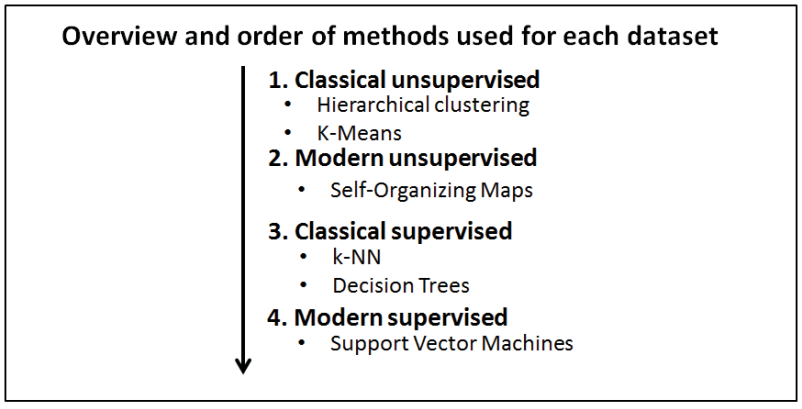


Figure 5.1 Overview and structure of the data organization methods used in this experiment

Unsupervised clustering methods, with the exception of SOM, require little or no training, while supervised classification methods in contrast require a training data set. As unsupervised

methods are applied first in this experiment, the training data sets are derived using a sub-sample of the optimal clustering solution for each data set. Such methodology has been applied in hybrid classification studies and is used when generating a training data set is labor intensive (Zhang and Xiao, 2012; Witten and Frank, 2005; Griffith et al., 2003). Derivation of the training data sets is based on the objectives to generate a valid representation of the data, to represent all classes of the data, and to form classes that manifest the highest variability (Bramer, 2007). Guidelines for selecting a training set of appropriate size follow Witten and Frank (2005) who suggest using a random split sample of 50% or larger for the training data set. .

5.1.2 Comparison and evaluation

Dataset organizations that result from the six methods are evaluated by common metrics. As described in Chapter 3, the Davies-Bouldin index, the Silhouette index, and the Dunn index will evaluate the unsupervised methods. A fourth metric, the Homogeneity and Separation metric will be discussed for the first data set for explanatory purposes, but dropped for the other data sets as it provides insufficient distinguishing power for any of the data sets. Cluster stability will be evaluated to establish a reasonable number of clusters for each dataset. Cross-validation measurements will be used to evaluate the supervised classification methods.

5.1.3 Software environments

Multiple software and programming languages are used. The R libraries "fastcluster" (http://cran.r-project.org/web/packages/fastcluster/), "fpc" (http://cran.r-project.org/web/packages/fpc/index.html) and "pvclust" (http://cran.rproject.org/web/packages/pvclust/index.html) are used for Hierarchical clustering. The Matlab Statistics Toolbox (http://www.mathworks.com/products/statistics/) and the Cluster Validity Analysis Platform (CVAP) (http://www.mathworks.com/matlabcentral/fileexchange/14620) are used for k-Means implementation. SOM analysis and evaluation is conducted in Matlab using the SOM toolbox

(http://www.cis.hut.fi/somtoolbox/). The software packages SPSS and R are used for evaluation of supervised classification algorithms. The R library "class" (http://cran.r-project.org/web/packages/class/index.html) is used for k-NN analysis and cross validation. Classification trees analysis is conducted with the statistics software package Statistica (http://www.statsoft.com). The "e1071" (http://cran.r-project.org/web/packages/e1071/index.html) library in R is used for Support Vector Machines analysis. Validation of supervised classification is being conducted using internal validation methods within each of the packages described above.

5.2 Full-text dataset organization

The first data set to be organized is the automatically indexed full text document. The index for this data set was generated by word stemming and term-weighting methods. The purpose of this clustering is to group cartographic journal articles so that similar articles are placed in the same cluster. Such a grouping might be used by a librarian wanting to create an online catalog which can be readily updated as new journal issues are published. The full data set with all cluster memberships is shown in Appendix (A).

5.2.1 Unsupervised methods

5.2.1.1 Cluster evaluation and selection

Multiple criteria are used to choose an optimal number of clusters (Milligan and Cooper, 1985; Everitt et al., 2001), including the local extrema (highest or lowest value of evaluation metrics), the cluster stability (the number of clusters at which item membership tends to stabilize), and the number of classes where the values of evaluation metrics tend to level off, indicating that the addition of more classes will not provide additional distinction of information within a data set. In order to select an optimal number of clusters, it is also important to take into account the purpose of the clustering, and this forms a fourth criterion. Several challenges are associated with using these criteria. Challenges occur in applying criteria

however. For example the local extrema do not automatically correspond to good clustering as these values usually occur at a very low or a very high number of clusters. Depending on the data set, some evaluation metrics do not show well defined local extrema or leveling off regions. For this reason, only evaluation metrics which show clearly identifiable extrema are used for cluster selection. To further explain how the evaluation criteria will be used, Figure 5.2 shows the four indices used for Hierarchical and k-Means clustering.

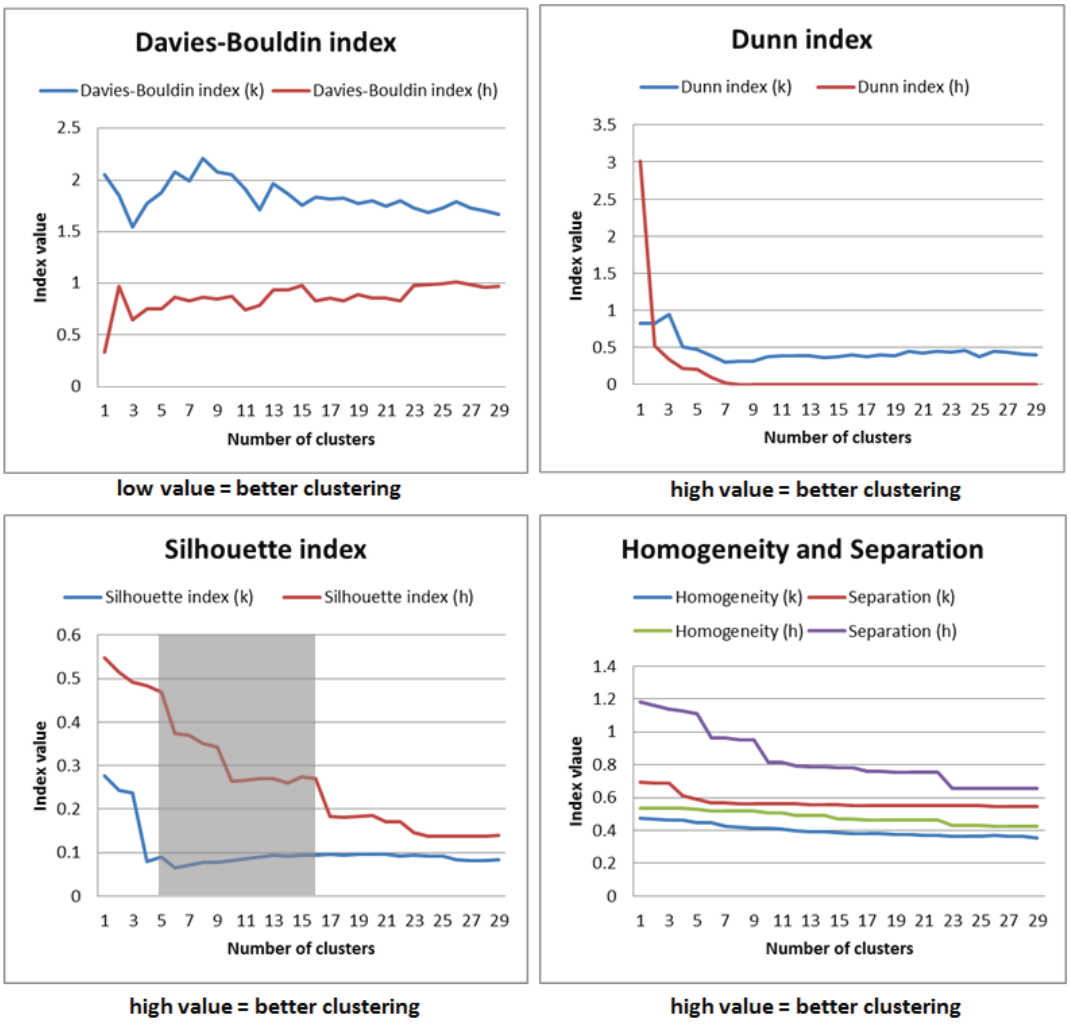


Figure 5.2 Evaluation metrics for unsupervised clustering of the full text data set. For these and subsequent data sets, the blue line shows the progression of values for k-Means clustering (k) and the red line reflects Hierarchical clustering (h) results. The area shaded in grey shows the

range of clusters chosen as optimal based on local extrema and leveling off region. This region will define a range of cluster solutions for cluster stability evaluation.

It can be observed from Figure 5.2 that the optimal number of clusters determined by each evaluation metric can differ; and a single criterion is insufficient to determine a single optimum for number of clusters since some metrics do not exhibit a distinct optimal value. For example if the choice of the optimal number of clusters would rely only on the extreme values, one to three clusters would be automatically selected using any of the evaluation metrics. These choices are obviously illogical, as evidenced by the subsequent rise in the index as the formation of additional clusters explains variation remaining in the full text dataset. The Davies-Bouldin and Dunn index values for both clustering methods level off quickly. The Homogeneity and Separation metrics do not show any extrema or distinct leveling off regions and are therefore unsuited for selection of regions for further evaluation.

The Silhouette index is chosen for defining leveling off regions and local extrema as indicated by the grey box in Figure 5.2. Based on the index, a range of 5 to 16 clusters has been chosen. After 5 clusters the Silhouette index for (k) drops off sharply, hitting its lowest (worst) value at 10 clusters; before leveling off till 16 clusters. After 16 clusters another drop can be observed.

a) Cluster stability in Hierarchical clustering

A representative sample of twelve different papers published by four different authors has been drawn from the full text data set to explore cluster stability and formation (Table 5.1). The papers by Hurni represent traditional cartographically themed papers; papers published by Cartwright highlight artistic aspects cartography; papers published by Burghardt report on cartographic generalization; and papers published by Crampton represent critical cartography. The title and abstract of the paper can be found by the ID listed in Appendix A. The numbers in the table correspond to the cluster membership each data object is assigned to for a given number of clusters.

Table 5.1 Hierarchical cluster membership stability and formation from 5 to 16 clusters.

		Number of clusters											
Author	ID	5	6	7	8	9	10	11	12	13	14	15	16
Hurni	569	1	1	1	1	1	1	1	1	1	1	1	1
	568	4	4	4	4	5	5	5	5	6	7	7	9
	1054	3	3	3	3	3	1	1	1	1	1	1	1
	1304	1	1	1	1	1	1	1	1	1	1	1	1
Cartwright	191	1	1	1	1	1	6	6	6	7	8	8	8
	192	1	1	1	1	1	1	1	1	1	1	1	1
	193	1	1	5	5	6	6	6	6	7	8	8	8
Burghardt	171	1	1	1	1	1	1	1	1	1	1	1	1
	790	1	1	1	1	1	3	3	3	3	4	4	4
Crampton	279	1	1	1	1	1	1	1	1	1	1	1	1
	280	1	1	1	1	1	1	1	1	1	1	1	1
	281	1	1	1	5	5	10	11	12	13	14	15	16

As can be seen in Table 5.1 cluster membership stabilizes at 10 clusters. At 9 clusters, papers published by the individual authors are assigned to different clusters while papers on the four very different topics are assigned the same cluster membership, indicating semantic instability. When moving up to 10 clusters, more variation is accounted for, and the papers become more meaningfully organized into separate categories. For 11, 12, and larger numbers of clusters, the organization of papers into clusters does not change, although the cluster ID does vary. In this way, one can say that cluster stability has been reached.

b) Cluster stability in k-Means clustering

Table 5.2 shows the cluster memberships after multiple k-Means clustering solutions have been applied. For each cluster solution, the k-Means clustering algorithm was run multiple times in order to avoid a bias towards random initial seed selection in the k-Means clustering process as well as to produce more stable results (Jain, 2009). Recall from Figure 5.2 that while the global optimum Silhouette value is reached at 6 clusters for the k-Means, the Davies-

Bouldin index reaches a local optimum at 12 clusters, calling for further examination. Examining cluster stability, it is apparent in the table that at twelve clusters three of the four articles published by Hurni fall into the same cluster together with the more "traditional" oriented cartographic article by Burghardt. Furthermore, both articles published by Crampton fall into a single class. When moving on to 13 clusters, no additional semantic stability is gained over the 12 cluster solution. In fact, some stability is lost as papers begin to drift into multiple clusters.

Table 5.2 k-Means cluster membership stability and formation from 5 to 16 clusters. Cluster formation is independent from those shown in Table 5.1.

		Number of clusters											
Author	ID	5	6	7	8	9	10	11	12	13	14	15	16
Hurni	569	3	2	4	2	1	6	4	7	5	8	8	1
	568	3	2	2	3	3	4	3	7	7	8	8	14
	1054	3	2	4	2	1	6	6	7	7	8	8	14
	1304	4	3	5	3	9	1	6	10	3	13	6	7
Cartwright	191	3	1	2	4	2	2	9	5	8	1	3	13
	192	5	4	1	1	5	10	3	2	4	7	12	15
	193	3	1	2	4	2	2	4	7	7	8	8	14
Burghardt	171	3	2	4	2	1	6	4	7	7	14	8	14
	790	3	1	2	4	2	2	9	5	5	8	3	1
Crampton	279	3	2	4	2	1	6	4	7	5	8	8	1
	280	3	2	1	1	2	3	3	7	7	10	11	8
	281	3	2	4	5	7	5	7	9	10	3	13	9

c) Self-Organizing Maps (SOM)

Unlike the previous classical clustering methods, the standard SOMs implementation can be interpreted as a fuzzy clustering method (Sarlin and Eklund, 2011). As can be seen from Figure 5.3, there are no discrete boundaries between clusters. SOM can be regarded as a method for regionalizing a multi-dimensional attribute space.

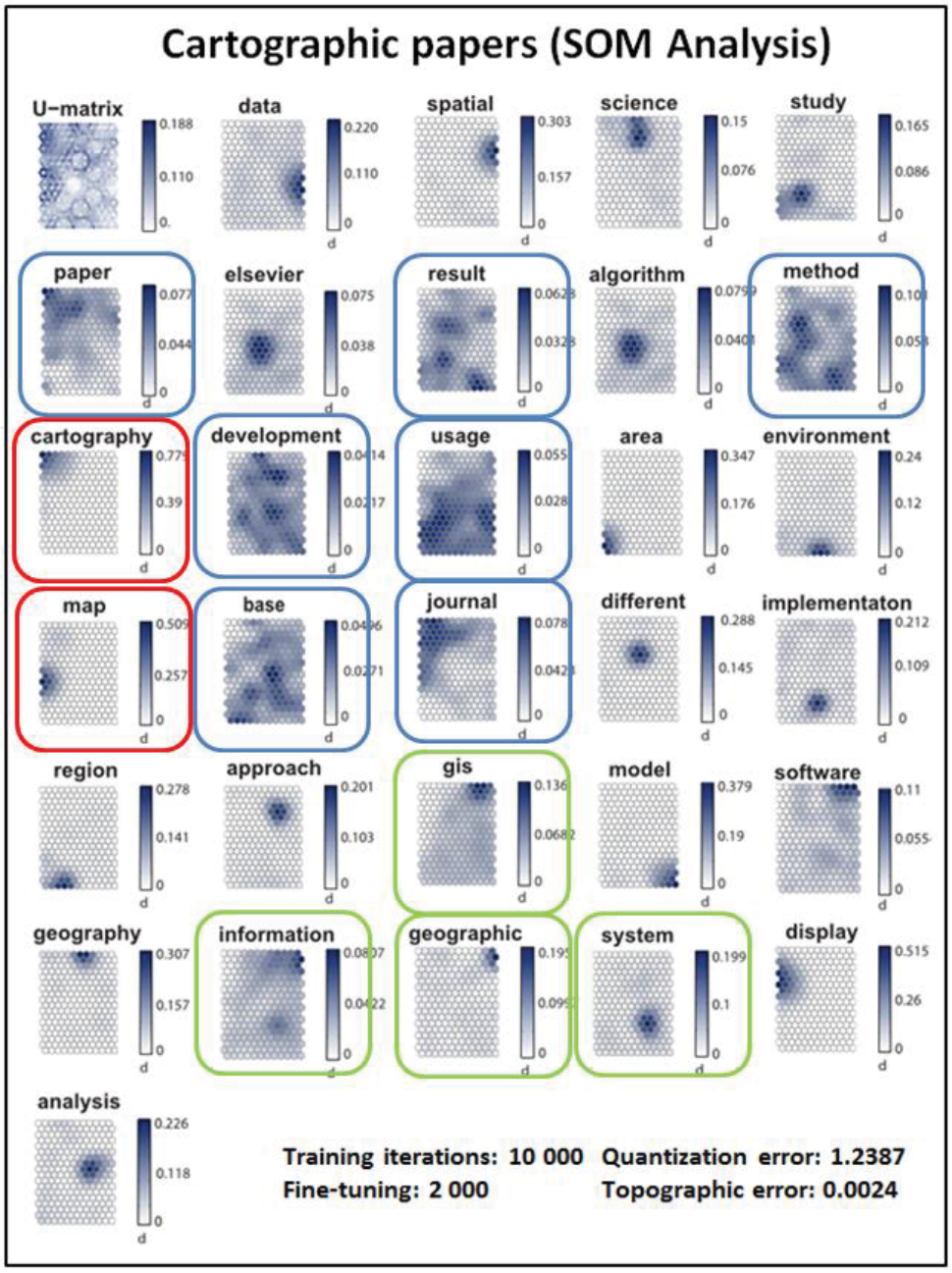


Figure 5.3 SOM visualization of the full-text data set. The U-matrix illustrates the overall compactness for all clusters as a whole and each variable of the dataset is represented in its own attribute plane. The red, blue, and green boxes highlight interesting patterns of coincidence, clustering, and scattering within the attribute space.

By comparing the SOM attribute planes, certain patterns emerge in the full-text dataset. Coincident dark shading in the attribute planes indicates semantic overlap between attributes.

For example, when analyzing the cartography and map attributes (red boxes) it can be seen that there is little coincidence in cluster signature. One possible interpretation is that authors usually select either cartography or mapping as the sole keyword. Also, darker SOM cells in the cartography cluster usually refer to older articles published before 2000, whereas the mapping terminology is often used in more recent papers describing web and mobile cartography.

Conversely, the green boxes indicate attributes which do coincide in the attribute space. Some articles use the term GIS instead of Geographic Information Systems. The largest overlap occurs between the attributes "geographic" and "GIS", whereas the attribute "system" is more localized and information is spread over the whole attribute plane. One possible reason for this is that the words "information" and "system" are also used in other contexts and are not only connected to the term "GIS". The term "geographic" is however mostly connected to the whole term of "Geographic Information System". It is also apparent that other attributes do not have a single uniquely located cluster signature and are scattered across the whole attribute space (blue boxes). These stemmed keywords are non-specific and are present in nearly all papers.

In order to meaningfully compare SOM to classical clustering results a linear SOM approached has been implemented next where each SOM cell corresponds to a cluster. In a linear SOM approach the vertical axis of the SOM shape is always kept at 1 and SOM shapes such as 1x8 or 1x10 are generated. This approach was successfully implemented by Wang et al. (2013) and has been applied in many different domains. Table 5.3 shows the cluster memberships after multiple linear SOM solutions have been applied. Examining cluster stability, it is apparent in the table that at eleven clusters three of the four articles published by Hurni fall into the same cluster together with the more "traditional" oriented cartographic article by Burghardt. Furthermore, both articles published by Crampton fall into a single cluster. When moving on to 12 clusters, no additional semantic stability is gained over the 11 cluster solution. In fact, some stability is lost as papers begin to drift into multiple clusters. These results are very similar to the clustering results from k-Means clustering. By reducing the SOM

output space where each cell corresponds to one cluster, multiple characteristics with k-Means are shared, such as cluster convexity (Wang et al., 2013). Therefore it is expected that the linear SOM method will archive similar results as k-Means clustering.

Table 5.3 Cluster membership stability and formation from 5 to 16 clusters. Cluster formation is independent from those shown in Table 5.1 and Table 5.2.

		Number of clusters											
Author	ID	5	6	7	8	9	10	11	12	13	14	15	16
Hurni	569	2	4	4	5	5	5	6	7	8	8	8	8
	568	3	4	4	6	5	5	7	6	8	7	9	9
	1054	3	4	4	5	5	5	6	6	8	8	7	8
	1304	5	1	1	1	1	10	6	12	13	14	1	2
Cartwright	191	1	6	6	7	8	3	4	5	3	6	10	11
	192	3	3	5	4	7	5	3	2	2	2	14	15
	193	2	5	4	6	5	2	4	6	4	7	9	10
Burghardt	171	3	4	5	5	6	5	7	7	7	8	8	9
	790	2	5	5	6	5	4	5	6	4	7	9	10
Crampton	279	3	4	4	5	5	5	7	7	8	8	7	8
	280	3	4	4	5	4	6	7	7	9	9	6	7
	281	2	4	5	6	6	4	4	4	6	4	12	13

5.2.1.2 Clustering results

As indicated by cluster stability analysis in the prior section, Hierarchical clustering generated more stable clusters which are an indication that the full text data set shows a hierarchy of the full text journal papers in the data set as indicated by Figure 5.4. Hierarchical clustering is used as the preferred classical clustering method for this data set. Figure 5.4 shows the dendrogram visualization for the full text document.

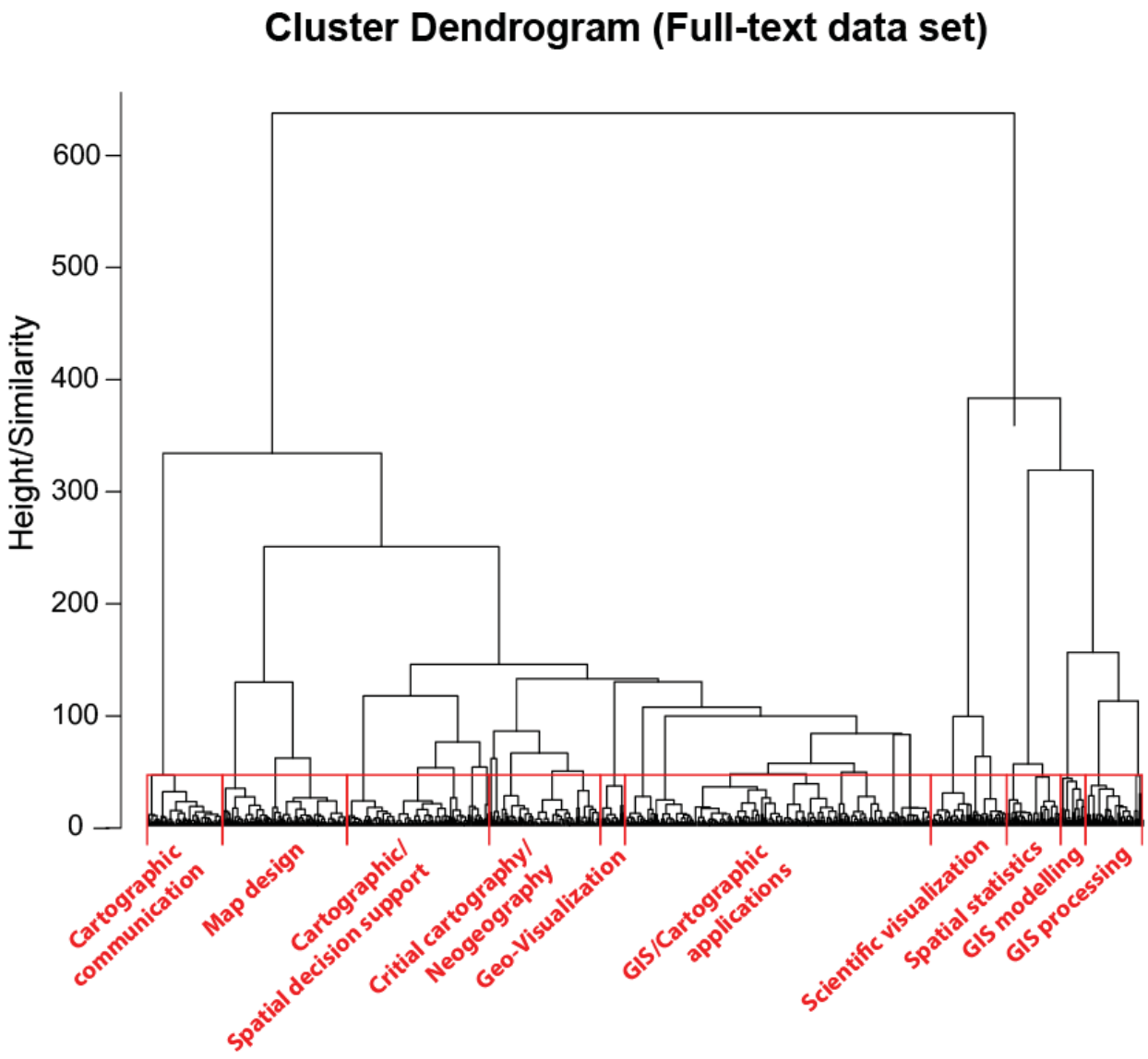


Figure 5.4 Dendrogram of the full-text data set showing class labels in red for the optimal 10 cluster solution.

As indicated by Figure 5.4, cluster size varies for the whole data set, with GIS and cartographic application themed journal papers forming the largest cluster, and GIS modeling publications forming the smallest cluster. From the dendrogram is it also apparent that two main groups are formed with cartographic themed papers on the left side and GIS themed papers on the right side of the dendrogram. On the cartography side three subgroups are formed, with one focused on cartographic communication, another focused on map design, and

the third focused on the application of cartographic methods. On the GIS side two larger groups are formed with scientific visualization on the left side and GIS modeling and spatial statistics on the right side.

For clustering of the full text data set, Hierarchical clustering tends to generate more stable clusters than does k-Means and linear SOM clustering. Cluster stability is established with fewer clusters than with k-Means and linear SOM clustering. Hierarchical clustering is already able to form stable clusters and correctly group the selected target publications at 10 clusters versus 11 with linear SOM clustering and 12 clusters with k-Means clustering. k-Means on the other hand generates well separated and compact clusters as indicated by the Silhouette index, Homogeneity and Separation indices. However, a 12 cluster solution is necessary to generate to establish stable k-Means clusters.

The traditional SOM approach (Figure 5.3) does not provide crisp clustering by itself. However, the SOM algorithm has the advantage that relationships among articles can be explored as distances between data objects. The degrees of membership can be established by using the BMU hierarchy of the SOM. For meaningful cluster stability comparison only the linear SOM approach will be applied for the remaining data sets.

5.2.2 Supervised methods

a) k-NN

Three training data sets are used ranging from 30%, 50% to 80% sample size. The training data sets are derived by random split sampling from the optimal clustering data set, specifically the 10-cluster solution generated by Hierarchical clustering. Cross-validation on the three training sets guides selection of an optimal number of k neighbors. Figure 5.6 shows the performance of k-NN algorithm on all three training samples using cross validation over 1,000 iterations. All variables are normalized to a common range from 0-1, using Euclidean distance.

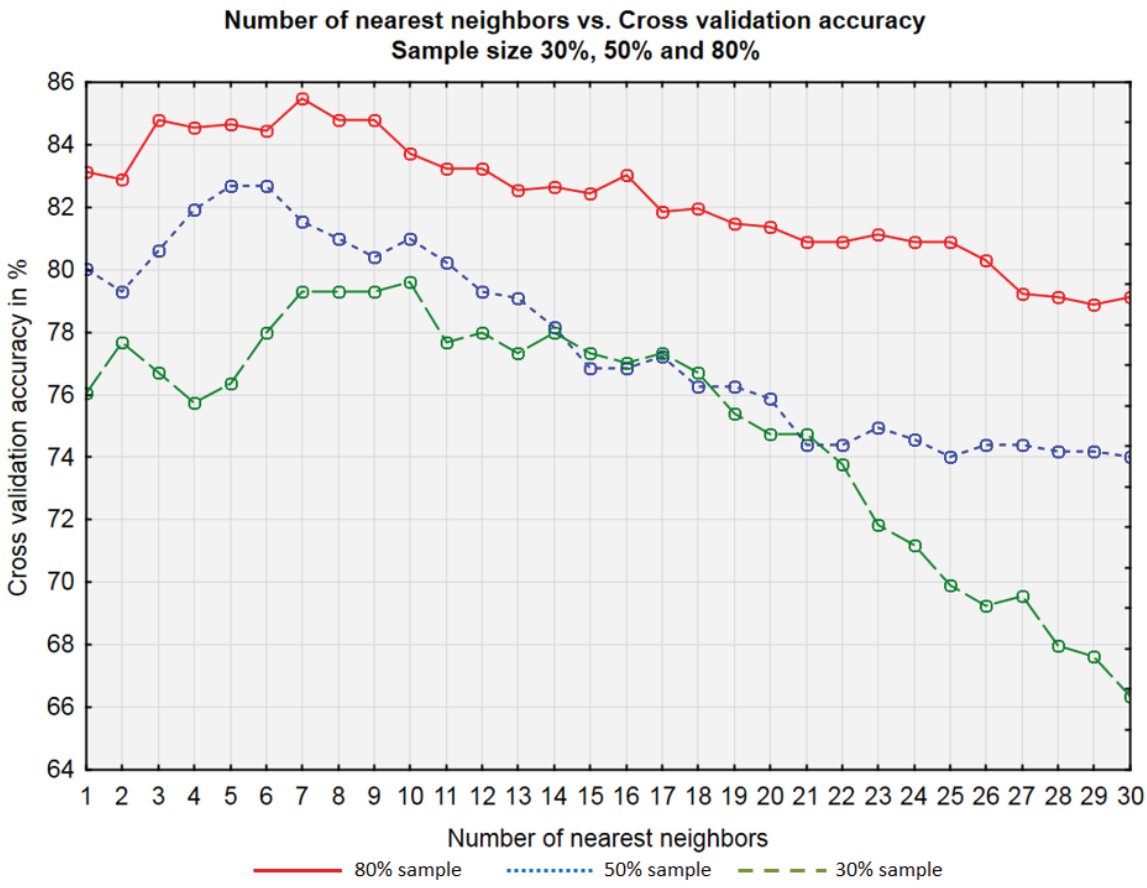


Figure 5.5 k-NN analysis of selecting the optimal number k of nearest neighbors for training of the classifier.

It can be seen from Figure 5.5 that with cross validation accuracy peaks at 7 nearest neighbors for the 80%, 5 to 6 for the 50% and 7-11 neighbors for the 30% training sample. Overall, the 30% training sample shows the lowest accuracy. The 50% and 30% training data sets show similar validation accuracy between 14 and 22 with the 50% training sample staying more stable while increasing the number of nearest neighbors. The highest cross validation accuracy of 85.2% is archived with the 80% training sample at 7 nearest neighbors. As the 80% training data set achieves the overall highest cross validation accuracy levels it is used for training of the classifier for the whole data set.

b) Classification trees

Figure 5.6 shows the cross validation for Random Forest classification over the course of 500 random trees, once again using three training data sets for selection of the optimal classifier. As described in Chapter 3, k-fold cross-validation on the training data set is used to determine the optimal choice of the Random Forest parameters. As more trees are added to the model, the misclassification rate for training data decreases.

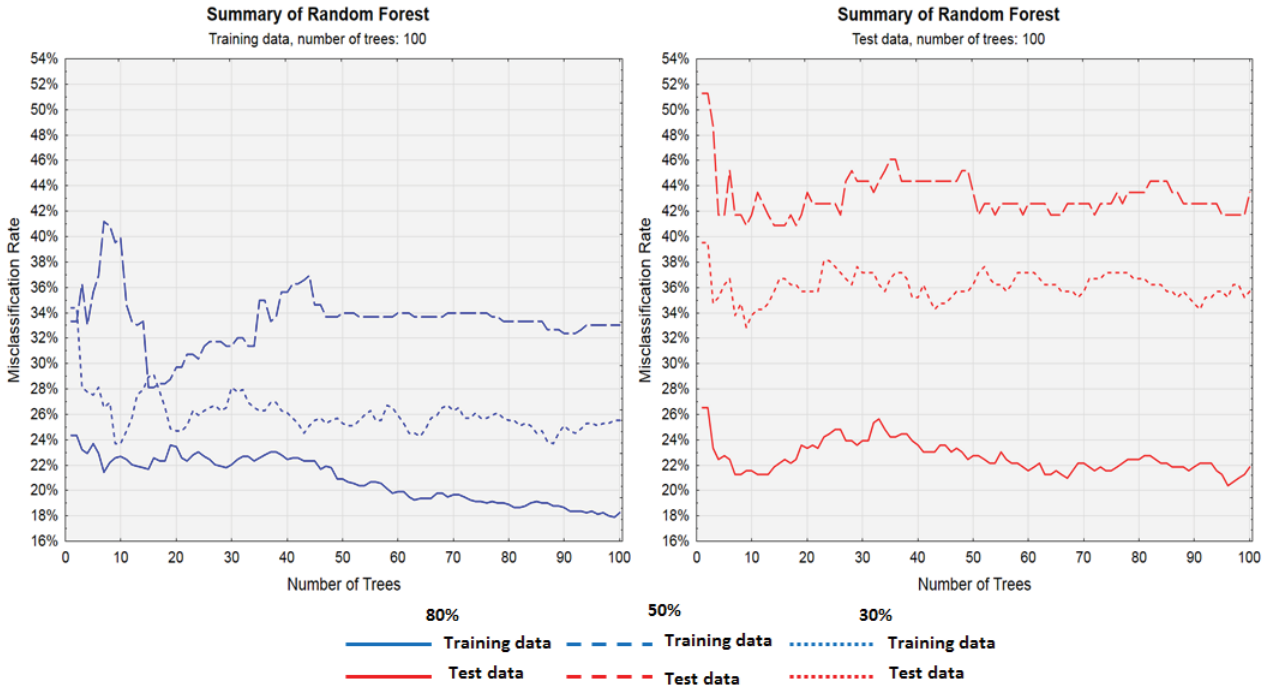


Figure 5.6 Summary of Random Forest analysis using three different training sample sizes to determine the optimal classifier.

As can be seen from Figure 5.7, the 80% training sample shows the lowest overall misclassification rate of the training data compared to the 50% and 30% sample. By further investigating the error rates it can be observed that the lowest error "margin", which is described by the difference between the misclassification rate of the test data and the training data, is achieved by the 80% sample and the largest error "margin" is generated by the 30% sample. As the 80% sample training data set archives the lowest misclassification rate as well as the lowest error "margin" it is applied to train the whole data set.

c) Support Vector Machines

An important step in working with Support Vector Machines (SVM) is the selection of the SVM kernel type. For the full test data set a Radial Basis Function (RBF) kernel is used. As indicated in the literature, an RBF kernel is preferred as it will work on both non-linear and linear separable data (Dioş et al., 2007). Table 5.4 shows the results from the SVM training on the three training samples.

Table 5.4 SVM training parameters using 80%, 50%, and 30% training sample. SVM per class indicates how many training vectors are calculated per class.

SVM (80% sample training set)		
SVM Type 1	Kernel type: RBF	561 support vectors
Cross validation accuracy: 83.5%	**Class accuracy:** 86.2%	
SVM per class: 25 (1), 32 (2), 217 (3), 26 (4), 17 (5), 68 (6), 33 (7), 41 (8), 40 (9), 62 (10)		

SVM (50% sample training set)		
SVM Type 1	Kernel type: RBF	380 support vectors
Cross validation accuracy: 79.9%	**Class accuracy:** 84.1%	
SVM per class: 152 (1), 50 (2), 20 (3), 29 (4), 19 (5), 43 (6), 19 (7), 10 (8), 13 (9), 25 (10)		

SVM (30% sample training set)		
SVM Type 1	Kernel type: RBF	236 support vectors
Cross validation accuracy: 76.6%	**Class accuracy:** 80.7%	
SVM per class: 99 (1), 25 (2), 17 (3), 25 (4), 16 (5), 11 (6), 12 (7), 11 (8), 7 (9), 13 (10)		

The 80% training sample produced the best overall cross validation accuracy of 83.5% and a class accuracy of 86.2%, followed in order by the 50% and 30% sample data set. When further exploring the training results it can be seen that for the 80% and 50% training samples class 1 shows the most SVM vectors per class meaning that this will be the largest predicted class. The likely reason for this is that the training data set is derived from the optimal Hierarchical clustering solution in which the "GIS/Application" cluster forms the largest clustering. The 80%

sample training data set is used to train the classifier for the whole data set with an overall class accuracy of 86.1%.

d) Comparison of classification results

The results of all three classification methods compared to the optimal clustering solution are shown in Table 5.5. It is assumed that the optimal clustering solution represent the 100% correctly classified data set. As before, the same selection of target journal articles is being explored for class membership assignment and misclassification.

Table 5.5 Classification results compared to the optimal Hierarchical clustering results. Misclassified journal papers are shown in red.

Author	ID	Hierarchical clustering	K-Nearest Neighbor	Random Forest	SVM
Hurni	569	1	1	1	1
	568	5	5	5	5
	1054	1	1	1	1
	1304	1	1	1	1
Cartwright	191	1	1	6	2
	192	6	1	1	1
	193	1	1	1	1
Burghardt	171	1	1	1	1
	790	3	3	1	1
Crampton	279	1	6	1	1
	280	1	1	1	1

One target paper was misclassified when k-NN classification was applied. One paper by Crampton (279) was placed in a separate class. Moving on to Random Forest it can be seen that two target papers got misclassified. Both articles (192 and 790) were moved into class 1. When further exploring Random Forest results it can be observed that class 1 is by far the largest class with the highest number of observations (Appendix A). When moving on to SVM classification three target papers in total were misclassified compared to the solution generated by Hierarchical clustering. Again, as before with Random Forest, two of the three misclassified

papers fall into class 1 which is also the largest class through all classification results of this data set. This might be due to the fact that the training data sample was derived from the optimal Hierarchical clustering results as well as the randomized training sample.

By comparing the results to the whole data set (Appendix A), k-NN classification performed best with 182 misclassified papers and an accuracy rate of 87.9%, followed by SVM with a total of 201 misclassified papers corresponding to an accuracy rate of 85.2%. Random Forest classification performed most poorly with 280 misclassified papers and an accuracy rate of 80.2%.

5.3. Spatial data set

The spatial data set is the largest dataset of the four and contains seven grid layers each with roughly 500,000 pixels. Continuous pixel values represent seven terrain and precipitation variables as described in Chapter 4. The original delineation of physiographic regions shows seven classes. The purpose of the clustering experiment is to explore if additional meaningful regions could be formed using the same set of input variables, especially to better distinguish regions in Midwest America, and to correct inconsistencies in the montane regions and the Southwest desert.

5.3.1 Unsupervised methods

5.3.1.1 Cluster evaluation and selection

As before, the clustering is first evaluated to select a range of appropriate clusters. Figure 5.8 shows the indices for Hierarchical and k-Means clustering. Only the Dunn index shows well defined extrema. The Dunn index is therefore used for determining a range of optimal clusters. Higher values account for better clustering. The grey box, as an indication of good clustering, spans from 5 to 12 clusters. Beyond 5clusters the Dunn index for (h) drops sharply. At 12 clusters the index for (k) shows a local maximum before leveling off. This also corresponds well with the DB index where both (k) and (h) indices start to stabilize after 12 clusters. One

additional criterion in defining the range of optimal cluster solutions, specific to this data set, is that only solutions for more than seven clusters are evaluated as the purpose of this clustering is to explore if more than the seven original clusters can be generated.

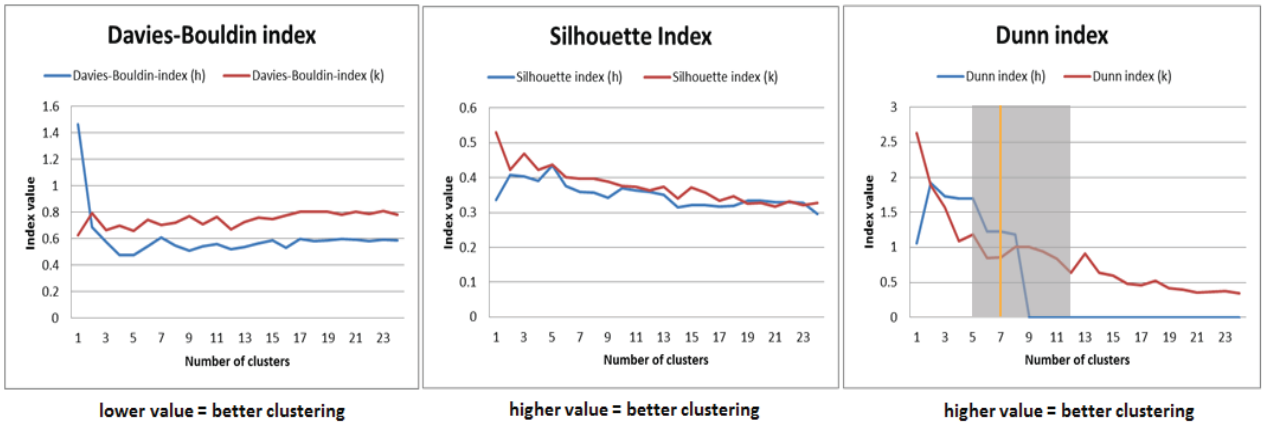


Figure 5.7 Evaluation metrics for Hierarchical and k-Means clustering. The area shaded in grey shows the range of optimal clusters based on local extrema and leveling off region. The orange line shows the seven class solution from prior analysis by Stanislawski (Stanislawski et al., 2010).

a) Cluster stability in Hierarchical clustering

Figure 5.8 shows multiple Hierarchical clustering solutions ranging from 7 to 12 clusters. It can be observed that the 7 and 8 cluster solutions form overly uniform clusters. For example both the humid and dry central plains fall into a single cluster. The high plains (light orange) cluster in the same group as most of Utah and Nevada (red box 1).

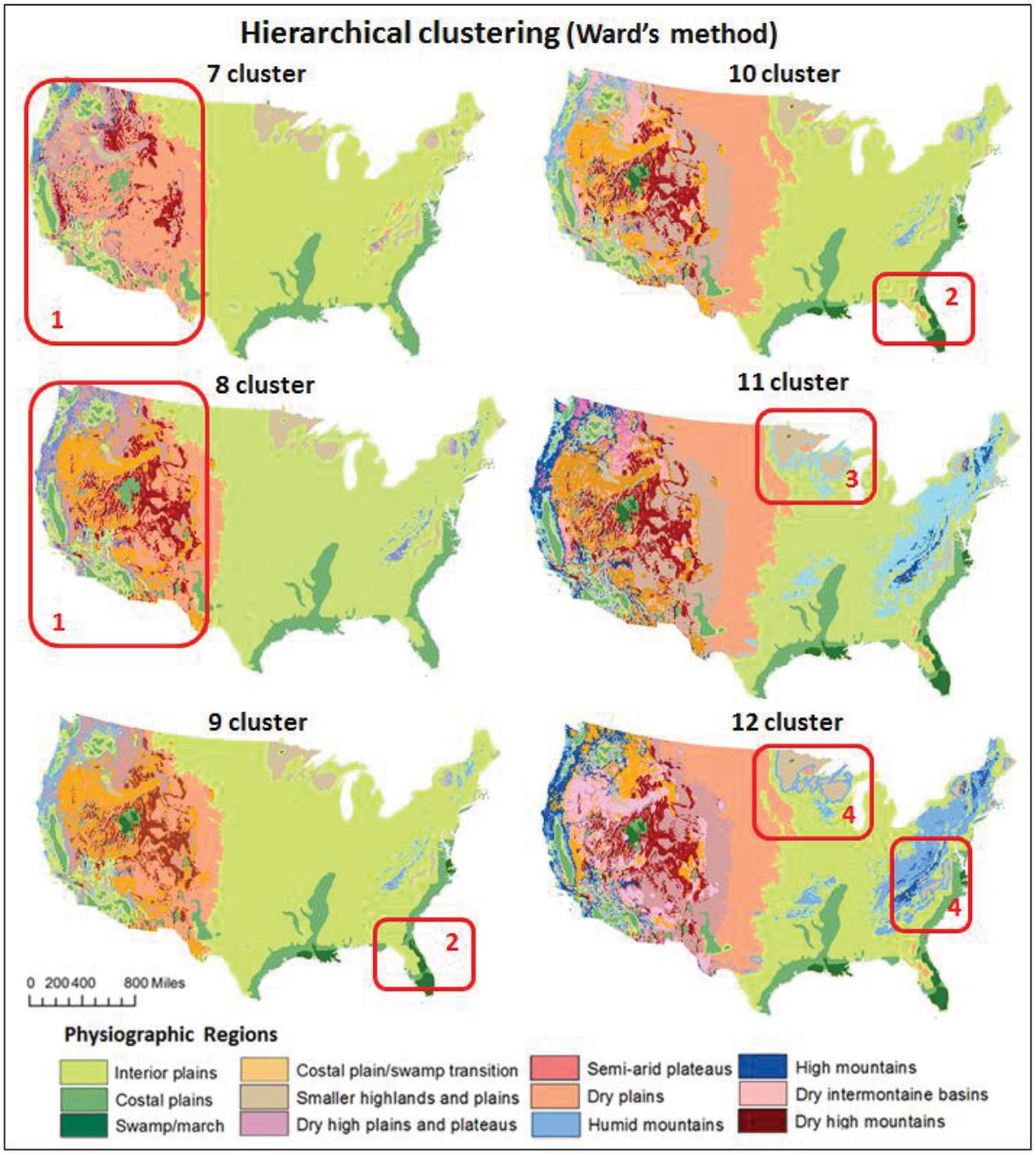


Figure 5.8 Range of solutions for Hierarchical clustering. The red boxes show areas of interest.

When moving on to the 9 and 10 cluster solution it can be observed that more variation is introduced. The 10 cluster solution shows a distinction between swamp areas and coastal plains as well as the drier inland area in Florida (red box 2). However, when increasing to 11 clusters, small banded artifacts emerge (red box 3). When further increasing the number of clusters

additional banding appears (red box 4). The reason for the appearance of these artifacts is due to the hierarchical nature of the clustering algorithm. Ten clusters have been selected as the optimal number of clusters for seven input variables as it represents most uniformity without the appearance of banding.

b) Cluster stability in k-Means clustering

Figure 5.9 shows the results for six different k-Means solutions. In comparison to Hierarchical clustering, k-Means clustering returns more uniform clusters without creating any banding. The 7 and 8 cluster solutions produce similar results and show overly uniform clustering in the western U.S. (red box 1), whose physiographic characteristics are highly varied. For example the cluster depicting the interior plains (light green) extends far west into most parts of Montana and even into Washington. In the 9 and 10 cluster solution more distinct regions are introduced. This introduced variation is most noticeable in the Pacific Northwest (red box 2). For example, there is now a clear distinction between high mountain regions (dark blue) and the humid coastal mountains (light blue). New clusters are introduced in the Gulf of Mexico and Florida regions at 11 and 12 clusters. There is now a distinction between swamp areas and coastal plains (red box 3) as well as more diversification in the mountainous west. When increasing the number of clusters to 12, a third cluster, describing the transition zones between the eastern coastal plains and the swap marsh areas is introduced (red box 4).

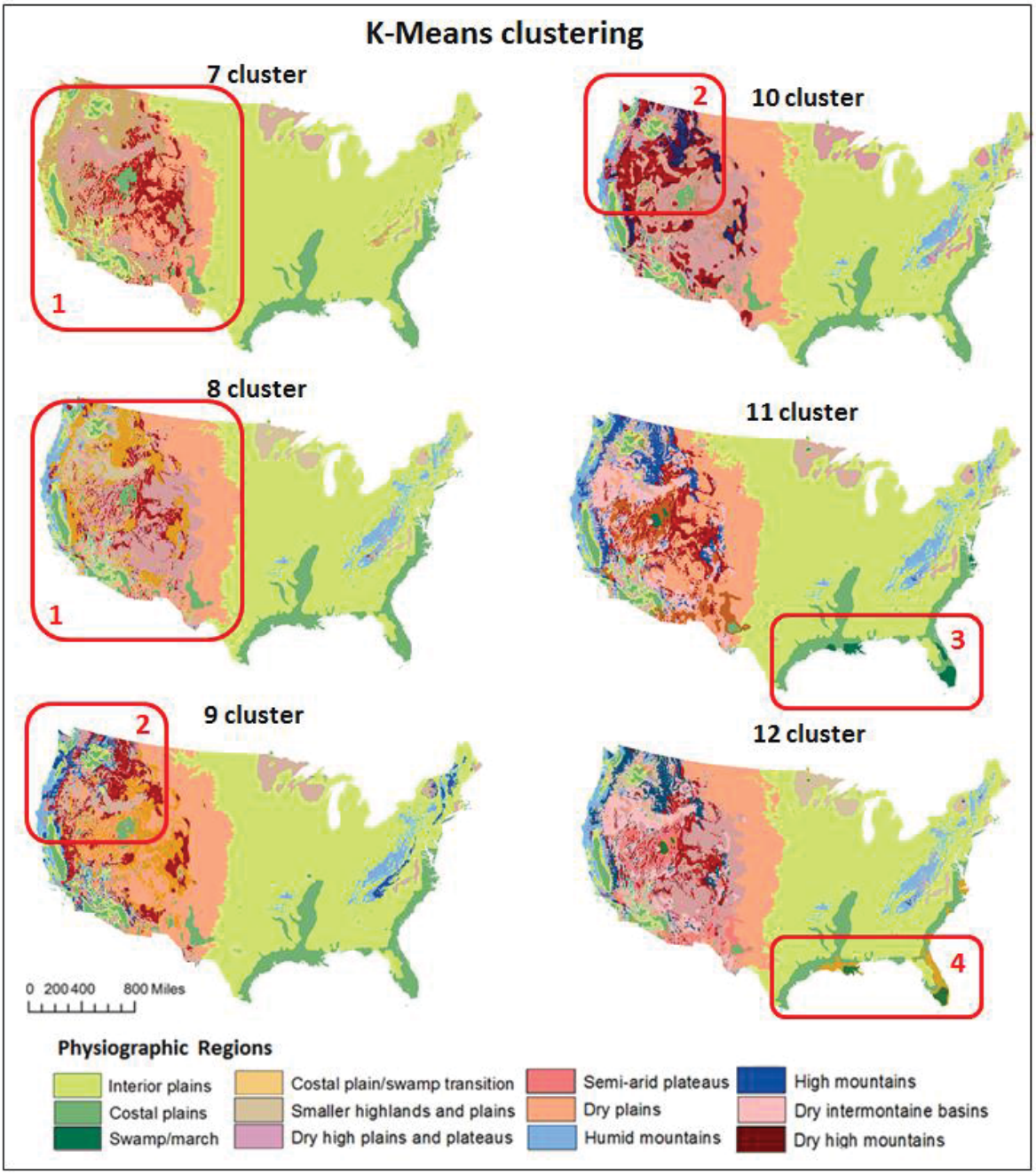


Figure 5.9 Range of physiographic cluster solutions for k-Means clustering.

The k-Means 12 cluster solution is chosen as the optimal cluster solution for classical clustering methods. It is able to extend the original 7 class solution the furthest without creating artifacts.

c) Self-Organizing Maps (SOM)

In comparison to the full-text data set, three different SOM clustering strategies are used for the spatial data set. Creating multiple SOM clustering solutions is possible due to the larger size of the data set as well as the purpose of this clustering, to determine additional clusters for the seven layer data set. The first approach uses the linear SOM method as seen before where each SOM cell corresponds to a cluster and each SOM shape is defined as linear (1x8, 1x9 …).

The second approach (Figures 5.10 a) and b)) follows a similar principle as the linear SOM approach where each SOM cell corresponds to one cluster. However, instead of defining a linear SOM shape, a non-symmetrical SOM shape as recommended by Vesanto (2005) is defined. This approach is referred to as clustering by BMU (Best Matching Unit) and is commonly applied in engineering (Vesanto and Alhoniemi, 2000; Come et al., 2011). Figure 5.10 shows SOM clustering by BMU for an 8 and 12 cluster solution. By using this method, crisp cluster boundaries can be generated. However, when increasing the SOM size to more than the number of input variables, degrees of membership become visible as indicated in the 12 cluster solution. This is due to the fact that the 7 variables are now spread over 12 SOM cells. In both BMU SOMs the U-Matrix is not well defined to form regions. The quantization error is reduced by increasing SOM size while the topographic error stays stable across SOM sizes.

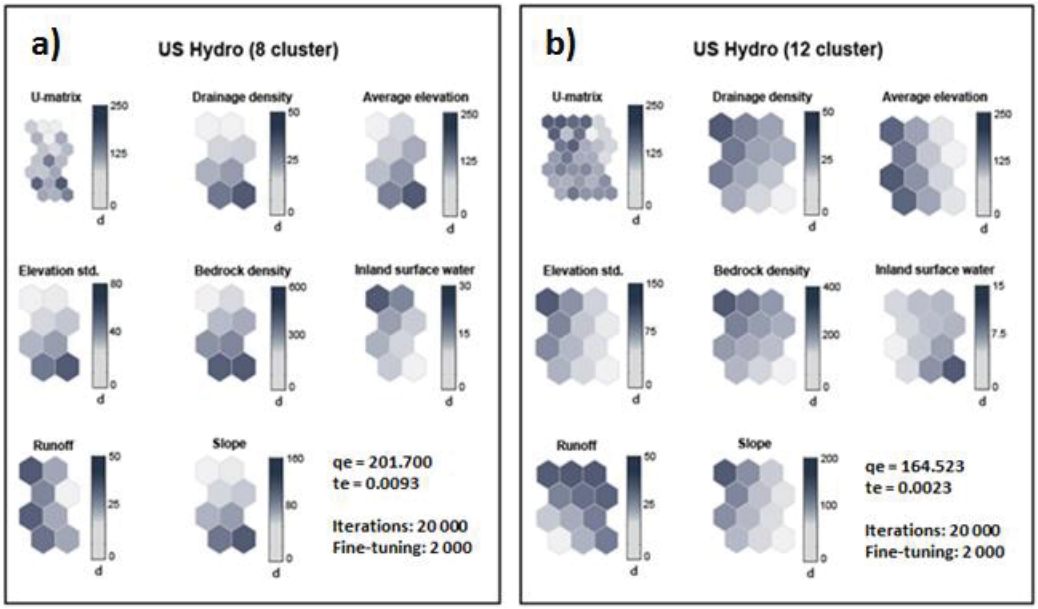


Figure 5.10 SOM BMU clustering solutions applied to the spatial data set. The reader is cautioned that the y-axes in the three panels are not scaled uniformly.

Figure 5.11 shows a comparison between the linear SOM and SOM BMU clustering. It is apparent that SOM BMU depicts better overall clustering. From the 8 cluster solution (red box 1) it can be seen that in the linear SOM the western part of the US is assigned to the same cluster as the costal plans of the eastern part of the US. The SOM BMU solution is able to correctly distinguish these groups but overall clusters are too uniform in the mountainous western part of the US. Moving on to the 12 cluster solution it can be observed that in both SOM methods small band like artifacts emerge (red box 2).Furthermore, the linear SOM is not able to correctly assign humid and high mountains regions (blue cluster) which can be observed in the high plains region. Overall linear SOM produce less meaningful clusters than SOM BMU clustering.

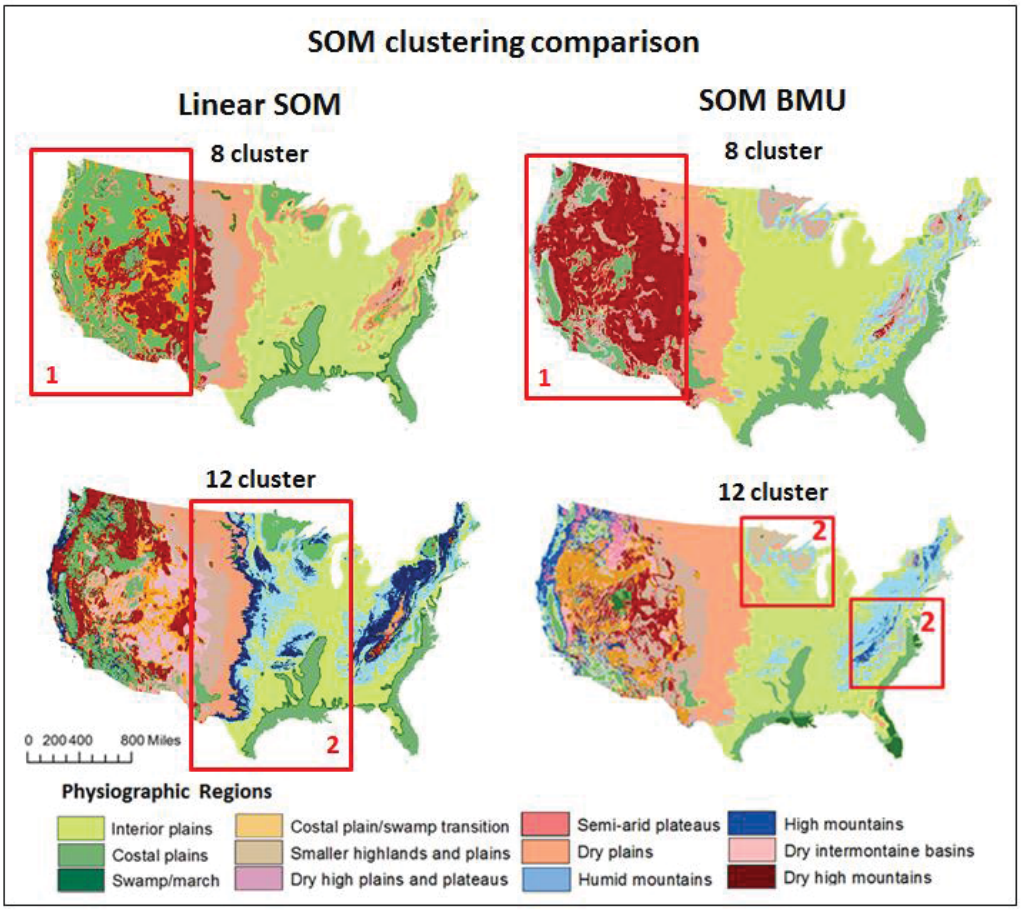


Figure 5.11 Comparison of linear SOM and SOM BMU clustering.

A third SOM approach first applies the standard SOM method and then clusters the SOM output using k-Means clustering to create crisper cluster boundaries. k-Means and SOM plus k-Means nearly produce similar results in the western part of the country. The increased computations required for running a SOM first does not provide any additional distinguishing information to the clustering process (Figure 5.12). In fact, when moving to the Gulf of Mexico region (red box 1) it can be seen that the k-Means solution depicts one more cluster showing a transition area between the swamp region and coastal plains. In the SOM BMU clustering solution, it can be seen that similar band-like features, as already seen in Hierarchical clustering results, are appearing around major clusters (red boxes 2). By increasing the number of cells (BMUs) in the SOM to more than the number of input variables, the SOM tries to assign degrees of membership to each cluster by trying to map 7 clusters onto a 12 cell grid.

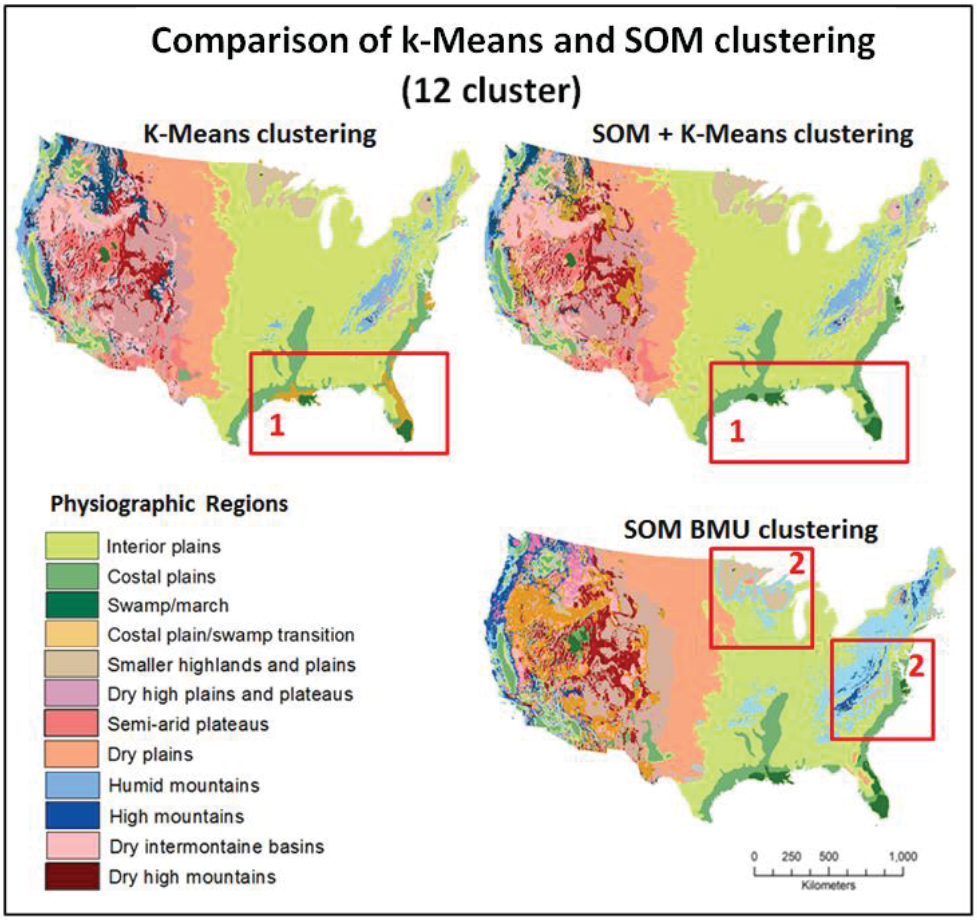


Figure 5.12 Optimal k-Means classical clustering solution and SOM clustering.

5.3.1.2 Clustering results

Overall, k-Means clustering generates the most clearly defined 12-cluster solution for this dataset. Furthermore it is also the most stable method indicated by the evaluation indices. No artifacts are present, even when increasing the number of clusters.

5.3.2 Supervised methods

a) k-NN

For the spatial data set, 30%, 15% and 7% samples are used to train the data. In contrast to the full-text document data set, an 80% or even a 50% sample is highly unlikely. Smaller training sets provide a more realistic set of training samples as training data sets for spatial data

are labor intensive to generate as well as are usually manually generated. Figure 5.13 shows the cross validation accuracy for the 30%, 15%, and 7% sample over a range of 1 to 60 k nearest neighbors. The upper limit of k nearest neighbors can be defined as $k = \sqrt{n}$ where n is the number of data points in a data set (Segaran, 2007).

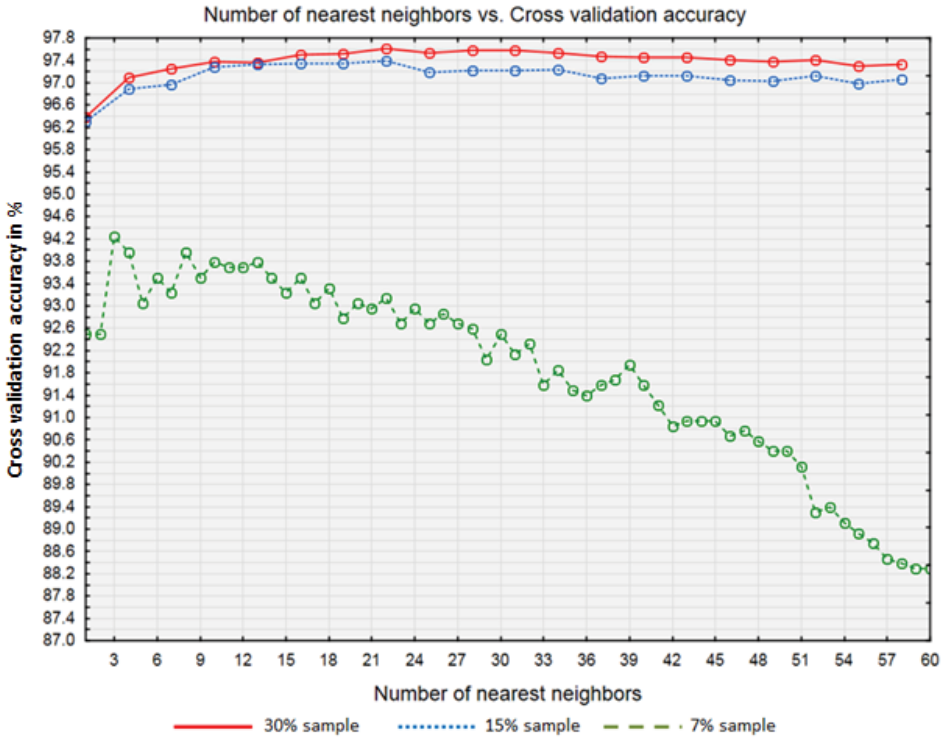


Figure 5.13 k-NN analysis for selecting the optimal number of k neighbors for three training data sets. Due to the increased complexity and computing time the 30% and 15% training samples show cross validation accuracy fore every other nearest neighbors while the 7% sample shows all 60 nearest neighbors for determining the number of nearest neighbors.

Even with a 7% training sample data set, a high cross validation accuracy of 94.2% at 5 nearest neighbors can be achieved. The cross validation accuracy drops steadily after a k value of 8 indicating that using this training data set to establish the classifier will not be optimal. The 30% and 15% training samples perform similarly and both achieve a high cross validation accuracy of 97.3% at k=22 for the 15% sample and 97.4% cross validation accuracy at k=22 for the 30% sample. The cross validation accuracy stabilizes for k values larger than 14 which is an

indication that the classifier can be robustly trained with both training sample sizes. As both training samples achieve essentially the same cross validation accuracy the 15% sample training data set will be used to train the classifier for the whole data set.

b) Classification trees

Figure 5.14 shows the misclassification rate for the 30%, 15%, and 7% training data set.

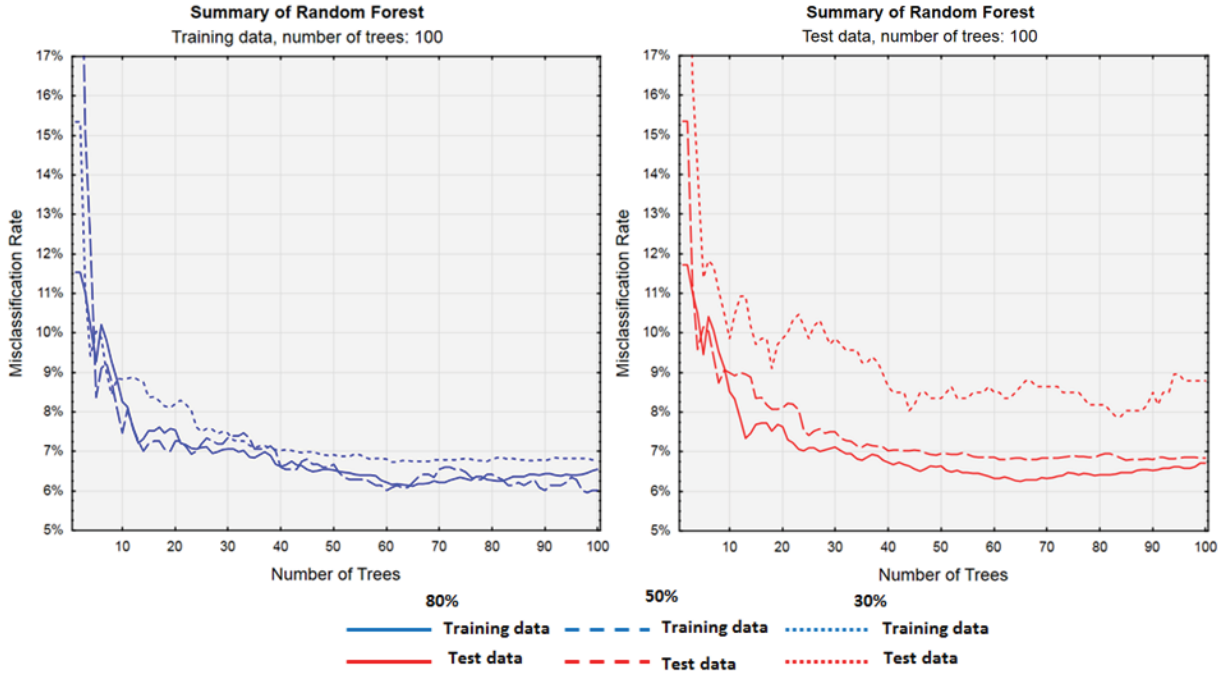


Figure 5.14 Summary of Random Forest analysis using three different training sample sizes to determine the optimal classifier for the spatial data set.

As with k-NN, the 30% and 15% training sample data perform very similarly. There is only a 1% increase in misclassification between the 30% training data set and the 15% training data set. The 7% training sample achieves the lowest overall misclassification rate and also shows the largest error margin compared to the 30% and 15% training samples. The error margin is defined as the discrepancy between training and test sample misclassification rates. As there is only a slightly lower misclassification rate of the 30% training sample, the classifier for the whole data set will be trained using the 15% training sample. The final Random Forest results are shown in section d).

c) Support Vector Machines (SVMs)

As with k-NN and Classification Trees the same three training data sets are used. Table 5.6 shows the different training parameters used to establish the optimal SVM classifier. The 30% training data set performed best with a cross validation accuracy of 98.9% and a class accuracy of 99.2%, followed by the 15% and 7% sample data set. However, there is only a 1.2% increase in cross validation accuracy and a 0.4% increase in class accuracy when moving from a 15% to a 30% training sample. As the increased computational complexity does not justify doubling training sample size, the 15% training data set is used for establishing the classifier for the whole spatial data set.

Table 5.6 SVM training parameters using 30%, 15%, and 7% training samples.

SVM (30% sample training set)		
SVM Type 1	Kernel type: RBF	10688 support vectors
Cross validation accuracy: 98.9%	**Class accuracy:** 99.2%	
SVM vectors per class: 1431 (1), 527 (2), 744 (3), 511 (4), 348 (5), 637 (6), 921 (7), 964 (8), 95 (9), 1868 (10), 1977 (11), 665 (12)		

SVM (15% sample training set)		
SVM Type 1	Kernel type: RBF	7614 support vectors
Cross validation accuracy: 98.7%	**Class accuracy:** 98.8%	
SVM vectors per class: 1461 (1), 457 (2), 454 (3), 224 (4), 509 (5), 371 (6), 599 (7), 365 (8), 62 (9), 1500 (10), 974 (11), 638 (12)		

SVM (7% sample training set)		
SVM Type 1	Kernel type: RBF	185 support vectors
Cross validation accuracy: 95.4%	**Class accuracy:** 95.9%	
SVM vectors per class: 161 (1), 55 (2), 60 (3), 51 (4), 40 (5), 17 (6), 34 (7), 60 (8), 7 (9), 43 (10), 160 (11), 97 (12)		

c) Comparison of classification results

Figure 5.15 shows the classification results and the misclassification and accuracy rates for the three classification methods. Misclassification is calculated based on the comparison to the optimal clustering method (12 cluster, k-Means) which was used to generate the training data.

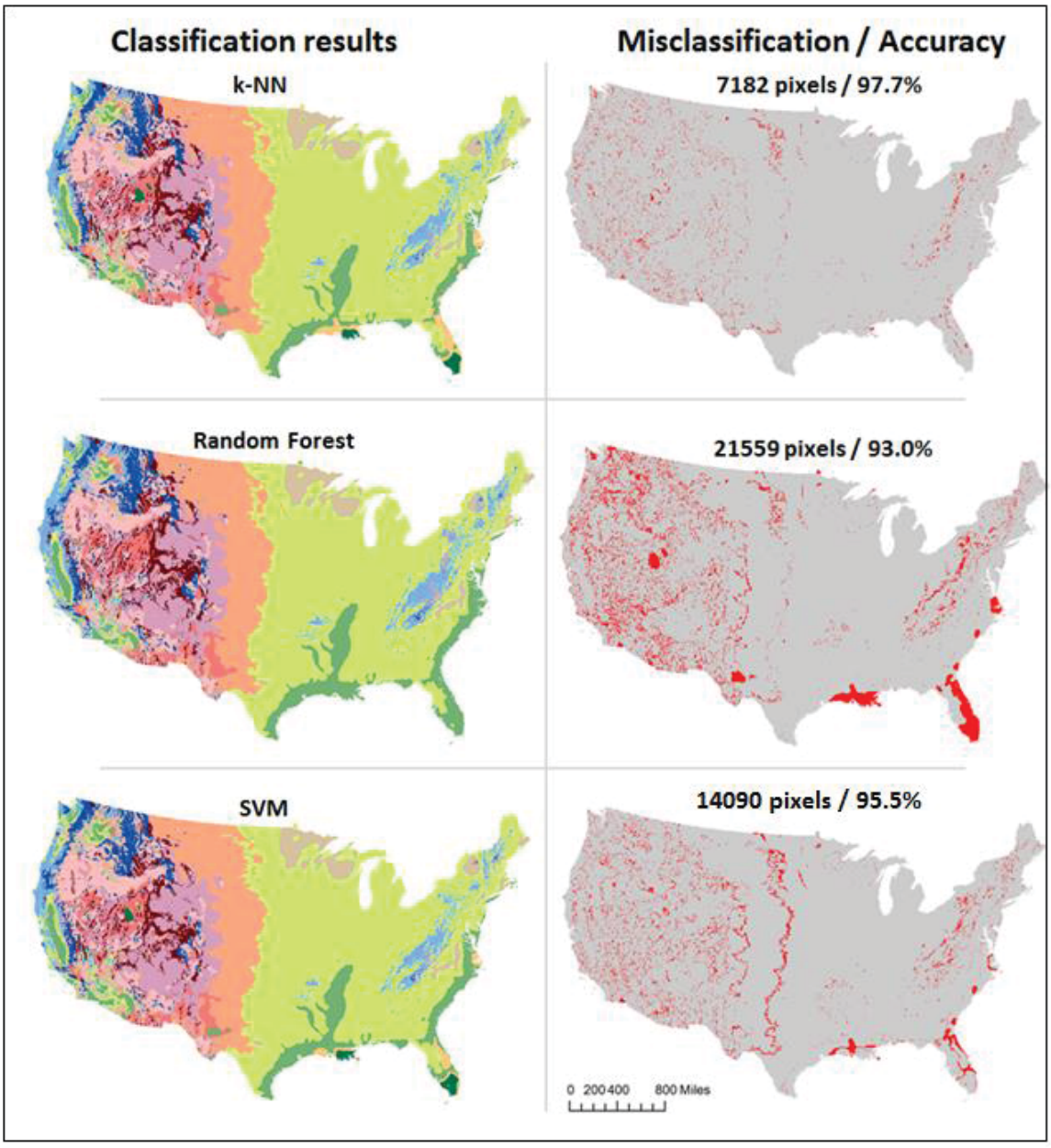


Figure 5.15 Comparison of supervised classification results. Red areas are misclassified.

Regions marked in red on the maps on the right side of the figure show misclassification. k-NN classification produces the best classification. Overall, only 7,182 pixels are misclassified which corresponds to an accuracy rate of 97.7%. Random Forest classification was unable to detect one class completely which is indicated by the larger red area present along the Gulf coast, the Florida Atlantic coast, parts of South and North Carolina, Texas and the Salt Flats in

Utah. These areas are the wettest spots in the country and that all of these regions are falling in the same cluster might be an indication that the inland surface water input variable is not providing sufficient information, relative to other variables, or that more input variables are needed. Besides the missing class, Random Forest also shows higher misclassification throughout the U.S. than k-NN classification. Overall 21,569 pixels are misclassified which corresponds to a total accuracy rate of 93%.The SVM classifier is able to classify 95.5% of the pixels correctly and detects all classes correctly. Only slight misclassification along the borders of each class occurs and this effect reflects the banding effects seen in the unsupervised clustering results. Most misclassification can be found along the coastal plain areas along the Gulf regions as well as a larger band of misclassification along the transition zone between the Great Plains and the High Plains in the middle of the country. Overall, 4,050 pixels are misclassified which corresponds to an overall accuracy of 95.5%.

In conclusion, k-NN neighbor is the best performing supervised classifier for this dataset. The optimal number of classes already determined by k-Means clustering produces well defined clusters. K-Means clustering and k-NN classification are the optimal methods to use for this data set and the results of these methods are an improvement over the original classification.

5.4 GIS Commands

The GIS commands dataset is the smallest data set in this experiment and it is also the only true binary data set in the experiment. The purpose of the clustering is to organize GIS commands into homogeneous groups, as for example might be required for a software command library, or to organize a command table of contents or online help interface.

5.4.1 Unsupervised methods

5.4.1.1 Cluster evaluation and selection

Figure 5.16 shows the evaluation indices for Hierarchical and k-Means clustering. Both clustering algorithms perform in a similar fashion for two metrics, with Hierarchical clustering generating overall lower indices values for the Davies-Bouldin (DB) and Silhouette index. As neither the DB nor Silhouette index show major extrema, the Dunn index will be used for defining an optimal range of clusters based on local extrema and leveling off regions.

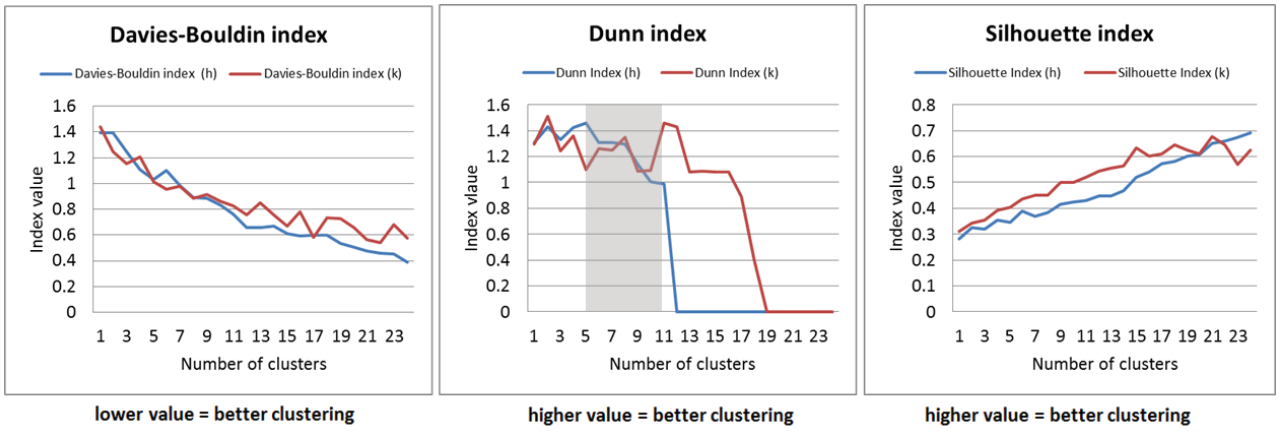


Figure 5.16 Evaluation indices for Hierarchical and k-Means clustering. The area shaded in grey shows the range of optimal clusters based on local minima and leveling off region.

The region for optimal clustering is defined by the highest Dunn index value for Hierarchical clustering at 5 clusters and a local extrema at 11 clusters for k-Means clustering (indicated by the grey box). The Dunn index drops to 0 at 12 clusters for hierarchical clustering and at 19 clusters for k-Means clustering.

a) Cluster stability in Hierarchical clustering

Hierarchical clustering shows remarkable consistency throughout the range of optimal clusters, in comparison to the previous two data sets. A sample of representative GIS commands has been selected to explore cluster stability and formation. The sample includes GIS commands relating to data management, spatial statistics, and flow analysis. Cluster solutions

ranging from 5 to 11 clusters have been created for Hierarchical clustering (Table 5.7). The data management functions "copy", "delete" and "rename" form a single cluster early on, at 5 clusters; and the spatial statistics commands cluster into two groups as well. The hydrological flow commands remain stable through 8 clusters. For more than 8 clusters, illogical groupings and inconsistencies begin to appear. For example, the commands "fill" and "flow accumulation" which are both flow analysis commands separate into 2 clusters. In the 12 cluster solution, the data management functions separate into two clusters indicating semantic instability at this level of clustering.

Table 5.7 Cluster membership assignment for GIS commands using Hierarchical clustering

	Number of clusters						
GIS command name	**11**	**10**	**9**	**8**	**7**	**6**	**5**
copy	5	5	5	5	5	4	1
delete	5	5	5	5	5	4	1
rename	5	5	5	5	5	4	1
fill	7	7	7	3	3	2	2
flow accumulation	3	3	3	3	3	2	2
spatial statistics clustering	9	8	1	1	1	1	1
spatial statistics hot spots	8	6	6	6	5	4	4
spatial statistics mean center	8	6	6	6	5	4	4
spatial statistics Moran	9	8	1	1	1	1	1
spatial statistics nearest neighbor	8	6	6	6	5	4	4

b) Cluster stability in k-Means clustering

Table 5.8 shows the cluster memberships after multiple k-Means clustering solutions have been applied. As with Hierarchical clustering, the data management and spatial statistics commands stabilize early on into one and two clusters respectively. The flow analysis commands destabilize at 7 clusters and then temporarily form a single group for the 10 cluster solution, after which they separate again into two groups. As compared to the Hierarchical clustering results, k-Means clusters are less stable throughout the range of optimal classes.

Table 5.8 k-Means cluster membership stability and formation from 5 to 11 clusters. Cluster formation is independent from those shown in Table 5.6.

	Number of clusters						
GIS command name	**11**	**10**	**9**	**8**	**7**	**6**	**5**
copy	6	6	6	6	6	6	4
delete	6	6	6	6	6	6	4
rename	6	6	6	6	6	6	4
fill	11	5	7	7	3	3	3
flow accumulation	7	5	3	8	7	3	3
spatial statistics clustering	1	1	1	1	1	5	5
spatial statistics hot spots	5	7	5	7	3	1	2
spatial statistics mean center	5	7	5	7	3	1	2
spatial statistics Moran	1	1	1	1	3	1	2
spatial statistics nearest neighbor	5	7	5	7	3	1	2

c) Cluster stability in Self-Organizing Maps

The linear SOM clustering of the GIS commands dataset shows similar results as seen with k-Means clustering. Table 5.9 shows the cluster memberships after multiple SOM clustering solutions have been applied. The data management functions "copy", "delete" and "rename" form a single cluster throughout the whole range of clusters. The spatial statistics commands cluster into two groups as well; however the GIS command "spatial statistics clustering" seems to jump clusters early on before stabilizing at 8 clusters. The hydrological flow commands remain stable through 8 clusters. Identical to the hierarchical clustering results, for more than 8 clusters, illogical groupings and inconsistencies begin to appear. For example, the commands "fill" and "flow accumulation" which are both flow analysis commands separate into 2 clusters. Overall the linear SOM clustering of this dataset produces more stable results that k-Means clustering.

Table 5.9 SOM cluster membership stability and formation from 5 to 11 clusters. Cluster formation is independent from those shown in Table 5.7 and 5.8.

	Number of clusters						
GIS command name	**11**	**10**	**9**	**8**	**7**	**6**	**5**
copy	1	1	9	1	1	6	5
delete	1	1	9	1	1	6	5
rename	1	1	9	1	1	6	5
fill	8	7	4	6	5	2	1
flow accumulation	8	8	3	6	5	2	1
spatial statistics clustering	6	5	6	4	4	3	3
spatial statistics hot spots	6	6	5	5	4	3	2
spatial statistics mean center	6	6	5	5	4	3	2
spatial statistics Moran	5	5	6	4	3	4	3
spatial statistics nearest neighbor	6	6	5	5	4	3	2

5.4.1.2 Clustering results

Hierarchical clustering generated more stable clusters than k-Means clustering and is used as the clustering method for this data set. Linear SOM clustering nearly produced identically results. However, due to the increased computational intensity hierarchical clustering is used as the optimal clustering method. Figure 5.17 shows the dendrogram visualization for the GIS commands. Each cluster has been analyzed and labeled. Cluster labels for the 8 cluster solution are displayed in red. The selected GIS commands, presented in Table 5.6 and 5.7, are labeled in black.

Hierarchical clustering returns clusters which are semantically logical. Similar commands group together. For example, the data management functions group together as do commands for aggregation, interpolation and zonal statistics. The terrain and flow functions group into a single cluster. Spatial statistics functions fall into three different clusters, one constituted by analysis of geometry and neighborhood functions; a second by distance based functions such as Moran's I, and the third formed by local functions.

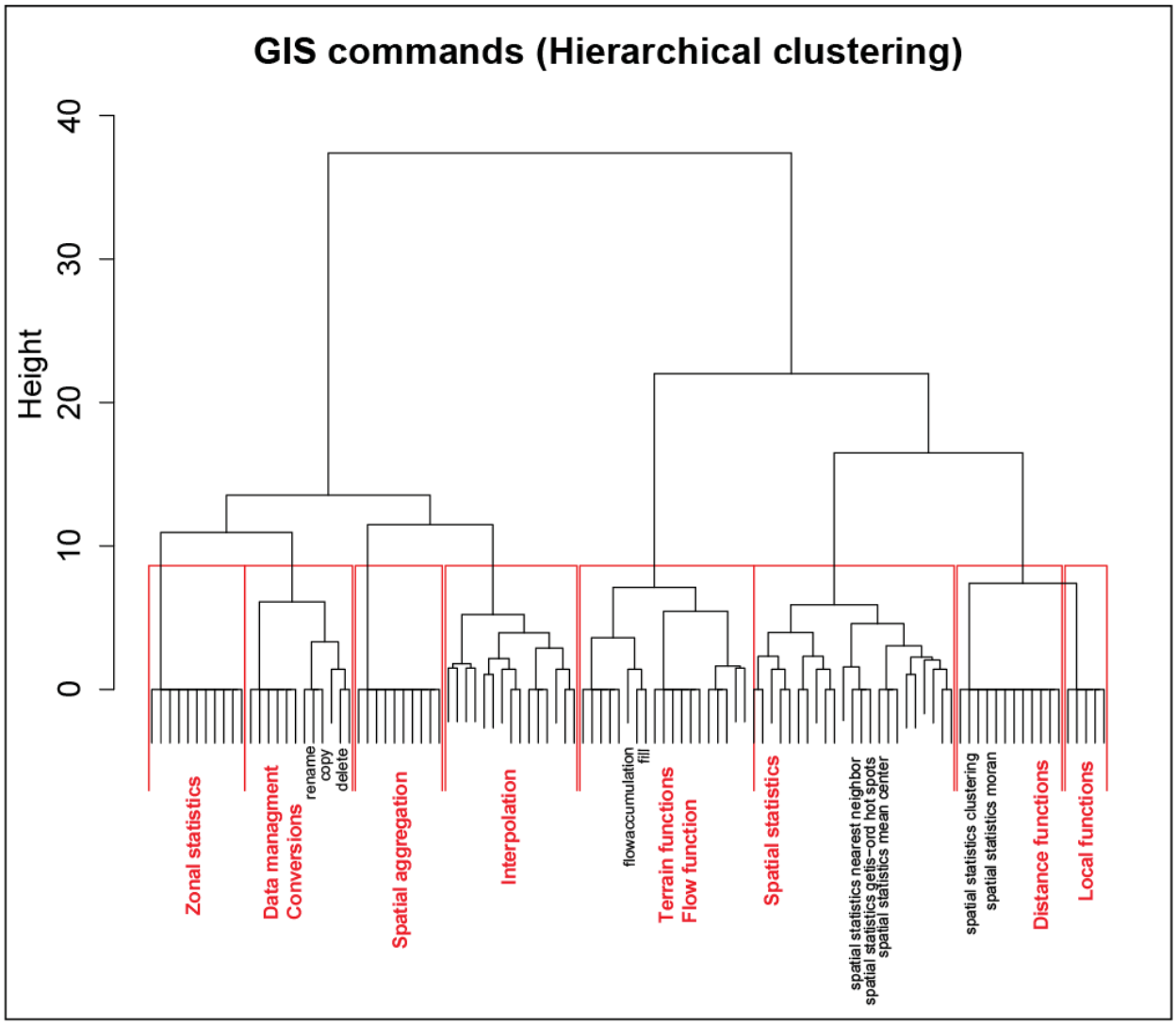


Figure 5.17 GIS commands dendrogram, clusters are shown in red.

For clustering the GIS commands data set, Hierarchical clustering tends to generate more stable clusters than k-Means clustering, and moreover, cluster stability is established with fewer clusters than with k-Means clustering. k-Means on the other hand generates well separated and compact clusters as indicated by the Silhouette index. The recommended clustering method to use for this data set and purpose of this clustering is Hierarchical clustering using 8 clusters.

As before, SOM does not provide crisp clustering by itself. However, the SOM algorithm has the advantage that relationships between GIS commands can be explored as distances between data objects.

5.4.2 Supervised methods

a) k-NN

Figure 5.18 shows the cross validation accuracy for 30%, 50%, and 80% training samples derived from the optimal clustering solution.

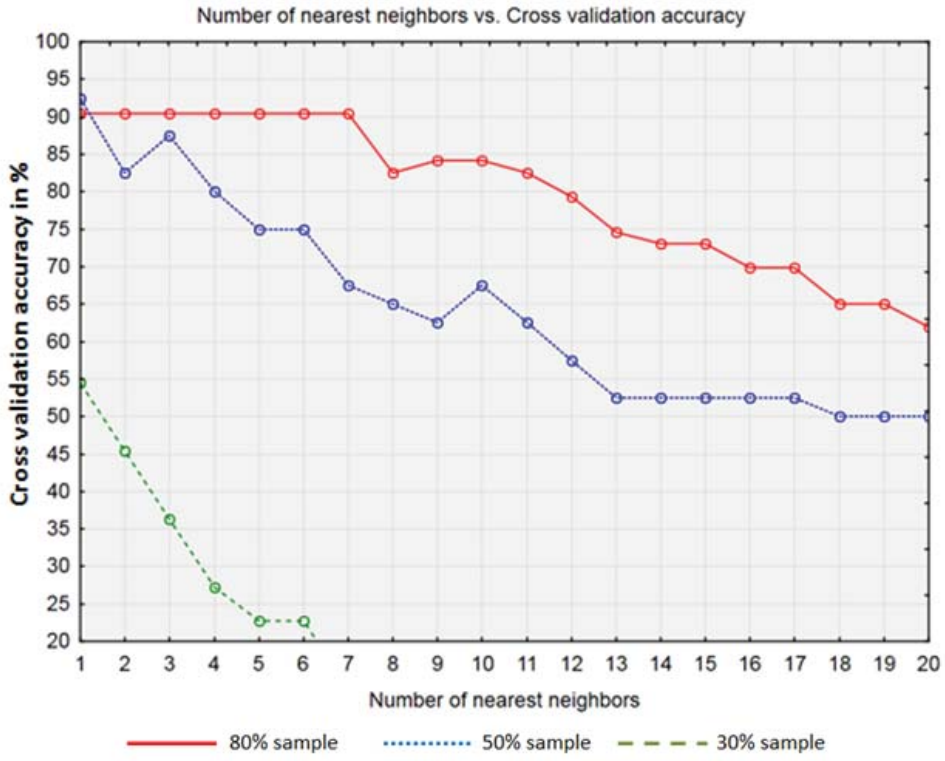


Figure 5.18 k-NN analysis of selecting the optimal number of k for multiple training data sets

The highest accuracy (93.8 %) is achieved at 1 nearest neighbor with the 50% training data set. For more than 1 neighbor, the cross validation accuracy for the 50% training sample drops steadily. The 80% training sample in contrast shows more stable results over a wider range of k values (1 to 7 nearest neighbors). For more neighbors, the cross validation accuracy starts to drop. The 30% training sample shows the least accuracy and only delivers a maximum accuracy rate of 55% which is due to the small data set size. The 30% training data will not be used in resuming classification methods for this data set. These findings are in part due to the very

small size of the data set. The 80% training data set is used for classifier training as it achieves the most stable cross validation results for the largest range of k values.

b) Classification trees

Figure 5.19 shows the Random Forest misclassification rates spanning 100 random trees for the three sample training sizes.

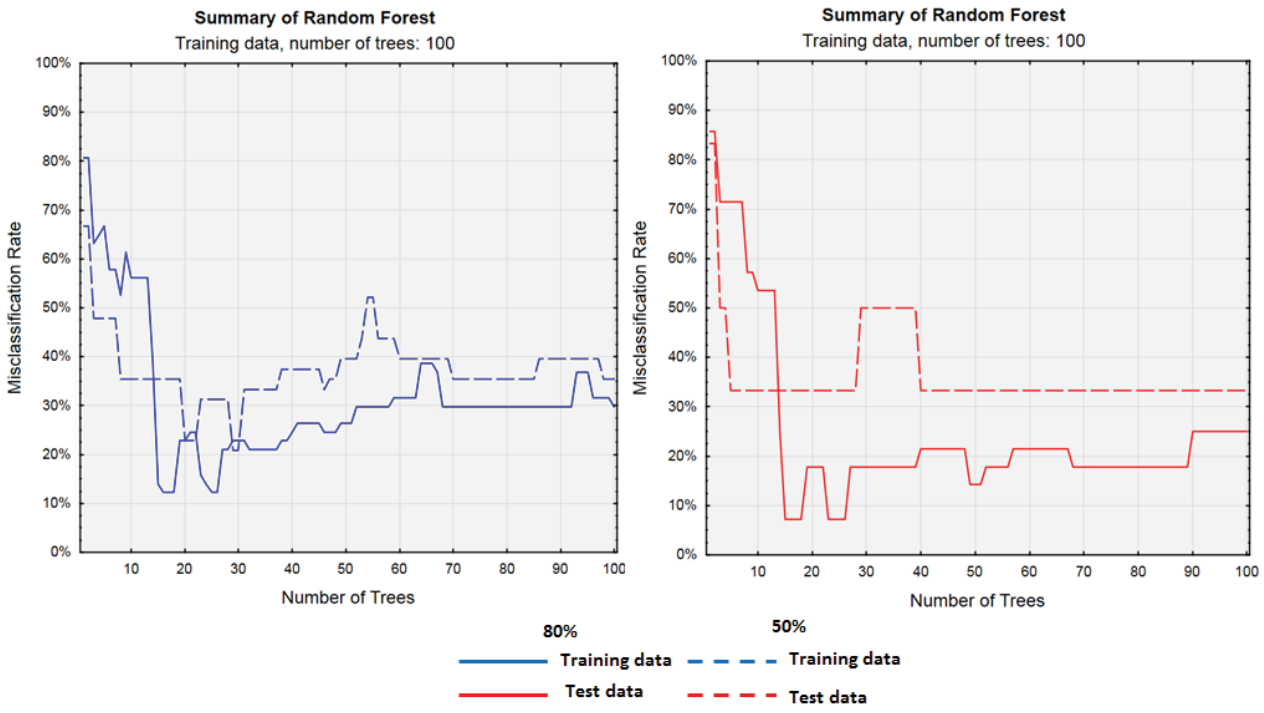


Figure 5.19 Summary of Random Forest analysis using two different training sample sizes.

Due to the small data set size only the 80% training data set is able to produce meaningful classification trees. The 80% trainings sample with a lowest misclassification rate of 8% for the test data set and a misclassification rate of 12% for the training data is used to train the classifier for the whole data set. The 50% sample exhibits a lowest misclassification rate of 34% for the test data set and 23% for the training data set. Due to its poor performance, the 30% training set was dropped from the analysis.

c) Support Vector Machines

Table 5.10 shows the different training parameters used to establish the optimal SVM classifier.

Table 5.10 SVM parameters and statistics for the two training data sets.

SVM (80% sample training set)		
SVM Type 1	Kernel type: RBF	45 support vectors
Cross validation accuracy: 93.6%	**Class accuracy:** 100.0%	
SVM vectors per class: 7 (1), 7 (2), 5 (3), 4 (4), 7 (5), 7 (6), 6 (7), 2 (8)		

SVM (50% sample training set)		
SVM Type 1	Kernel type: RBF	28 support vectors
Cross validation accuracy: 90.0%	**Class accuracy:** 98.1%	
SVM vectors per class: 4 (1), 4 (2), 6 (3), 2 (4), 6 (5), 3 (6), 2 (7), 1 (8)		

From the training experiment is can be seen that the 80% training data set performed best with a cross validation accuracy of 93.6% and an overall class accuracy of 100%. Again, this is probably a consequence of the small size of the GIS commands data.

d) Comparison of classification results

Table 5.11 shows the classification results of all three methods compared to the optimal clustering solution (Hierarchical clustering, 8 clusters). k-NN and SVM were able to classify most all GIS commands correctly. For k-NN, only 3 GIS commands ("intersection", "spatial statistics centroid", "streamlink") got misclassified which corresponded to an overall accuracy rate of 97.2%. SVM achieved the highest accuracy level as well as lowest misclassification rate with only 1 misclassified GIS command ("generalize") which corresponds to a total accuracy rate of 99.6% (Appendix B). The Random Forest classification performed more poorly with 42 commands misclassified in the whole data set which corresponds to an overall accuracy rate of only 63%. In summary, SVM is the best performing supervised classifier for this dataset.

Table 5.11 Classification results compared to the optimal clustering results for the selected GIS commands summarized for unsupervised grouping. Misclassified commands are shown in red.

GIS command name	Hierarchical clustering	K- Nearest Neighbor	Random Forest	SVM
copy	5	5	5	5
delete	5	5	5	5
rename	5	5	5	5
fill	3	3	3	3
flow accumulation	3	3	3	3
spatial statistics clustering	1	1	1	1
spatial statistics hot spots	6	6	4	6
spatial statistics mean center	6	6	4	6
spatial statistics Moran	1	1	1	1
spatial statistics nearest neighbor	6	6	4	6

5.5 Algorithms

The cartographic algorithm data set is the last data set in this experiment and forms the continuum of indexability described earlier in this dissertation. The purpose of grouping the algorithms data set is to separate cartographic generalization articles from the complete article data set, and to organize similar generalization papers into homogeneous groups, as for example in preparation for distributing an online knowledge base on generalization algorithms, or to contribute to a shared information exchange on the topic.

5.5.1 Unsupervised methods

5.5.1.1 Cluster evaluation and results

As before, clustering is first evaluated by local extrema as well as leveling off regions. Figure 5.20 shows the validation indices for Hierarchical and k-Means clustering.

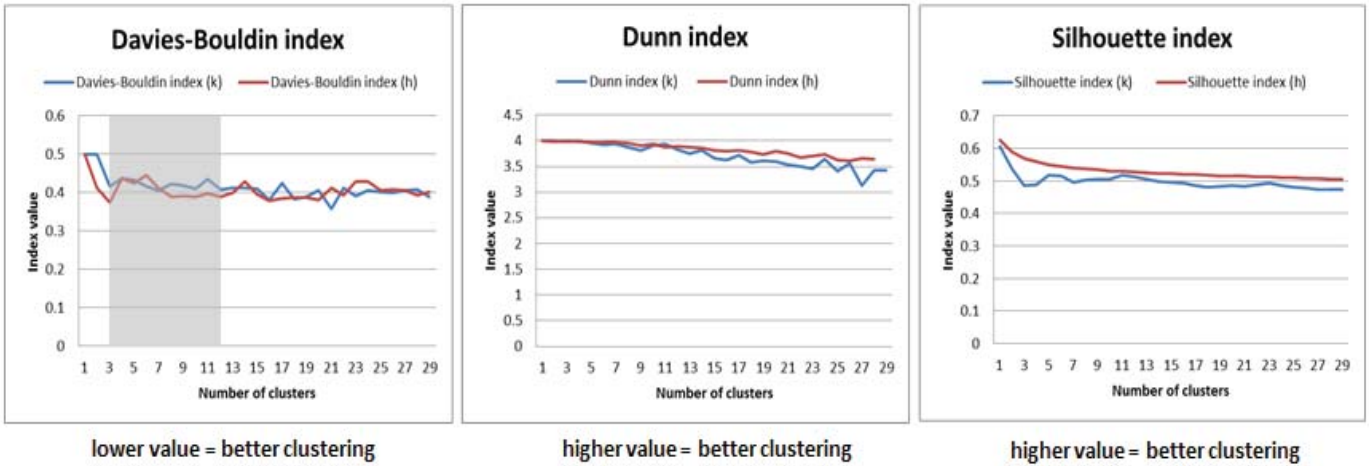


Figure 5.20 Evaluation indices for the algorithm data set.

In contrast to the previous data sets, the progression of cluster indices for k-Means and Hierarchical clustering is nearly identical for all three evaluation metrics. From the progression of the Dunn and Silhouette index it is not possible to define a region of optimal clustering. The Davies-Bouldin index is the only index which shows local extrema and as such will be used to establish a region for optimal clustering. The region for optimal clustering spans from 3 to 12 clusters as indicated by the grey box. At 3 clusters, both k-Means and Hierarchical clustering reach local minima after a steep drop. At 12 clusters, the DB for Hierarchical clustering rises before it starts to fluctuate which can be described as an instability in clustering solutions.

a) Cluster stability in Hierarchical clustering

Nine articles have been selected to explore cluster stability.. Articles by Cromley, Ratschek, Burghardt, and Buttenfield have been choosen to show a representative sample for simplification specific cartographic publications. The paper by Li is selected as a representative sample for Web generalization. Two papers by Visvalingam represent line feature modeling, and papers by Mason and Xie have been chosen as a representative sample of interpolation.

As before with the full text data set, the title and abstract of the papers can be found by the ID in the Appendix section (Appendix C). Clusters over a range of 3 to 16 clusters, indicated by prior analysis, are evaluated for stability (Table 5.10).

Table 5.12 Cluster membership assignments for Hierarchical clustering.

			Number of clusters													
Topic	Author	ID	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Simplification	Cromley	289	2	4	5	6	7	8	8	9	10	11	12	12	12	12
	Ratschek	1137	2	2	2	2	3	3	3	3	3	3	3	3	3	3
	Burghardt	182	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Buttenfield	187	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Web generalization	Li	827	1	1	1	1	1	1	2	2	2	2	2	2	2	2
Line generalization	Visvalingam	1413	2	2	2	2	3	3	3	3	6	6	6	13	14	14
		1414	2	2	2	2	3	3	3	3	6	6	6	6	6	6
Interpolation	Mason	330	2	2	2	2	2	2	4	4	4	8	8	10	10	10
	Xie	1490	3	4	4	4	4	4	4	4	4	8	9	9	9	9

As indicated in Table 5.12, nine clusters were selected as the optimal number of clusters. Addition of more clusters does not modify the grouping, except for the papers by Visvalingam, Mason and by Xie. In the 9 cluster solution, it can be observed that for example, both interpolation algorithms fall into one cluster as well as the web generalization article by Li gets separated from the simplification algorithms and forms its own cluster. Throughout the whole range of cluster solutions the simplification themed articles are split into 3 different clusters. The reason for this is that, different keywords are used to describe the simplification methods. The line generalization themed articles by Visvalingam (1413, 1414) stay stable in one cluster till 14 clusters. Furthermore the article by Ratschek (1137) is placed in cluster 3 together with line generalization articles by Visvalingam due to similar keywords describing the simplification algorithms.

b) Cluster stability in k-Means clustering

Table 5.13 shows cluster solutions ranging from 3 to 16 clusters.

Table 5.13 Cluster membership assignments from k-Means clustering.

			Number of clusters													
Topic	Author	ID	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Simplification	Cromley	289	3	2	5	5	6	6	4	8	11	11	1	1	10	9
	Ratschek	1137	3	2	4	3	6	3	7	8	6	2	10	2	6	8
	Burghardt	182	3	2	4	3	6	3	7	8	6	2	7	2	11	16
	Buttenfield	187	2	4	3	6	4	1	1	9	9	5	6	3	1	6
Web generalization	Li	827	2	4	3	3	6	1	9	4	6	3	4	10	5	4
Line generalization	Visvalingam	1413	3	2	4	3	6	3	2	8	6	2	7	2	13	3
		1414	3	2	4	3	6	3	7	8	6	2	7	2	4	3
Interpolation	Mason	330	3	2	4	3	6	3	9	7	10	7	4	10	5	4
	Xie	1490	3	2	2	1	3	4	3	7	10	7	5	4	14	7

Ten clusters were selected as the solution for which cluster stability is achieved. By ten clusters, the simplification algorithm themed articles start to form two clusters before breaking up into individual clusters. The article by Buttenfield is separated from the other simplification themed articles due to the use of different vocabulary as this article is describing the special case of simplification based on physiographic regions which uses different terminology than the other simplification articles. The web generalization article by Li starts to form its own cluster at ten clusters. However, at lower cluster numbers this article jumps clusters between the simplification and interpolation themed cluster. The line generalization articles by Visvalingam start to form clusters with simplification themed algorithms at 10 clusters and stays stable to 13 clusters, which is where the Davies-Bouldin index starts to fluctuate widely. The interpolation algorithm themed articles sit in a single group at 10 clusters but break into two groups for 13 to 16 clusters. This also corresponds with the fluctuating Davies-Bouldin values for that range. The full data set with all cluster memberships is shown in Appendix C.

c) Self-Organizing Maps (SOM)

The linear SOM clustering of the algorithm dataset shows similar fluctuating results as seen with k-Means clustering. Table 5.14 shows the cluster memberships after multiple SOM clustering solutions have been applied. Throughout the whole range of clusters the four simplification articles are split into their own clusters. The web generalization algorithm by Li starts forming its own cluster at 10 clusters. As seen before with k-Means clustering at lower cluster numbers this article jumps clusters between simplification and interpolation themed clusters. The line generalization articles by Visvalingam are separated between 3 to 9 and 13 to 16 clusters but form one cluster and stabilize between 10 and 13 clusters. The same is true for the interpolation algorithms themed papers which are split into different clusters but form one cluster between 10 and 11 clusters and between 15 and 16 clusters.

Table 5.14 SOM cluster membership stability and formation from 3 to 16 clusters.

			Number of clusters													
Topic	Author	ID	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Simplification	Cromley	289	1	3	3	4	6	5	7	6	6	6	7	9	7	10
	Ratschek	1137	1	4	4	3	4	3	5	4	4	8	8	6	11	6
	Burghardt	182	3	3	1	6	1	8	1	10	11	1	13	1	1	16
	Buttenfield	187	3	3	1	6	7	7	8	1	9	3	5	10	3	14
Web generalization	Li	827	3	3	1	6	7	7	9	9	9	3	1	2	3	14
Line generalization	Visvalingam	1413	1	1	4	3	4	3	5	2	3	9	11	6	11	3
		1414	1	3	5	2	3	2	4	2	3	9	11	4	13	5
Interpolation	Mason	330	1	1	4	2	4	2	6	8	8	10	5	10	10	12
	Xie	1490	3	2	2	5	7	6	9	8	8	4	1	14	10	12

5.5.1.2 Clustering results

As indicated by cluster stability analysis in the prior section, Hierarchical clustering generates the most stable clusters. K-Means and linear SOM clustering show similar fluctuations in cluster stability while k-Means produces slightly more stable cluster results. Hierarchical clustering is used for cluster visualization and as an input for deriving training

data sets for the classification methods. Figure 5.21 shows the dendrogram visualization for the algorithm data set. Cluster size varies for the data set, with simplification algorithms forming the largest cluster, separated from line generalization algorithms at the lowest clustering level. The dendrogram can be divided into two groups at the highest clustering level, with publications focused on algorithms for GIS applications on the left side and cartographically themed generalization algorithms on the right side. On the cartographic generalization side, seven clusters emerge, dividing generalization algorithms into specific methods ("interpolation", "classification", and "simplification") and specific data types ("raster", "network", "shape", "points", and "line"). The simplification cluster shows strongly hierarchical patterns. Lower levels in this sub-hierarchy are partition into e.g. "smoothing", "coordinate reduction", or "collapse".

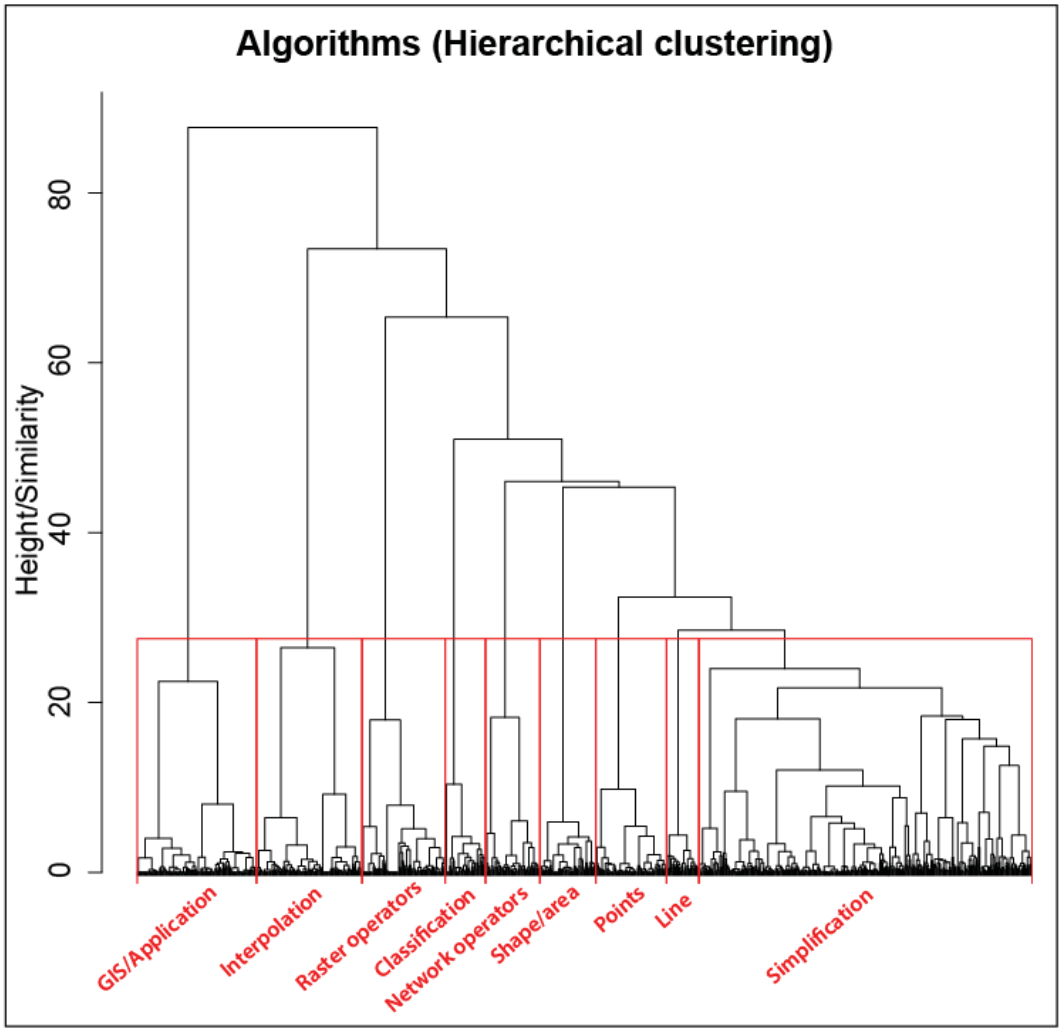


Figure 5.21 GIS commands dendrogram, clusters and cluster labels are shown in red.

Hierarchical clustering is able to differentiate well between the target publications. It also produces stable clustering results for a range of clusters and is therefore used as the preferred clustering method for this data set.

2.5.2 Supervised methods

a) k-NN

Figure 5.22 shows the cross validation accuracy for the three different training samples for the algorithms data set. The training data set is derived from the optimal classical clustering methods described in the previous section.

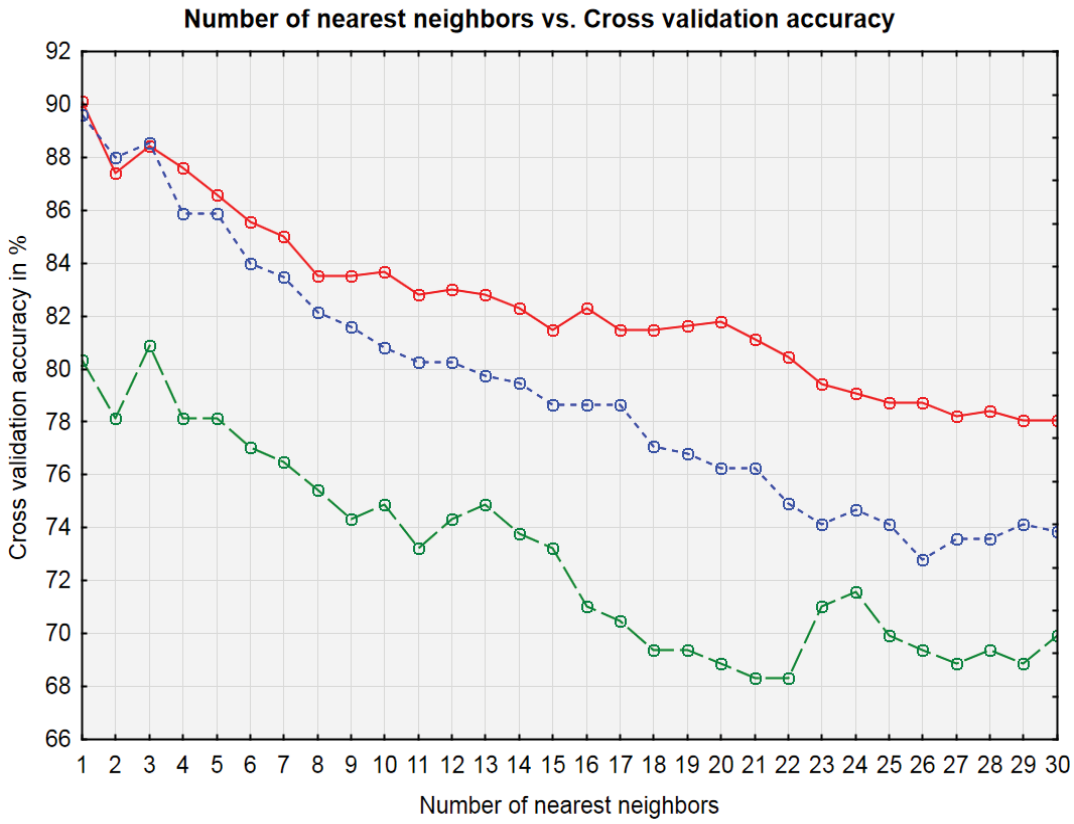


Figure 5.22 Cross validation accuracy of the three training samples for the algorithms data set.

It can be seen from Figure 5.26 that between 1 to 8 nearest neighbors the 80% and 50% training data sets perform in a similar way, with 80% training sample performing at higher accuracy overall. The 80% training data set performs best at k=1 with a cross validation accuracy of 90.7%, followed by the 50% training data set with a cross validation accuracy of 90.4% at k =1. At 3 nearest neighbors, both the 80% and the 50% training data set achieve the same accuracy value of 88.4%. The 30% training data set performed worst with a maximum cross validation accuracy of 80.8% at 3 nearest neighbors. As both the 80% and 50% perform very similarly, the 50% training data set is used for classifier training of the whole data set as the accuracy gain of less than 1% does not justify the increased computational complexity in establishing the classifier.

b) Classification trees

Figure 5.23 shows the misclassification rates for a Random Forest classifier across the three training sample sizes.

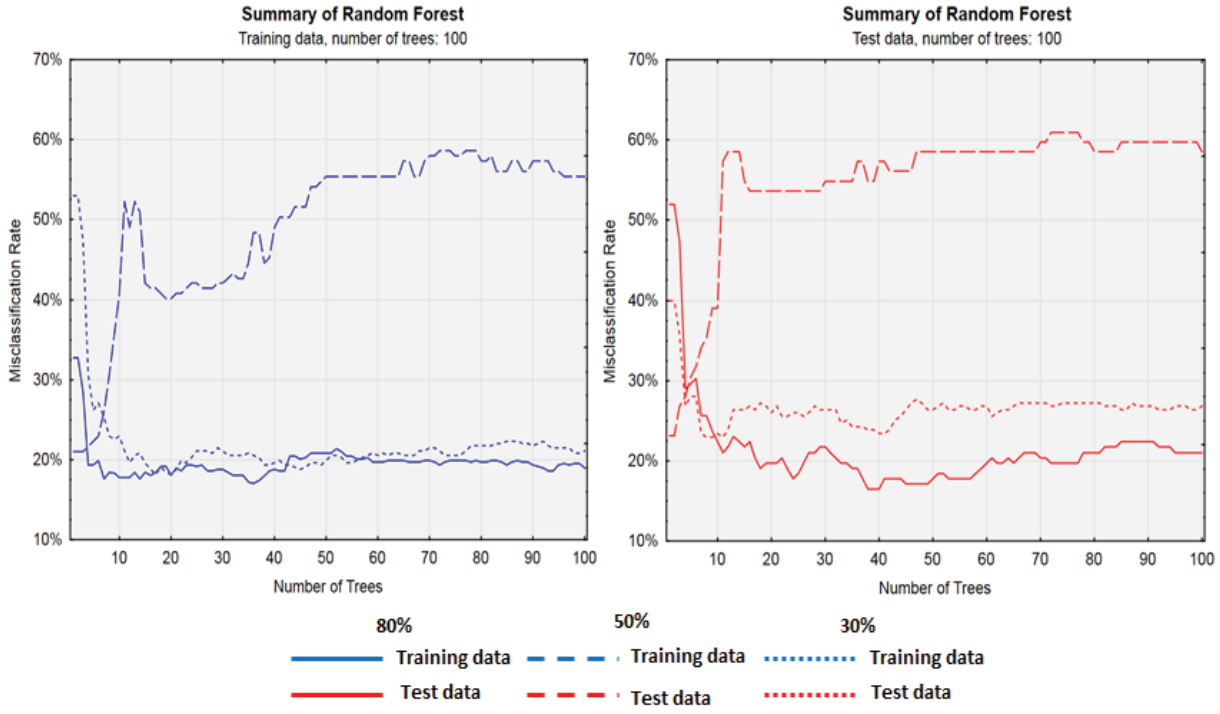


Figure 5.23 Summary of Random Forest analysis using three different training sample sizes.

From Figure 5.27 it can be seen that the 80% training sample performs best, displaying the lowest misclassification rate of 18% for the test and 16% for the training data set. The 50% training sample performs with a lowest misclassification rate of 19% for the test data and 23% for the training data. The 30% training sample never achieves a better misclassification rate than 20% for both the training and test data. As the 80% training sample achieves the lowest misclassification rate, it is used to train the classifier.

c) Support Vector Machines

Table 5.15 shows the training parameters that establish the optimal SVM classifier.

Table 5.15 SVM parameters and statistics for the 80% and 50% training data sets.

SVM (80% sample training set)		
SVM Type 1	Kernel type: RBF	340 support vectors
Cross validation accuracy: 87.5%	**Class accuracy:** 88.3%	
SVM per class: 33 (1), 41 (2), 35 (3), 113 (4), 40 (5), 28 (6), 15 (7), 30 (8), 5 (9)		

SVM (50% sample training set)		
SVM Type 1	Kernel type: RBF	241 support vectors
Cross validation accuracy: 86.7%	**Class accuracy:** 88.1%	
SVM per class: 27 (1), 25 (2), 19 (3), 26 (4), 82 (5), 29 (6), 14 (7), 17 (8), 2 (9)		

SVM (30% sample training set)		
SVM Type 1	Kernel type: RBF	122 support vectors
Cross validation accuracy: 78.2%	**Class accuracy:** 79.9%	
SVM per class: 36 (1), 16 (2), 14 (3), 8 (4), 12 (5), 11 (6), 12 (7), 12 (8), 1 (9)		

From the training experiment is can be seen that the 80% training data set performed best with a cross validation accuracy of 87.5 % and an overall class accuracy of 88.3%. As there is only a 0.8% increase in cross validation accuracy and a 0.2% increase in class accuracy by moving from the 50% training data set to the 80% training data set, the 50% training data set will be used for training of the SVM classifier as the small increase in accuracy does not justify the increased computational complexity. As before, the 30% training data set performed worst overall with a cross validation accuracy of 78.2% and an overall class accuracy of 79.9%.

d) Comparison of classification results

Table 5.16 shows the classification results of all three methods compared to the optimal clustering solution. k-NN only misclassified one target algorithm (827) from Li on web

generalization, placing it in the same class as most simplification algorithms. Overall k-NN was able to correctly classify 899 journal papers which correspond to an accuracy rate of 91.8% for the whole data set. Three target algorithms (181, 827 and 330) were misclassified by Random Forest and placed into class 1 and 3. Random Forest was only able to classify 761 journal papers correctly for the whole data set achieving an accuracy rate of 77.8%. SVM was able to correctly classify 832 journal papers which correspond to an accuracy rate of and a total accuracy rate of 85.1% for the whole data set. Class memberships for the whole data set can be found in Appendix C. Overall k-NN performed best and is the optimal supervised method to choose for this data set.

Table 5.16 Classification results compared to the optimal clustering results.

Topic	Author	ID	Hierarchical clustering	K-NN	Random Forest	SVM
Simplification	Cromley	289	8	8	8	8
	Ratschek	1137	3	3	3	3
	Burghardt	182	1	1	3	3
	Buttenfield	187	1	1	1	1
Web generalization	Li	827	2	1	1	3
Line generalization	Visvalingam	1413	3	3	3	3
		1414	3	3	3	3
Interpolation	Mason	330	4	4	3	4
	Xie	1490	4	4	4	4

5.6 Summary of the grouping experiment

The grouping experiment applied clustering and classification methods to organize the four exemplar data sets. Depending on the data set and the indexing method applied, some clustering and classification methods performed better than others. Each clustering method has been evaluated by the purpose of the clustering, local extrema, leveling off regions and cluster stability for each data set. While local extrema and leveling off regions were derived mathematically by common cluster evaluation methods, cluster stability was assessed in an exploratory fashion meaning that only the selected target data objects for each data set have

been manually evaluated for stability. Specifically, cluster stability was assessed thorough changes and shifts in cluster membership for the range of all cluster solutions for each data set. Techniques for formally addressing cluster stability have been proposed by Albatineh et al. (2006) but have not been implemented in this experiment.

Table 5.17 shows all four data sets and the optimal clustering and classification methods applied as well as the choice of number of clusters for each data set. The reader should keep in mind that the number of clusters and classes for each data set was given by the criteria set for clustering, for purposes of comparison.

Table 5.17 Overview of all data sets and methods used in this experiment.

Data set	Optimal clustering method	Optimal classification method	Number of classes
Full text (Cartographic publications)	Hierarchical clustering	k-Nearest Neighbor	10
Spatial (Physiographic regions)	k-Means	k-Nearest Neighbor	12
GIS Commands (Hydrological analysis)	Hierarchical clustering	SVM	8
Algorithms (Cartographic generalization)	Hierarchical clustering	k-Nearest Neighbor	9

Discussion of the results and findings of this experiment in regards to the research questions asked will be presented in Chapter 6. Furthermore the next chapter will explore limitations and further extensions of the results and methods presented in this experiment.

CHAPTER VI

Discussion of results

This chapter is divided into four sections. The first section provides a summary of the results from the indexing and grouping experiment; and the second discusses the results in regard to the research questions asked in this dissertation. The third section develops a critique of the experimental design and outlines possible enhancements that could be made in future research. The concluding section will summarize the findings of the dissertation.

6.1 Results from the organization experiment

The discussion of results focuses on the general overview of the experiment and provides a synoptic view of the outcomes of each data set before the results are discussed in regard to the research questions posed.

This dissertation developed a set of experiments to evaluate the suitability of classical and modern methods of grouping for differently indexed data types commonly found in geographic analysis. Multiple evaluation indices for unsupervised and supervised methods have been implemented in this research. Guidelines on optimal cluster selection have been compiled into four criteria guided by recommendations found in the literature (Everitt et al., 2001; Jain, 2009; McDavid et al., 2011; Rendón et al., 2011) and have been applied to all four data sets. The compiled criteria for unsupervised clustering include the purpose of the grouping task, the local extrema and leveling off regions for selecting an appropriate number of clusters (and classes), and the semantic stability of the groups which form. Supervised classification was guided by using differently sized training data sets as well as assessment of misclassification and accuracy levels by cross validation. While estimation studies of different clustering and classification methods have previously been reported, this experiment was designed to compare the effectiveness of grouping methods across data sets and different indexing strategies that have not been reported previously in the literature. Information gained for best usage of those

methods applied on different data sets is especially valuable to domain and cross-domain Ontology research as recommendation for best methods usage for data and multi data type systems can be given which will be relevant in developing future Ontology and information systems. Before diving into the discussion of the results from this experiment in regard to the research questions asked, the following paragraphs give a concise overview of the results of the four data sets from the experiment in this dissertation.

The data sets were chosen to span the spectrum of indexability ranging from fully automatic indexing to manual keyword generation. The full text data set was automatically indexed by text stemming methods without human intervention. Hierarchical clustering performed best as a method for clustering, while k-NN achieved the lowest misclassification rate and the highest accuracy level for supervised classification, using a 50% training data sample. The spatial data set, the largest data set in this experiment, was indexed on the raw data values, also without human intervention. Overall k-Means performed best for clustering and k-NN performed best for classification using a 15% training data sample. However, SVM classification was able to produce similar results while being more computationally efficient than the k-NN method. The GIS commands data set was the only manually indexed and binary data set in this experiment. Hierarchical clustering performed best as a method of clustering and SVM performed best for classification with an 80% training data sample. The algorithm data set was indexed semi-automatically by including manually derived taxonometric generalization keywords into the stemming process. For this data set Hierarchical clustering performed best as a method of clustering and k-NN delivered the optimal classification solution with a 50% training data sample.

The results of the grouping experiment indicate that some methods perform better than others, as indicated by the systematic evaluation of cluster indices and classification measurements summarize above. Overall, for clustering methods, Hierarchical clustering performed well for smaller data sets and data sets with an inherent underlying taxonometric

structure such as the cartographic generalization algorithm data set. Hierarchical clustering stands in contrast to k-Means, which performed well for larger non-hierarchical data sets such as the spatial data set. K-Means also performed better for data sets that could support larger number of clusters across all data sets such as the spatial data set, as indicated by higher and more stable values in the validation indices. Furthermore, the k-Means algorithm performed better for less noisy data sets such as the spatial data set. It can be concluded that in the domain of Geography, k-Means might be best suited for data sets representing natural features such as the spatial data set which was derived from features representing physiographic regions with gradual changes between classes and only a limited number outliers present in the data. These findings are also concluded by similar comparative studies in the literature on cluster evaluation (Budayan et al., 2009; Jain, 2009; Abbas, 2008).

Not all methods can be natively compared directly. For example, the standard SOM algorithm, with the exception of the linear SOM and SOM BMU approaches, does not by itself generate clusters in the sense of strict boundaries as present in Hierarchical clustering and k-Means clustering. Instead of clusters, SOM generates regions in attribute space. Therefore, common cluster indices cannot be applied to SOM. In order to delimit crisp boundaries the SOM is usually clustered by a second method, which was not implemented in this experiment. However, two special cases of SOM where each SOM cell represents one cluster, namely linear SOM and SOM BMU clustering, have been implemented. As indicated by the experiment, both SOM variants achieve similar results as k-Means clustering. In reducing the SOM size to the number of clusters requested, characteristics such as cluster convexity is shared with k-Means clustering. SOM is different from traditional clustering methods in effectively separating patterns form random fluctuations (Wang et al., 2013). This can be observed with the full-text and algorithm data set where SOM clustering performed overall better than k-Means clustering which is sensitive to outliers in the data set.

Classification methods have been evaluated by misclassification and accuracy rates using cross validation. Training data sets were established by the clustering method determined to be optimal. All supervised classification methods can be compared using the same metrics. Across all data sets k-NN and SVM performed better than Random Forest. As indicated by evaluation indices, k-NN established the overall highest accuracy levels. One drawback of the k-NN method is its sensitivity to data set size and to the selection of nearest neighbors. SVM on the other hand is the most stable method in regard to training data size and even with standard parameters performs very well. Furthermore, it is also suitable for large data sets. Random Forest classification overall performed worst. Random Forest is also very sensitive to data set size. Random Forest classification did not perform well on small data sets as it was not able to meaningfully classify the GIS commands data set.

6.2 Discussion of results and answers to the research questions

In order to place findings gained in this experiment into broader perspective this section discusses the results from this experiment in regard to the research questions asked. As already described above, the performance of the methods used in this experiment was highly dependent on the data set and purpose of the grouping.

This experiment was designed to present a representative sample of multiple data sets commonly used in the geographic domain. The results gained from this experiment could support the development of a generalized framework for organizing different types of data by using common methods of indexing and organizing them in a combined fashion as it was presented here. Such a framework is an important innovation on how to master complex data organization problems as well as systematic investigations of different kinds of data in combination. This is particularly important for the deployment of domain and cross-domain Ontologies and ontological applications. Such systems require being flexible and being able to combine multiple domains into one framework (Zablith, 2008). The information gained in this experiment supports studies on optimal usage of organization methods in cross domain

application for multiple data types such as presented in Tang et al. (2012) who conducted a survey on clustering methods for ontological knowledge and Salem and AbdelRahman (2010) who conceptualize work on a multi-domain Ontology builder.

Furthermore, from the experiment presented here it can be concluded that the spatial data set achieved very different results from the rest of the data sets. While still following the proposed continuum of indexability the goal of this data set was different, namely delineating physiographic regions from the seven input data sets. The other data sets, especially the full-text and algorithm data followed a different principle of indexing and grouping which leans more towards the domain of information retrieval. By building such a semantic reference system for each data set, recommendations by data type can be formulated. Recommendations for best usage of grouping methods are extremely beneficial as setting the correct parameters in the clustering and classification process is extremely time consuming. This is particularly relevant in the geographic domain where multiple data types are commonly used together in one analysis. In addition, the findings presented here could be directly applied in formulating information systems which could help build and improve present ontological applications such as the Semantic Web.

Furthermore, as the full-text and algorithm data sets nearly share the same methodologies, both could be combined into one generalized framework and act as a cross-domain reference system by sharing multiple properties as it was suggested by Zablith (2008). Such cross domain application in the field of Geography could be a catalog incorporating multiple data types while sharing the same keywords for indexing. For example, in the field of cartographic generalization such a system could contain both, full text articles and algorithms. As demonstrated in this experiment the same keyword sets with slight modifications could be used to host both data sets in one catalog while sharing common semantics.

The following two sections answer the research questions proposed in Chapter 1 of this dissertation.

1. For a given indexing scheme does a particular organization method link clearly to an indexing method and why?

As indicated by the grouping results in this experiment, the type of indexing can be linked to some extent to the success or failure of a particular organization method. From the grouping experiment, it can be argued that modern methods of clustering and classification will perform better on automatically indexed data, while classical methods work better on manually derived indexing schemes. Furthermore, it can be concluded that data set size and data type dictates the indexing methods which again links to an organization method.

The failure of success of an indexing method is highly dependent on the underlying ontological structure of a data set. For example, Hierarchical clustering performed well on the cartographic algorithm dataset. This is primarily due to the fact that this data set consists of a clear hierarchical structure which is due to the domain it represents and how it was indexed. In the case of cartographic generalization algorithms there is a clear taxonometric hierarchy. For example the group of simplification algorithms can be split into different groups such as line or polygon simplification algorithms and these in turn can break down into different types of line simplification (corridor tolerancing, coordinate weeding, etc.). These groups can then be further divided into sub-groups. This stands in contrast to the spatial data set which is represented by physiographic regions. These natural features show a gradual change in value and when a strict taxonometric hierarchy is applied, misclassification (as indicated by Hierarchical clustering, linear SOM and SOM BMU results) is present.

From this experiment it can be concluded that there is a link between indexing method and organization method. It can be said that the indexing method dictates the ontological structure of a data set which is ultimately responsible for the failure or success of an organizational method as mentioned above.

Furthermore it can be concluded from this experiment that a manually derived indexing scheme which was not derived by crowd-sourcing will most likely generate a relatively small

data set that consists of categorical values. Selecting keywords manually, as demonstrated with the GIS commands data set, resulted in a small binary characterization scheme. Also, manually indexed data sets are most likely to be limited in size due to the manual workload. This stands in contrast to automatically derived indexing schemes that can be scaled to nearly any data set size. Through automatic indexing and term weighting schemes this type of data usually consist of continuous values.

2. What systematic recommendations can be established for organizing data by unsupervised or supervised methods?

The experiment in this dissertation has implemented and provided an evaluation scheme for optimal selection of the clustering parameters. Additionally, by applying multiple methods for clustering and classification, recommendations for an optimal method can be established based on the data type. As already indicated in the answer to the first research question, recommendations can be giving by data type and structure of the data it represents. However, recommendations have to be formulated differently for unsupervised and supervised methods.

As indicated by the experiment, Hierarchical clustering performed well on datasets which have an underlying taxonometric structure which benefits the nature of the clustering algorithm. K-Means performed well on smooth continuous data sets where only a few outliers were present. In contrast, SOM was less sensitive to outliers and therefore outperformed k-Means for the other data sets. Besides the data type, the purpose of grouping is also an important factor which dictates the selection of the number of groups and has to be taken into account. This is particularly important for establishing a framework which can encompass multiple data types. It can be argued that organized data sets with different "levels of detail" (number of groups) will be harder to integrate into one common framework.

Recommendations for supervised methods can be given in regard to training size selection and model parameter selection. Three different training sizes ranging from 30% to 80% were applied with the exception of the spatial data set, for which training data set size ranges from

7% to 30%. As indicated by the experiment carried out in this dissertation, the smallest data set required an 80% training set to produce acceptable classification accuracy. For the medium sized data set a 50% training set was sufficient as indicated by misclassification and accuracy assessment. The largest data set could be successfully trained with a 15% training sample. This stands in accordance with the central limit theorem which states that in order for a sample to accurately represent the population it needs to be of a certain size. Therefore it can be argued that the training size can decrease with an increasing size of the data set.

6.3 Limitation of the experiment

A number of areas in both the indexing and the grouping experiment could be improved. Improvements pertain to validation and organization methods as well as data size selection. The three areas are discussed in the following sections.

a) Evaluation of methods

For determining the optimal number of clusters only internal evaluation methods have been applied. It could be argued that by applying both, internal and external evaluation, more confidence in cluster selection as well as more stable results would have been generated. However, the highly exclusive topic matter of the data sets did not support the use of an external data set for validation. Some of the data sets used, such as the cartographic literature, GIS commands, and algorithm data set are self-compiled and unique in nature. It could not be determined if similar open access data sets exist. While this would be a very interesting area to explore in further research, the time necessary to create similar data sets would be beyond the scope of this dissertation research.

b) Methods of indexing and grouping

This dissertation implemented one indexing strategy for each of the four data sets. The indexing method was chosen based on the assumption that an automatic indexing method is preferred. However, throughout the experiment description, other indexing strategies have

been suggested. This experiment could be extended to implement multiple indexing strategies for each data set. By doing so, multiple indexing schemes by data type could be evaluated by the various grouping methods. Additional recommendations could be established to comparatively evaluate each grouping method for a range of indexing methods applied to a single data type which could lead to more precise recommendation for cross domain systems.

For example, the algorithm data set was indexed using a modified approach of stemming in order to distinguish generalization specific algorithms from other publications in the data set. This process was done by manually including keywords from multiple taxonomies in the stemming process. By doing this, the automatic stemming process was converted into a semi-automatic process with limited human intervention. However, as this data set closes the continuum of indexability, multiple indexing strategies could have been feasible. For example, it would be interesting to explore if this data set would return results similar to those of the GIS commands data set, when indexed by manually created keywords into a binary matrix. This could show whether a data set would be flexible enough to be included in a cross-domain framework were multiple data sets are combined and flexibility is necessary (Zablith, 2008). Furthermore, it would be interesting to explore the stability of a single grouping method by applying multiple indexing strategies per data set. One might postulate that the manually derived keyword set for the GIS commands data set would behave similarly to a larger manually indexed data set. Recommendations for best usage of the grouping methods could be extended by a direct comparison of methods on one data set as well as across data sets.

c) Data set size and relevance

Data set size plays an important role in both indexing and grouping methods. The four data sets, while relatively small in the context of information science or big data, are nevertheless representative of data sets sizes commonly used in the field of GIScience. The type of indexing is determined for each data type and represents all commonly used types including continuous data and binary data. Data set size ranges from small, as represented by the manual indexed

GIS commands data set, to medium represented by the automatically indexed spatial data set. It would also be useful to explore how the findings from this experiment correspond to large data sets commonly used in information science research or in Semantic Web research.

6.4 Future work

This dissertation research could be extended in multiple directions. The first possible extension of the work presented here would be to use the same methodology and apply it to different data sets, preferable larger ones. It would be interesting to explore the scalability of the results presented here. It would be particularity of interest to evaluate the effectiveness of classical and modern methods in regard to increasing data set size. The new data sets could be chosen with more relevance to information systems and Semantic Web research.

A second possible extension of this research project would be to test the flexibility of each indexed and organized data set as already suggested earlier. The experiment only regarded each data set as one entity without exploring the possibility if the developed indexing and organization framework would be flexible enough to be included in a cross –domain organization framework for multiple data types. Findings from such an extension of this research would have a large impact on all relevant areas of information systems research as well as current research on Semantic Web technology. While such a system has been explored conceptually before, no complete evaluation of such an implemented system could be found in the literature.

A third possible area in further extending this research could be the inclusion of more spatial data sets into the analysis. In the current research, only one spatial data set is included which also stands out from the other data sets in its purpose of organization. It would be interesting to explore how spatial data types, other than the used raster data, would correspond to the framework established here and if the findings could be still conclude the same results.

6.5 Conclusion

This dissertation designed and implemented approaches to assess the suitability of commonly used unsupervised and supervised grouping methods on different indexing strategies. Four different data sets commonly used in geographic research have been indexed by fully automatic and manual keyword generation; and different classical and modern methods from clustering and classification have been successfully implemented. Depending on the data set and the indexing method applied, some clustering and classification methods performed better than others.

This dissertation demonstrates that by systematic evaluation of clustering and classification indices, recommendations for organizing data can be formulated by data type. Furthermore, through systematic evaluation and application of the six clustering and classification methods it is possible to link a particular organization method to a specific indexing method. These findings are of use for complex data organization problems and in the development of information systems which encompasses multiple types of data.

References

Abbas, O. (2008). Comparisons Between Data Clustering Algorithms. *The international Arab Journal of Information Technology*. **5** (3): 320 – 325.

Agarwal, P. and Skupin, A. (2008). Self-Organizing Maps: Applications in Geographic Information Science. Chichester, England; Hoboken, NJ, Wiley.

Akerkar, R. and Lingras, P. (2008). Building an Intelligent Web. Sundbury, MA. Jones and Bartlett Publishers.

Albatineh, A. et al. (2006). On Similarity Indices and Correction for Chance Agreement. *Journal of Classification*. **23**: 301-313.

Augen, G. (2005). Bioinformatics in the Post-Genomic Era. Addison Wesley.

Avancini, H. et al. (2004). Organizing Digital Libraries by Automated Text Categorization. *Proceedings of ICDL*. **2004**: 919.

Bação, F. et al. (2004). Geo-Self-Organizing Map (Geo-SOM) for Building and Exploring Homogeneous Regions. *Lecture Notes in Computer Sciences*. Egenhofer, M et al. (Eds.). Berlin. Springer. **GIScience 2004**: 22-37.

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). Modern Information Retrieval. Addison-Wesley Professional.

Bansal, N. et al. (2004). Correlation Clustering. *Machine Learning Journal (Special Issue on Theoretical Advances in Data Clustering)*: 86–113.

Beale, E. L. (1969). Euclidean cluster analysis. In: Bulletin of the International Statistical Institute. *Proceedings of the 37th Session (London)*. Book 2: 92-94.

Beard, M. K. and Mackaness, W. (1991). Generalization Operators and Supporting Structures. Proceedings Auto Carto. **10**: 29-45.

Belkin, J. and Croft, B. (1992) Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*. **35** (12): 29-38.

Bennett, K. and Campbell, C. (2000). Support vector machines: Hype or hallelujah? SIGKDD Exploration, **2**: (2).

Berners-Lee, T. (1989). Information Management: A Proposal. W3C. (http://w3.org/History/1989/proposal.html). Last time accessed November, 2010.

Berners-Lee, T. (2001). The Semantic Web. Scientific American Magazine.

Blanken, H. et al. (2007). Multimedia retrieval. Berlin, New York. Springer.

Bird, S. et al. (2009). *Natural Language Processing with Python*. Sebastopol. O'Reilly.
Bonner, E. (1964). On some clustering techniques. *International Business Machines Journal of Research and Development*. **8**: 22-32.

Bush, V. (1945). As We May Think. *Atlantic Monthly*. (http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm). Last time accessed November, 2011.

Butler, M. (2002). Barriers to real world adoption of semantic web technologies. Technical Report. HP Labs. Bristol, UK.

Bramer, M. (2007). Principles of Data Mining. Berlin, New York. Springer.

Budayan, C. et al. (2009). Comparing the performance of traditional cluster analysis, self-organizing maps and fuzzy c-means methods for strategic grouping. *Expert Systems with Applications*. **36**:11772-11781.

Burges, C. (1999). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*. **2**: 121-167.

Buttenfield, B.P., Stanislawski, L.V. and Brewer, C.A. (2010). Multiscale Representations of Water: Tailoring Generalization Sequences to Specific Physiographic Regimes. *Proceedings GIScience 2010*, Zurich, Switzerland, September 2010.

Buttenfield, P.B. and Mark, D.M. (1991). Expert Systems in Cartographic Design. *In*: Taylor, D.R.F. Geographic Information Systems: The Microcomputer and Modern Cartography. Oxford: Pergamon Press: 129-150.

Buttenfield, P. B. and McMaster R.B. (1991) Map generalization: making rules for knowledge representation. New York. Longman.

Bo, X. et al. (2009). Climate Prediction by SVM based on Initial Conditions. *Fuzzy Systems and Knowledge Discovery*. **5**: 578-581.

Cattell, B. (1966). The Meaning and Strategic Use of Factor Analysis. In: *Handbook of Multivariate Experimental Psychology*. Chicago, Rand McNally.

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

Chebotko, A. and Lu. S. (2010). Querying the Semantic Web: An Efficient Approach Using Relational Databases. LAP Lambert Academic Publishing.

Chisholm, R. (1996). A Realistic Theory of Categories—An Essay on Ontology. Cambridge University Press.

Conklin, J. (1987). Hypertext: An Introduction and Survey. *Computer.* **20** (9): 17–41.

Couclelis, H. (1998). Worlds of information: the geographic metaphor in the visualization of complex information. *Cartography and Geographic Information System*. **25** (4): 209–220.

Come, E. et al. (2011). Aircraft Engine Fleet Monitoring Using Self-Organizing Maps and Edit Distance. In: Laaksonen, J and Honkela, T. *WSOM 2011*. LNCS **6731**: 298-307.

Cortes C. and Vapnik, V. (1995). Support-Vector Networks, Machine Learning, **20** (3):273-297.

Cleverdon, C. W. (1967). The Cranfield tests on index language devices. *Aslib Proceedings*. **19**(6): 173-194.

Cormen, T. H. et al. (2009). Introduction to Algorithms. Cambridge, MA. The MIT Press.

Croft, W., and Harper, D. (1979). Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*. **35**: 285–295.

Cromack, M. (1971). A review of classification. *Journal of the Royal Statistical Society*. **134**: 321-367.

Davies, D. and Bouldin, W. (1979). A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **2**: 224.

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

Douglas, J. (2004). Self-Organizing Maps: A Tourist's Guide to Neural Network. University of California - Santa Barbara. 2004, December.

Dunn, J. C. (1974). Well separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*. **4**: 95-104.

Dioş, L. et al. (2007). Improving SVM Performance using a Linear Combination of Kernels. *Adaptive and Natural Computing Algorithms*. LNCS. **4432**: 218-227.

Erkar, R and Lingras, P. (2008). Building an Intelligent Web. Jones and Barlett Publishers.

Everitt, S. et al. (2001). Cluster Analysis. New York. Oxford University Press.

Estvill-Castro, V. (2002). Why so many clustering algorithms. *ACM SIGKDD Exploration Newsletter.* **4**. 65.

Exchangeable Image File Format. (http://www.exif.org). Last time accessed November, 2011.

Fabrikant, S.I. (2000). Spatial metaphors for browsing large data archives. (Unpublished doctoral dissertation). University of Colorado, Boulder.

Fabrikant, S. I. and Buttenfield, B. P. (2001). Formalizing Spaces for Information Access. *Annals of the Association of American Geographers.* **91**: 263-280.

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

Facebook. (http://www.facebook.com). Last time accessed October, 2011.

Fellbaum, C. (2005). WordNet and wordnets. *In:* Brown, Keith et al.: Encyclopedia of Language and Linguistics. Second Edition. Oxford. Elsevier: 665-670.

Fenneman, N.M. and Johnson, D.W. (1946). Physical Divisions of the United States: Washington, D.C., USGS special map series, scale 1:7,000,000. (http://water.usgs.gov/GIS/metadata/usgswrd/XML/ physio.xml) Last time accessed November, 2011.

Frakes, W.B. and Fox, C.J. (2003*).* Strength and Similarity of Affix Removal Stemming Algorithms. *SIGIR Forum*. **37**: 26-30.

Gärdenfors, P. (2000). Concept combination: a geometrical model. In: Cavedon, L., Blackburn, P., Braisby, N. and Shimojima, A.: *Logic language and Computation Vol. 3*. 129-146.

Gärdenfors, P. (2004). Conceptual spaces as a framework for knowledge representation. *Behavioral and brain sciences*. **27**. 403-403.

Gärdenfors, P. (2004). How to make the Semantic Web more semantic. *Formal Ontology in Information Systems: Proceedings of the third international conference*. 2004: 17–34.

Geisser, S. (1993). Predictive Inference: An Introduction. Chapman & Hall, London.

Geisler, N. (1999). Baker Encyclopedia of Christian Apologetics. Baker Books: 446.

Getty Images. (http://www.gettyimages.com). Last time accessed November, 2011.

Gonzales, R. et al. (2004). Digital Image Processing using MatLab. Prentice Hall.

Goodman, L. and Kruskal, W. (1954). Measures of associations for cross-validations. *Journal of the American Statistical Association*. **49**: 732-764.

Google Picasa. (http://picasaweb.google.com). Last time accessed October, 2011.

Google Search. (http://www.google.com/howgoogleworks/). Last time accessed March, 2011.

Gorden, D. (1999). Classification. Boca Raton, FL. Chapman and Hall/CRC.

Goswami G. et al. (2007). Evolutionary Monte Carlo Methods for Clustering. *Journal of Computational and Graphical Statistics*. **16.** (4): 1-22.

Graves, A., & Lalmas, M. (2002). Video Retrieval using an MPEG-7 based inference network. *Proceedings of the 25th ACM-SIGIR Conference*.

Griffiths, T. L., et al. (2003). Semi-Supervised Learning with Trees. *NIPS 2003*.

Gruber, T. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*. **5**. (2): 199-220.

Gruber, T. (2008). Despite our Best Efforts, Ontologies are not the Hard Part. *AAAI Spring Symposium*. March 2008.

Hamming, R. W. (1950). Error-detecting and error-correcting codes. *Bell System Technical Journal*, **29.** (2): 147–160.

Handl, J. et al. (2005). Computational cluster validation in post-genomic data analysis. *Bioinformatics*. **21**. (15): 3201-3212.

Hardy, A. and Lallemand, P. (2004). Clustering of symbolic objects described by multi-valued and modal variables. In: Studies in Classification, Data Analysis, and Knowledge Organization. *Proceedings of the IFCS 04 Conference*: 325-332.

Hastie, T. et al. (2001). Elements of Statistical Learning. New York. Springer.

Hinton, G. and Sejnowski, T.J. (1999). *Unsupervised Learning: Foundations of Neural Computation*. Cambridge, MA: MIT Press.

Heery, R. and Wagner, H. (2002). A Metadata Registry for the Semantic Web. D-Lib Magazie. **8** (5).

Honarkhah, M. and Caers, J. (2010). Stochastic Simulation of Patterns Using Distance-Based Pattern Modeling. *Mathematical Geosciences*. **42**: 487 – 517.

Honkela, T. (1999). Connectionist Analysis and Creation of Context for Natural Language Understanding and Knowledge Management. CONTEXT 1999: 479-482.

Honkela, T. et al. (1996). Newsgroup exploration with WEBSOM method and browsing interface. Technical Report A32, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo.

Honkela, T. et al. (2008). Simulating processes of concept formation and communication. *Journal of Economic Methodology*.**15**. (3): 245-259.

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

Hull, D.A. (1996*).* Stemming Algorithms – A Case Study for Detailed Evaluation. *JASIS*. **47**. (1): 70-84.

Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. *Bulletin de la Societe Vaudoise des Sciences Naturelles.* **44**: 223-370.

Jaccard, P. (1912). The distribution of flora in the alpine zone. *New Phytologist*. **11**: 37–50.

Jain, A. and Dubes, R. (1988). Algorithms for clustering data. Upper Saddle River, N.J. Prentice-Hall.

Jain, A. et al. (2000). Statistical pattern recognition: A review. *IEEE Transaction of Pattern Analysis and Machine Intelligence.* **22**: 4-37.

Jain, A. et al. (2009). Data clustering: 50 years beyond k-Means. *Pattern Recognition Letters*. **31** (8): 651-666.

Jensen, J.R. (2005) Introductory Image Digital Image Processing. 3rd Edition. Upper Saddle River, N.J. Prentice-Hall.

Jolliffe, T. (1986). Principal Component Analysis. Berlin, New York. Springer-Verlag.

Joyce, J. (2006). Pandora and the Music Genome Project. *Scientific Computing.* **23** (10): 40–41.

Kaski, S. et al. (1998). Bibliography of self-organizing map (SOM) papers: 1981-1997. *Neural Computing Surveys*. **1** (3&4): 1-176.

Kaplan, R. and Bresnan, J. (1982) Lexical-functional grammar: A formal system for grammatical representation. In: Joan Bresnan, editor, *The Mental Representation of Grammatical Relations*. 173–281. The MIT Press, Cambridge, Mass.

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

Kohonen, T. (1995). Sel-Organizing Maps. Berlin. Springer.

Kohonen, T. (2001). Self-Organizing Maps. Third, extended edition. Berlin. Springer.

Kosambi, D. D. (1943). Statistics in Function Space, *Journal of Indian Mathematical Society*. **7** (46): 76-88.

Klapuri, A. (1999). Sound Onset Detection By Applying Psychoacoustic Knowledge. *IEEE International Conference on Acoustics*. Speech and Signal Processing. ICASSP.

Kiviluoto, K. (1998). Predicting bankruptcies with the self-organizing map. *Neurocomputing*. **21**(1-3): 191-201.

Kobayashi, M. and Takeda, K. (2000). Information retrieval on the web. ACM Computing Surveys (ACM Press). **32** (2): 144–173.

Kohonen, T. (2001). Self-organizing maps. Berlin; New York. Springer.

Koren, Y. and Bell, R. (2009). Matrix Factorization techniques for recommender systems. *IEEE Computer Society*: 30-37.

Kruskal, J.B. et al. (1978). Multidimensional scaling. Beverly Hills, Calif. Sage Publications.

Lovins, J. B. (1968). Development of a Stemming Algorithm. *Mechanical Translation and Computational Linguistics*. **11**: 22—31.

Luhn, H. P. (1957). A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, **1** (4).

Lyman, P. and Varian, R. (2003). How much information.
(http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/)
Last time accessed August, 2011.

MacQueen, B. (1967). Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability.* Berkeley. University of California Press. **1**:281-297.

Maron , M. J. and Kuhns, J. K. (1960). On Relevance, Probabilistic Indexing and Information Retrieval. *Journal of the ACM (JACM)*. **7** (3): p.216-244.

Mayer, R. et al. (2008). Map-based Interfaces for Information Management in Large Text Collections. *JDIM*: 294-302.

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

McDavid, A. et al. (2011). Normalized Mutual Information to evaluate overlapping community finding algorithms. *Proceedings of CoRR. 2011*.

McMaster, R. (1991). Conceptual frameworks for geographic knowledge. In: Buttenfield, P. B. and McMaster R.B. (1991) *Map generalization: making rules for knowledge representation*. New York. Longman: 21-39.

McMaster, R. and Shea, K. (1992). Generalization in Digital Cartography. Washington, DC: Association of American Geographers Press.

Meila, M. (2007). Comparing clustering and information based distance. *Journal of Multivariate Analysis*. **98** (5): 873-895.

Milligan, G. W. and Cooper, M. C. (1985). And examination of procedures for determining the number of clusters in a data set. Psychometrika. **50**: 159-179.

Mingoti, S. and Lima, J. (2006). Comparing SOM neural network with Fuzzy c-means, K-Means and traditional hierarchical clustering algorithms. *European Journal of Operational Research*. **147**: 1742-1759.

Moerchen, F. et al. (2005). Databionic visualization of music collections according to perceptual distance. *Proceedings 6th International Conference on Music Information Retrieval (ISMIR 2005)*. London, UK. 396-403.

Mountrakis, G. et al. (2010). Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*. **66**: 3. 247-258.

Navigli, R. and Velardi, P. (2004). Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. *Computational Linguistics.* **30**. (2): 151–179.

Nguyen, N.T. et al. (2008). Agent and Multi-Agent Systems: Technologies and Applications, Berlin, Heidelberg. Springer.

Oberle, D. et al. (2009). What is an ontology? *In*: Handbook on Ontologies". Springer, 2nd edition.

Oja, M. et al. (2003). Bibliography of self-organizing map (SOM) papers: 1998-2001 addendum: *Neural Computing Surveys*. **3** (1): 1–156.

Openshaw, S. et al. (1980). Functional Regions for the 1981 Census of Britain. A User's Guide to the CURDES Definitions. *Discussion paper*, R-Report.

Openshaw, S. (1983). Multivariate analysis of census data. In A Census User's Handbook, ed. D. W. Rhind, 243–263. London: Methuen & Co.

OpenStreetMap (http://www.openstreetmap.org/) Last time accessed July, 2012

Pacual, D. et al. (2010). Cluster validation using information stability measures. *Pattern Recognition Letters.* **31**: 454-461.

Pandora. (http://www.pandora.com). Last time accessed September, 2011.

Paice, C.D. (1990). Another Stemmer, *SIGIR Forum*. **24**: 56-61

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*. **14** (3): 130–137.

Policy Aware Web Project. (http://www.policyawareweb.org). Last time accessed August, 2013).

Powers, D. M. (2011). Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies.* **2** (1): 37-63.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*. 846-850.

Raghavan, V. and Wong, S. (1986). A critical analysis of vector space model for information retrieval. *Journal of the American Society for Information Science*. **33**: 279 – 287.

Regnauld, N. and McMaster, R. (2007). A Synoptic View of Generalization Operators. In: Mackaness, W., Ruas, A. and Sarjakowski, L.: *Generalization of Geographic Information: Cartographic Modeling and Applications*. Oxford, UK. Elsevier.

Rendón, E. et al. (2011). Internal and External cluster validation indexes. *International Journal of Computers and Communications*. **5** (1): 27-34.

Rigaux, P. et al. (2002). Spatial Databases with Application to GIS. San Francisco. Morgan Kaufmann Publishers.

Robert, M. et al. (1973). Textual Features for image classification. *IEEE Transaction on Systems, Man, and Cybernetics*. **3**: 610-621.

Rousseeuw, P. (1987). Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis. Computational and Applied Mathematics **20**: 53–65.

Rosenberg, A. and Hirschberg, J. (2007). V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL).* Prague, Czech Republic, ACL.

Salem, S. and AbdelRahman, S. (2010). A Multiple-Domain Ontology Builder. *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. 967-975.

Salton et al. (1975). A Vector Space Model for Automatic Indexing. *Communications of the ACM*. **18** (11): 613–620.

Salton, G. (1968). Automatic information organization and retrieval. New York. McGraw-Hill.

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

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

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*. **24** (5): 513–523.

Sammon, J. (1969). A Nonlinear Method for Data Structure Analysis. *IEEE Transactions on Computers.* **C-18** (5): 401-409.

Sarlin, P. and Eklund, T. (2011). Fuzzy clustering of the Self-Organizing Map: Some Applications on Financial Time Series. In: Laaksonen, J. and Honkela, T.: *Advances in Self-Organizing Maps. Lecture Notes in Computer Science.* **6731**: 40-50.

Segaran, T. (2007). Programming Collective Intelligence. Sebastopol. O'Reilly Media.

Segaran, T. (2008). Programming the Semantic Web. Sebastopol. O'Reilly Media.

Singhal, A. (2001) Modern information retrieval: A brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*. **24** (4): 35–43.

Shepard, R. N. (1962). The Analysis of Proximities: Multidimensional Scaling with an Unknown Distance Function. *Psychometrika.* **27**: 125-139 and 219-246.

Smith B. and Welty, C. (2001). Ontology: Towards a new synthesis. *In*: Formal Ontology in Information Systems.

Skupin, A. and Buttenfield, B. P. (1997). Spatial Metaphors for Visualizing Very Large Data Archives. *Proceedings, AUTO-CARTO 13*. Seattle, WA, Apr. 7-10, **1997**: 116-125.

Skupin, A. (1998).Organizing and visualizing hypermedia information spaces. (Unpublished Dissertation). Buffalo. SUNY-Buffalo.

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

Skupin, A. (2007) Spatialization. In: Kemp, K. *Encyclopedia of Geographic Information Science*. Sage Publications Inc. 418-422.

Skupin, A. (2010) Tri-Space: Conceptualization, Transformation, Visualization. Sixth International Conference on Geographic Information Science (GIScience 2006), Zürich, Switzerland, September 2010.

Sneath P. (1957). The application of computer to taxonomy. *Journal of General Microbiology*.

Sneath, P. and Sokal, R. (1963). Principles of Numerical taxonomy. W.H. Freeman.

Sparck-Johnes, K. and Willet, P. (1997). Readings in information retrieval. San Francisco. Morgan Kaufmann Publishers Inc.

Stanislawski, L.V., Finn, M.P. and Buttenfield, B.P (2010). Integrating Hydrographic Generalization over Multiple Physiographic Regimes. In: Buttenfield, B.P. and Mackaness, W. (eds.) *Generalization and Data Integration.* (manuscript accepted; book in preparation).

STATA online help. (http://www.stata.com/links/resources1.html). Last time accessed October, 2011.

Stats Counter Online Search Engines. (http://gs.statcounter.com/#search_engine-ww-monthly-201010-201012). Last time accessed November, 2011.

Steger, C. (1998). An unbiased detector of curvilinear structures. *IEEE Transactions on Pattern Recognition and Machine Intelligence*. **20**: 113-125.

Strobl, C. et al. (2009). An Introduction to Recursive Partitioning: Rational, Application, and Characteristics of Classification and Regression Trees, Bagging, and Random Forests. *Psychological Methods*. **14** (4): 323-348.

Soboroff, I et al. (2001). Ranking Retrieval Systems without Relevance Judgments. *Proceedings of the 24th Annual International Conference on Research and Development in Information Retrieval (SIGIR 2001)*, New Orleans, LA.

Tang, Y. et al. (2012). Survey on clustering methods for ontological knowledge. *International Technology Alliance:* 1-8.

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*. **22** (2): 283-295.

Taylor, A and Joudrey, D. (2009). The organization of Information. Westport, Conn. Libraries Unlimited.

Tobler W. (1970) A computer movie simulating urban growth in the Detroit region. *Economic Geography*. **46** (2): 234-240.

Topchy, A. and Punch, W. (2003). Combining multiple weak clustering. *Proceeding of the Third IEEE International Conference on Data Mining (ICDM'03)*: 331-338.

Torgerson, W.S. (1952). Multidimensional scaling .1. Theory and methods. *Psychometrika*. **17** (4): 401–419.

Torgerson, W.S. (1958). Theory and Methods of Scaling. New York. Wiley.

Turk, M. and Pentland, A. (1991). Face Recognition using Eigenfaces. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*: 586-591.

Turtle, H. and Croft, B. (1991). Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems*. **9** (3):187–222.

Ultsch, A. and Siemon, P. (1990). Kohonen's self-organizing feature maps for exploratory data analysis. *Proc. INNC'90, Int. Neural Network Conference: Dordrecht*, Netherlands, 1990. 305-308.

USGS Earth Explorer. (http://www.earthexplorer.usgs.gov/). Last time accessed November 2011.

USGS Libraries Program. (http://library.usgs.gov/). Last time accessed June 2011.

Van der Heijden, F. (1994). Image based measurements systems: Object recognition and parameter estimation. Chichester, England. Wiley.

Vapnik, V. and Lerner, A. (1963). Pattern recognition using generalized portrait method. *Automation and Remote Control*. **24**: 774–780.

Vapnik, V. (2000). *The Nature of Statistical Learning Theory.* New York. Springer-Verlag.

Vesanto, J. and Alhoniermi, E. (2000). Clustering of the Self-Organizing Map. *IEEE Transactions on Neural networks **11** (3): 586-600.*

Vesanto, J. (2005). SOM implementation in SOM Toolbox. (http://www.cis.hut.fi/somtoolbox/documentation/somalg.shtml). Last time accessed August, 2011.

Viger, R.J. (2008). The GIS Weasel: an interface for the treatment of geographic information in modeling. *Computers & Geosciences*. **34** (8): 891–901.

Viger, R. (2011). What is the meaning of a GIS procedure? Reasoning about geographic concepts using a geoprocessing framework. (Unpublished doctoral dissertation). Department of Geography. University of Colorado at Boulder. Boulder, Colorado.

Wang, C. (2010). An automatic documentation generator based on model-driven techniques. *Computer Engineering and Technology (ICCET).* **4**: 174-179.

Wang, K. (2008). Cluster Validation Toolbox for MatLab (http://www.mathworks.com/matlabcentral/fileexchange/authors/24811). Last time accessed November, 2011.

Wang, N. et al. (2013) Visualizing Gridded Time Series Data with Self-Organizing Maps: An Application to Multi-Year Snow Dynamics in the Northern Hemisphere. *Computers, Environment and Urban Systems*. **39.** (5): 107-120.

Werbos, P. (1994). The Roots of Backpropagation. From Ordered Derivatives to Neural Networks and Political Forecasting. New York, NY: John Wiley & Sons, Inc.

Wendel, J. et al. (2009). Spatialization of a GIS Software Toolbox. *Kartographische Nachrichten*. Jahrgang. **59**: Heft 5.

Wendel, J. and Buttenfield B. P. (2011). Flexible Characterization of Cartographic Generalization Resources for a Self-Organizing Online Catalog. *Proceedings ICC 2011*. Paris, France. 7th July 2011.

Wise, J.A. (1999). The ecological approach to text visualization. *Journal of the American Society for Information Science*. **50** (13): 1224–1233.

Witten, I.H. and Frank, E. (2005). Data Mining: Practical Machine Learning Tools and Techniques. San Francisco. Morgan Kaufman.

Wong, S. et al. (1989). Extended Boolean query processing in the generalized vector space model. *Proceedings of Information Systems*. **1989**: 47-63.

Ye, H. and Lo, N. (2001). Towards a self-structuring software library. *IEE Proceedings-Software*. 148.2: 45–55.

Youtube. (http://www.youtube.com). Last time accessed September, 2011.

Zablith, F. (2008). Dynamic ontology evolution. *In*: ISWC 2008 Doctoral consortium. 25 Oct. 2008. Karlsruhe, Germany.

Zanoni, C. et al. (2011). A semi-automatic source code documentation method for small software development teams. *Computer Supported Cooperative Work in Design (CSCWD). 2011*: 113-119.

Zhang, X. and Xiao, W. (2012). Clustering based Two-Stage Text Classification Requiring Minimal Training Data. *Computer Science and Information Systems*. **9**: 4, 1627-1644.

Zdziarski, J.A. 2005 Ending spam: Bayesian content filtering and the art of statistical language classification. San Francisco: No Starch Press.

Zhou, Y. and Davis, J. (2006). Analysis of Weblog Link Structure: A Community Perspective. *Proceedings of the International Conference on Web Information Systems and Technologies*. Setubal, Portugal. April 11-13, 2006. 13-20.

Appendix A – Full-text data set

This appendix contains two tables. The first, Table A-1 lists all 1410 full-text documents in the data set. The second table, Table A-2, shows the result of all clustering and classification methods. The full-text documents can be identified by ID.

Table A-1 Complete full text data set

www.geo-think.net/dissertation/literature_data_complete.htm

Table A-2 Cluster and class membership for the full-text data set

http://www.geo-think.net/dissertation/full_text_cluster_membership.htm

Appendix B – GIS commands data set

This appendix contains the clustered and classified GIS commands data set.

Table B-1 Cluster and class membership for the GIS commands data set

www.geo-think.net/dissertation/GIS_commands_cluster_class_membership.htm

Appendix C – Algorithm data set

This appendix contains two tables. The first, Table C-1 lists all 1607 algorithm related documents before filtering of the cartographic specific keywords was conducted. The second table, Table C-2, shows the result of all clustering and classification methods of the final data set which consists of 979 publications. The algorithm publications can be identified by ID.

Table C-1 Complete algorithm data set

www.geo-think.net/dissertation/Algorithms_complete_dataset.htm

Table C-2 Cluster and class membership for the algorithm data set

http://www.geo-think.net/dissertation/algo_cluster_membership.htm