ROBUST ANALYSIS OF NON-PARAMETRIC SPACE-TIME CLUSTERING

by

Xin Huang

APPROVED BY SUPERVISORY COMMITTEE:

Yulia R. Gel, Chair

Yifei Lou

Yongwan Chun

Yuly Koshevnik

Copyright © 2018

Xin Huang

All rights reserved

To my supervisor, my family, and myself.

ROBUST ANALYSIS OF NON-PARAMETRIC SPACE-TIME CLUSTERING

by

XIN HUANG, BS, MS

DISSERTATION

Presented to the Faculty of
The University of Texas at Dallas
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY IN
STATISTICS

THE UNIVERSITY OF TEXAS AT DALLAS
August 2018

ACKNOWLEDGMENTS

I would like to express my deep gratitude to my supervisor, Professor Yulia R. Gel, for her constant mentorship, dedication and help throughout this research process. Three years of research being with Professor Yulia R. Gel allowed me to gain the best academic training in becoming a doctor, not only in research but also in presentation and communication skills. When I look at the past three years of research with my supervisor, I can remember countless moments, especially the nights when we worked together to meet submission deadlines for conferences until up to 2 AM in the morning; the 40 suggestions she gave to me after hearing the rehearsal for my first-time presentation in a conference; and her inspiration and encouragement when I hesitate about going for onsite interview and conferences. I really enjoyed those three years of research life with Professor Yulia R. Gel. Without her support and help this dissertation work would not have been possible.

I would also like to express my gratitude to my committee members, Professor Yifei Lou, Professor Yongwan Chun, and Professor Yuly Koshevnik for serving on my committee and providing their support in completing this work. I would also like to express my thanks to Professor Michael Baron for his fantastic lectures on probability and the recommendation letter he wrote for me for my summer internship application. Without his recommendation letter I would not have found my second internship at Procter & Gamble.

I would like to express my special appreciation to my family for their unconditional love and enormous support throughout these years. I want especially thank my grandmother Guangping Deng who always supports me, believes in me, and loves me. I am also very grateful to my friends Xiao Zhang, Tao Wang, and Yuchu Chen for always being there, listening to me, and encouraging me while I am in a difficult position. I would also like to thank Jeremy M. Testa, Karianne Bergen and Cuneyt Akcora for stimulating discussions in writing papers.

I would like to thank all professors and staff in the Department of Mathematical Sciences. I would also like to thank the department for the five-year teaching assistantships. I really enjoyed studying and working here over the past five years.

May 2018

ROBUST ANALYSIS OF NON-PARAMETRIC SPACE-TIME CLUSTERING

Xin Huang, PhD
The University of Texas at Dallas, 2018

Supervising Professor: Yulia R. Gel, Chair

Recently, the rampant growth of various remote sensing technologies has resulted in a spike of interest in space-time data mining and particularly clustering of environmental time series and spatio-temporal processes.

Remarkably, the dynamic data-driven clustering procedures for space-time data that allow the number, shape and distributional properties of clusters to vary, have received a flare of interest in recent years. Despite the potential of the dynamic data-driven clustering procedures, the price for their flexibility is usually a set of parameters that control clustering performance and are to be user-specified – for instance, the value similarity threshold δ in TRUST; the maximum radius of the neighborhood Eps in DBSCAN; the steepness parameter ξ in OPTICS; and the kernel smoothing parameter h in DENCLUE. The choice of these parameters can noticeably impact the number and shape of detected clusters, and ideally should be approached in an objective manner. The goal of this dissertation is to address those challenges by developing new nonparametric data-driven approaches in space-time clustering.

First, we propose a new data-driven procedure for optimal selection of these tuning parameters in dynamic clustering algorithms, using the notion of stability probe. We study finite sample performance of DR in conjunction with DBSCAN and TRUST in application to clustering synthetic times series and yearly temperature records in Central Germany. We also utilized DR in studying the ecological trends and water quality in Chesapeake Bay and legislative rhetoric data in the U.S. Senate.

Second, when it comes to optimal selection of tuning parameters in density-based clustering procedures such as DBSCAN, OPTICS, and DENCLUE, some additional problems such as existence of clusters with varied densities and existence of outliers need to be addressed. Therefore, we develop a new density-based clustering algorithm named CRAD which is based on a new neighbor searching function with a robust data depth as the dissimilarity measure. Our experiments prove that the new CRAD is highly competitive at detecting clusters with varying densities, compared with the existing algorithms such as DBSCAN, OPTICS and DBCA.

TABLE OF CONTENTS

ACKNOWLEDGMENTS v

ABSTRACT vii

LIST OF FIGURES xi

LIST OF TABLES xiii

CHAPTER 1 INTRODUCTION 1

- 1.1 Motivation 1
  - 1.1.1 Dynamic Data-driven Clustering Procedures for Space-time Data . . 1
  - 1.1.2 Data Depth based Spatial Clustering 2
- 1.2 Dissertation Outline 3
- 1.3 Contributions 5

CHAPTER 2 RIDING DOWN THE BAY: SPACE-TIME CLUSTERING OF ECOLOGICAL TRENDS 7

- 2.1 Introduction 7
- 2.2 Downhill Riding Procedure with Clustering Stability Probe 11
- 2.3 Clustering Algorithms 16
  - 2.3.1 The TRUST Algorithm 16
  - 2.3.2 The DBSCAN Algorithm 18
- 2.4 Numerical Experiments 18
  - 2.4.1 Benchmark Iris Data 18
  - 2.4.2 Synthetic data 19
- 2.5 Case Study I 21
- 2.6 Case Study II 26
- 2.7 Conclusion 31

CHAPTER 3 CRAD: CLUSTERING WITH ROBUST AUTOCUTS AND DEPTH 33

- 3.1 Introduction 33
- 3.2 Our Algorithm 36
  - 3.2.1 A Robust Data Depth Based Dissimilarity 37
  - 3.2.2 A New Neighbor Searching Algorithm 38

3.2.3 An Extension to DBSCAN 41
3.3 Determining the Parameter *StepSize* and *Nbin* 43
3.4 Experimental Evaluation 45
3.4.1 Synthetic Data 45
3.4.2 Read-World Multivariate Data 50
3.4.3 Real-World Time Series Data 51
3.5 Conclusion 54
CHAPTER 4 POLITICAL RHETORIC THROUGH THE LENS OF NONPARAMETRIC STATISTICS: ARE OUR LEGISLATORS THAT DIFFERENT? 55
4.1 Introduction 55
4.2 Related Work 59
4.3 Data Generation 62
4.3.1 Rhetoric in Committee 62
4.3.2 Natural Language Processing of Committee Rhetoric 64
4.3.3 Re-Indexing Time 68
4.3.4 Statistical Validation and Reliability Testing 69
4.4 Cluster Analysis of Political Rhetoric 71
4.5 Case Study 74
4.5.1 The Full Membership Dataset — High Number of Senators, Shorter Period 78
4.5.2 The Full Term Dataset — Longer Period, Lower Number of Senators 81
4.6 Discussion 83
CHAPTER 5 CONCLUSION AND FUTURE WORK 86
5.1 Conclusion 86
5.2 Future Work 88
REFERENCES 89
BIOGRAPHICAL SKETCH 103
CURRICULUM VITAE

LIST OF FIGURES

2.1 Aggregated dynamics of NMI and ACD (smooth lines) and dynamics for a single data realization (dashed lines) with clustering by TRUST. The metrics are scaled to fit on one graph. The Downhill Riding (DR) procedure selects the TRUST tuning parameters v (e.g., the value-similarity threshold δ) corresponding to the first right-hand side local minimum of ACD. 14

2.2 Clustering of weather stations in time period 4 (year 46-60) by TRUST. 22

2.3 Clustering of weather stations in time period 1 and 4 by TRUST. 24

2.4 Clustering of weather stations in time period 1 and 4 by DBSCAN. 25

2.5 Clustering of the Chesapeake Bay water quality monitoring stations using TRUST, with cluster assignments denoted by color. Black indicates the most populous cluster in each sub-period; "2" denotes the second populous cluster in 1985–2000, and "*" denotes upper Rappahannock River stations. 29

2.6 Time series of the standardized TSS concentrations in the main cluster, along with the loess smoothing curves. 29

2.7 Time series of the upper Rappahannock River stations (marked with "*" in Figure 2.5) that exited the main cluster in 2001–2016, along with a loess smoothing curve. 30

2.8 Time series of cluster 2 (see Figure 2.5) that joined the main cluster in 2001–2016, along with a loess smoothing curve. 30

3.1 Clustering Performance of CRAD, DBCA, DBSCAN and OPTICS on the Toy Example. 36

3.2 A contour plot based on traditional (black dash line) and robust (blue solid line) Mahalanobis depth function. 38

3.3 Neighbor search for a given point (the yellow dot) in the toy example. The red dots are the neighbors, identified by CRAD and DBCA under their best clustering performances, shown in top (bottom) right. The histogram shows the optimal value of the cut-off parameter, $h_{opt}(i)$ (the red vertical line) for CRAD and θ (the blue vertical line) for DBCA. 40

3.4 Clustering Performance of CRAD, CRAD-DBSCAN, DBCA, DBSCAN (EU), and OPTICS on $S1$. 47

3.5 Clustering Performance of CRAD, CRAD-DBSCAN, DBCA, DBSCAN (EU), and OPTICS on $S2$. 47

3.6 Best AMI of CRAD, CRAD-DBSCAN, DBCA, DBSCAN (EU) and OTPICS on synthetics $S1$ and $S2$. Each AMI score is an average result over 10 trials. 48

4.1 Example of supervised classification of the energy-related political rhetoric. . . . 67

4.2 Sketch of the R-TRUST clustering approach for a toy example with 6 senators. Circles $S_1, \ldots, S_6$ denote times series corresponding to rhetoric of each senator over a period of one slide. 73

4.3 Summary of clustering results for the Full Term (6 periods) and Full Membership (4 periods) datasets. Each period is 12 months. Note that since the Full Membership dataset is limited to only 4 periods, no clustering exists for periods 5 and 6. 75

4.4 Dynamic Clustering of Legislative Rhetoric in the Full Membership dataset. . . 79

4.5 Sensitivity analysis for Full Membership Dataset. Number of crossvalidation replications is 10. 81

4.6 Dynamic Clustering of Legislative Rhetoric in the Full Term dataset. 82

4.7 Sensitivity analysis for Full Term Dataset. Number of crossvalidation replications is 10. 84

LIST OF TABLES

2.1 Performance of DBSCAN on Iris data in terms of NMI, with Eps-opt selected using Downhill Riding and with Eps-kdist selected by conventional *k-dist graph*. 19

2.2 Cluster configuration of 16 simulated time series. 20

2.3 Performance of TRUST with value similarity threshold δ_{opt} and connectivity parameter θ_{opt} selected by Downhill Riding and TRUST with δ_{oracle}, θ_{oracle}. Number of Monte Carlo experiments is 100. Number of cross-validation splits T is 300. . 21

2.4 MAPE of 4 time periods by TRUST and DBSCAN on observed temperature data. 26

2.5 MAPE of 4 time periods by TRUST and DBSCAN on scaled temperature data. 26

3.1 Clustering Performance on 5 UCI Multivariate Datasets. The winner method on each dataset is highlighted. 48

3.2 Clustering Performance for the 29 UCR Time Series Datasets. Two ratios, which are over 0.9 and 0.8 in column "CRAD / k-means" and column "Empirical CRAD / CRAD", are highlighted, respectively. 49

3.3 Clustering Performance of CRAD, DBCA and DBSCAN on 5 UCR Time Series Datasets. The winner method on each dataset is highlighted. 53

CHAPTER 1

INTRODUCTION[1]

1.1 Motivation

1.1.1 Dynamic Data-driven Clustering Procedures for Space-time Data

Clustering of time series has received considerable attention in the last two decades both in data mining and statistical literature (Lux and Marchesi, 2000; Keogh and Lin, 2005; Ratanamahatana et al., 2005; Wei and Keogh, 2006; Euan et al., 2015), with applications ranging from finance and communication sciences to neuroscience and geology. Most recently, the rampant growth of various remote sensing technologies has resulted in a spike of interest in space-time data mining and particularly clustering of environmental time series and spatio-temporal processes (Phoha et al., 2003; Lozano et al., 2009; Urbancic et al., 1992; Rashidi and Cook, 2010). However, many currently existing clustering procedures for space-time data are either based solely on geographical proximity, which does not account for drifts in space-time data distribution, or are restricted to a relatively small domain to avoid high spatial heterogeneity (Stein, 2005; Beelen et al., 2013). Furthermore, the number of possible

---

[1] This chapter includes verbatim excerpts from

Reprinted from Huang, X., Iliev, I., Brenning, A., Gel, Y. (2016) Space-Time Clustering with Stability Probe while Riding Downhill. *Proc.22nd ACM SIGKDD Workshop on Mining and Learning from Time Series (MiLeTS), 2016*, `http://www-bcf.usc.edu/~liu32/milets16/#papers`.

©2017 John Wiley and Sons. Reprinted, with permission, from Huang, X., Iliev, I., Lyubchich, V., Gel, Y. Riding down the Bay: Space-time clustering of ecological trends. *Environmetrics, e2455, 2017*, `https://onlinelibrary.wiley.com/doi/abs/10.1002/env.2455`.

©2017 IEEE. Reprinted, with permission, from Huang, X., Gel, Y. CRAD: Clustering with Robust Autocuts and Depth. *Proc. 17th IEEE International Conference on Data Mining (ICDM), 2017*, `https://ieeexplore.ieee.org/document/8215579/`

Under Invited Revision. Iliev, I., Huang, X., Gel, Y. (2017) Political Rhetoric through the Lens of Nonparametric Statistics: Are Our Legislators That Different?

clusters is often fixed a-priori, which substantially limits the utility of such clustering procedures in environmental applications that are typically characterized by spatio-temporal non-stationarity and non-separability (Gneiting, 2002; Sherman, 2011).

Remarkably, the number of dynamic data-driven clustering procedures for space-time data that allow the number, shape and distributional properties of clusters to vary, still remains quite limited. However, this research direction has received a flare of interest in recent years (Gaber et al., 2005; Cao et al., 2006; Banerjee et al., 2014). Two such dynamic clustering procedures are an efficient space-time data mining procedure (TRUST) of Ciampi et al. (2010) that is based on interleaving spatial clustering and temporal trend detection, and a hierarchical spectral merger algorithm to cluster brain connectivity (Euan et al., 2015). Alternatively, we can adjust various density-based clustering procedures such as DBSCAN (Kogan et al., 2006; Ester et al., 1996), OPTICS (Ankerst et al., 1999), DENCLUE (Hinneburg and Keim, 1998) etc, to a space-time context.

Despite the potential of these dynamic clustering procedures, the price for their flexibility is usually a set of parameters that control clustering performance and are to be user-specified – for instance, the value similarity threshold δ in TRUST (Ciampi et al., 2010); the maximum radius of the neighborhood Eps in DBSCAN (Ester et al., 1996); the steepness parameter ξ in OPTICS (Ankerst et al., 1999); and the kernel smoothing parameter h in DENCLUE (Hinneburg and Keim, 1998). The choice of these parameters can noticeably impact the number and shape of detected clusters, and ideally should be approached in an objective manner.

Therefore, the first goal of our research is the optimal selection of these parameters to achieve both accurate and robust clustering performances for space-time data using the notion of stability probe.

1.1.2 Data Depth based Spatial Clustering

Data depth methodology is a widely employed nonparametric tool in multivariate and functional data analysis, with applications ranging from outlier detection to clustering and visualization (Liu et al., 1999; Cuevas et al., 2007; Li et al., 2012). Depth measures the "centrality" (or "outlyingness") of a given object with respect to an observed data cloud (Mosler, 2013; Zuo and Serfling, 2000). Many desirable properties of data depth such as affine invariance, robustness, and center maximality have earned it an increasing attention in the machine learning and statistics communities in the last decade. There exist numerous clustering and classification methods, based on a data depth concept (Jörnsten, 2004; Mosler, 2013; Pokotylo et al., 2016). Most such methods, however, rely on the knowledge of a true number of clusters k. Most recently, Jeong et al. (2016) proposed a Depth Based Clustering Algorithm (DBCA) and showed benefits of a data depth for clustering spatial data. To the best of our knowledge, Jeong et al. (2016) is the first and only reference introducing a data depth concept into clustering analysis of spatial data. However, DBCA cannot handle a case of clusters with varying densities, which is a common issue in density-based clustering domain. In addition, the problem how to select the tuning parameter, which highly impacts the clustering result, remains unstudied.

Therefore, the second goal of our research aims to answer following major challenges in density-based clustering: (1) Based on data depth, can we propose an algorithm that delivers more robust performance under the existence of clusters with varying densities? (2) Based on the proposed algorithm, how can we select the true underlying parameter in the real-world clustering when the ground truth is not given? (3) Can the density-based algorithm be extended to multivariate time-series clustering, without a-priori knowledge of the number of clusters?

1.2 Dissertation Outline

Space-Time Clustering with Stability Probe while Riding Downhill Motivated by Section 1.1.1, we propose a new data-driven procedure for optimal selection of tuning parameters in dynamic clustering algorithms, using the notion of stability probe (Chapter 2).

Due to the shape of the stability probe dynamics, we refer to the new clustering stability procedure as Downhill Riding (DR). We study finite sample performance of DR in conjunction with DBSCAN and TRUST in application to clustering synthetic times series and benchmark data (Section 2.4). We also utilized DR in analysing yearly temperature records in Central Germany (Section 2.5) and studying the ecological trends and water quality in Chesapeake Bay[2] (Section 2.6).

CRAD: Clustering with Robust Autocuts and Depth Motivated by Section 1.1.2, we develop a new density-based clustering algorithm named CRAD which is based on a new neighbor searching function with a robust data depth as the dissimilarity measure (Chapter 3). Our experiments prove that the new CRAD is highly competitive at detecting clusters with varying densities, compared with the existing algorithms such as DBSCAN, OPTICS and DBCA (Section 3.4.1, 3.4.2). Furthermore, a new effective parameter selection procedure is developed to select the optimal underlying parameter in the real-world clustering, when the ground truth is unknown (Section 3.3). Lastly, we suggest a new clustering framework that extends CRAD from spatial data clustering to time series clustering without a-priori knowledge of the true number of clusters (Section 3.4.3). The performance of CRAD is evaluated through extensive experimental studies.

Political Rhetoric through the Lens of Nonparametric Statistics: Are Our Legislators That Different? Not limited to environmental space-time data in Chapter 2, the dynamic clustering procedure combined with DR procedure can also be extended to unstructured space-time data—legislative rhetoric data in the U.S. Senate. We present a novel statistical analysis of legislative rhetoric in the U.S. Senate that sheds a light on hidden patterns in the behavior of senators as a function of their time in office (Chapter 4). Using natural language processing, we create a novel comprehensive dataset based on the speeches

---

[2] The Chesapeake Bay Program, initiated in 1983, is a regional partnership between several state governments, federal agencies and advisory groups, that is involved in the clean-up and restoration of the Bay.

of all senators who served on the U.S. Senate Committee on Energy and Natural Resources in 2001–2011 (Section 4.3.1). We develop a new measure of congressional speech, based on senators' attitudes toward the dominant energy interests (Section 4.3.2, 4.3.3, 4.3.4). To evaluate intrinsically dynamic formation of groups among senators, we adopt a model-free unsupervised space-time data mining algorithm that has been proposed in the context of tracking dynamic clusters in environmental geo-referenced data streams (Section 4.4). Our approach based on a two-stage hybrid supervised-unsupervised learning methodology is innovative, data-driven and transcends conventional disciplinary borders. We discover that legislators become much more alike after the first few years of their term, regardless of their partisanship and campaign promises.

To ensure that each chapter is self-contained, the preliminary information and notations are mentioned in every chapter in which new methodology is proposed.

1.3 Contributions

- We present a new data-driven procedure for optimal selection of tuning parameters in dynamic clustering algorithms called Downhill Riding (DR), using the notion of stability probe (Chapter 2). We study finite sample performance of DR in conjunction with DBSCAN and TRUST in application to clustering synthetic times series and yearly temperature records in Central Germany, and studying the ecological trends and water quality in Chesapeake Bay. The results of this project are published in (Huang et al., 2016) and (Huang et al., 2017).

- We develop a new density-based clustering algorithm named CRAD which is based on a new neighbor searching function with a robust data depth as the dissimilarity measure (Chapter 3). The findings of this project have been published in (Huang and Gel, 2017).

- We present a novel statistical analysis of legislative rhetoric in the U.S. Senate that sheds a light on hidden patterns in the behavior of senators as a function of their time in office (Chapter 4). The findings have been summarized in a manuscript which undergoes an invited revision and has been presented at Joint Statistical Meetings (JSM) conference 2016.

CHAPTER 2

RIDING DOWN THE BAY: SPACE-TIME CLUSTERING OF ECOLOGICAL TRENDS[1]

2.1 Introduction

Chesapeake Bay is the largest estuary in the United States and one of the largest in the World. It stretches for over 200 miles from Maryland to Virginia and is home to a large number of plants, animals, and people. The Bay has been impacted by numerous contaminants and ecological threats. Currently, about three-quarters of its waters are considered impaired by chemical contaminants such as pesticides, pharmaceuticals, and metals (U.S. Environmental Protection Agency, 2012). These contaminants can harm the health of humans and wildlife alike. The Chesapeake Bay Program is a regional partnership between several state and local governments, federal agencies, academic institutions, and advisory groups, aiming at the restoration of the Bay and the clean-up of pollutants. Investigating ecological trends, such as the concentrations of pollutants, can allow for the better management of resources and a more precise geographic focus of the program. Studying ecological trends requires a data-driven procedure that can identify spatial and temporal clustering in a selected area. We propose such a procedure for dynamic clustering that allows the automatic optimal selection of the tuning parameters using a *clustering stability probe*. By probe here we understand an

---

[1] This chapter includes verbatim excerpts from

Reprinted from Huang, X., Iliev, I., Brenning, A., Gel, Y. (2016) Space-Time Clustering with Stability Probe while Riding Downhill. *Proc.22nd ACM SIGKDD Workshop on Mining and Learning from Time Series (MiLeTS), 2016*, `http://www-bcf.usc.edu/~liu32/milets16/#papers`.

©2017 John Wiley and Sons. Reprinted, with permission, from Huang, X., Iliev, I., Lyubchich, V., Gel, Y. Riding down the Bay: Space-time clustering of ecological trends. *Environmetrics, e2455, 2017*, `https://onlinelibrary.wiley.com/doi/abs/10.1002/env.2455`.

exploratory measure (or indicator) that can be utilized to assess stability and accuracy of clustering performance.

Clustering of time series has received considerable attention in the last two decades both in data mining and statistical literature (for overviews see, e.g., Wei and Keogh, 2006; Mueen et al., 2011; Silva et al., 2013; Euan et al., 2015, and references therein), with applications ranging from finance and communication sciences to neuroscience and geology. Most recently, the rampant growth of various remote sensing technologies has resulted in a spike of interest in space-time data mining and particularly clustering of environmental time series and spatio-temporal processes. However, many currently existing clustering procedures for space-time data are either based solely on geographical proximity, which does not account for drifts in space-time data distribution, or are restricted to a relatively small domain to avoid high spatial heterogeneity. Furthermore, the number of possible clusters is often fixed a-priori, which substantially limits the utility of such clustering procedures in environmental applications that are typically characterized by spatio-temporal non-stationarity and non-separability (Sherman, 2011).

Our initial interest in the topic was motivated by studies of the impact of climate change on insurance claim dynamics and early recognition of areas with the highest vulnerability to adverse weather conditions, particularly, the so-called "normal" extreme weather, with a low individual but high cumulative impact (for overview see, e.g., Lyubchich and Gel, 2017a; Scheel et al., 2013; Soliman et al., 2015, and references therein). Remarkably, attribution analysis of such "normal" weather on the insurance industry is largely unexplored (Curry et al., 2012; CIA, 2014; Lyubchich and Gel, 2017a). Since there exist multiple factors contributing to elevated insurance risks, e.g., city infrastructure, building codes, socio-demographics, landscape, as well as numerous latent variables, areas that are similar in their sensitivity to adverse weather are not necessarily close geographically. At the same time, the number of clusters, or areas with similar levels of vulnerability, is unknown and can vary over different periods driven, for example, by the El Niño-Southern Oscillation (ENSO) cycles and other forcings. Moreover, the choice of optimal number of clusters is a longstanding problem in climate sciences (see, for instance, discussion by Werner and Gerstengarbe, 1997; Mahlstein and Knutti, 2010; Heikkilä and Sorteberg, 2012). How can we approach this problem then?

Nevertheless, the number of dynamic data-driven clustering procedures for space-time data that allow the number, shape, and distributional properties of clusters to vary, still remains quite limited. Two such dynamic clustering procedures are an efficient space-time data mining procedure TRUST of Ciampi et al. (2010) that is based on interleaving spatial clustering and temporal trend detection; and a hierarchical spectral merger algorithm to cluster brain connectivity (Euan et al., 2015). Alternatively, we can adjust various density-based clustering procedures, such as DBSCAN (Ester et al., 1996; Kogan et al., 2006), OPTICS (Ankerst et al., 1999), and DENCLUE (Hinneburg and Keim, 1998), to a space-time context.

Despite the potential of these dynamic clustering procedures, the price for their flexibility is usually a set of parameters that control clustering performance and are to be user-specified – for instance, the maximum radius of the neighborhood Eps in DBSCAN; the steepness parameter ξ in OPTICS; the value similarity threshold δ in TRUST; and the kernel smoothing parameter h in DENCLUE. The choice of these parameters can noticeably impact the number and shape of detected clusters, and ideally should be approached in an objective manner.

In this paper we propose a new data-driven and computationally efficient procedure for optimal selection of clustering tuning parameters using a clustering stability probe. Our approach is rooted in the so-called clustering (in)stability criteria (Wang, 2010; Ben-David et al., 2006; Ben-David and Von Luxburg, 2008; Dudoit and Fridlyand, 2002), based on the intuitive idea that if we randomly split our data into two non-overlapping subsets, then a good clustering algorithm should deliver similar clustering results. Hence, the idea is to perform multiple splits, using cross-validation, and search for the case with the most similar (on average) partitions.

Clustering (in)stability has gained an increased interest in machine learning and statistical sciences for identification of the optimal number of clusters, typically in conjunction with k-means (Wang, 2010; Ben-David et al., 2006; Ben-David and Von Luxburg, 2008; Dudoit and Fridlyand, 2002). The field has attracted a lot of attention in the last couple of years (Ben-David and Reyzin, 2014; Jia et al., 2014; Nikulin, 2015), especially in terms of consensus clustering, which aims to find a single partitioning that is as similar as possible to existing basic partitions (see Lock and Dunson, 2013; Niu et al., 2016, and references therein).

Our approach, however, has two main advantages over conventional clustering (in)stability. First, instead of measuring the distance between each two partitions, which is a very computationally demanding if not prohibitive step, we select a clustering probe and define stability only based on the distance between univariate probes. In this paper, we are primarily interested in the utility of a number of clusters as a probe. Second, we advance the idea of a clustering (in)stability criterion to choose the optimal parameters in TRUST, DBSCAN and other dynamic clustering algorithms. Due to the shape of the stability probe dynamics, we refer to the new clustering stability procedure as Downhill Riding (DR). We outline the theoretical properties of the new DR procedure and evaluate its finite sample performance for dynamic clustering using synthetic time series. We also illustrate the DR procedure in application to data on water quality in Chesapeake Bay for a 32-year period (1985–2016).

The paper is organized as follows. The new stability probe approach, DR algorithm, is presented in Section 2.2. In Section 2.3, we discuss TRUST and DBSCAN, i.e., the two primary clustering methods we focus on. The proposed DR algorithm is then evaluated by extensive numerical studies in Section 2.4. We illustrate applications of new Downhill Riding (DR) procedure to analysis of the yearly temperature records in Central Germany in

Section 2.5 and the water quality in Chesapeake Bay in Section 2.6. The paper is concluded by discussion in Section 2.7.

2.2 Downhill Riding Procedure with Clustering Stability Probe

Preliminaries Let Ω_N be the observed data set that contains N multivariate items, i.e., $\Omega_N = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N\}$ and $\mathbf{a}_i = (a_{i1}, a_{i2}, \ldots, a_{it})^T$, $1 \leqslant i \leqslant N$. (For instance, $\mathbf{a}_i$ may represent an i-th observed time series up to time point t.) Our goal is to partition Ω into subsets $C_1, C_2, \ldots, C_K$ such that $\bigcup_{k=1}^{K} C_k = \Omega_N$ and $C_i \bigcap C_j = \emptyset$ for $i \neq j$. A number of clusters K is unknown a-priori. To achieve such partition C, we use a clustering algorithm $M(\nu, \Omega)$, where ν is a generic notation for a set of tuning parameters that controls partitioning of Ω, and is usually pre-specified by user. (For simplicity, we start from a case of a single parameter ν but the idea can be extended to a more general case.) The resulting clustering performance is typically evaluated using standard information criteria such as the Normalized Mutual Information (NMI), the Jaccard Index, or Rand Index (RI, Wagner and Wagner, 2007; Meilă, 2007).

The Downhill Riding (DR) Algorithm As discussed by Kogan et al. (2006), choice of a tuning parameter ν, such as Eps in DBSCAN or steepness parameter ξ in OPTICS, may substantially impact the resulting partitioning C. How can we choose ν in an objective manner while achieving the optimal clustering performance? In a nutshell, our intuitive idea is to look at the stability of the number of detected clusters as indicator for the underlying "ground truth".

In particular, let us select the number of clusters $\hat{K}$ as a clustering probe; obviously, $\hat{K}$ is a function of ν and Ω_N (i.e., $\hat{K}(\nu, \Omega_N)$). Suppose that Ω_N is a sufficiently large data set such that each true cluster is well represented in Ω_N. We now randomly split Ω_N into two subsets $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ of equal cardinality. If we partition $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ using the same clustering algorithm $M(\nu, \cdot)$, we intuitively expect that, if the tuning parameter ν is selected correctly, such partitions should be relatively similar, homogeneous or, at least,

$$|\hat{K}(\nu, \Omega^1_{N/2}) - \hat{K}(\nu, \Omega^2_{N/2})| \approx 0. \tag{2.1}$$

Hence, by viewing (2.1) as a function of ν, we can look at its minimum as indicator of correctly selected parameter ν. We define the function in (2.1) as the Cluster Deviation:

$$CD(\nu) = |\hat{K}(\nu, \Omega^1_{N/2}) - \hat{K}(\nu, \Omega^2_{N/2})|.$$

Smaller CD indicates more steady clustering performance. However, there exist two extreme scenarios when $CD(\nu) \approx 0$. First, when all $N/2$ items in $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ are partitioned into $N/2$ individual clusters. Second, when all data are grouped into a single cluster. Hence, we search for the local minimum in $CD(\nu)$ as the indicator of "truth". Since estimation uncertainty due to a single split of Ω_N into $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ might be high, we use the V-fold cross-validation procedure with multiple splits and define a new measure for the stability of a clustering algorithm, Average Cluster Deviation (ACD):

$$ACD(\nu) = \frac{1}{B} \sum\nolimits_{b=1}^{B} \left| \hat{K}(\nu, \Omega^1_{N/2}, b) - \hat{K}(\nu, \Omega^2_{N/2}, b) \right|,$$

where B is the number of splits in cross-validation, and $\hat{K}(\nu, \Omega^1_{N/2}, b)$ and $\hat{K}(\nu, \Omega^2_{N/2}, b)$ are the number of clusters delivered by M in application to the b-th split of $(\Omega^1_{N/2}, b)$ and $(\Omega^2_{N/2}, b)$. The *optimal empirical estimate* $\hat{\nu}^e$ is the parameter ν corresponding to the local minimum of ACD (see Algorithm 1).

Note that the idea is intrinsically linked to the notion of clustering (in)stability (Wang, 2010; Ben-David et al., 2006; Ben-David and Von Luxburg, 2008; Dudoit and Fridlyand, 2002; Bubeck and Luxburg, 2009; Von Luxburg, 2010). However, in contrast to the earlier stability approaches, we do not aim to evaluate closeness of cluster assignments of each observation but focus on a distance between probes.

Algorithm 1: Downhill Riding (DR)

```
**Input** : $\Delta = \{\nu_n, n = 1, 2, \ldots, L\}$, $\Omega$, $B$
**Output**: *optimal empirical estimate* $\hat{\nu}^e$
for each $\nu_n \in \Delta$ do
    Compute $ACD(\nu_n)$;
    $\mathbb{A} \leftarrow \{\mathbb{A} \cup ACD(\nu_n)\}$;
end
Find $\nu_n^*$ such that $ACD(\nu_n^*)$ is a local minimum in $\mathbb{A}$;
if the number of $\nu_n^*$ equals 1 then
    return $\nu_n^*$;
else
    return $\arg\min \nu_n^*$;
end
```

To get an initial validation insight into this idea, we now consider a relationship between ACD and NMI (Vinh et al., 2010; Rand, 1971), which is a robust evaluation metric especially when the number of cluster is large. Given a set Ω_N of N observations, let us consider two partitions of Ω_N, namely $U = \{U_1, U_2, \ldots, U_R\}$ with R clusters, and $V = \{V_1, V_2, \ldots, V_C\}$ with C clusters. NMI is defined as:

$$NMI(U, V) = \frac{MI(U, V)}{(H(U) + H(V))/2}, \tag{2.2}$$

where $H(U) = -\sum_{i=1}^{R} P(i) \log(P(i))$, $MI(U, V) = \sum_{i=1}^{R} \sum_{j=1}^{C} P(i, j) \log \frac{P(i,j)}{P(i)P'(j)}$, $P(i) = |U_i|/N$, $P'(j) = |V_j|/N$, and $P(i, j) = |U_i \cap V_j|/N$. NMI has a range $[0, 1]$ with larger values of NMI indicating better clustering performance.

Figure 2.1 shows the dynamics (aggregated and for a single synthetic data set) of ACD and NMI in application to the TRUST clustering algorithm (Section 2.3.1). Figure 2.1 suggests that the local minimum for ACD indeed is well aligned with the global maximum of NMI. Note that as expected, ACD is close to 0 at lower or higher values of ν. Lower values of ν tend to correspond to a higher number of clusters, up to an extreme case of each sample forming a single cluster, which leads to lower (or even zero) values of ACD but also lower NMI. At the same time, higher values of ν lead to a lower number of clusters, up to an extreme case of all samples being in one group, which again leads to lower (or even zero) values of ACD but low NMI. Based on the $\bigwedge$-shape of ACD and our search for its right-hand side minimum, we call our algorithm a *Downhill Riding (DR)* procedure.

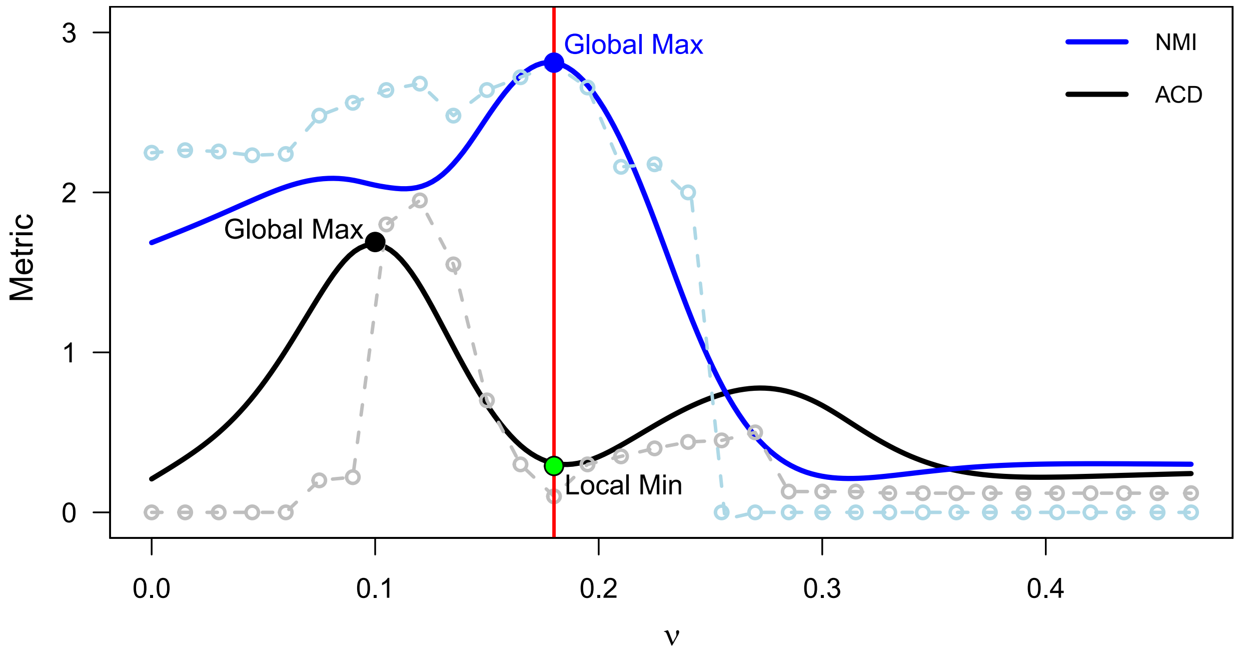


Figure 2.1. Aggregated dynamics of NMI and ACD (smooth lines) and dynamics for a single data realization (dashed lines) with clustering by TRUST. The metrics are scaled to fit on one graph. The Downhill Riding (DR) procedure selects the TRUST tuning parameters υ (e.g., the value-similarity threshold δ) corresponding to the first right-hand side local minimum of ACD.

The DR procedure discussed above is defined by (Huang et al., 2016) for the optimal selection of a single parameter ν with a searching space of one-dimensional array. However, most dynamic clustering algorithms have two tuning parameters rather than one. The DR procedure can be extended to the optimal selection of two parameters ν_1, ν_2 with a searching space of a two-dimensional matrix. Specifically, for each combination of values of ν_1 and ν_2 (i.e., each entry in the two-dimensional matrix of searching space), an ACD can be calculated. The procedure produces a two-dimensional surface of ACD rather than a one-dimensional curve. A saddle point is defined as the point where the value of ACD is the smallest in a predefined neighborhood. The saddle point for the ACD surface, an analogy to the local minimum for one-dimensional ACD curves, is used to select the optimal parameters. Note that there is a possibility of non-existence of these saddle points in the implementation. In such cases, the smallest non-zero value in the ACD surface can be selected as the optimal position, due to the fact that zero values of ACD correspond to extreme cases of each sample forming a single cluster or all samples being in one group (Nykamp, 2016; Cormen et al., 2009).

Asymptotics Properties To proceed with theoretical properties of the DR procedure, we adopt notions of clustering consistency and stability discussed by (Bubeck and Luxburg, 2009) and (Ben-David et al., 2006).

Definition Assume that the observed data Ω_N has been sampled from an underlying population Ω according to some probability measure P. Let Q be a clustering loss function on the set $\mathbb{S}$ of all partitions of the population Ω. Let $C^*(\Omega)$ be a unique minimizer of Q. A clustering algorithm $M(\nu)$ is called *asymptotically consistent* if it delivers a partition $C(\Omega_N)$ such that $Q(C(\Omega_N))$ converges to $Q(C^*(\Omega))$ as $N \to \infty$. Note that this concept is intrinsically connected to uniqueness of optimal partitioning of a population set, discussed by (Pollard, 1981) in a context of k-means clustering.

Proposition 1. *Let M be an asymptotically consistent clustering algorithm such that $Q(C(\Omega_N))$ converges to $Q(C^*(\Omega))$ at rate $r_{N,k}$ where $r_{N,k}$ is a nonincreasing sequence of positive numbers. Let ν^* be a a value of a tuning parameter ν that delivers $M(\Omega) = C^*$. Then, in probability*

$$\hat{\nu}^0 \underset{N\to\infty}{\to} \nu^*,$$

where $\hat{\nu}^0$ is the argument of the local minimum of the oracle loss function, or the Expected Cluster Deviation,

$$E\left|\hat{K}(\nu, \Omega^1_{N/2}, b) - \hat{K}(\nu, \Omega^2_{N/2}, b)\right|.$$

Proof of Proposition 1 is approached similarly to Theorem 1 of (Wang, 2010).

Now, let $\hat{\nu}^e$ be the empirical counterpart of $\hat{\nu}^0$, that is,

$$\hat{\nu}^e = \arg\min \frac{1}{B} \textstyle\sum_{b=1}^{B} \left| \hat{K}(\nu, \Omega^1_{N/2}, b) - \hat{K}(\nu, \Omega^2_{N/2}, b) \right|.$$

The next proposition states that difference between $\hat{\nu}^e$ and $\hat{\nu}^0$ is asymptotically negligible.

Proposition 2. *Let $\Omega_N = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N\}$ be a sample from an underlying population Ω, where $\mathbf{a}_i$, $i = 1, \ldots, N$ are mutually independent random vectors. Let M be an asymptotically consistent clustering algorithm, and let $C^*(\Omega) = \{C^*_1, \ldots, C^*_K\}$ be the true clustering of Ω. Let N_i be the number of items in Ω_N corresponding to the true cluster C^*_i in Ω. Then, if $K \ll N_i$, $N \to \infty$, and $N_i \to \infty$*

$$|\hat{\nu}^e - \hat{\nu}^0| \to 0,$$

in probability.

The proof of Proposition 2 is based on the Chebyshev inequality and approached in a similar manner as Theorem 3 by (Bickel and Gel, 2011).

2.3 Clustering Algorithms

We now discuss the two main clustering algorithms that we illustrate application of the Downhill Riding procedure to, that is, TRUST and DBSCAN.

2.3.1 The TRUST Algorithm

The TRUST algorithm is an unsupervised clustering algorithm designed for space-time data streams by Ciampi (Ciampi et al., 2010; Appice et al., 2015). Specifically, TRUST integrates spatial clustering and temporal trend detection with a goal to continuously group geo-referenced data according to a similar temporal trajectory in time $t_1, t_2, \ldots, t_p$, where p is pre-specified by the user. TRUST has the following advantages:

- does not require a number of clusters a-priori as opposed to k-means;
- can detect arbitrarily shaped clusters;
- can dynamically detect the drift of space-time data distributions by using a sliding window moving from past to recent.

Let the data be stored in a matrix $\Omega = [\mathbf{a}_1, \dots, \mathbf{a}_N]$, where $\mathbf{a}_t = [a_{t1}, \dots, a_{tm}]^\mathsf{T}, (1 \leqslant t \leqslant N)$ corresponds to observed time series data from m sensor devices at a time point t. Each $\mathbf{a}_t\,(1 \leqslant t \leqslant n)$ is called a *layer* (corresponding to a time point) and several layers constitute a *slide* (corresponding to a time period). We segment data set Ω into several slides such that TRUST performs clustering on each slide (*slide-level* clustering). Then a sliding window, moving from past to recent, generates slide-level clustering sets to obtain final trend-clusters (*window-level* clustering).

The core clustering performed by TRUST is *slide-level* clustering. The TRUST algorithm splits the data Ω into segments of p time points, and then divides the m time series, each of length p, into clusters. Specifically, for one slide $\Omega_1 = [\mathbf{a}_1, \dots, \mathbf{a}_p]$, where $\mathbf{a}_t = [a_{t1}, \dots, a_{tm}]^\mathsf{T}, (1 \leqslant t \leqslant p)$ and $p < N$, TRUST randomly starts with one time series u as an initial point (seed time series), and searches time series v for close relations in terms of E_δ^θ (referred to as neighbors). This is formalized as:

$$E_\delta^\theta = \left\{ \langle u, v \rangle \in E \,\middle|\, \sum_{i=1}^{p} \psi_\delta(u[t], v[t]) \geqslant \theta \times p \right\},$$

where

$$\psi_\delta(u[t], v[t]) = \begin{cases} 1, & \text{if } \frac{\|u[t]-v[t]\|_1}{\beta-\alpha} \leqslant \delta \\ 0, & \text{otherwise} \end{cases}$$

where E is a set of time series, $u[t]$ ($v[t]$) is the value at t-th time point, θ is the slide-level trend continuity threshold in $[0, 1]$, p is the slide size, $[\alpha, \beta]$ is the domain of slide Ω_1, and δ is the value-similarity threshold in $[0, 1]$.

An initial cluster is formed by the seed time series and its neighbors. Each neighbor in this cluster is recursively chosen as a seed time series and applied to the same neighborhood construction described above. Initial clusters are merged to form a bigger one if they share some time series. Once all time series are classified into a cluster, the algorithm stops by returning Γ as a slide-level clustering set for slide Ω_1. The slide size p, slide-level trend continuity threshold θ, and value-similarity threshold δ need to be pre-specified by users when TRUST is run. (For more details on pseudo code of TRUST algorithm and definitions of concepts see (Ciampi et al., 2010), an R code is available from (Lyubchich and Gel, 2017b).)

2.3.2 The DBSCAN Algorithm

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) of (Ester et al., 1996) and (Kogan et al., 2006) is one of the most widely used clustering algorithms and recipient of the 2014 SIGKDD Test of Time Award. Similar to TRUST, DBSCAN does not require a pre-defined number of clusters and can detect arbitrarily shaped clusters. The core idea of DBSCAN is as follows: given a set of points in some space, it groups together points that are in a high-density region (i.e., neighbors of the points are close to each other), and marks points as outliers if they lie alone in low-density regions (whose nearest neighbors are far away). DBSCAN requires two parameters: the maximum radius of the neighborhood, Eps, and the minimum number of points required to form a dense region, minPts. Selection of both Eps and minPts is typically performed in a subjective manner.

2.4 Numerical Experiments

2.4.1 Benchmark Iris Data

We evaluate the performance of Downhill Riding by applying it to DBSCAN and comparing its performance against DBSCAN with pre-selected Eps on real benchmark data Iris (Fisher,

Table 2.1. Performance of DBSCAN on Iris data in terms of NMI, with Eps-opt selected using Downhill Riding and with Eps-kdist selected by conventional *k-dist graph*.

MinPts	Eps-opt	Eps-kdist
5	0.76	0.61
6	0.76	0.60
7	0.76	0.58

1936; Bache and Lichman, 2013). Despite the popularity of DBSCAN in spatial clustering, determination of its parameter Eps is mostly based on heuristic and non-automatic methods – *sorted k-dist graph* (Ester et al., 1996). Using Iris data, which contains 150 samples with 4 variables for 3 clusters, we compare the performance of DBSCAN with Eps selected using Downhill Riding, and with conventional selection (through *sorted k-dist graph*). The pre-selected Eps is set as 0.5 (Hahsler, 2015). As evaluation metric we consider Normalized Mutual Information (NMI). Table 2.1 shows the comparative results: for each possible value of parameter MinPts, DBSCAN with Downhill Riding outperforms DBSCAN with the conventional selection. This is a notable result: an algorithm using our data-driven mechanism for selection of optimal clustering parameters outperforms the same algorithm that uses predefined parameters based on a-priori knowledge.

2.4.2 Synthetic data

To further evaluate the performance of the Downhill Riding procedure, we proceed with a series of Monte Carlo simulations, where the performance with TRUST is evaluated with the selection of two parameters. (For a detailed discussion of a single parameter selection with TRUST and DBSCAN and an extensive comparison study, see (Huang et al., 2016).)

We produce a data stream of 16 time series, denoted by Y, and obtained by sequencing two consecutive periods (slides) of 60 time points (layers). The cluster configurations of Y are shown in Table 2.2.

Table 2.2. Cluster configuration of 16 simulated time series.

Time series model	Time series
AR(1), $\phi_1 = 0.2$, $\varepsilon_t \sim N(0,1)$	1, 5, 4, 12
AR(2), $\phi_1 = 0.1, \phi_2 = -0.2$, $\varepsilon_t \sim N(1,1)$	3, 9, 11, 15
MA(1), $\theta_1 = 0.4$, $\varepsilon_t \sim N(2,1)$	10, 13, 14, 16
ARMA(1,1), $\phi_1 = 0.3, \theta_1 = 0.2$, $\varepsilon_t \sim N(3,1)$	2, 6, 7, 8

To evaluate the finite sample performance of Downhill Riding procedure with TRUST, we calculate ACD and Rand Index with different values of the similarity threshold δ and the connectivity parameter θ (Table 2.3). To compute the Rand Index, a cluster label cl_1 returned by TRUST and the ground truth cluster label cl_2 are needed. Let A be the number of pairs of time series in Y that are in the same cluster in both cl_1 and cl_2, B be the number of pairs of time series that are in different clusters in both cl_1 and cl_2, C be the number of pairs of time series in the same cluster in cl_1 but in different clusters in cl_2 and D be the number of pairs of time series in different clusters in cl_1 but in the same cluster in cl_2. Then the Rand Index can be computed as follows:

$$Rand\ Index = (A + B)/(A + B + C + D). \tag{2.3}$$

The Rand Index has a range $[0, 1]$ with larger values indicating better clustering performance. In contrast, smaller values of ACD indicate more steady clustering performance. The number of detected clusters for TRUST corresponds to slide-level clustering by setting slide size $p = 60$.

We find that TRUST parameters δ_{opt} and θ_{opt}, selected using Downhill Riding, are close to δ_{oracle} and θ_{oracle} and yield the Rand Index comparable to the highest empirically achievable (Table 2.3). These findings show that our automatic data-driven parameter selection

Table 2.3. Performance of TRUST with value similarity threshold δ_{opt} and connectivity parameter θ_{opt} selected by Downhill Riding and TRUST with δ_{oracle}, θ_{oracle}. Number of Monte Carlo experiments is 100. Number of cross-validation splits T is 300.

	δ_{opt},θ_{opt}	δ_{oracle},θ_{oracle}
Average RI	$0.80_{(0.13)}$	$0.94_{(0.04)}$
Average ACD	$1.48_{(1.00)}$	$2.54_{(0.33)}$

procedure tends to deliver close to the empirically achievable levels of clustering performance despite the lack of a-priori knowledge.

Combined with the results from the benchmark Iris data study above, the results show that our automatic procedure is not just on par with the competition that has the informational advantage, but at times better. These findings imply that the new DR algorithm can be particularly useful in studies where there is no knowledge of the parameters or number of clusters, such as when exploring environmental, insurance, or social science data, without imposing considerable performance trade-offs.

2.5 Case Study I

Observed temperature data We applied TRUST and DBSCAN to yearly temperature records from 167 weather stations in Central Germany in a 60-year period 1951–2010 (Deutscher Wetterdienst Data Archive, 2015). Analyzing temperature data for such a long period can provide some important insights into climate change and the differences of these effects in the various geographic areas. The controlling parameters δ for TRUST and Eps for DBSCAN are set as 0.036 and 6.5 by "Downhill Riding". We select 15-year intervals as a time period to perform clustering since the climate of Europe exhibits cycles of 12-16 years (Vines, 1985). Thus, the 60 year temperature data is segmented into 4 non-overlapped time periods, each of which is clustered by TRUST and DBSCAN, respectively. We set the layer size p as 40, the window size ω as 3 (with step size as 3), and set MinPts of DBSCAN as 3.

The clustering results based on TRUST and DBSCAN show similar patterns where elevation is a dominant factor: elevation of weather stations – one of the key factors in temperature differences – is found to be relatively homogeneous within each cluster. Figure 2.2 shows the results of TRUST clustering in time period 4 in a topographic map. The contour lines show places of equal elevation. Different clusters are labeled with different colors. The weather stations in the yellow cluster are mostly located in areas below 300 m; while the weather stations in the red cluster are mostly located in areas around 500-600 m. The fact that elevation strongly affects temperature is well known in climate sciences. Hence, we are interested to investigate potential less explicit latent factors affecting temperature dynamics and segmentation.

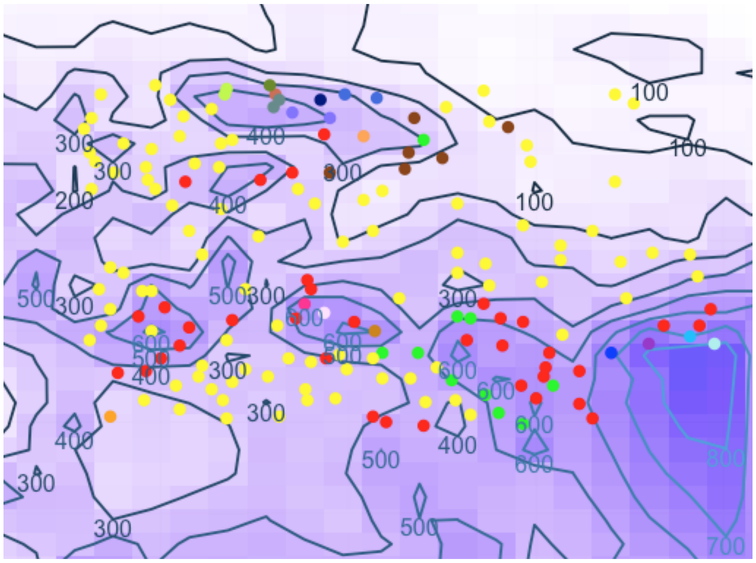


Figure 2.2. Clustering of weather stations in time period 4 (year 46-60) by TRUST.

Elevation Scaled temperature data We now consider elevation scaled temperature where the impact of elevation has been removed according to (Barry, 1992; Daly et al., 2008). In particular, let X be elevation and Y be temperature, then:

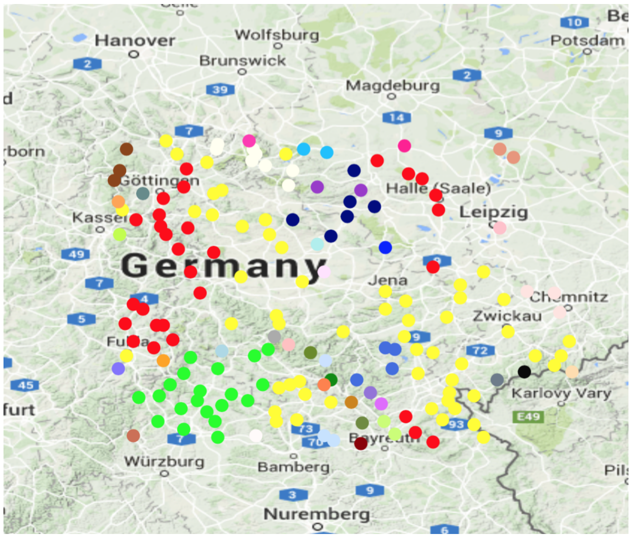
$$Y_n^t = \beta_{t0} + \beta_{t1} X_n + \epsilon_n^t, n = 1, 2, \ldots, 167, t = 1, 2, \ldots, 60 \tag{2.4}$$

The residuals ϵ_n^t $(n = 1, 2, \ldots, 167, t = 1, 2, \ldots, 60)$ from linear regression are combined into a new data set where TRUST and DBSCAN are applied with the same framework setting as in previous temperature data clusterings. Optimal Downhill Riding parameters are δ and Eps, set as 0.1 and 0.65.

The resulting patterns are different from the ones observed in the temperature clustering. Figure 2.3a and Figure 2.3b show the clustering results for periods 1 and 4 by TRUST in terrain maps, respectively. Climate stations in Halle (Saale) area are grouped together in both time periods 1 and 4 (red dots in Figure 2.3a and navy dots in Figure 2.3b). Average residuals of the two clusters are 0.2 and 0.6 respectively, which makes sense because Halle (Saale) area corresponds to the dry region of Central Germany. In addition, a handful of stations north/northwest of Karlovy Vary show unique patterns: individual weather stations form their own mini-clusters. For example, black and grey dots in Figure 2.3a, and light pink, rosy dots in Figure 2.3b. These weather stations are all in a part of a mountain range called the Ore Mountains where they are probably located in fairly unique topographic situations, e.g. mountain top, or valley. In mountain areas, the orientation of a valley can have a large influence on the movement of air masses, so valleys of different orientations may be distinct enough to be placed in different clusters. Similar patterns are observed in the DBSCAN clustering results depicted in Figure 2.4.

Between periods 1 and 4, we observe how the cluster patterns change dynamically. Weather stations in the southwest area are grouping into one big yellow cluster, changing from a mean residual -0.23 for the red cluster and -0.01 for the green cluster in period 1 to -0.02 for the yellow cluster in period 4. And weather stations in the west are grouping into another cluster (red) with a mean residual 0.12. Remarkably, the TRUST algorithm identifies partly changing clusters of temperature residuals in the early and late periods. Spatially

(a) Period 1 (Year 1-15).



(b) Period 4 (Year 46-60).

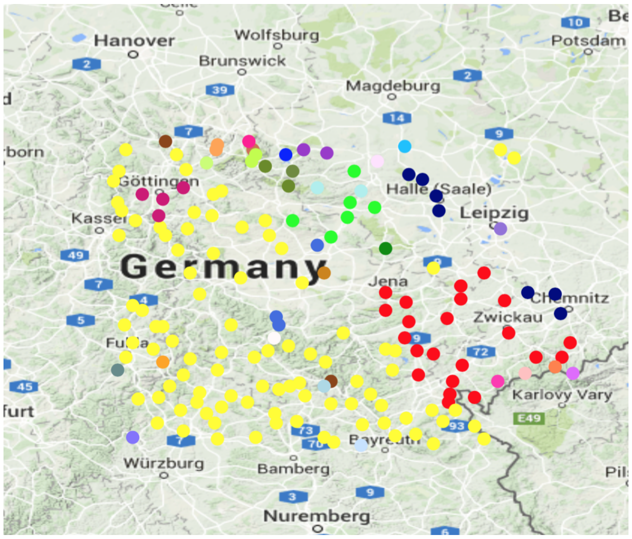
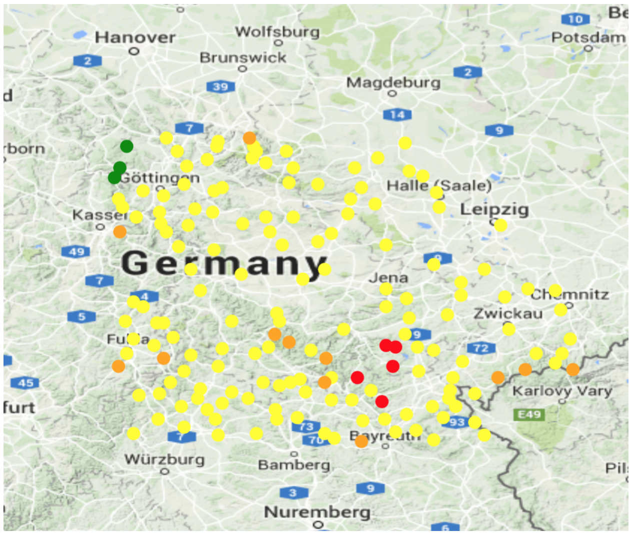


Figure 2.3. Clustering of weather stations in time period 1 and 4 by TRUST.

varying climatic changes have been observed elsewhere before (Anisimov et al., 2013); such patterns that would explain these observed changes in clustering may potentially be related to the complexity of topography in the studied region (orographic effects), changes in cloud

(a) Period 1 (Year 1-15).



(b) Period 4 (Year 46-60).

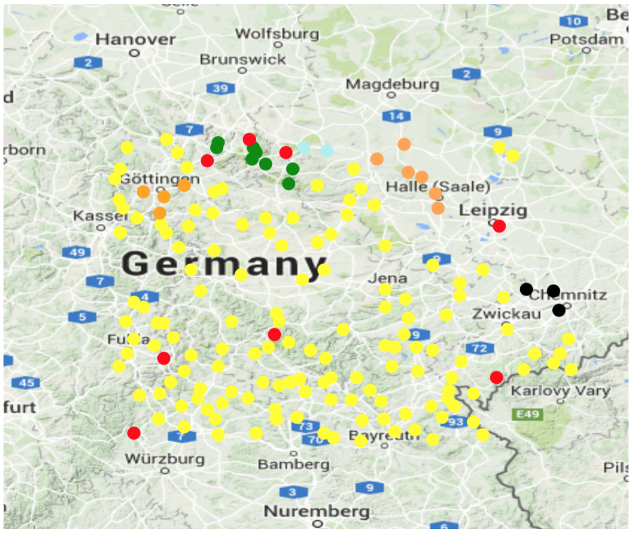


Figure 2.4. Clustering of weather stations in time period 1 and 4 by DBSCAN.

cover and atmospheric dust content due to reduced industrial emissions first in West and later in East Germany, or confounding with spatially varying changes in precipitation, for example. While such explanations are not immediately evident from the clusters produced

Table 2.4. MAPE of 4 time periods by TRUST and DBSCAN on observed temperature data.

TRUST MAPE (Year 1-15)	TRUST MAPE (Year 16-30)	TRUST MAPE (Year 31-45)	TRUST MAPE (Year 46-60)
0.07	0.07	0.06	0.03
DBSCAN MAPE (Year 1-15)	**DBSCAN MAPE (Year 16-30)**	**DBSCAN MAPE (Year 31-45)**	**DBSCAN MAPE (Year 46-60)**
0.11	0.11	0.10	0.09

Table 2.5. MAPE of 4 time periods by TRUST and DBSCAN on scaled temperature data.

TRUST MAPE (Year 1-15)	TRUST MAPE (Year 16-30)	TRUST MAPE (Year 31-45)	TRUST MAPE (Year 46-60)
0.63	1.12	1.10	0.94
DBSCAN MAPE (Year 1-15)	**DBSCAN MAPE (Year 16-30)**	**DBSCAN MAPE (Year 31-45)**	**DBSCAN MAPE (Year 46-60)**
1.69	1.28	1.74	1.75

by TRUST, this knowledge discovery technique provides a starting point for further climatological analyses of local patterns of climate change. Knowledge of the existence and location of regions with homogeneous patterns may furthermore be instrumental in the geostatistical interpolation of instationary random fields of climatic parameters (Guinness et al., 2013).

MAPE values for 4 periods of observed temperature data and scaled temperature data are shown in Table 2.4 and Table 2.5. TRUST outperforms DBSCAN in each of the 4 periods (with smaller MAPE) on both observed temperature data and scaled temperature data.

2.6 Case Study II

In this section, we illustrate the suggested approach of finding the optimal classification with TRUST by applying it in a spatio-temporal analysis of water quality. We use the example of Chesapeake Bay, which is one of the most important estuaries in the USA in terms of its size, ecosystem diversity, and economic impact of the developed fishing and tourism industries.

At the same time, the bay has been known for severe problems of water pollution, which negatively affect the populations of bay inhabitants.

The Chesapeake Bay Agreement of 1983 laid down a basis for the Chesapeake Bay Program to monitor, improve, and protect the water quality and living resources of the Chesapeake Bay. As one of the results, we are able to use the publicly available dataset[2] of water quality parameters recorded at more than a hundred monitoring stations spread out throughout the watershed. Particularly, we consider pollution with suspended matter (suspended solids or sediment), the primary sources of which in the Chesapeake bay watershed are agriculture and urban runoff (59.8% and 23.9% of the total suspended solids, TSS, load in 2015[3]).

The largest spatial extent of the continuous records is available since 1985, thus, we use 32 years (1985–2016) of surface TSS concentrations (mg/L) from 133 stations. Biweekly measurements are aggregated to monthly averages; missing values (approximately 12% of the data points) are filled in using information from other stations in the radius of 15 km (considering stations only in the same tributary, not crossing the land) or up to the nearest station, whichever is greater. In the analyzed period, the method for measuring TSS concentrations did not change, however, many monitoring stations experienced changes of the laboratories, which might have slightly different implementations of the same measuring method. To account for the shifts possibly caused by the change of analytical laboratories at each station, we adjusted the pre-change measurements by the difference of medians calculated for one year before and after the change.

Similarly to (Schaeffer et al., 2016), we split the data in two sub-periods of 16 years each (1985–2000 and 2001–2016) that correspond to the time before and after adopting the

---

[2] http://data.chesapeakebay.net/WaterQuality [Accessed on January 13, 2017]

[3] http://chesapeakeprogress.com/clean-water/water-quality/watershed-implementation-plans [Accessed on January 23, 2017]

Chesapeake 2000 agreement. This agreement brought a new wave of restoration activities to the bay watershed, and here we assess how it changed the space-trend panorama.

Since we are more interested in trends rather than particular TSS values, we scale the data for each station to zero mean and unit variance. The TRUST clustering procedure with Downhill Riding parameter selection is applied to each of the two non-overlapping sub-periods of 16 years. The controlling TRUST parameters δ and θ selected by Downhill Riding are 0.16 and 0.63 for 1985–2000; and 0.15 and 0.63 for 2001–2016.

The results yield one dominant cluster (comprises about half of the stations) and several small clusters in each sub-period. Many of the small clusters contain only one station. Notably, Figure 2.5 shows clear groupings of the stations, even though the time series were scaled and no spatial information (longitude and latitude) was supplied to the algorithm. Similar dynamics in each cluster, however, does not imply statistical significance of the trends, which should be assessed separately by other methods.

The size of the dominant cluster reduced from 77 stations in 1985–2000 to 59 stations in 2001–2016 (stations from this cluster are shown in black in Figure 2.5). To assess the cluster dynamics, we obtain an aggregated series as a median of concurrent values in the cluster and apply loess smoothing. Figure 2.6 shows that time series from the main cluster exhibit almost no change in the long-term perspective, except the variance declined in the second sub-period. The fact that this cluster decreased in size tells us that the overall dynamics became more disparate.

Remarkably, by comparing which black dots in Figure 2.5 turned out to be colored in 2000–2016 (i.e., exited the main cluster), we notice that the main changes occur in the bay tributaries and partly in the upper bay, which can be seen as an effect of implementing the restoration activities (restoration activities take place in the streams, not in the bay itself). Examples include the Rappahannock, York, and James rivers. Thus, the tidal fresh Rappahannock River stations (TF3.1B, TF3.1E, and TF3.1F; marked with "*" in

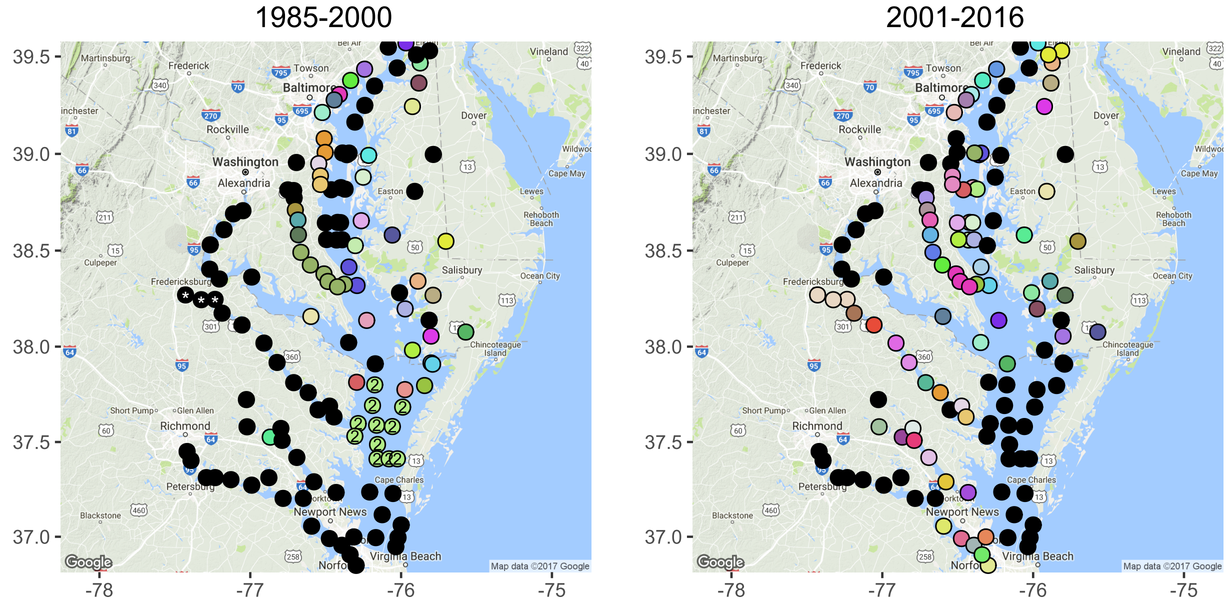


Figure 2.5. Clustering of the Chesapeake Bay water quality monitoring stations using TRUST, with cluster assignments denoted by color. Black indicates the most populous cluster in each sub-period; "2" denotes the second populous cluster in 1985–2000, and "*" denotes upper Rappahannock River stations.

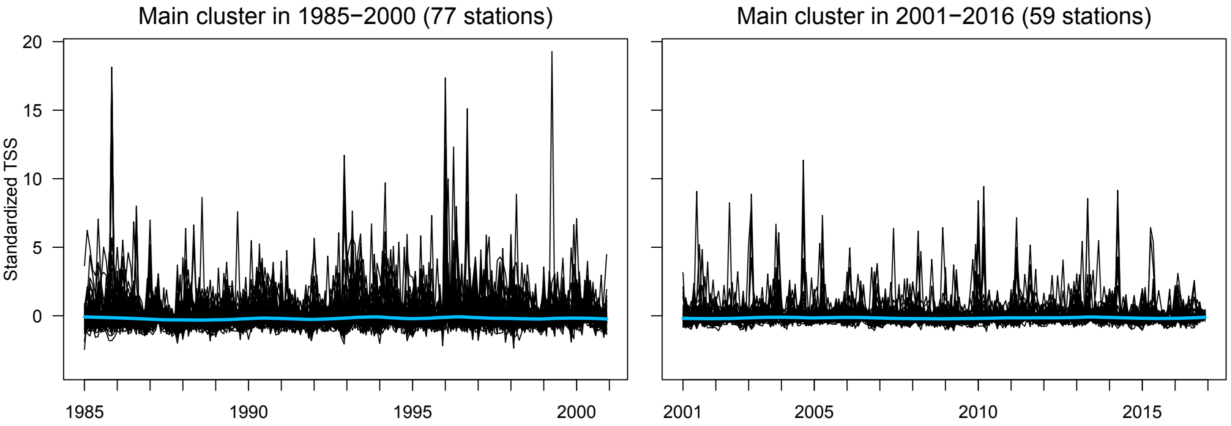


Figure 2.6. Time series of the standardized TSS concentrations in the main cluster, along with the loess smoothing curves.

Figure 2.5) belong to the main cluster in 1985–2000, but exhibit a non-linear dynamics of TSS concentrations in the later years. The rise in 2001–2002 then steady decline of concentrations in these three stations (Figure 2.7) do not match the flat loess curve in Figure 2.6 and make these stations cluster separately. Similar case is observed in lower James River, with a jump and decline of TSS concentrations in 2011–2016.

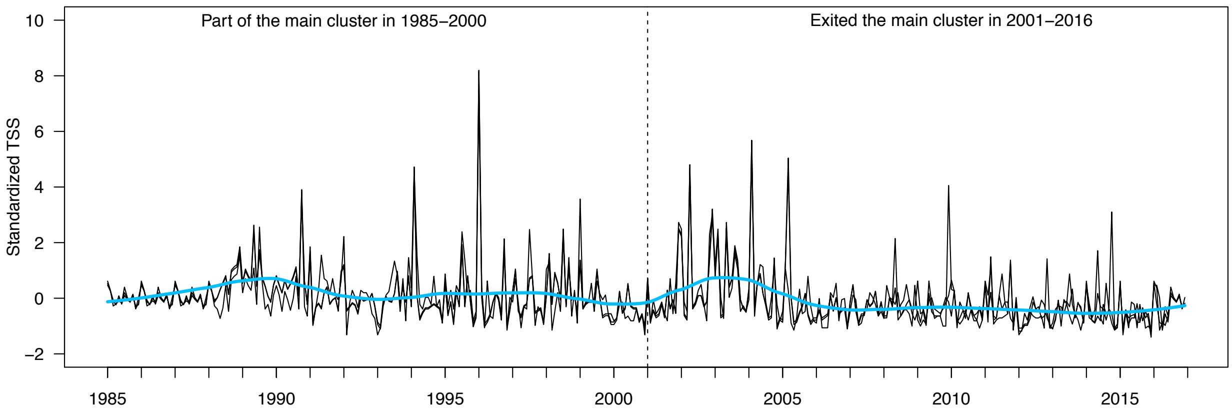


Figure 2.7. Time series of the upper Rappahannock River stations (marked with "*" in Figure 2.5) that exited the main cluster in 2001–2016, along with a loess smoothing curve.

A different example is the cluster 2 in 1985–2000 (Figure 2.5), which joined the main cluster in 2001–2016. Dynamics at these 11 stations is characterized by profound stabilization of the concentrations, both in terms of their mean and variance. Figure 2.8 shows how increasing trend changed its direction and matched the main cluster behavior with no or subtle long-term changes.

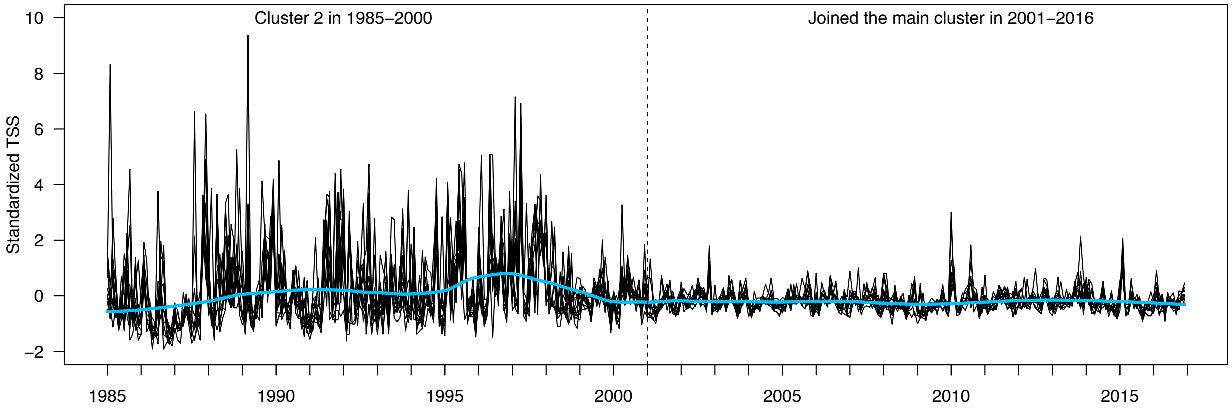


Figure 2.8. Time series of cluster 2 (see Figure 2.5) that joined the main cluster in 2001–2016, along with a loess smoothing curve.

Overall, the monthly TSS concentrations in the main cluster exhibit little or no central tendency, however, the variance visibly decreased in the last 16 years (Figure 2.6), which can be seen as an effect of storm water best management practices developed for sediment and nutrient retention and reducing the discharge of pollutants into the streams, particularly, during the storm events. Clustering results help us to see the broader picture and joint dynamics across stations that now can be analyzed together, according to their cluster associations. The reasons and implications of changing cluster associations should be assessed individually for each group of stations. The considered examples of Rappahannock River and lower main stem demonstrate that both exiting and joining the main cluster can be associated with improvements of the water quality (decline of TSS concentrations).

2.7 Conclusion

In this paper we advance the idea of clustering (in)stability from a case of selecting a "true" number of clusters to a choice of multiple optimal tuning parameters in a broad range of dynamic clustering algorithms. We propose a new data-driven and computationally efficient procedure called Downhill Riding (DR) for optimal selection of clustering tuning parameters in dynamic clustering algorithms like TRUST and DBSCAN using a *clustering stability probe*. Using simulations, as well as real data, we show the effectiveness of the new procedure for selection of optimal parameters. The finite sample performance of DR for dynamic clustering of synthetic time series is close to the optimal for these algorithms. Furthermore, the performance of clustering algorithms using DR against competing algorithms that have a-priori knowledge of the parameters shows that our procedure is a viable alternative, and often performs better.

We also illustrate the Downhill Riding procedure in dynamic cluster detection in yearly temperature records among 167 stations in Central Germany over year period (1951–2010) and in monthly average concentrations of suspended solids across 133 stations in Chesapeake

Bay for a 32-year period (1985–2016). Particularly on the case of Chesapeake Bay, we find remarkable patterns in the data that can provide an insight into the management of resources in the area and the effects of the restoration activities over time. Based on our clustering results, we discover a dynamic pattern, which is useful when studying spatially varying ecological changes. The identification of clusters in the water quality in the Chesapeake Bay has a number of applications and can help address problems in the area. Identifying concentrations of pollutants can aid in determining sources of contamination and assessing which parts of the Bay are at risk. The clustering results provide a clearer picture of the environmental impact of various activities in the area, and can aid future restoration efforts in creating targeted interventions for specific parts of the Bay.

In the future, we plan to extend the use of the new DR procedure to other dissimilarity measures and stability probes, and investigate the utility of DR in other clustering algorithms.

CHAPTER 3

CRAD: CLUSTERING WITH ROBUST AUTOCUTS AND DEPTH[1]

3.1 Introduction

Data depth methodology is a widely employed nonparametric tool in multivariate and functional data analysis, with applications ranging from outlier detection to clustering and visualization (Liu et al., 1999; Cuevas et al., 2007; Li et al., 2012). Depth measures the "centrality" (or "outlyingness") of a given object with respect to an observed data cloud (Mosler, 2013; Zuo and Serfling, 2000). Many desirable properties of data depth such as affine invariance, robustness, and center maximality have earned it an increasing attention in the machine learning and statistics communities in the last decade. There exist numerous clustering and classification methods, based on a data depth concept (Jörnsten, 2004; Mosler, 2013; Pokotylo et al., 2016). Most such methods, however, rely on the knowledge of a true number of clusters k. Most recently, (Jeong et al., 2016) proposed a Depth Based Clustering Algorithm (DBCA) and showed benefits of a data depth for clustering spatial data. To the best of our knowledge, (Jeong et al., 2016) is the first and only reference introducing a data depth concept into clustering analysis of spatial data. However, DBCA cannot handle a case of clusters with varying densities, which is a common issue in density-based clustering domain. In addition, the problem how to select the tuning parameter, which highly impacts the clustering result, remains unstudied.

The current paper is motivated by three over-arching major challenges in density-based clustering: (1) Based on data depth, can we propose an algorithm that delivers more ro-

---

[1] This chapter includes verbatim excerpts from

©2017 IEEE. Reprinted, with permission, from Huang, X., Gel, Y. CRAD: Clustering with Robust Autocuts and Depth. *Proc. 17th IEEE International Conference on Data Mining (ICDM), 2017*, `https://ieeexplore.ieee.org/document/8215579/`

bust performance under the existence of clusters with varying densities? (2) Based on the proposed algorithm, how can we select the true underlying parameter in the real-world clustering when the ground truth is not given? (3) Can the density-based algorithm be extended to multivariate time-series clustering, without a-priori knowledge of the number of clusters? We address these three major problems by proposing a new clustering algorithm, named Clustering with Robust Autocuts and Depth (CRAD).

One of the key benefits of the new CRAD algorithm is its ability to detect clusters with varying densities. Let us start with a simple yet typical dataset to shed some light on the difference between our algorithm and some existing algorithms such as DBSCAN (Ester et al., 1996), OPTICS (Ankerst et al., 1999), and DBCA (Jeong et al., 2016) in addressing this type of problem. As shown in Figure 3.1(a), the toy dataset includes two dense clusters (clusters 1 and 2), and one sparse cluster (cluster 3). The number of observations in cluster 3 is larger than that in clusters 1 and 2. The result of each algorithm is selected by searching the best clustering performance on a wide range of possible combinations of its tuning parameters. Clustering results are shown in Figure 3.1. Currently available methods such as DBCA, DBSCAN, and OPTICS, all fail to separate the cluster 1 and 2; in contrast, our new CRAD algorithm is able to detect both. The reason for this phenomenon is that both DBSCAN and DBCA use globally-defined parameters (i.e., ϵ and θ, respectively) to find clusters, thus lacking the flexibility to adjust their value when clusters have different densities. Even OPTICS, which is proposed to solve this density variation problem, still does not deliver competitive clustering performance on the toy example. Our algorithm, in contrast, uses a *locally-defined* parameter to customize the neighbor searching function for each observation, based on a notion of density level. As a result, CRAD is able to deliver a competitive performance in separating clusters with varying densities.

This paper makes the following novel contributions to spatial and temporal clustering:

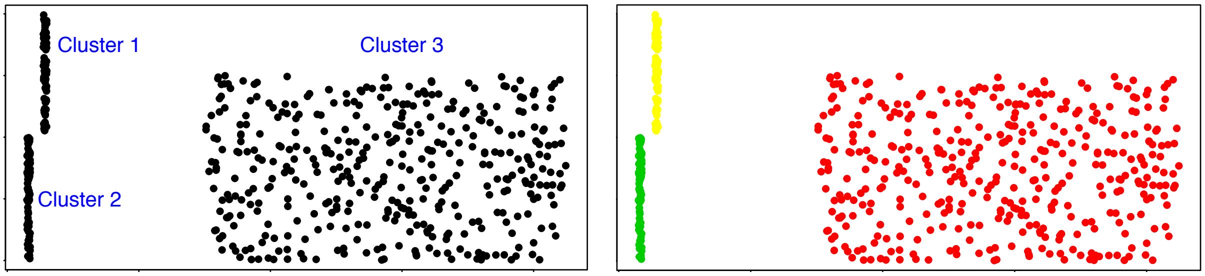
1. We propose a new robust density-based clustering algorithm (CRAD), using a notion of statistical *data depth* as the dissimilarity measure, and further augment the depth-based clustering analysis with an outlier-resistant and highly computationally efficient estimator of multivariate scale, namely, the Minimum Covariance Determinant (MCD). Our experiments prove that the new algorithm CRAD is highly competitive at detecting clusters with varying densities, compared with the existing algorithms such as DBSCAN, OPTICS and DBCA.

2. Furthermore, we show that a hybrid combination of our new robust depth-based neighbor searching algorithm and conventional DBSCAN, allows to significantly improve clustering performance of DBSCAN. This is an important standalone step toward future extension of DBSCAN to non-Euclidian spaces and functional data clustering.

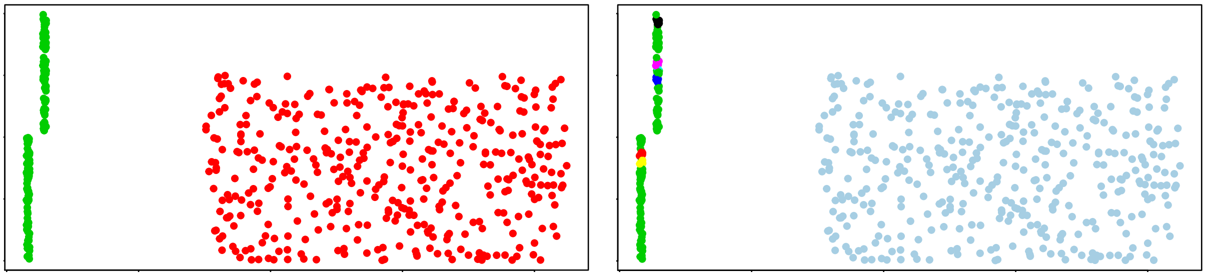
3. We develop a new effective parameter selection procedure to select the optimal underlying parameter in the real-world clustering, when the ground truth is unknown.

4. We suggest a new clustering framework that extends CRAD from spatial data clustering to time series clustering without a-priori knowledge of the true number of clusters. Performance of CRAD in time series clustering is evaluated with extensive experiments on benchmark data.

The paper is organized as follows. In Section 3.2 we present the new algorithm CRAD. In Section 3.3 an effective parameter selection procedure is proposed to select the parameters in CRAD. We evaluate CRAD through extensive numerical studies in Section 3.4. The paper is concluded with discussion and future research directions in Section 3.5.



(a) Raw Data (b) CRAD



(c) DBCA & DBSCAN (d) OPTICS

Figure 3.1. Clustering Performance of CRAD, DBCA, DBSCAN and OPTICS on the Toy Example.

3.2 Our Algorithm

We start from providing a direct insight into our algorithm, with a particular emphasis on introducing the distinguishing features of CRAD, namely, the dissimilarity measure and the neighbor searching function.

Before proceeding to details, we first review the general structure for density-based clustering algorithm. Let the data be stored as an $n \times p$-matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^t$ with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^t$ the i-th observation, and n be a sample size. The core idea behind all density-based algorithms is to assign a $\{0, 1\}$-relationship between all observations in $\mathbf{X}$, based on how close the two observations are, in terms of a given dissimilarity measure. That is, a neighbor searching function $\mathrm{NBR}(\mathbf{x}_i), i = 1, 2, \ldots, n$ is needed such that $\mathbf{x}_i$ and $\mathbf{x}_j$ are 1-related if $\mathbf{x}_j \in \mathrm{NBR}(\mathbf{x}_i)$ and 0-non-related if $\mathbf{x}_j \notin \mathrm{NBR}(\mathbf{x}_i)$. These results are stored in a $\{0, 1\}$-adjacency matrix A, and a breadth-first search is then applied to A to generate the final clustering partition of $\mathbf{X}$. What discriminates the clustering algorithms, however, is the *dissimilarity measure* and *neighbor searching function.* We present the distinguishing features of CRAD in terms of these two as follows.

3.2.1 A Robust Data Depth Based Dissimilarity

A data depth is a function that quantifies how closely an observed point $x \in \mathbb{R}^d$, $d \geq 2$, is located to the "center" of a finite set $\mathcal{X} \in \mathbb{R}^d$, or relative to a probability distribution P in $\mathbb{R}^d$. A data depth shall satisfy the following desirable properties (Mosler, 2013; Zuo and Serfling, 2000): affine invariant; upper semi-continuous in x; quasiconcave in x; (i.e., having convex upper level sets) vanishing as $||x|| \to \infty$ (Mosler, 2013; Zuo and Serfling, 2000).

We propose to utilize a robust Mahalanobis depth function, with the Minimum Covariance Determinant (MCD) as an outlier-resistant and highly computationally efficient estimator of multivariate scale, as an alternative clustering dissimilarity measure. That is, let the data be stored as an $n \times p$-matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^t$ with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^t$ the i-th observation, and n be a sample size. The Robust Mahalanobis depth function can be defined as:

$$RM_d(\mathbf{x}_j|\mathbf{x}_i) = [1 + (\mathbf{x}_j - \mathbf{x}_i)^T \mathbf{\Sigma}^{-1} (\mathbf{x}_j - \mathbf{x}_i)]^{-1}, \tag{3.1}$$

where

$$\mathbf{\Sigma} = c_1 \frac{1}{n} \sum_{i=1}^{n} W(d_i^2)(\mathbf{x}_i - \hat{\mathbf{u}}_{MCD})(\mathbf{x}_i - \hat{\mathbf{u}}_{MCD})^T,$$

and $\hat{\mathbf{u}}_{MCD} = \sum_{i=1}^n W(d_i^2)\mathbf{x}_i / \sum_{i=1}^n W(d_i^2)$; $d_i = \sqrt{(\mathbf{x}_i - \hat{\mu}_o)^T \hat{\mathbf{\Sigma}}_{\mathbf{o}}^{-1} (\mathbf{x}_i - \hat{\mu}_o)}$; W is an appropriate weight function; $\hat{\mu}_0$ and $\hat{\mathbf{\Sigma}}_o$ are sample mean and sample covariance matrix, respectively; and c_1 is a consistency factor (Hubert and Debruyne, 2010). The MCD covariance estimator has been proven to significantly outperform the Minimum Volume Ellipsoid (MVE) covariance estimator, that is used by (Jeong et al., 2016), both in terms of statistical efficiency and computation (see, e.g., Van Aelst and Rousseeuw (2009)). The high computational efficiency of MCD makes it a preferred method over MVE, especially in modern high dimensional problems.

Now for each $\mathbf{x}_i$, we calculate a robust Mahalanobis depth vector $\mathbf{RM}_d(\mathbf{x}_i) = \langle RM_d(\mathbf{x}_1|\mathbf{x}_i), \ldots, RM_d(\mathbf{x}_n|\mathbf{x}_i)\rangle$, measuring the "outlyingness" of every other observation with respect to $\mathbf{x}_i, i = 1, 2, \ldots, n$. The depth vector $\mathbf{RM}_d(\mathbf{x}_i)$ provides a center-outward ordering of the data and serves as a topological map. The effect of traditional and robust Mahalanobis depth function is visualized in Figure 3.2, where the solid red dot represents the observation $\mathbf{x}_i$ (center) and each contour corresponds to a depth value. Armed with a robust depth-based dissimilarity measure (3.1), we now proceed to clustering.

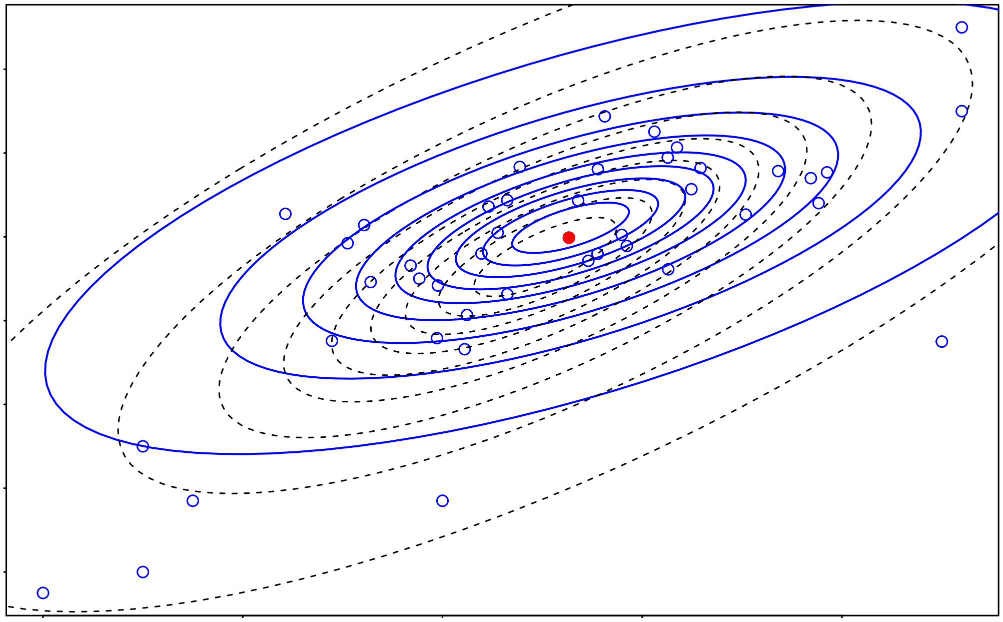


Figure 3.2. A contour plot based on traditional (black dash line) and robust (blue solid line) Mahalanobis depth function.

3.2.2 A New Neighbor Searching Algorithm

Who is Your Closest Neighbor?

In CRAD, we use a robust depth-based dissimilarity measure (3.1), and the neighbor searching function is defined as:

$$\mathrm{NBR}(\mathbf{x}_i) = \{\mathbf{x}_j \colon RM_d(\mathbf{x}_j|\mathbf{x}_i) \geq h_{opt}(i), j = 1, \ldots, n\}, \tag{3.2}$$

where $RM_d(\mathbf{x}_j|\mathbf{x}_i)$ is defined in (3.1) and $h_{opt}(i)$ is the cut-off parameter. The novel part of our neighbor searching function is that for each observation $\mathbf{x}_i, i = 1, \ldots, n$, the cut-off parameter $h_{opt}(i)$ is *locally* rather than globally defined, and accounts for different density level around it. E.g., if a person resides in Manhattan, his closest neighbor is likely in the same apartment complex; but if he lives in Dallas, TX, the closest neighbor might be miles away.

In contrast, DBCA uses a globally defined parameter θ in its neighbor searching function:

$$\text{NBR}(\mathbf{x}_i) = \{\mathbf{x}_j \colon RM_d(\mathbf{x}_j|\mathbf{x}_i) \geq \theta, j = 1, \ldots, n\}, \tag{3.3}$$

Similarly, DBSCAN uses a globally-defined parameter ϵ in its neighbor searching function:

$$\text{NBR}(\mathbf{x}_i) = \{\mathbf{x}_j \colon \|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \epsilon, j = 1, \ldots, n\}. \tag{3.4}$$

With an additional requirement on the minimum number of observations $MinPts$ in each cluster, elements A_{ij} of the adjacency matrix for DBSCAN are defined as 1 if $\mathbf{x}_j \in \text{NBR}(\mathbf{x}_i)$ and $|\text{NBR}(\mathbf{x}_i)| > MinPts$, and 0, otherwise. Here $|X|$ denotes the cardinality of a set X.

The parameter in NBR is critical in detecting cluster patterns. A globally-defined parameter cannot find all intrinsic clusters with varying densities. The example in Figure 3.3 best illustrates the idea: we take the toy data in Section 3.1 and investigate the neighbor searching process of an observation for CRAD and DBCA (Jeong et al., 2016). The observation is labeled as the yellow dot in Figure 3.3(a). The reader could visualize the difference between the neighbor observations (red dots) found by the locally-adjusted parameter $h_{opt}(i)$ in CRAD and the globally-defined parameter θ in DBCA, as shown in the Figure 3.3(c), (d). We find that the DBCA incorrectly includes the observations in the nearby cluster of the yellow dot as its neighbors, thus leading to the inaccurate clustering result. Given the ground truth, the parameter θ in DBCA is selected by searching the best clustering result over a wide range of values $[0.80, 0.81, \ldots, 1]$. In contrast, parameter $h_{opt}(i)$ of CRAD is selected by an automatic self-searching algorithm based on a notion of density level (see Algorithm 3).

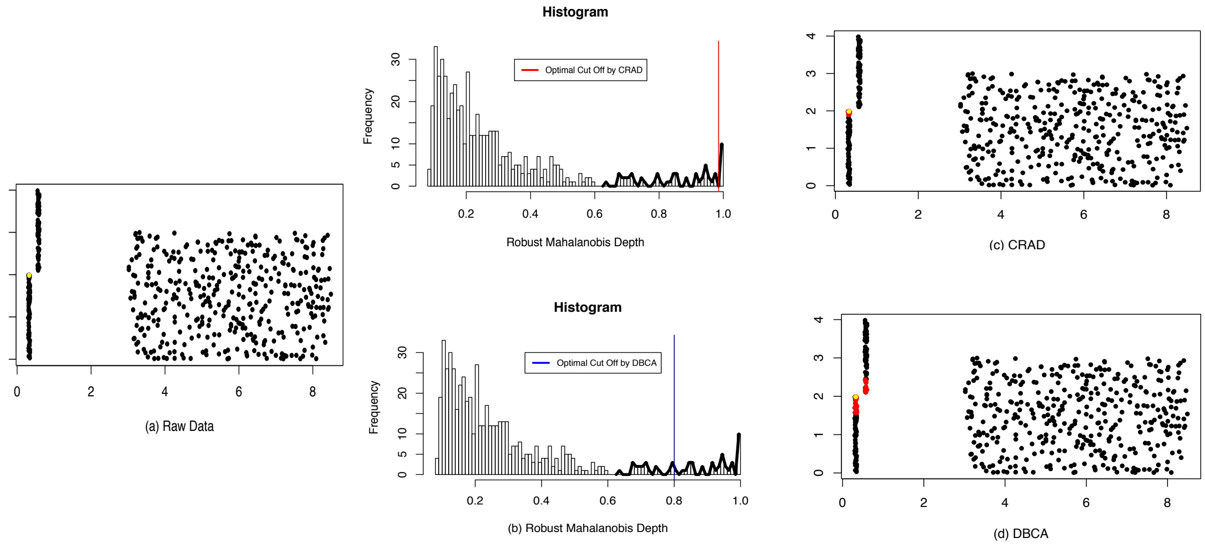


Figure 3.3. Neighbor search for a given point (the yellow dot) in the toy example. The red dots are the neighbors, identified by CRAD and DBCA under their best clustering performances, shown in top (bottom) right. The histogram shows the optimal value of the cut-off parameter, $h_{opt}(i)$ (the red vertical line) for CRAD and θ (the blue vertical line) for DBCA.

An Automatic Self-Searching Algorithm for Finding $h_{opt}(i)$

The idea is that the neighbor searching function of each observation should depend on the *relative change of the density level* around it. The term "relative" accounts for the customization for each observation. As mentioned before, for each $\mathbf{x}_i$ we calculate a robust Mahalanobis depth vector $\mathbf{RM}_d(\mathbf{x}_i) = \langle RM_d(\mathbf{x}_1|\mathbf{x}_i), \ldots, RM_d(\mathbf{x}_n|\mathbf{x}_i)\rangle$, measuring the "outlyingness" of every other observation with respect to $\mathbf{x}_i, i = 1, 2, \ldots, n$ (Figure 3.2).

Armed with $\mathbf{RM}_d(\mathbf{x}_i)$ of $\mathbf{x}_i$, we create a vector of histogram $\mathbf{H} = \langle h_{width}, h_{2*width}, \ldots, h_1\rangle$, where $h_j = \sum_{k=1}^{n} \mathbb{1}_{j-width} < RM_d(\mathbf{x}_k|\mathbf{x}_i) \leq j$. Parameter $width = 1/Nbin \in (0, 1)$, where $Nbin$ is the number of bins in $\mathbf{H}$ and is user pre-defined. Analogous to the definition of density for a substance, $\rho = m/V$ mass (m) per unit volume (V), we define the density level of a point as N/d, number of observations (N) per unit depth distance (d). If we choose the unit depth distance as the parameter $width$, then the reverse order of $\mathbf{H}$: $h_1, h_{1-width}, h_{1-2*width}, \ldots, h_{width}$ are the density levels around x_i in a center-outward order. A higher value of $h_k, k = 1, 1 - width, 1 - 2 * width, \ldots, width$ indicates a denser region and a lower value corresponds to a sparser region. Thus, starting from h_1 we search for the first local minimum h_{opt} over $\mathbf{H}$. The value of $h_k, k = 1, 1-width, 1-2*width, \ldots, width$ decreasing from h_1 to h_{opt} indicates that the density level around $\mathbf{x}_i$, in a center-outward manner, changes from dense to sparse. The observations in the sparse region do not have the same property as the observations in the dense region. Thus, the first local minimum h_{opt} could serve as the cut-off depth value to select the neighbors of $\mathbf{x}_i$. For each $\mathbf{x}_i, i = 1, 2, \ldots, n$, a locally-defined $h_{opt}(i)$ is selected. Thus, neighbor observations of $\mathbf{x}_i$ can be found from (3.2). Figure 3.3(b) shows how the neighbor searching parameter $h_{opt}(i)$ (red vertical line) is selected for each $\mathbf{x}_i$. Note, the DBCA does not include a similar self-searching step. For better comparison and visualization purpose we put the selected θ (blue line) in the histogram plot.

The CRAD algorithm is summarized in Algorithm 2. The neighbor searching function and the automatic self-searching method are described in Algorithm 3. A user pre-defined parameter *StepSize* is required to decide the size of neighbor buckets in $\mathbf{H}$ to compare for each $h_i, i = 1, 1-width, 1-2*width, \ldots, width$. Another user pre-defined parameter is the number of bins *Nbin* in generating $\mathbf{H}$ (for details on *Nbin* selection see Section 3.3). The upper bound for time complexity of CRAD is $O(n^2)$, and complexity can be further lowered to achieve $O(n \log n)$, by using an accelerating index structure for the data in two dimensional spaces (Ester et al., 1996; Gan and Tao, 2015). The source code of CRAD algorithm is available from `https://github.com/DataMining-ClusteringAnalysis/CRAD-Clustering/`.

3.2.3 An Extension to DBSCAN

Since the essential difference between CRAD and DBSCAN is the neighbor searching function, a hybrid combination of our new robust depth-based neighbor searching algorithm and conventional DBSCAN is generated by replacing the neighbor searching function (3.4) in DBSCAN with the proposed new function (3.2). We name the hybrid algorithm as CRAD-DBSCAN. Our experiments show that with a replacement of a neighbor searching function,

Algorithm 2: CRAD Algorithm

```
Input: A finite set of observations $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^t$ with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^t$ the $i$th
       observation; $n$: Sample size; $Nbin$: Number of bins; $StepSize$: Size of
       neighbor buckets to compare.
Output: ClVec: Cluster ID of each observation.
Initialization: ClVec = $[-1] * n$; label := 0; $A := \{0\}^{n \times n}$;
Compute Robust Mahalanobis depth vector for each observation:
$\mathbf{RM}_d = \langle \mathbf{RM}_d(\mathbf{x}_1), \mathbf{RM}_d(\mathbf{x}_2), \ldots, \mathbf{RM}_d(\mathbf{x}_n) \rangle$;
// Compute the adjacency matrix $A$
for $i := 1$ to $n$ do
    AdjIndex := NBR($\mathbf{RM}_d(\mathbf{x}_i)$, $Nbin$, $StepSize$);
    for $\forall$ ind $\in$ AdjIndex : $A[i, \text{ind}] := 1$;
end
for $i := 1$ to $n$ do
    if $ClVec[i] == -1$ then
        nbrs := neighbor IDs of observation i;
        if nbrs. size() == 1 then
            ClVec[$i$] := 0; // single cluster
        end
        else
            label := label + 1;
            for $\forall$ nbrId $\in$ nbrs : ClVec[nbrId] := label;
            nbrs. remove($i$);
            while nbrs is not empty do
                CurrentPoint := nbrs. get()
                Snbrs := neighbor IDs of CurrentPoint;
                if Snbrs. size() > 1 then
                    for $x$ in Snbrs do
                        if $ClVec[x] == -1$ then
                            ClVec[$x$] := label; nbrs. add($x$);
                        end
                    end
                end
            end
        end
    end
end
```

Algorithm 3: NBR($\mathbf{RM}_d(\mathbf{x}_i), Nbin, StepSize$)

```
Input: $\mathbf{RM}_d(\mathbf{x}_i)$: Robust Mahalanobis depth vector of observation $i$; $Nbin$: Number
       of bins; $StepSize$: Size of neighbor buckets to compare.
Output: nbrIds: Neighbor IDs of observation $i$.
Initialization: $width := 1/Nbin$.
Compute histogram $H$ based on $\mathbf{RM}_d(\mathbf{x}_i)$: $H = \langle h_{width}, h_{2*width}, \ldots, h_1 \rangle$.
for $j := H.size() - StepSize$ to $1 + StepSize$ do
    Boolean b := An empty array;
    for $z := 1$ to $StepSize$ do
        if $H[j] < H[j+z]$ and $H[j] < H[j-z]$ then
            $b.append$(TRUE);
        end
        else
            $b.append$(FALSE);
        end
    end
    if $(b == TRUE).size() == StepSize$ then
        $h_{opt} := 1 - (H.size() - j + 1) * width$;
        Break;
    end
end
nbrIds := $\{l : RM_d(\mathbf{x}_l|\mathbf{x}_i) > h_{opt}, l = 1, \ldots, n\}$.
```

CRAD-DBSCAN significantly outperforms DBSCAN (see Section 3.4.1). This is an important standalone step toward future extension of DBSCAN to non-Euclidian spaces and functional data clustering. That is, the DBSCAN approach and its adaptations, such as CRAD-DBSCAN, with a suitable metric as a dissimilarity measure (e.g., band depth), can be further advanced to clustering of functional curves in Hilbert spaces.

3.3 Determining the Parameter *StepSize* and *Nbin*

Our CRAD algorithm requires two parameters, *StepSize* and *Nbin*, both of which are used in the automatic self-searching algorithm in Algorithm 3. The goal is to select optimal *StepSize* and *Nbin* to help CRAD achieve the highest quality of clustering results. There are two kinds of evaluation metrics to measure the quality of clustering results, external and internal metrics. An external metric, such as Rand Index (RI) (Rand, 1971; Jain et al., 1999) and Adjusted Mutual Information (AMI) (Meilă, 2007), is a measure of agreement between the result obtained from a clustering algorithm and the ground truth. Since the ground truth is not available in the real-world clustering, we use the internal metric, which measures the goodness of clustering without external information (Rousseeuw, 1987; Caliński and Harabasz, 1974; de Amorim and Hennig, 2015), to serve as a validation tool for selecting optimal *StepSize* and *Nbin*. If we assume larger values of the metric indicate better clustering results, $StepSize_{opt}$ and $Nbin_{opt}$ are then defined as:

$$StepSize_{opt}, Nbin_{opt} := \underset{StepSize, Nbin}{\arg\max}\ M(\mathbf{X}, \mathrm{ClVec}), \tag{3.5}$$

where ClVec is the clustering result returned by Algorithm 2. Here we consider the Calinski-Harabasz (CH) score as the internal metric M, which evaluates the clustering quality based on the average between- and within-cluster sum of squares (Caliński and Harabasz, 1974; Frakes and Baeza-Yates, 1992).

The CH score is defined as:

$$CH(\mathbf{X}, ClVec) = \frac{trace\mathbf{B}/(k-1)}{trace\mathbf{W}/(n-k)}, \tag{3.6}$$

where $\mathbf{B}$ is the error sum of squares between different clusters (between-cluster),

$$trace\mathbf{B} = \sum_{m=1}^{k} |\overline{C}_m| \|\overline{C}_m - \bar{\mathbf{x}}\|_2, \tag{3.7}$$

and $\mathbf{W}$ is the squared differences of all objects in a cluster from their respective cluster center (within-cluster)

$$trace\mathbf{W} = \sum_{m=1}^{k} \sum_{i=1}^{n} w_{m,i} \|\mathbf{x}_m - \overline{C}_m\|. \tag{3.8}$$

Here $|\bar{C}_m|$ and $\bar{\mathbf{x}}$ are the sample mean of mth cluster and the data set $\mathbf{X}$, respectively; n is sample size; k is the number of clusters in $ClVec$, and $w_{m,i}$ is the weight function. The larger value of CH, the better clustering performance (Caliński and Harabasz, 1974).

Our simulations show that optimal performance can be achieved with $StepSize \in \{1, 2\}$ and $Nbin \in (0.2 * n - 100, 0.2 * n + 100)$, where n is the sample size. Thus, we fix $StepSize$ as 1 and search $Nbin_{opt}$ based on (3.5) (For details see Section 3.4).

3.4 Experimental Evaluation

3.4.1 Synthetic Data

We evaluate performance of CRAD with respect to DBCA (Jeong et al., 2016), DBSCAN (Ester et al., 1996), and OPTICS (Ankerst et al., 1999). The DBSCAN has two versions: 1. Original DBSCAN with Euclidean distance as the dissimilarity measure, i.e., DBSCAN (EU); 2. An extension version of DBSCAN (CRAD-DBSCAN), which is a hybrid combination of our new robust depth-based neighbor searching algorithm and conventional DBSCAN, as discussed in Section 3.2.3. We show that with a simple replacement of a neighbor searching function, CRAD-DBSCAN significantly outperforms DBSCAN in the considered set of synthetic data.

The evaluation is first conducted on 2 synthetic data sets, $S1$ and $S2$. To visualize the improved effects of our algorithm on DBCA, we extend the data sets in (Jeong et al., 2016) so that they exhibit the challenging properties on which we focus. Specifically, for $S1$ we generate a mixture of clusters from both normal and uniform distributions (with varying density among clusters) by replacing the “circles” shaped clusters with the “cassini” cluster structure. In addition, we decrease the distance bewteen clusters, which makes it harder to detect true patterns. All the sample data are from the mlbench (Leisch and Dimitriadou, 2010; Lichman, 2013). The extended dataset $S1$ is shown in Figure 3.4(a). Lastly, we explore the performance of algorithms under the existence of noises. Dataset $S2$ is generated by adding a number of noises with 2% noise to signal ratio, shown in Figure 3.5(a).

As evaluation metric we consider Adjusted Mutual Information (AMI) (Vinh et al., 2010; Meilă, 2007), which is a robust adjustment of the Mutual Information (MI) score.

Given a set X of n observations $(x_1, x_2, \ldots, x_n)$, let us consider two partitions of X, namely $U = \{U_1, U_2, \ldots, U_R\}$ with R clusters, and $V = \{V_1, V_2, \ldots, V_C\}$ with C clusters. The AMI is defined as follow:

$$AMI(U,V) = \frac{MI(U,V) - \mathbb{E}(MI(U,V))}{\max\{H(U), H(V)\} - \mathbb{E}\{MI(U,V)\}}, \tag{3.9}$$

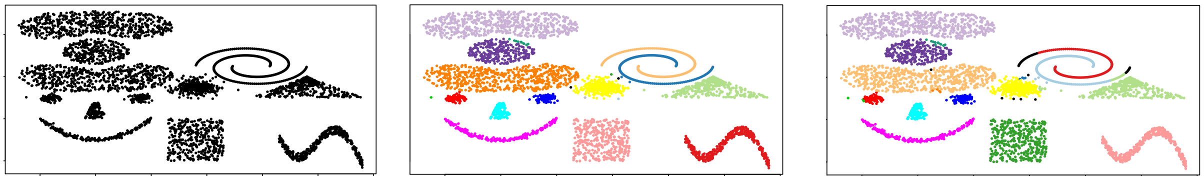
where

$$\begin{aligned} H(U) &= -\sum_{i=1}^{R} P(i)\log(P(i)), \\ MI(U,V) &= \sum_{i=1}^{R}\sum_{j=1}^{C} P(i,j)\log\frac{P(i,j)}{P(i)P'(j)}, \end{aligned}$$

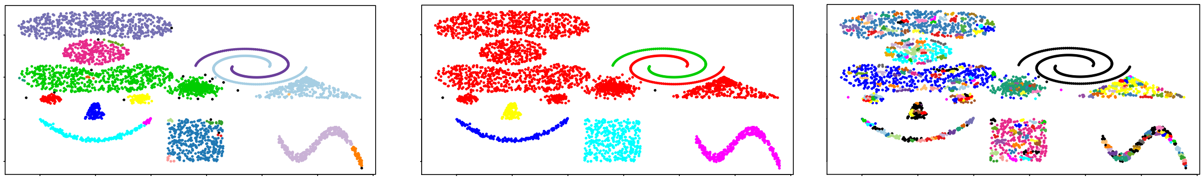
$P(i) = |U_i|/N$, $P'(j) = |V_j|/n$ and $P(i,j) = |U_i \cap V_j|/n$. In contrast to MI, the value of AMI between two random clusterings takes on a constant value, especially when the two partitions have a larger number of clusters (Vinh et al., 2010).

For each clustering algorithm, we search the best achievable clustering result in a wide range of combinations of its parameters. The search range of $Nbin$ in CRAD and CRAD-DBSCAN is in $\{80, 90, \ldots, 700\}$, and $StepSize$ is set as 1. The search range of θ in DBCA is in $\{0.80, 0.82, \ldots, 1\}$. In DBSCAN (EU), ϵ is selected from minimum to the half of the maximum value of pairwise dissimilarity in the given dataset. Parameter $MinPts$ for CRAD-DBSCAN, DBCSAN (EU) and OPTICS is selected from $\{2, 3, \ldots, 6\}$, and $\xi \in \{0.01, 0.02, \ldots, 0.99\}$.

The clustering results on $S1$ and $S2$ are shown in Figure 3.6, where each number is an average result over 10 trails. The clustering results on $S1$ and $S2$ are visualized in Figure 3.4 and Figure 3.5. For $S1$, we can see that our new CRAD achieves the best clustering performance, with an almost perfect detection result. In addition, CRAD-DBSCAN produces almost the same result as CRAD with a minor misclassification on the boundaries of the "spiral" cluster (top right). DBSCAN (EU), in contrast, fails to separate most of the clusters, which well demonstrates the competitive performance of our new neighbor searching

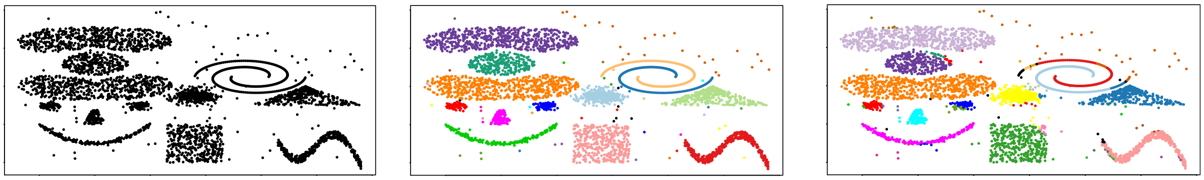


(a) Raw Data (b) CRAD (c) CRAD-DBSCAN

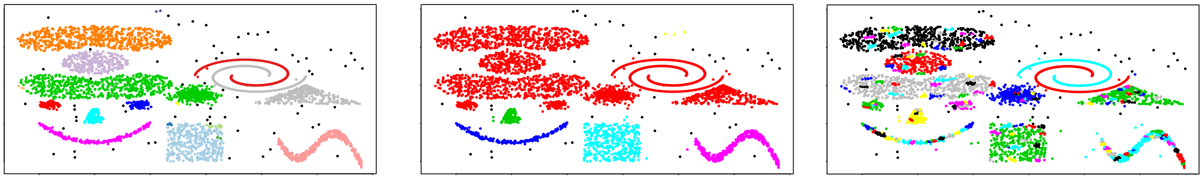


(d) DBCA (e) DBSCAN (EU) (f) OPTICS

Figure 3.4. Clustering Performance of CRAD, CRAD-DBSCAN, DBCA, DBSCAN (EU), and OPTICS on $S1$.



(a) Raw Data (b) CRAD (c) CRAD-DBSCAN



(d) DBCA (e) DBSCAN (EU) (f) OPTICS

Figure 3.5. Clustering Performance of CRAD, CRAD-DBSCAN, DBCA, DBSCAN (EU), and OPTICS on $S2$.

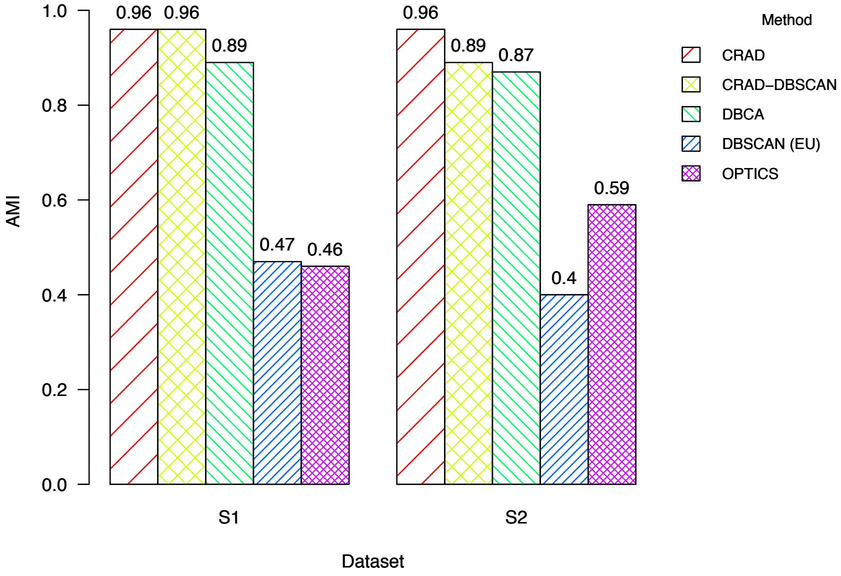


Figure 3.6. Best AMI of CRAD, CRAD-DBSCAN, DBCA, DBSCAN (EU) and OTPICS on synthetics $S1$ and $S2$. Each AMI score is an average result over 10 trials.

algorithm. DBCA has a slightly better performance than DBSCAN (EU) but still cannot recognize the “cassini” cluster (top left) and the “spiral” cluster (top right) with their nearby clusters. Lastly, OPITCS delivers the poorest performance among the five considered methods. Same conclusion is obtained for $S2$, that is, CRAD and CRAD-DBCSAN outperform all the other competing methods, showing highly competitive performance in detecting intrinsic clusters with varying densities under the existence of noises.

Table 3.1. Clustering Performance on 5 UCI Multivariate Datasets. The winner method on each dataset is highlighted.

Dataset	Rand Index				
	CRAD	CRAD -DBSCAN	DBCA	DBSCAN (EU)	OPTICS
Banknote	**0.86**	0.79	0.52	0.81	0.53
Iris	0.77	**0.78**	0.36	**0.78**	0.75
Blood Transf.	**0.64**	**0.64**	0.46	**0.64**	0.49
Occupancy	**0.77**	0.72	0.67	0.67	0.66
Seeds	0.68	0.67	0.33	**0.69**	**0.69**

Table 3.2. Clustering Performance for the 29 UCR Time Series Datasets. Two ratios, which are over 0.9 and 0.8 in column "CRAD / *k*-means" and column "Empirical CRAD / CRAD", are highlighted, respectively.

Dataset (# of Clusters)	CRAD /*k*-means	Empirical CRAD /CRAD	CRAD	Empirical CRAD	# of Time Series /Length
50words (9)	**0.96**	0.35	0.90	0.31	905/271
Beef (5)	**1.09**	**0.80**	0.76	0.62	60/471
BirdChicken (2)	**1.09**	**1.00**	0.54	0.54	40/513
Car (4)	**1.10**	0.64	0.75	0.48	120/578
Coffee (2)	0.88	**1.00**	0.74	0.74	56/287
Cricket-X (9)	**1.01**	**0.87**	0.88	0.77	780/301
Cricket-Y (9)	**1.02**	**0.82**	0.89	0.73	780/301
Cricket-Z (9)	**1.02**	**0.94**	0.88	0.83	780/301
ECG200 (3)	**0.97**	**0.87**	0.61	0.53	200/97
ECG-FiveDays (2)	**1.00**	**0.82**	0.98	0.80	5000/141
FaceFour (4)	**0.95**	**0.98**	0.94	0.92	112/351
FISH (7)	**1.00**	**0.99**	0.84	0.83	350/464
Gun-Point (2)	**1.03**	0.79	0.77	0.62	200/151
Ham (2)	**0.96**	**1.00**	0.56	0.56	214/432
Haptics (5)	**1.10**	**0.88**	0.75	0.66	463/1093
Herring (2)	**0.96**	**0.93**	0.53	0.49	128/513
InlineSkate (7)	**1.09**	0.31	0.82	0.25	650/1883
Lighting2 (3)	**1.04**	**0.93**	0.54	0.50	121/638
Lighting7 (7)	**1.01**	**1.00**	0.82	0.82	143/320
Meat (3)	**0.97**	**1.00**	0.78	0.78	120/449
OSULeaf (6)	**1.03**	**0.87**	0.80	0.70	442/428
Plane (7)	**0.98**	**0.96**	0.95	0.91	210/145
SonyAIBORobot (2)	**0.94**	**1.04**	0.67	0.70	621/71
Synthetic-Control (6)	**0.96**	**1.00**	0.86	0.86	600/61
ToeSegmentation1 (2)	**0.95**	0.77	0.81	0.62	268/278
ToeSegmentation2 (2)	**1.00**	**0.95**	0.89	0.84	166/344
Trace (4)	**0.99**	**1.00**	0.99	0.99	200/276
Worms (5)	**1.13**	0.66	0.75	0.49	258/901
WormsTwoClass (2)	**0.97**	**0.98**	0.54	0.53	258/901

3.4.2 Read-World Multivariate Data

We now evaluate CRAD on real-world multivariate data, with the same competing methods and experiment settings, as in Section 3.4.1. The evaluation is conducted on 5 benchmark multivariate datasets from UCI (Lichman, 2013): *Banknote Authentication*, *Iris*, *Blood Transfusion*, *Occupancy Detection*, and *Seeds*. A brief description of each dataset is shown as follows.

- *Banknote Authentication*: contains 762 and 610 observations for each of two classes of banknotes, respectively. Each observation has 4 attributes, which are extracted features from the image of the banknote-like specimen.
- *Iris*: contains 3 classes of 50 observations each, where each class refers to a type of iris plant. The number of attributes for each observation is 4.
- *Blood Transfusion*: contains 570 and 178 observations for each of two classes of people, respectively. Each class represents whether he/she donated blood in March 2007. The number of attributes for each observation is 4.
- *Occupancy Detection*: contains 7703 and 2049 observations for each of two classes of office rooms, respectively. Each class indicates whether the office room is occupied. Each observation has 5 attributes, which are temperature, humidity, light, CO2 and humidity ratio in the room.
- *Seeds*: contains 3 classes of 70 observations each, where each class refers one kind of wheat. Each observation has 7 attributes, which are geometric descriptions of the wheat kernel.

As Table 3.1 indicates, for all datasets, except *Seeds*, CRAD and CRAD-DBSCAN rank 1st/2nd among all the methods. In particular, CARD and CRAD-DBSCAN significantly outperform DBCA and OPTICS for *Banknote Authentication*, *Iris*, *Blood Transfusion*, and

Occupancy Detection. Furthermore, CRAD outperforms DBSCAN (EU) for *Banknote Authentication* and *Occupancy Detection* and delivers a comparable performance for *Iris* and *Blood Transfusion*. For *Seeds*, CRAD and CRAD-DBSCAN slightly underperform, comparing to DBSCAN (EU) and OPTICS, but still significantly outperform DBCA.

3.4.3 Real-World Time Series Data

We now evaluate the utility of CRAD for time series clustering. Time series data usually contain noises, dropouts, or extraneous data, existence of which can greatly limit the accuracy of clustering (Ye and Keogh, 2009; Mueen et al., 2011; Hartmann et al., 2010). Thus, we apply a time-series based feature-extraction technique, named U-Shapelets (Zakaria et al., 2012) to filter out noises in data in the first place. The idea of the U-Shapelets is to search for small subsequences of a few time series, named U-Shapelets, that best represent the entire time series data and then to use those subsequences as features. Since the number of extracted U-Shapelets is small (usually < 10), dimension of time series data is highly reduced.

Based on the extracted U-Shapelets, we evaluate our CRAD algorithm with respect to the "U-Shapelets + k-means" methodology, where k denotes the true number of clusters (Zakaria et al., 2012). The choice for the study settings is addressed as follows.

1. First, we select "U-Shapelets + k-means" as the competing clustering method, because as demonstrated by (Zakaria et al., 2012), it is the winner method over other clustering methods with the state-of-the-art feature-extraction techniques.

2. Second, the density-based clustering algorithms such as DBSCAN, DBCA, and OPTICS are not included as competing methods, as none of these methods are designed for time series clustering. Furthermore, we focus on the following practical problems in this section:

    (a) Without any a-priori knowledge on a number of clusters, can our CRAD algorithm detect the true number of clusters and the correct partitions?

(b) Without knowing the true parameters, $Nbin$ and $StepSize$, can our parameter selection procedure, from Section 3.3, assist the clustering method to achieve a satisfactory clustering performance?

The evaluation is conducted on 29 benchmark datasets from the UCR time series archive (Chen et al., 2015), in terms of RI (Rand, 1971) and is consistent with the evaluation of "U-Shapelets + k-means" in (Zakaria et al., 2012). The datasets include various domains, i.e., from finance to neuroscience to geology (see Table 3.2). The column "CRAD" and "k-means" denote the best achievable RI by searching clustering results over a wide range of parameters and possible combinations of U-Shapelets features. As shown in (Zakaria et al., 2012), 1 or 2 U-Shapelets are sufficient to achieve the best clustering result in most cases. Hence, the upper limit on a number of U-Shapelets is set to 2. The column "Empirical CRAD" is the RI achieved by CRAD, using the new parameter selection procedure in Section 3.3. Compared with the best achievable "CRAD", the "Empirical CRAD" is more important since we will not know the ground truth in real-data clustering and thus being able to select the right parameter is critical in achieving a good clustering result. Two ratio indicators, that is "CRAD / k-means" and "Empirical CRAD / CRAD", are presented to simplify the comparison among methods.

For the ratio "CRAD / k-means", 28 (out of 29) datasets are over 0.9, among which 16 datasets deliver a ratio of more than 1, indicating the competitive performance of our CRAD algorithm. Note, our benchmark method "U-Shapelets + k-means" has a critical advantage of knowing the true number of clusters in datasets, thus operating with more information. Despite this, the new CRAD algorithm still delivers a quite close performance with the benchmark method and even outperforms it in half of the datasets. For the ratio "Empirical CRAD / CRAD", 23 (out of 29) datasets are over 0.8, among which 16 datasets yield a ratio more than 0.9, and 13 datasets deliver a ratio more than 0.95. These findings indicate a high practical utility of CRAD in the real-world time-series clustering, which is typically performed without any prior information on the true number of clusters and cluster density.

Finally, we assess performance of CRAD with respect to other density-based clustering algorithms. The competing methods are DBCA and DBSCAN. All the methods are performed on U-Shapelets extracted from the data, following the framework of clustering time series in (Zakaria et al., 2012), i.e., "U-Shapelets + a clustering method". The time series data are selected from Table 3.2. Each dataset contains two versions: a raw dataset and a noisy dataset which is obtained by adding a random noise $N(0, 0.2)$ on each observation of time series in the raw dataset.

Table 3.3. Clustering Performance of CRAD, DBCA and DBSCAN on 5 UCR Time Series Datasets. The winner method on each dataset is highlighted.

Dataset	Rand Index		
	CRAD	DBCA	DBSCAN
Coffee	0.74	0.59	**0.75**
Noisy Coffee	**0.65**	0.58	0.55
FaceFour	0.94	0.91	**0.98**
Noisy FaceFour	**0.92**	0.86	0.85
SonyAIBO	0.67	**0.68**	0.53
Noisy SonyAIBO	**0.67**	0.56	0.52
ToeSegm1	**0.81**	0.71	0.72
Noisy ToeSegm1	**0.80**	0.68	0.68
Trace	0.99	0.99	**1.00**
Noisy Trace	**0.95**	0.93	0.89

As Table 3.3 indicates, on the raw datasets CRAD outperforms DBCA and DBSCAN for *ToeSegm1* and delivers a comparable performance for *Coffee*, *FaceFour*, *SonyAIBO*, and *Trace*. However, under noised scenarios, CRAD outperforms DBCA and DBSCAN on all the five considered datasets. These findings are consistent with conclusions in previous sections, that is, our new CRAD algorithm delivers a more competitive performance for data that contain noise, outliers and of varying densities.

3.5 Conclusion

We propose a new robust data depth based clustering algorithm CRAD with a locally-defined neighbor searching function. Besides robustness to outliers, we show that the new CRAD algorithm is highly competitive in detecting clusters with varying densities, compared with the existing algorithms such as DBSCAN, OPTICS and DBCA. Furthermore, the performance of DBSCAN is shown to be effectively improved, by replacing its original neighbor searching function with the new locally tuned neighbor searching algorithm. In addition, we propose a new effective parameter selection procedure, to select the optimal underlying parameter in the real-world clustering, when the ground truth is unavailable. In the future, we plan to investigate the utility of other data depth functions as dissimilarity measures and extend the CRAD idea to functional data clustering.

CHAPTER 4

POLITICAL RHETORIC THROUGH THE LENS OF NONPARAMETRIC STATISTICS: ARE OUR LEGISLATORS THAT DIFFERENT?[1]

4.1 Introduction

What are politicians really telling us? They sound different, but are they really? The 2016 campaign for U.S. president presented us with two distinct candidates: Hillary Clinton and Donald Trump, who had very different messages for the voters. The outsider message was a prominent feature of the 2016 elections, used by both presidential and congressional candidates, on both sides of the isle (Lowry, 2016; Healy, 2015). The Democrats and Republicans in the U.S. Congress are also fighting for the increasingly divided electorate. Using their speech, politicians from both sides constantly send (often contradictory) signals to the voters and financial supporters. Such political sentiments have a tremendous impact on all aspects of our society, and better understanding of the behavior of politicians when in office is critical for both voters and interest groups. This is particularly acute as political discourse is rarely driven by compromise and cooperation, especially in the currently highly polarized U.S. Congress. Politicians running for office often promise to be different from their colleagues, but do they keep those promises when in office? Is their rhetoric really that distinct, or do they cater to similar interests? And how can statistical analysis help us to unveil and quantify political perception?

Detecting hidden underlying dynamics in political attitudes and expressions, as well as similarities between them is crucial for our society but such a challenging task is impossible

---

[1] This chapter includes verbatim excerpts from

Under Invited Revision. Iliev, I., Huang, X., Gel, Y. (2017) Political Rhetoric through the Lens of Nonparametric Statistics: Are Our Legislators That Different?

without advanced statistical methods and data mining tools. And while politicians increasingly use data analytics in their election campaigns (see overviews in Issenberg, 2012; Kaye, 2014; Grajales, 2014; Markman, 2016, and references therein), can we uncover political patterns with modern statistical methodology? We attempt to address this challenging problem by proposing a novel study of political rhetoric in congressional committees.

We develop a mixed supervised-unsupervised approach for tracking changes in speech over time and detecting common features and behavioral clustering among legislators. We combine automated content analysis of legislative speech with spatio-temporal dynamic clustering and a data-driven (in)stability criteria to select optimal clustering input parameters. While some of the tools that we use existed before, the combination of these methods is novel, as well as their application. The application of these algorithms to the study of politicians' rhetoric and their dynamic behavior with respect to their time in office is innovative, and such an analysis is only possible through the use of advanced statistical methods. The statistical contribution stems from the combining of three methods that existed separately to solve a complex problem in energy policy, which could not be analyzed using a traditional approach. That is, we show how statistical thinking and statistical algorithms and analyses can play a vital role in enhancing our understanding of intrinsic mechanisms of legislative politics and can benefit our society in general. Furthermore, the need and application of these advanced statistical and data mining methods renders our study beyond the traditional scope of a single discipline. Our approach is not exclusive to political science, but can be used as a template to solve similarly complex multi-stage problems in a wide array of disciplines.

While committee text has been used previously (Yano et al., 2012; Nowlin, 2015; Stramp and Wilkerson, 2015; Talbert et al., 1995), the focus has been on legislative bills or specific issue definitions. Rhetoric in congressional committees has not been analyzed before, to our knowledge, in a systematic and objective way using advanced statistical methods. Here, we primarily focus on the arguably most dominant area of the U.S. and worldwide politics, the energy sector. We create a novel comprehensive dataset based on the statements from congressional hearings of all senators who served on the U.S. Senate Committee on Energy and Natural Resources in 2001–2011. Our dataset can be used to quantitatively evaluate various expressions of political behavior or attitudes through rhetoric. Our analytic approach of combining supervised natural language processing and aggregate classification with unsupervised dynamic clustering has numerous applications well beyond the energy sector and range from analysis of political statements when soliciting campaign contributions, discussing foreign policy issues, or during contentious political campaigns. The results can in turn lead to a deeper understanding of political rhetoric, and facilitate the comprehensive analysis that can inform the voters, as well as a broad range of policy-makers.

The first stage of our algorithm is an automated content analysis of legislative speech in a supervised context. We are interested in scaling attitudes towards pre-specified interest groups and in classifying *document category* proportions. Notice that we do not focus on individual documents since we study attitudes in the aggregate. Thus, the first stage of our analysis is based on a supervised natural language processing (NLP) method for aggregate classification developed by (Hopkins and King, 2010). Since our current focus is on the U.S. Senate Committee on Energy and Natural Resources and the energy sector explicitly and implicitly affects all sectors of the economy, including non-energy related ones, from national security to the environment, we propose a novel measure for legislative rhetoric — *attitudes toward the dominant energy interests*. In a statistical sense, the measure can be viewed as a baseline. The data are aligned so that all rhetoric begins at month one of the senators' tenure in office within the period that we analyze, allowing us to trace the speech trajectories as they progress in office.

The second stage of our analysis aims to uncover clusters in the behavior of legislators in terms of their expressed rhetoric identified in the first stage. However, the process of political group formation is intrinsically dynamic, and the number and shape of the clusters are unknown a-priori, since they could be based on partisanship, geography, constituency, and other unknown subjective factors. To address these challenges, we adopt an unsupervised spatio-temporal data mining algorithm for discovering dynamic clusters of arbitrary shape in environmental geo-referenced data, namely, TRend based clUstering algorithm for Spatio-Temporal data stream (TRUST) (Ciampi et al., 2010). We then employ data-driven (in)stability criteria to select optimal clustering input parameters, based on the crossvalidation argument (Dudoit and Fridlyand, 2002; Ben-David et al., 2006; Ben-David and Von Luxburg, 2008; Wang, 2010; Huang et al., 2016) that allow for more objective and automatic choice of parameters in contrast to a traditional user pre-specified option. Such a data-driven approach to clustering allows us to retain the dynamic component of political speech — group membership and the number of groups can evolve over time, thus the number of clusters are selected automatically.

The results of the study suggest that in the beginning of their tenure, senators tend to noticeably differ from each other (i.e., exhibit distinct rhetoric), as usually promised on the campaign trail — there is a higher number of clusters with smaller membership. However, as senators spend more time in the institution, becoming more institutionalized, these differences begin to diminish and their rhetoric becomes more alike — as a result, we observe less clusters and cluster membership tends to be higher. The cluster formation shows complex dynamics, and not separation based on party. For instance, the senators cluster based on their seniority in the committee, the significance of the energy industry in their state, and other connections to the sector, but surprisingly not on partisanship. Our data-driven dynamic clustering approach allows us to explore these political formation dynamics in a more objective way while minimizing model and data constraints.

The paper is organized as follows. We proceed with an overview of related methodology for the analysis of political text in Section 4.2. Next, in Section 4.3 we discuss the natural language processing algorithm that we employ and the rhetoric data set produced by it. Section 4.4 is devoted to the dynamic clustering algorithm TRUST in application to committee rhetoric. We discuss our case study and findings on the dynamics of legislative rhetoric in the U.S. Senate Committee on Energy and Natural Resources in Section 4.5. The paper is concluded with the closing discussion and overview of future work in Section 4.6.

4.2 Related Work

Stringent seniority deference norms are suggested to influence the behavior (Sinclair, 2016, 1983), which is in line with the traditional view of Congressional norms. Analysis of floor speeches suggests that senators tend to differentiate themselves later in their terms (Quinn et al., 2010). However, an important aspect of modern campaigns in the candidate positioning as an outsider, or someone who is different from an otherwise unpopular institution. This phenomenon, which shapes Congressmen's behavior is what is known as Fenno's paradox — voters' generally disapprove of Congress as a whole, but support the Congressmen from their own district (Fenno, 2002). Anti-establishment politics are not new (Horwitz, 2013; Barr, 2009), and can currently be observed both in the conservative right with the Tea Party members (Boykoff and Laschever, 2011; Skocpol and Williamson, 2012), as well as in the liberal left (Bolton, 2016). However, members of Congress become institutionalized and their behavior is shaped by their institution (Binder, 2015; Hibbing and Theiss-Morse, 1995; Canon, 1989) even as they campaign as outsiders (Herrnson, 2007; Burden, 2004). The literature provides evidence that members become more alike the more time they spend in the institution (Cox and McCubbins, 2005) and that general election competition exerts pressure toward convergence (Hirano et al., 2010).

Rhetoric is an important political *commodity* that carries value for both the speaker and the audience, and it can be used as a strategic tool (Mayhew, 2004). The need for credibility makes the signals costly and not just "cheap talk". Interest groups pay attention to the way their preferences are discussed and adjust their responses. Examples come from every policy area, ranging from oil companies getting involved in lobbying during an oil spill (OpenSecrets.org, 2011) to IT companies fighting for net neutrality (Ars Technica, 2015) and for changes in the H1B visa process (San Jose Mercury News, 2015). The evolving patterns of legislative behavior can alert these groups when and how to get involved into the political processes. The diverse patterns of legislative behavior can be based on party lines, geographic differences, or the composition of the particular constituents. The numerous sources that influence the behavior and associated uncertainties create underlying patterns that are hardly detectable without a deeper statistical analysis.

Individual Document Classification vs. Category Proportions Unlike other fields, content analysis in the social sciences is often focused on category proportions and generalizations rather than individual document classifications (Hopkins and King, 2010; Grimmer, 2010). Political scientists are generally interested in the attitudes of a senator or a presidential candidate in the aggregate, rather than the classification of any specific speech or statement. Hopkins and King (2010) make an apt analogy — "*policy makers or computer scientists may be interested in finding the needle in the haystack..., but social scientists are more commonly interested in characterizing the haystack*". Grimmer (2010) proposes a hierarchical structure with political statements at the bottom, and their author at the top, which we employ in our analysis. The focus on political actors and not on their individual expressions is crucial, as we are interested in the overall legislative behavior of these actors. Document classification would require frequency-based methods, while our focus on the speakers themselves requires a method that takes the category proportions into account. In addition, the link between legislative sentiment and legislative text, and its use to explain and predict roll call voting, is discussed, for instance, by (Gerrish and Blei, 2011, 2012).

Analysis of the Dynamics The behavior of political actors is dynamic in nature and studying it requires statistical solutions that capture its temporal and spacial components. One such solution is provided by the dynamic networks literature (Loglisci, 2013; Loglisci et al., 2015; Beykikhoshk et al., 2015; Loglisci and Malerba, 2015). The focus is on the dynamics of evolving (heterogeneous) structured data (Loglisci et al., 2015), as well as the dynamics of the content of textual data (Loglisci, 2013; Beykikhoshk et al., 2015). Beykikhoshk et al. (2015) propose a flexible solution that does not restrict the type of changes that the model can capture, while Loglisci and Malerba (2015) develop a method based on two notions of patterns, emerging patterns and periodic changes. The emphasis is on discovering complex structural changes in the dynamic network across the temporal dimension. These solutions are useful when the focus of the analysis is on networks, where the complex linkages between individual nodes are of interest. When the dynamics of the behavior are studied at the group level, i.e. similar behavior within a group over time, clustering methods are appropriate. The number of dynamic data-driven clustering procedures for space-time data that allow the number, shape and distributional properties of clusters to vary, remains limited, despite receiving interest in recent years (Gaber et al., 2005; Cao et al., 2006; Banerjee et al., 2014). Two such dynamic clustering procedures are a space-time data mining procedure (TRUST) that is based on interleaving spatial clustering and temporal trend detection (Ciampi et al., 2010), and a hierarchical spectral merger algorithm to cluster brain connectivity (Euan et al., 2015).

Supervised vs. Unsupervised Political Content Analysis Measuring the intensity of expressed attitudes through political speech patterns provides rich data, and allows for a better measure of support or opposition that is not necessarily visible in the final vote. Automated analysis of political text is a relatively new field, which poses specific problems and requires a problem-specific validation (Hopkins and King, 2010; Grimmer, 2010; Gerrish and Blei, 2011; Grimmer and Stewart, 2013). Within the field, the two main categories of analysis are *ideological scaling* and *classification mechanisms*, both of which can be supervised or unsupervised (for a more extended overview see Grimmer and Stewart, 2013, and references therein). *Ideological scaling* offers a specific scale (for example 0 to 100 scale of liberalism), while *classification mechanisms* have separate categories (liberal and conservative), without offering a scale. *Supervised* and dictionary-based learning methods assume defined sets of categories allowing for the selection of a particular focus of the analysis rather than relying on underlying categories. If such a predetermined categorization scheme is missing or the goal is to explore unknown classifications, *unsupervised* methods can be helpful. According to (Grimmer and Stewart, 2013), while supervised and unsupervised methods are often seen as competing, they are in fact context-dependent, and can even be complementary. We apply this logic to our study, and utilize the benefits of both methods in a new two-stage procedure to better capture the dynamics of legislative behavior over time.

4.3 Data Generation

4.3.1 Rhetoric in Committee

Our focus on committee rhetoric is novel both in political science and in the statistics and machine learning contexts, providing multi-fold benefits. Despite some indication that the role of committees in the legislative process in Congress is evolving (Sinclair, 2016; Schickler, 2001), Congressional committees continue to be an important component of that process. First, most of the interactions take place within committees, providing richer information on its members. Second, legislative hearings in committee allow for daily measures of changes in rhetoric. Such highly disaggregated data are still underused in political science, where measures of committee speech are absent. Third, understanding dynamics of committee work is essential because policy preferences are expressed during committee hearings, and those discussions shape future bills. While a large number of proposed bills never make it past that stage, they still contain indispensable information about policy positions. Finally, despite the various audiences that scrutinize the behavior in committees, they provide a lower visibility setting where legislative interactions can occur with less attention from the public. Thus, the agenda in congressional committees is more fluid, and it is easier for interest groups to get involved and insert their preferences (Hall and Wayman, 1990; Hojnacki and Kimball, 1998).

Despite the importance of congressional rhetoric, and the fact that the records of the committee hearings are public, to the best of our knowledge, this wealth of data has never been systematically organized and analyzed. This is largely due to the fact that committee hearings span over hundreds of thousands of pages and hand-coding of the data, as is customary in the social sciences, is not a feasible task. Hence, such a dataset on committee rhetoric simply did not exist up to this point. We develop a novel methodical procedure to analyze political rhetoric, with an immediate application for committee speech, but with a multitude of applications beyond that.

Our source of legislative rhetoric is the U.S. Government Printing Office (GPO), which has an archive of transcripts from congressional hearings (U.S. Government Printing Office, 2014). We use data on hearings from 2001 to 2011. However, the unit of analysis is individual rhetoric, which is not readily obtainable. Senators can make a number of statements within the same hearing, thus we utilize a Java algorithm to parse single uninterrupted remarks from the full text of the hearings. The algorithm separates statements by members of the committee from other statements such as testimonies from witnesses, statements by foreign delegations, and others. The algorithm also takes into account changing chairmanship of the committee, and different members sharing the same last name such is the case with Frank Murkowski of Alaska who was the chairman in the beginning of the time period in question, and his daughter Lisa Murkowski, who replaced him in 2002.

The transcripts from the committee include statements by senators (the focus here), as well as statements from witnesses and expert testimony. The witness statements are oftentimes in the form of written prepared testimony. Senators can sometimes submit written remarks as well, usually when they are not present at the hearing. These types of statements occur rather infrequently and since most introductory statements are prepared in advance, the dynamics and inherent structure are similar in written and spoken statements.

The resulting dataset includes 40,525 individual statements from congressional hearings from 2001 to 2011 that are organized temporally. The rhetoric data spans over more than 10,000 pages. Due to the large amount of text that was analyzed, we employ a natural language processing algorithm devised by (Hopkins and King, 2010).

4.3.2 Natural Language Processing of Committee Rhetoric

Computerized text analysis falls within the broader field of pattern recognition and is a rather new field, especially for the social sciences (Monroe and Schrodt, 2008; Pang and Lee, 2008; Shellman, 2008). We conducted an extensive search of appropriate language processing tools, and a method developed by (Hopkins and King, 2010) is particularly suitable for our objectives. The method is a supervised learning approach, allowing us to specify categories of interest (attitudes towards the energy industry), and to estimate document category proportions instead of individual document scores. Our focus is not to find random inherent patterns in the congressional speeches, but to measure changes in specified categories over time that constitute a basis for subsequent quantitative and qualitative analysis. We study document category proportions instead of scoring individual texts — the proportion of speeches within each month that fall within the predetermined categories because the focus is on the speakers and not the speeches themselves. The natural language processing algorithm is used to produce proportion level estimates per senator per month (a senator-month measure).

Let us select a training set with documents D_i, where $i = 1, \ldots, I$. We choose a training sample size I in accordance with the guidelines specified by Hopkins and King (2010, pp.241-242), who recommend 500 documents for minimizing the root mean square error. Our training set is almost double the recommended value and is based on I of 947 statements. The documents in the training set are selected using simple random sampling from the full set of speeches, and hand coded into four categories. The documents are labeled with a label j, such that $D_i = j$. In our case, with four categories, $j = 1, \ldots, 4$. The categories that we use are "*pro-lobby*" (category 1), "*neutral*" (category 0), "*procedural*" (category 9) and "*anti-lobby*" (category -1). An "anti-lobby" statement, for example, would be one that proposes a cut in the subsidies for the oil industry. A "pro-lobby" statement would be one that proposes an increase in the number of drilling permits for off-shore oil drilling. A "procedural" speech is one that thanks an outside expert for being present during a hearing, while an example for a "neutral" statement would be the discussion of the creation of a memorial.

The energy industry is dominated by the fossil fuel and electric utilities sub-industries both in terms of interest group activity, and production. These sub-industries represented 78% of the total energy production in 2010 (EIA (U.S. Energy Information Administration), 2000) and around 92% of the lobbying and campaign contributions from the energy sector (OpenSecrets.org, 2013). Thus, our focus here is on those interests, and the attitudes we measure in the rhetoric are towards these groups. Multivariate classification across the sub-interests (oil, coal, etc.) is preferable, but there is not enough variation in the data to support coding multiple dimensions.

The classification algorithm summarizes the text of the labeled documents D_i using word stems S_{ik}, $k = 1, \ldots, K$, where K is a number of word stems. Thus, the text of each labeled document is represented by a $K \times 1$-vector of word stems, where word stem $S_{ik} = 1$, if the particular word stem is used at least once in document D_i, and $S_{ik} = 0$ otherwise. The selection mechanism and size of the subsets of words that are used for word stems K are outlined in Hopkins and King (2010, p.237).

The population set of documents, D_l, $l = 1, \ldots, L$, includes the full 40,525 speeches for all the members of the committee from 2001 to 2011, which are then subsetted by speaker and time. The documents in the population set have an unobserved classification $D_l = j$, $j = 1, \ldots, 4$, and are described using word stems, similarly as for the training set. The quantity of interest for the algorithm is the aggregate proportion of all of the population documents that fall into each category ($P(D) = \{P(D = 1), \ldots, P(D = 4)\}'$). For example, $P(D' = 1)$ is the estimate of the proportion of documents in category 1, and the true proportion is $P(D = 1)$. Proportion $P(D)$ is a 4×1-vector, where each element is computed as follows

$$P(D = j) = \frac{1}{L} \sum_{l=1}^{L} 1(D_l = j). \tag{4.1}$$

The algorithm estimates the proportion of documents in various categories using

$$P(D' = j) = \sum_{j'=1}^{J} P(D' = j | D = j') P(D = j'). \tag{4.2}$$

Hence, the aggregate proportion of all of the population documents that fall into each category, $P(D_l = j)$, is computed according to

$$\underset{2^k \times 1}{P(S)} = \underset{2^k \times J}{P(S|D)} \underset{J \times 1}{P(D)}, \tag{4.3}$$

where $P(S)$ is a probability of each of the 2^K possible word stem profiles occurring and $P(S|D)$ is a probability of each of the word stem profiles occurring within the documents in category $D_l = j$. Note that $P(S|D)$ signifies that the attitude of the senators towards the energy industry comes before the words that they use in their statements. (We follow Mahatma Gandhi's paradigm "*Your beliefs become your thoughts, Your thoughts become your words...*") For a further discussion of the algorithm, see (Hopkins and King, 2010).

Figure 4.1 illustrates how the algorithm works in practice. A statement such as "*Exxon should be subsidized*" would be labeled as being a category 1 statement, while a statement in the population set, "more subsidies should be given to Exxon" would have an unobserved classification D_l. Word stems S_{i1}, S_{i2}, S_{i3}, S_{i4} appear in the training set, and the unclassified stems (S_{i5}) do not.

Labeled set - $\underbrace{\mathit{Exxon}}_{S_1}\underbrace{\mathit{should}}_{S_2}\underbrace{\mathit{be}}_{S_3}\underbrace{\mathit{subsidized}}_{S_4} \rightarrow$ category

$D_i = 1$

Population set with unobserved classification D_l -

$\underbrace{\mathit{More}}_{\text{unclassified}}\ \underbrace{\mathit{subsidies}}_{S_4}\underbrace{\mathit{should}}_{S_2}\ \underbrace{\mathit{be}}_{S_3}\ \underbrace{\mathit{given}}_{\text{unclassified}}\ \underbrace{\mathit{to}}_{\text{unclassified}}\ \underbrace{\mathit{Exxon}}_{S_1}$

Thus, $S_{i1}, S_{i2}, S_{i3}, S_{i4} = 1$, while $S_{i5} = 0$

Figure 4.1. Example of supervised classification of the energy-related political rhetoric.

The syntax of a language oftentimes needs to be annotated as part of natural language processing. The annotation usually includes part of speech tagging, phrase structure, and dependency structure. Various corpora include parts of speech tags as part of the annotation process. Such a step is particularly important for parsers. Additional annotation for semantic content is also possible, allowing an algorithm to distinguish more complex structures. Finally, annotation can help identify the document level semantic properties implied by a text, such as free-text annotations (Branavan et al., 2009). While annotations can be crucial for certain applications, the so called "bag of words" simplification can be highly effective (Pang et al., 2002). Hopkins and King (2010, p.232) discuss the issue at length and conclude that upon empirical testing, annotation is not necessary for the particular application of the algorithm.

Certain senators might make a higher number of interrupted remarks (such as the chairman) versus longer uninterrupted statements within the same hearing. The final scores are not affected by the length of the individual statements, nor by whether they are interrupted or not because we are not interested in classification of individual statements, but in classifying statement category proportions within a given month. There are different reasons why a senator would be a member of the committee but not make a statement in a given time period. There might not have been a committee hearing in that period (during recesses for example), thus the senators were unable to express their attitudes and we can extend the last available expression of their attitudes to the period with the missing data. The same logic applies in cases of absence from a hearing. However, the cases when there was a committee hearing and a senator did not make a speech contain information because not making a speech while being present on the committee can represent a certain expression of attitude and we cannot assume that he or she simply retained the previously expressed attitude. In these cases, we cannot extend their previous available scores without inducing bias. Instead a score of 0 for all possible categories of rhetoric in that period would signify that there was a hearing, but attitude was not expressed in either category. Ignoring the different reasons for the missing data would induce bias. We handle the missing rhetoric using a combination of extending previous rhetoric scores in cases when a hearing did not occur and assigning a score of 0 when a senator was present, but did not speak.

4.3.3 Re-Indexing Time

The final measures are monthly time series for each member of the committee in the period 2001-2011. For our analysis, we select the pro-lobby rhetoric for all senators who were members of the committee for at least 48 months during the period of interest, and produce two datasets with a focus on Full Membership and Full Term representation. The Full Membership data include 31 senators over 48 months, while the Full Term data includes a smaller number of senators (19) over a longer time period (72 months). The supportive rhetoric measure is on a scale of 0.00 to 1.00.

The data presents a time indexing problem. If we use the “real world clock”, we would account for events affecting the policy environment (oil spills, wars, etc.) and the political environment (divided government, etc.). However, this “clock” does not align with the time the senators have spent on the committee or the time that has passed since their previous election because they are elected at different times. We also cannot account for the time that has passed since they were first elected (their time in the Senate) because some of the members that are included had been in Congress for decades before our period of interest. Finally, aligning the senators based on when they appeared in the data removes the substantive meaning that we are interested in.

Ultimately, our substantive focus is on their behavior based on length of time since their last election. Electoral strategies and concerns have an impact on legislative behavior as evidenced by the "electoral connection" literature (Mayhew, 2004; Rothenberg and Sanders, 2000). We measure the impact of the electoral connection on legislative behavior by re-indexing the data so that all rhetoric begins at month one in office following a senator's election (time-since-last-election). Thus, for both datasets, x_{tm} is rhetoric expressed by the m-th senator at time point t, where $t = 1, \ldots, T$, $m = 1, \ldots, M$. For the Full Membership dataset, $T = 48$, and $M = 31$, while for the Full Term dataset, $T = 72$, and $M = 19$.

4.3.4 Statistical Validation and Reliability Testing

The two main requirements for any automated natural language processing algorithm are reliability and utility over hand-coding. In our case, hand-coding tens of thousands of pages of text is not feasible, so the algorithm clearly adds utility. However, validation and reliability concerns need to be addressed by any researchers that utilize automated text analysis (Grimmer and Stewart, 2013). Biases in supervised learning stem mostly from the human coding that "teaches" the algorithm (Hopkins and King, 2010). A critical component of content analysis is the measure of intercoder reliability or coder agreement (Lombard et al., 2002). Intercoder reliability refers to the degree to which independent coders evaluate a feature (in this case, text) and reach the same conclusion. We perform an intercoder reliability test with two coders who coded the training set of 947 statements. A widely used measure for evaluating intercoder reliability is Krippendorff's α which can be used regardless of the number of observers, levels of measurement, sample sizes, and presence or absence of missing data (Hayes and Krippendorff, 2007; Krippendorff, 2004). The general form of Krippendorff's α is

$$\alpha = 1 - D_o/D_e, \tag{4.4}$$

where D_o is the disagreement observed, and D_e is the disagreement expected by chance (for a detailed discussion, see Krippendorff, 2004). Values of α above 0.8 are generally considered to indicate a high intercoder reliability (Krippendorff, 2004, pp.241-243). In our case, the resulting Krippendorff's α is 0.91, which suggests a high intercoder agreement and a reliable coding scheme.

Furthermore, the algorithm needs to be validated as a reliable tool for replicating human coding (Grimmer and Stewart, 2013, p.271). Cross-validation is a commonly used method for assessing reliability of automated natural language processing algorithms. The idea is to partition data into complementary non-overlapping subsets, perform analysis on one subset, and validate the results on the other subset. The process involves multiple rounds using different partitions (known as folds), with the results averaged over the folds to produce a single estimation. V-fold cross-validation is a process, in which the original text data are randomly partitioned into V equally sized subsamples. We performed five-fold cross-validation, which randomly partitioned the training set into five groups. This allows us to compare the output of the machine coding to the output of the hand coding. The performance is assessed on each of the groups with predictions made on data out of sample. (For a complete discussion of this validation method, see Grimmer and Stewart, 2013, pp.279-280.) In our study, the resulting accuracy (i.e., proportion of correctly classified documents) score is 0.92, which indicates an accurate classifier.

4.4 Cluster Analysis of Political Rhetoric

Grimmer and King (2011) provide an overview of the existing automated methods for cluster analysis of political data, and propose a simultaneous unsupervised algorithm that combines text analysis and clustering, under the assumption of unknown document categories or topics. In our setting, however, we focus on detecting common features and a hidden structure among time series where each time series represents a senator's aggregated behavior in respect to a certain *known* topic, or a baseline. In our study, the pre-specified topic is attitudes towards energy, and the goal is to cluster senators' aggregated behavior with respect to the energy interests and then to trace the changes during their time in office.

Political clusters based on such "behavioral" time series are intrinsically dynamic, with time-varying distributional shapes and number of groups. Indeed, behavior of senators may dynamically evolve over time to maximize their interests at different stages. Hence, most conventional clustering algorithms are inapplicable in the current setting. Due to this complexity of the behavioral time series, it is crucial to choose a more flexible clustering method that can dynamically detect the intrinsic patterns. Some possible methods are clustering techniques from streamed data mining, where a window model is usually used to capture dynamics inside data (Aggarwal et al., 2003; Munro and Chawla, 2004; Cao et al., 2006; Aggarwal, 2007). However, these methods either require the number of clusters to be pre-specified (Aggarwal et al., 2003), thus increasing the level of a-priori subjectivity, or they do not address the temporal dynamics of the data within a cluster, which is required for our political science study (Munro and Chawla, 2004; Cao et al., 2006).

To address these challenges, we bring an idea from environmental studies and adopt a flexible data mining approach, TRend based clUstering algorithm for Spatio-Temporal data (TRUST) that is proposed by Ciampi et al. (2010); Appice et al. (2015) in the context of environmental space-time data. The key idea of TRUST is based on a temporal sliding window argument extended to multiple spatially distributed data sources such as, for instance, geo-referenced sensors. Spatial locations are then grouped together in terms of the proximity of their temporal trajectories in the recent past. We, in turn, apply TRUST by viewing each senator as a location (or "sensor"), and cluster them in terms of the similarity of their rhetoric. However, in the framework of political time series, there exists no spatial information, and senators are grouped together as long as they share similar temporal trajectories in their rhetoric. Therefore, our modified algorithm is targeted to detect dynamic clustering in time series only, and is referred as reduced-TRUST, or R-TRUST.

Below we present a schematic idea of the R-TRUST algorithm. (Since in general TRUST aims to preserve the space-time continuity of the observed data, its detailed description is tedious, and for more details we refer the reader to Ciampi et al., 2010, and discussion therein.)

For both datasets, x_{tm} is rhetoric expressed by the m-th senator at time point t, where $t = 1, \ldots, T$, $m = 1, \ldots, M$. (For the Full Membership dataset, $T = 48$, and $M = 31$, while for the Full Term dataset, $T = 72$, and $M = 19$.) Let X be a $T \times M$-matrix with elements x_{tm}. Each row of X, i.e. $x_{\cdot 1}, \ldots, x_{\cdot M}$, represents the rhetoric of all senators at a given time point and is called a *layer*. A set of p consecutive rows is called a *slide*, and a set of ω consecutive layers forms a *sliding window* of size ω.

The R-TRUST algorithm consists of two main steps. The first step is *slide-level* clustering. Let the i-th slide, denoted by X_i, be a $p \times M$ matrix, where $p < T$, $i = 1, \ldots, \lfloor T/p \rfloor$ and $\lfloor r \rfloor$ denotes the floor function, i.e., greatest integer less than or equal to r. Each column of X_i, i.e. $x_{1\cdot}, \ldots, x_{p\cdot}$ represents temporal behavior of one senator (i.e., a sequence of time series). R-TRUST randomly starts with one time series l as an initial point (i.e., *seed time series*) and searches time series m for close relations in terms of E_δ^θ (referred to as *neighbors*). The procedure is formalized as follows

$$E_\delta^\theta = \left\{ x_{\cdot m}, x_{\cdot l}, m, l = 1, \ldots, M \middle| \sum_{t=1}^{p} \psi_\delta(x_{tm}, x_{tl}) \geq \theta \times p \right\}, \tag{4.5}$$

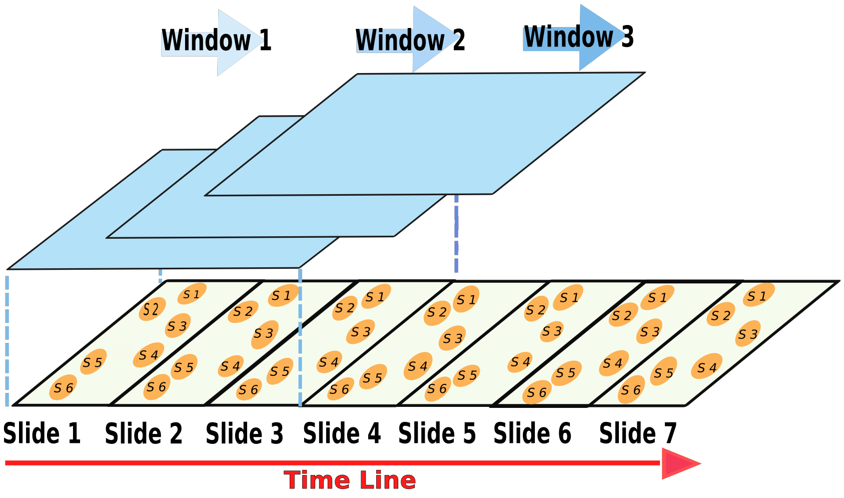


Figure 4.2. Sketch of the R-TRUST clustering approach for a toy example with 6 senators. Circles $S_1, \ldots, S_6$ denote times series corresponding to rhetoric of each senator over a period of one slide.

where

$$\psi_\delta(x_{tm}, x_{tl}) = \begin{cases} 1 & \text{if } \frac{|x_{tm} - x_{tl}|}{\beta - \alpha} \leq \delta \\ 0 & \text{otherwise,} \end{cases} \tag{4.6}$$

where $x_{\cdot m}$ and $x_{\cdot l}$ are rhetoric expressed by the m-th and l-th senators, respectively, over $t = 1, \ldots, T$; θ is the slide-level trend continuity threshold in $[0, 1]$; p is the slide size; $[\alpha, \beta]$ is the domain of slide X_i, and δ is the value-similarity threshold in $[0, 1]$ ($\theta, p, \alpha, \beta, \delta$ are user-prespecified parameters). Hence, the dissimilarity measure $\psi_\delta(x_{tm}, x_{tl})$ is a weighted l_1-distance between $x_{\cdot m}$ and $x_{\cdot l}$.

The seed time series and neighbors found by (4.5) form the initial cluster. Each neighbor time series is then chosen as seed time series and applied to (4.5) to further expand the cluster in an iterative fashion. Once each time series is assigned into a cluster, the algorithm stops and returns a Γ slide-level clustering set for slide X_i. (For more details on pseudocode of the algorithm and definitions of concepts, see (Ciampi et al., 2010; Appice et al., 2015).)

Second, *window-level* clustering: senators are identified to belong to the same slide-level cluster over a window of size ω if their rhetoric trajectories are clustered jointly in at least

$[\epsilon\omega]$ slides; here $\epsilon \in \mathbb{R}, 0 < \epsilon < 1$ and $[\cdot]$ denotes its integer part. Slide-level clustering can be performed as a stand-alone task or a preliminary stage for window-level clustering.

R-TRUST has multifold benefits: 1. it does not require a number of clusters a priori as opposed to k-means; 2. it can detect arbitrarily shaped clusters; 3. it can dynamically detect the drift of space-time data distributions by using a sliding window moving from past to recent, which provides flexibility to detect various behaviors and political expressions of senators at different time periods.

Following Ciampi et al. (2010), layer and window sizes are defined via expert knowledge input, e.g. corresponding to climate cycles. In our case, we set the slide size to four and the window size to three. These settings produce 12 month periods for the analysis. The 12 month period is a reasonable choice for the analysis because a Congressional session lasts for around a year or until the Congressional chambers decide to adjourn for that year. A full session of 12 months is a natural time period for position taking as it follows the cycle of the institution. Finally, we select δ using the clustering cross-validation based (in)stability principle (Dudoit and Fridlyand, 2002; Ben-David et al., 2006; Ben-David and Von Luxburg, 2008; Wang, 2010). In particular, we adopt the Downhill Riding (DR) procedure of (Huang et al., 2016) with 2-fold cross-validation averaged over 100 rounds.

4.5 Case Study

As outlined in Section 4.3, the natural language processing algorithm produced two datasets that differ in their membership and temporal composition. The Full Membership dataset includes 31 senators over 48 months, while the Full Term dataset contains a smaller number of senators (19) over a longer time period (72 months). The reasons for the inclusion of two separate datasets are both substantive and methodological. It allows us to evaluate the performance of the clustering algorithm with varying data structures — differences in the number of time points and in the estimates per time point. In particular, the Full

Membership dataset, which contains a higher number of senators, allows us to better evaluate the dynamics of expressed rhetoric due to variation in the types of included senators. This provides more generalizable results in terms of membership-related factors. For instance, some of the senators included in the Full Membership dataset are senators who retired in the period that we study; others were first time senators that just started their first term. The Full Term dataset, on the other hand, comprises one full term in the Senate — 72 months. While not every senator from the Full Membership dataset is included in the second data set, analysis of the 19 senators over their full term allows us to test how clusters evolve in the last years before an election. Hence, the combined conclusions from the results on both data sets allow us to paint a fuller and multi-perspective picture of the behavior in the Senate committee.

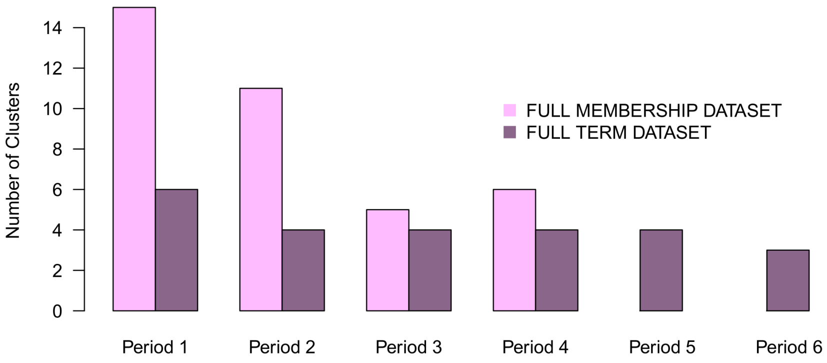


Figure 4.3. Summary of clustering results for the Full Term (6 periods) and Full Membership (4 periods) datasets. Each period is 12 months. Note that since the Full Membership dataset is limited to only 4 periods, no clustering exists for periods 5 and 6.

Figure 4.3 depicts clustering results for the Full Membership and Full Term datasets, obtained using TRUST. The results are comparable across the datasets — the number of clusters diminishes over time. In both datasets, there is a higher number of clusters in the beginning of the senators' terms than in later periods. The biggest changes occur in the earlier periods, and then the number of clusters stabilize. The dynamics are comparable in the two datasets despite the different membership and length of the analyzed period. The findings suggest an astonishing phenomenon — whether we are observing institutionalization or similarities in strategic behavior, legislative speech is congruous, and becomes more so as a senator's term progresses. Politicians, especially in Congress, run for office on a promise of being different from their colleagues, but we find that these differences greatly diminish soon after they take office. Our results are also a proof of concept, political speech can in fact be analyzed, and it does follow detectable patterns. However, simply analyzing the number of clusters does not show us the full picture. Hence, we proceed with a more fine-grained analysis of the composition and structure of the clusters in each dataset.

In certain contexts, unsupervised clustering models might include substantive clusters/sub-clustering specification that can help detect latent clustering effects (Womack et al., 2014; Yano et al., 2014; Gill and Casella, 2009; Sireci and Geisinger, 1992). Womack et al. (2014) propose a Bayesian approach where the random effects are modeled with a Dirichlet process mixture prior. The application of such specifications is for instances when "a variable could be a strong determinant of the outcome variable, but its effect is sufficiently heterogeneous across individuals that it does not appear statistically reliable in the model" (Womack et al., 2014, p.2). The specification accounts for latent clusters where the effect of the variable might differ between clusters, affecting the way the explanatory variable is assessed in the model summary. Additional modeling of the substantive/latent clusters can be beneficial when working with unsupervised clustering. However, such specifications rely on the inclusion of information from explanatory variables, which is beyond the scope of the analysis proposed here.

The issue of substantive clusters also relates to the identity of the nodes within the clusters. Of interest is which nodes are included in each cluster, and what is the unifying feature. Two aspects of our clustering analysis relate to that concept. Our focus is not on the relationships between the individual nodes (senators) or the network structure within the clusters, but the overall behavior over time (whether the nodes converge, not who converges). The unifying features are unspecified, which is a crucial component behind the selection of the TRUST algorithm – it is a dynamic data-driven clustering procedure that allows the *number, shape and distributional properties* of the clusters to vary. Additionally, the cluster numbers are not fixed between time periods – cluster number one in time period one is not necessarily the same as cluster number one in time period two so we are not observing the "evolution" of specific clusters over time. These aspects of the analysis were chosen deliberately due to our focus on the overall dynamics instead of the individual dynamics of specific nodes (senators).

Remark To evaluate the stability of our findings, we also consider multiple settings for slide and window sizes in the R-TRUST algorithm, and such additional studies yield qualitatively similar results. Furthermore, we also investigate the clustering dynamics of both data sets using Density-Based Spatial Clustering of Applications with Noise (DBSCAN) of Ester et al. (1996) and a more conventional k-means method. While neither DBSCAN nor k-means are aimed for cluster recovery in dynamic space-time data, DBSCAN is considered to be the most popular and most cited clustering algorithm for spatial data (Microsoft Academic Search, 2016); while k-means is (arguably) a conventional starting point in many clustering studies (Steinbach et al., 2000; Kanungo et al., 2002; Friedman et al., 2009). In the case of DBSCAN, we select optimal clustering parameters using cross-validation based (in)stability criterion (Huang et al., 2016). In the case of k-means, we choose the optimal number of clusters using the silhouette analysis (Rousseeuw, 1987). Both approaches deliver similar findings as TRUST, that is, the number of groups is highest in the beginning of the senators' terms and gradually diminishes over time. (The results for DBSCAN and k-means are omitted here but are available from the authors. The R code used for the analysis is made available by the authors as part of the *funtimes* R package (Lyubchich and Gel, 2017b).)

4.5.1 The Full Membership Dataset — High Number of Senators, Shorter Period

Our clustering findings for the Full Membership dataset are depicted in Figure 4.4. The circles represent the clusters, and their sizes convey the relative size of the clusters. The initials of the senators, their party and state are shown in the circles. The smaller clusters are individual senators whose speech is different from the rest of their colleagues. Their names are omitted from the circles for clarity.

The composition of the clusters reveals that not only the number of clusters diminishes over time, but the membership in the clusters becomes more concentrated in fewer clusters. Less clusters exist over time, but also more senators belong to fewer clusters with a small number of senators representing separate clusters, or forming small clusters with a few others. Overall, the senators become more alike.

We observe a high number of clusters in the first year in office with a small number of senators in each cluster, as illustrated in time period 1 in Figure 4.4. This is to be expected given the current levels of polarization in Congress, which is at an all time high — congressmen rarely agree on policies and there is not much compromise. A number of widely known surveys show that the institution is the least popular part of the government, surpassed even by the IRS (GALLUP, 2014; The Washington Post, 2011). Legislators are aware of it and often run their campaigns promising their constituents that they are different than their colleagues in office. This creates a situation where the voters like their congressman, but hate Congress (GALLUP, 2013; The Wall Street Journal, 2014). The results of our clustering analysis depict these attitudes and promises. In their first and second years in office, congressmen use their rhetoric to differentiate themselves from their colleagues and do not coalesce into big groups. This trend is sustained throughout their second year in office, when we still observe a relatively high number of groups, but the total number diminishes — they begin to form bigger clusters.

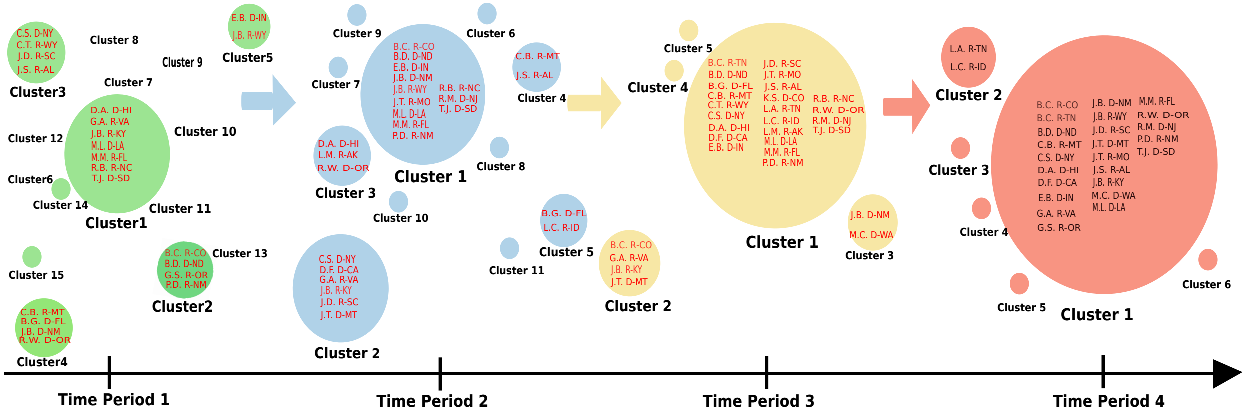


Figure 4.4. Dynamic Clustering of Legislative Rhetoric in the Full Membership dataset.

A significant change occurs after the second year, as seen in the remarkable change between time period 2 and time period 3 in Figure 4.4. The slow decrease in the number of total clusters that can be observed during year two in office is accelerated and the total number of clusters drops precipitously. One big cluster is formed in this period, as well as a number of smaller ones. The rhetoric becomes more alike and both Republicans and Democrats start speaking in similar ways. This type of behavior is associated with institutionalization or “marrying the locals” — it occurs in various bureaucracies where an outsider becomes more like those already in the institution. The trend peaks in the fourth year (time period 4 in Figure 4.4), when the largest cluster becomes even bigger and almost all senators are included. The rhetoric towards the energy sector is very similar across parties and geography.

The clusters are not based on simple party lines, but show more complex dynamics. These dynamics are more evident with the newer members of the committee — senators Maria Cantwell D-WA (joined in 2001; cluster numbers 7,6,3,1), Jon Tester D-MT (joined in 2007; clusters 12,10,2,1), and Bob Corker R-TN (joined in 2007; clusters 15,11,1,1), were not clustering with anyone else in their first one or two years in office. They are relatively new members of the committee and do not yet have well established relations with their colleagues, or with the various interest groups. Their rhetoric slowly converges with the rest of their colleagues, and by their fourth year, they become members of the biggest cluster (cluster 1). Similar dynamics can be observed with other relatively new senators who converge with the rest of the institution at various speeds — senators Robert Menendez D-NJ (joined in 2006, cluster numbers 8,1,1,1), James Talent R-MO (joined in 2002, clusters 9,1,1,1), and Jim Demint R-SC (joined in 2005, clusters 3,2,1,1), either do not group with others in their first year, or are not part of bigger groups in their first year, but in their second year start speaking in a similar way to the rest of their colleagues.

The senators who do not cluster with others towards their fourth year in office tell a compelling story too. Senator Lisa Murkowski R-AK (cluster numbers 10,3,1,3) was the minority leader in the energy committee, and her speech patterns reflect her special role in the committee. Most of the time she is not part of big groupings, and even in the fourth year in office has different speech patterns than the rest of her colleagues. Minority leaders in a polarized Congress are to be expected to make stronger and more divisive statements (Davidson et al., 2013). Another senator who does not appear in large clusters most of the time is Ken Salazar D-CO (cluster numbers 11,7,1,4) who served between 2005 and 2009, when he became Secretary of the Interior in President Obama's administration. He has strong ties with the coal and oil industries, and often supports these interests. His rhetoric and voting record exhibit complex patters, as a right-of-center Democrat, which explains why he does not cluster together with colleagues most of the time.

Furthermore, since any clustering algorithm can be sensitive to the underlying assumptions built into it, it is important to evaluate sensitivity of the drawn conclusions. We address the issue of clustering sensitivity, by performing a crossvalidation study, i.e., a standard data-driven validation routine in statistics. We find that the crossvalidation (CV) analysis, based on randomly selecting 40%, 60%, and 80% of Senators in the Full Membership Dataset over 10 CV replications, supports our conclusion of a decreasing number of clusters over time and institutionalizing of Senators over their term in office (see Figure 4.5).

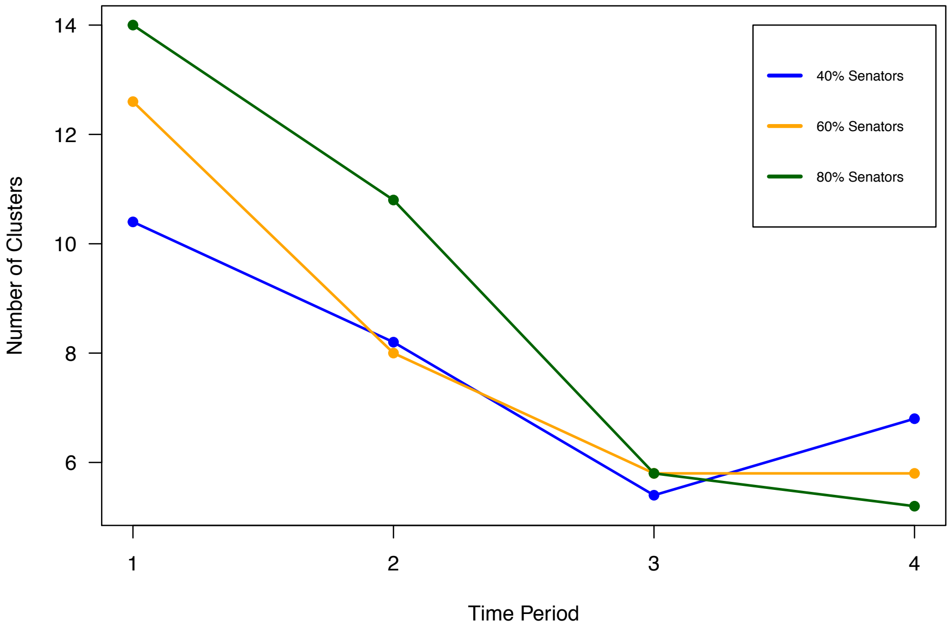


Figure 4.5. Sensitivity analysis for Full Membership Dataset. Number of crossvalidation replications is 10.

4.5.2 The Full Term Dataset — Longer Period, Lower Number of Senators

Figure 4.6 shows the clustering results for Full Term dataset. It includes more temporal points, but less senators per time period. Analyzing the full term allows us to study whether the dynamics exhibited in Full Membership dataset hold for the Full Term dataset despite the variation in the number of senators, and whether there exist differences in their last two years in office before an election. While the number of clusters that the algorithm captures differ between the datasets, the dynamics are similar. The number of groups decreases over time in both datasets, although we do not observe the same sharp drop in the Full Term dataset. Just like with the Full Membership dataset, in the Full Term dataset, the group membership becomes more concentrated in fewer clusters. After the first two years, most senators belong to one or two large clusters, which is sustained until the end of their term.

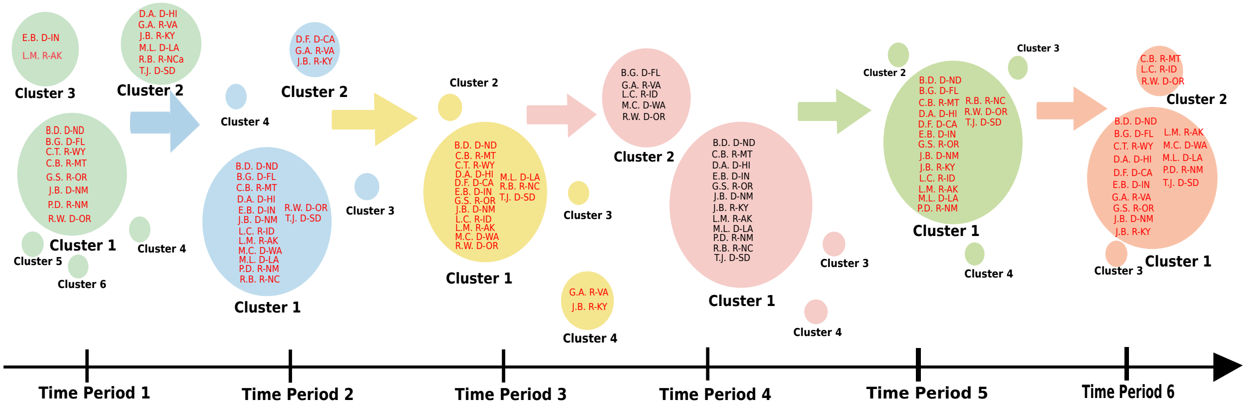


Figure 4.6. Dynamic Clustering of Legislative Rhetoric in the Full Term dataset.

As with the Full Membership dataset, we observe the most significant changes in the first few time periods. The tendency to cluster in bigger groups that we detect in the first dataset is also present here, but the "lumping" of senators occurs sooner in their term. Clusters in the Full Term dataset are not based on simple party lines, but once again on more complex dynamics. The members in this dataset are generally long-time senators that have worked with one another multiple times before and are well aware of what successful strategies look like. This can explain the differences in the results for the two datasets. The senators in the Full Term dataset have likely already been institutionalized in their previous terms, and they follow a familiar strategy — behave differently in the beginning of the term, but then start converging. These similarities in behavior occur frequently in various organizations and depict the "culture" of the institution. This type of homogeneous behavior is beneficial to the institutional agents because it allows them easier access to resources (successful legislation, campaign contributions, or earmarks) and creates informal mechanisms for their distribution. These patterns are not based on partisanship — both Republicans and Democrats follow similar strategies and the dynamics of the rhetoric towards the energy sector is very similar across parties and geography.

The overall dynamics in the data are clear — in the beginning of their tenure, most senators' speech and attitudes are distinct from their colleagues, as most politicians promise on the campaign trail, especially at times when the institution is highly polarized. However, after the first two years in office, the differences tend to diminish. These findings are of importance to interest groups because they can adjust their strategies based on these changes. Understanding these dynamics is crucial when strategizing when to get involved through lobbying and campaign contributions. The results show that legislative rhetoric is malleable and dynamic, and efforts from outside groups to influence and shape it could be more efficient at certain times during a senator's tenure. These conclusions are also meaningful for the voters because of the prevalent "we like our congressman, but hate Congress" attitudes (GALLUP, 2013, 2014, 2016).

Similarly to the Full Membership Dataset, the sensitivity analysis of the Full Term Dataset based the crossvalidation argument, i.e., randomly selecting 40%, 60%, and 80% of Senators, supports our conclusion of a decreasing number of clusters over time and associated institutionalizion of Senators in office (see Figure 4.7). Legislators run on the promise of being different, but the results suggest that their rhetoric when in office is not that different from one another.

4.6 Discussion

Legislative representation is crucial in a democratic society, and politicians typically run for office on the promise of change. Understanding their behavior when in office can help interest groups and voters make informed decisions. However, the hidden sophisticated structure of political time series poses a broad range of methodological challenges and cannot be assessed with conventional statistical procedures of time series analysis and clustering. In this paper, we develop an innovative two-stage hybrid supervised-unsupervised learning methodology to study dynamics in legislative rhetoric in congressional committees, without imposing restrictions on shape, number and structure of clusters. We construct an innovative measure of political rhetoric and produce two datasets with varying structures, both in terms of the

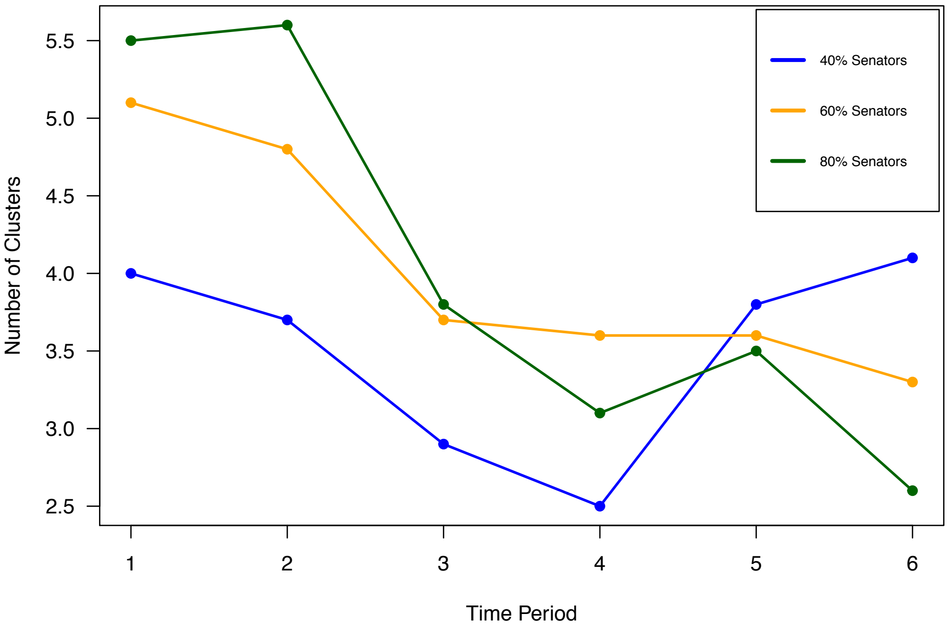


Figure 4.7. Sensitivity analysis for Full Term Dataset. Number of crossvalidation replications is 10.

time component and membership. Such investigation of legislative behavior at the committee level was not available before, and it serves as a proof of concept, allowing for other political speech to be analyzed in a systematic way, and for the uncovering of hidden structures within such data.

The results from our clustering analysis depict compelling dynamics. As the legislative term progresses, senators, despite their party membership and campaign promises, tend to increasingly group together over time. The similarities in their behavior become apparent after the first year or two in office, and near the last year, there are only few groupings that describe the speech. Our findings are in contrast with the promises that politicians usually make on the campaign trail. They usually emphasize how different they are from their opponents, and their future colleagues. As we showed, voters are generally supportive of their own representative, but not of the institution as a whole. We find that these perceived dissimilarities are in fact greatly exaggerated, and politicians are in fact more alike then they are trying to project.

In the future, we plan to incorporate a network component into a rhetoric analysis and to evaluate how the structure of legislative networks evolves over time. Possible algorithms for this future work can be found in the dynamic networks literature (Beykikhoshk et al., 2015; Loglisci, 2013). This is of particular interest in analyzing presidential elections in the United States, as well as statements by terrorist organizations, and foreign political and social leaders, to name a few. Another future component is analysis at the document level, which would require a frequency-based solution. Such classification would be a beneficial addition to the current analysis and would allow us to study the effect of specific events such as terrorist attacks or international conflicts.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

This dissertation opened up new horizons for more robust non-parametric analysis of space-time clustering procedures.

In particular, we propose a new data-driven and computationally efficient procedure called Downhill Riding (DR) for optimal selection of clustering tuning parameters in dynamic clustering algorithms like TRUST and DBSCAN using a clustering stability probe (Chapter 2). Using simulations, as well as real data, we show the effectiveness of the new procedure for selection of optimal parameters. The finite sample performance of Downhill Riding for dynamic clustering of synthetic time series is close to the optimal for these algorithms. Furthermore, the performance of clustering algorithms using Downhill Riding against competing algorithms that have a-priori knowledge of the parameters, shows that our procedure is a viable alternative, and often performs better. Lastly, We illustrate the Downhill Riding procedure in dynamic cluster detection in yearly temperature records among 167 stations in Central Germany over 1951–2010. Based on the clustering results of TRUST and DBSCAN, not only do we discover a well known pattern but also a dynamic pattern, which is useful when studying spatially varying climatic changes. We also illustrate the DR procedure in dynamic cluster detection in monthly average concentrations of suspended solids across 133 stations in Chesapeake Bay for a 32-year period (1985–2016). We find remarkable patterns in the data that can provide an insight into the management of resources in the area and the effects of the restoration activities over time. Based on our clustering results, we discover a dynamic pattern, which is useful when studying spatially varying ecological changes. The identification of clusters in the water quality in the Chesapeake Bay has a number of applications and can help address problems in the area. Identifying concentrations of pollutants can aid in determining sources of contamination and assessing which parts of the Bay are at risk. The clustering results provide a clearer picture of the environmental impact of various activities in the area, and can aid future restoration efforts in creating targeted interventions for specific parts of the Bay.

Furthermore, we propose a new robust data depth based clustering algorithm CRAD with a locally-defined neighbor searching function (Chapter 3). Besides robustness to outliers, we show that the new CRAD algorithm is highly competitive in detecting clusters with varying densities, compared with the existing algorithms such as DBSCAN, OPTICS and DBCA. Furthermore, the performance of DBSCAN is shown to be effectively improved, by replacing its original neighbor searching function with the new locally tuned neighbor searching algorithm. In addition, we propose a new effective parameter selection procedure, to select the optimal underlying parameter in the real-world clustering, when the ground truth is unavailable.

Lastly, not only limited to environmental space-time data in Chapter 2, the dynamic clustering procedure combined with DR procedure can also be extended to unstructured space-time data—legislative rhetoric data in the U.S. Senate. We develop an innovative two-stage hybrid supervised-unsupervised learning methodology to study dynamics in legislative rhetoric in congressional committees, without imposing restrictions on shape, number and structure of clusters. We construct an innovative measure of political rhetoric and produce two datasets with varying structures, both in terms of the time component and membership. Such investigation of legislative behavior at the committee level was not available before, and it serves as a proof of concept, allowing for other political speech to be analyzed in a systematic way, and for the uncovering of hidden structures within such data.

5.2 Future Work

- We plan to extend the new Downhill Riding procedure (Chapter 2) from univariate parameter selection to multivariate parameters selection and investigate the utility of Downhill Riding in other clustering algorithms.

- We intend to investigate the utility of other data depth functions as dissimilarity measures such as simplicial volume depth, L^p depth, and projection depth (Mosler, 2013; Zuo and Serfling, 2000) and extend the CRAD idea (Chapter 3) to functional data clustering.

- We also plan to incorporate a network component into a rhetoric analysis and to evaluate how the structure of legislative networks evolves over time. Possible algorithms for this future work can be found in the dynamic networks literature (Beykikhoshk et al., 2015; Loglisci, 2013). This is of particular interest in analyzing presidential elections in the United States, as well as statements by terrorist organizations, and foreign political and social leaders, to name a few. Another future component is analysis at the document level, which would require a frequency-based solution. Such classification would be a beneficial addition to the current analysis and would allow us to study the effect of specific events such as terrorist attacks or international conflicts.

REFERENCES

Aggarwal, C. C. (2007). *Data streams: models and algorithms*, Volume 31. Springer Science & Business Media.

Aggarwal, C. C., J. Han, J. Wang, and P. S. Yu (2003). A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases-Volume 29*, pp. 81–92. VLDB Endowment.

Anisimov, O., V. Kokorev, and Y. Zhil'tsova (2013). Temporal and spatial patterns of modern climatic warming: case study of northern eurasia. *Climatic Change 118*(3-4), 871–883.

Ankerst, M., M. M. Breunig, H.-P. Kriegel, and J. Sander (1999). Optics: ordering points to identify the clustering structure. In *Proc. SIGMOD'99 Int. Conf. on Management of Data*, Volume 28, pp. 49–60. ACM.

Appice, A., A. Ciampi, and D. Malerba (2015). Summarizing numeric spatial data streams by trend cluster discovery. *Data Mining and Knowledge Discovery 29*(1), 84–136.

Ars Technica (2015, 12). Tech companies urge congress to drop fight against net neutrality rules. `http://arstechnica.com/tech-policy/2015/12/tech-companies-urge-congress-to-drop-fight-against-net-neutrality-rules/`, Accessed: 2016-02-12.

Bache, K. and M. Lichman (2013). UCI machine learning repository.

Banerjee, S., B. P. Carlin, and A. E. Gelfand (2014). *Hierarchical modeling and analysis for spatial data*. CRC Press.

Barr, R. R. (2009). Populists, outsiders and anti-establishment politics. *Party Politics 15*(1), 29–48.

Barry, R. G. (1992). *Mountain weather and climate*. Psychology Press.

Beelen, R., G. Hoek, D. Vienneau, M. Eeftens, K. Dimakopoulou, X. Pedeli, M.-Y. Tsai, N. Künzli, T. Schikowski, A. Marcon, et al. (2013). Development of NO_2 and NO_x land use regression models for estimating air pollution exposure in 36 study areas in europe–the escape project. *Atmospheric Environment 72*, 10–23.

Ben-David, S. and L. Reyzin (2014). Data stability in clustering: A closer look. *Theor. Computer Sci. 558*, 51–61.

Ben-David, S. and U. Von Luxburg (2008). Relating clustering stability to properties of cluster boundaries. In *Proc. COLT*, Volume 2008, pp. 379–390.

Ben-David, S., U. Von Luxburg, and D. Pál (2006). A sober look at clustering stability. In *Learning Theory*, pp. 5–19. Springer.

Beykikhoshk, A., O. Arandjelović, S. Venkatesh, and D. Phung (2015). Hierarchical dirichlet process for tracking complex topical structure evolution and its application to autism research literature. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 550–562. Springer.

Bickel, P. and Y. Gel (2011). Banded regularization of covariance matrices in application to parameter estimation and forecasting of time series. *Journal of the Royal Statistical Society, Ser. B 73*(5), 711–728.

Binder, S. (2015). The dysfunctional congress. *Annual Review of Political Science 18*, 85–101.

Bolton, A. (2016, 3). Anti-establishment mood roils senate democratic primaries. `http://thehill.com/homenews/campaign/271897-anti-establishment-mood-roils-democratic-primaries`.

Boykoff, J. and E. Laschever (2011). The tea party movement, framing, and the us media. *Social Movement Studies 10*(4), 341–366.

Branavan, S., H. Chen, J. Eisenstein, and R. Barzilay (2009). Learning document-level semantic properties from free-text annotations. *Journal of Artificial Intelligence Research 34*, 569–603.

Bubeck, S. and U. v. Luxburg (2009). Nearest neighbor clustering: A baseline method for consistent clustering with arbitrary objective functions. *The Journal of Machine Learning Research 10*, 657–698.

Burden, B. C. (2004). Candidate positioning in u.s. congressional elections. *British Journal of Political Science 34*(02), 211–227.

Caliński, T. and J. Harabasz (1974). A dendrite method for cluster analysis. *Commun. Stat. Theory Methods 3*(1), 1–27.

Canon, D. T. (1989). The institutionalization of leadership in the u.s. congress. *Legislative Studies Quarterly*, 415–443.

Cao, F., M. Ester, W. Qian, and A. Zhou (2006). Density-based clustering over an evolving data stream with noise. In *Proc. SDM*, Volume 6, pp. 328–339. SIAM.

Chen, Y., E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista (2015, July). The ucr time series classification archive. `www.cs.ucr.edu/~eamonn/time_series_data/`.

CIA (2014). *Water damage risk and Canadian property insurance pricing.* Canadian Institute of Actuaries.

Ciampi, A., A. Appice, and D. Malerba (2010). Discovering trend-based clusters in spatially distributed data streams. In *International Workshop of Mining Ubiquitous and Social Environments*, pp. 107–122.

Cormen, T. H., C. E. Leiserson, R. L. Rivest, and C. Stein (2009). *Introduction to Algorithms* (3rd ed.). The MIT Press.

Cox, G. W. and M. D. McCubbins (2005). *Setting the agenda: Responsible party government in the US House of Representatives.* Cambridge University Press.

Cuevas, A., M. Febrero, and R. Fraiman (2007). Robust estimation and classification for functional data via projection-based depth notions. *Comput. Stat. 22*(3), 481–496.

Curry, L., A. Weaver, and E. Wiebe (2012). *Determining the impact of climate change on insurance risk and the global community. Phase I: Key climate indicators.* Sponsored by the American Academy of Actuaries' Property/Casualty Extreme Events Committee, CAS, CIA, and SOA, and with input from CIWG.

Daly, C., M. Halbleib, J. I. Smith, W. P. Gibson, M. K. Doggett, G. H. Taylor, J. Curtis, and P. P. Pasteris (2008). Physiographically sensitive mapping of climatological temperature and precipitation across the conterminous united states. *Int. J. of Climatology 28*(15), 2031–2064.

Davidson, R. H., W. J. Oleszek, F. E. Lee, and E. Schickler (2013). *Congress and its Members.* CQ Press.

de Amorim, R. C. and C. Hennig (2015). Recovering the number of clusters in data sets with noise features using feature rescaling factors. *J. Inf. Sci. 324*, 126–145.

Deutscher Wetterdienst Data Archive (2015). `http://www.dwd.de`, Accessed: 2016-05-26.

Dudoit, S. and J. Fridlyand (2002). A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology 3*(7), research 0036.1–0036.21.

EIA (U.S. Energy Information Administration) (2000, 07). Federal financial interventions and subsidies in energy markets 1999: Primary energy. Web. URL: http://www.eia.gov/oiaf/servicerpt/subsidy/index.html. Accessed: 2014-03-16.

Ester, M., H.-P. Kriegel, J. Sander, and X. Xu (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. KDD'96*, Volume 96, pp. 226–231. ACM.

Euan, C., H. Ombao, and J. Ortega (2015). Spectral synchronicity in brain signals. *arXiv preprint arXiv:1507.05018*.

Fenno, R. F. (2002). *Home Style: House Members in Their Districts (Longman Classics Series)*. Longman Publishing Group Harlow.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics 7*(2), 179–188.

Frakes, W. B. and R. Baeza-Yates (1992). Information retrieval: data structures and algorithms.

Friedman, J., T. Hastie, and R. Tibshirani (2009). *The elements of statistical learning, 2nd ed.*, Volume 1. Springer series in statistics Springer, Berlin.

Gaber, M. M., A. Zaslavsky, and S. Krishnaswamy (2005). Mining data streams: a review. *ACM SIGMOD Record 34*(2), 18–26.

GALLUP (2013, 05). Americans down on congress, ok with own representative. `http://www.gallup.com/poll/162362/americans-down-congress-own-representative.aspx`, Accessed: 2016-02-11.

GALLUP (2014, 06). Public faith in congress falls again, hits historic low. `http://www.gallup.com/poll/171710/public-faith-congress-falls-again-hits-historic-low.aspx`, Accessed: 2016-02-11.

GALLUP (2016, 10). Congress and the public. `http://www.gallup.com/poll/1600/congress-public.aspx`, Accessed: 2016-10-11.

Gan, J. and Y. Tao (2015). Dbscan revisited: mis-claim, un-fixability, and approximation. In *SIGMOD*, pp. 519–530.

Gerrish, S. and D. M. Blei (2011). Predicting legislative roll calls from text. In *Proceedings of the 28th international conference on machine learning (icml-11)*, pp. 489–496.

Gerrish, S. and D. M. Blei (2012). How they vote: Issue-adjusted models of legislative behavior. In *Advances in Neural Information Processing Systems*, pp. 2753–2761.

Gill, J. and G. Casella (2009). Nonparametric priors for ordinal bayesian social science models: Specification and estimation. *Journal of the American Statistical Association 104*(486), 453–454.

Gneiting, T. (2002). Nonseparable, stationary covariance functions for space–time data. *Journal of the American Statistical Association 97*(458), 590–600.

Grajales, C. A. G. (2014, 11). How statisticians have changed elections. `http://www.statisticsviews.com/details/feature/6931581/How-statisticians-have-changed-elections.html`, Accessed: 2016-10-09.

Grimmer, J. (2010). A bayesian hierarchical topic model for political texts: Measuring expressed agendas in senate press releases. *Political Analysis 18*(1), 1–35.

Grimmer, J. and G. King (2011). General purpose computer-assisted clustering and conceptualization. *Proceedings of the National Academy of Sciences 108*(7), 2643–2650.

Grimmer, J. and B. M. Stewart (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*.

Guinness, J., M. L. Stein, et al. (2013). Interpolation of nonstationary high frequency spatial–temporal temperature data. *Annals of Applied Statistics 7*(3), 1684–1708.

Hahsler, M. (2015). *dbscan: Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms.* R package version 0.9-6.

Hall, R. L. and F. W. Wayman (1990). Buying time: Moneyed interests and the mobilization of bias in congressional committees. *American political science review 84*(03), 797–820.

Hartmann, B., I. Schwab, and N. Link (2010). Prototype optimization for temporarily and spatially distorted time series. In *AAAI Spring Symp.: It's All in the Timing.*

Hayes, A. F. and K. Krippendorff (2007). Answering the call for a standard reliability measure for coding data. *Communication Methods and Measures 1*(1), 77–89.

Healy, P. (2015, 10). Democrats find that anti-establishment isn't just a g.o.p. theme. `https://www.nytimes.com/2015/10/04/us/insurgent-candidacies-shaking-up-the-gop-also-dog-democrats.html?_r=0`.

Heikkilä, U. and A. Sorteberg (2012). Characteristics of autumn-winter extreme precipitation on the norwegian west coast identified by cluster analysis. *Climate Dynamics 39*(3-4), 929–939.

Herrnson, P. S. (2007). *Congressional elections: Campaigning at home and in Washington.* Cq Press.

Hibbing, J. R. and E. Theiss-Morse (1995). *Congress as public enemy: Public attitudes toward American political institutions.* Cambridge University Press.

Hinneburg, A. and D. A. Keim (1998). An efficient approach to clustering in large multimedia databases with noise. In *KDD*, Volume 98, pp. 58–65.

Hirano, S., J. M. Snyder, S. Ansolabehere, and J. M. Hansen (2010). Primary elections and partisan polarization in the us congress. *Quarterly Journal of Political Science 5*(2), 169–191.

Hojnacki, M. and D. C. Kimball (1998). Organized interests and the decision of whom to lobby in congress. *American Political Science Review 92*(04), 775–790.

Hopkins, D. J. and G. King (2010). A method of automated nonparametric content analysis for social science. *American Journal of Political Science 54*(1), 229–247.

Horwitz, R. B. (2013). *America's right: Anti-establishment conservatism from Goldwater to the Tea Party.* John Wiley & Sons.

Huang, X. and Y. R. Gel (2017). Crad: Clustering with robust autocuts and depth. In *Data Mining (ICDM), 2017 IEEE International Conference on*, pp. 925–930. IEEE.

Huang, X., I. R. Iliev, A. Brenning, and Y. R. Gel (2016). Space-time clustering with stability probe while riding downhill. In *Proceedings of SIGKDD Mining and Learning from Time Series.*

Huang, X., I. R. Iliev, V. Lyubchich, and Y. R. Gel (2017). Riding down the bay: Space-time clustering of ecological trends. *Environmetrics*.

Hubert, M. and M. Debruyne (2010). Minimum covariance determinant. *Wiley Interdiscip. Rev. Comput. Stat. 2*(1), 36–43.

Issenberg, S. (2012, 12). How obama's team used big data to rally voters. `https://www.technologyreview.com/s/509026/how-obamas-team-used-big-data-to-rally-voters/`.

Jain, A. K., M. N. Murty, and P. J. Flynn (1999). Data clustering: a review. *ACM Comput. Surv. 31*(3), 264–323.

Jeong, M.-H., Y. Cai, C. J. Sullivan, and S. Wang (2016). Data depth based clustering analysis. In *SIGSPATIAL*, pp. 29.

Jia, H., S. Ding, X. Xu, and R. Nie (2014). The latest research progress on spectral clustering. *Neural Computing and Applications 24*(7-8), 1477–1486.

Jörnsten, R. (2004). Clustering and classification based on the l1 data depth. *J. Multivariate Anal. 90*(1), 67–89.

Kanungo, T., D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence 24*(7), 881–892.

Kaye, K. (2014, 11). GOP provided social analytics to key senate campaigns. `http://adage.com/article/datadriven-marketing/gop-provided-social-data-tech-key-senate-campaigns/295573/`, Accessed: 2016-10-09.

Keogh, E. and J. Lin (2005). Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems 8*(2), 154–177.

Kogan, J., C. Nicholas, and M. Teboulle (2006). *Grouping multidimensional data*. Springer.

Krippendorff, K. (2004). *Content analysis: An introduction to its methodology*. Sage.

Leisch, F. and E. Dimitriadou (2010). *mlbench: Machine Learning Benchmark Problems*. R package version 2.1-1.

Li, J., J. A. Cuesta-Albertos, and R. Y. Liu (2012). Dd-classifier: nonparametric classification procedure based on dd-plot. *JASA 107*(498), 737–753.

Lichman, M. (2013). UCI machine learning repository.

Liu, R. Y., J. M. Parelius, K. Singh, et al. (1999). Multivariate analysis by data depth: descriptive statistics, graphics and inference. *Ann. Stat. 27*(3), 783–858.

Lock, E. F. and D. B. Dunson (2013). Bayesian consensus clustering. *Bioinformatics*, btt425.

Loglisci, C. (2013). Time-based discovery in biomedical literature: mining temporal links. *International Journal of Data Analysis Techniques and Strategies 5*(2), 148–174.

Loglisci, C., M. Ceci, and D. Malerba (2015). Relational mining for discovering changes in evolving networks. *Neurocomputing 150*, 265–288.

Loglisci, C. and D. Malerba (2015). Mining periodic changes in complex dynamic data through relational pattern discovery. In *International Workshop on New Frontiers in Mining Complex Patterns*, pp. 76–90. Springer.

Lombard, M., J. Snyder-Duch, and C. C. Bracken (2002). Content analysis in mass communication: Assessment and reporting of intercoder reliability. *Human communication research 28*(4), 587–604.

Lowry, R. (2016, 1). Where is the republican establishment? `http://www.nationalreview.com/article/429596/2016-presidential-race-all-about-anti-establishment-candidate`.

Lozano, A. C., H. Li, A. Niculescu-Mizil, Y. Liu, C. Perlich, J. Hosking, and N. Abe (2009). Spatial-temporal causal modeling for climate change attribution. In *Proc. KDD'09*, pp. 587–596. ACM.

Lux, T. and M. Marchesi (2000). Volatility clustering in financial markets: a microsimulation of interacting agents. *International Journal of Theoretical and Applied Finance 3*(04), 675–702.

Lyubchich, V. and Y. R. Gel (2017a). Can we weather proof our insurance? *Environmetrics 28*(2), e2433.

Lyubchich, V. and Y. R. Gel (2017b). *funtimes: Functions for Time Series Analysis.* R package ver. 4.0.

Mahlstein, I. and R. Knutti (2010). Regional climate change patterns identified by cluster analysis. *Climate Dynamics 35*(4), 587–600.

Markman, J. (2016, 08). Big data and the 2016 election.
`http://www.forbes.com/sites/jonmarkman/2016/08/08/big-data-and-the-2016-election`.

Mayhew, D. R. (2004). *Congress: The electoral connection*, Volume 26. Yale University Press.

Meilă, M. (2007). Comparing clusterings – an information based distance. *J. of Multivariate Analysis 98*(5), 873–895.

Microsoft Academic Search (2016, 10). Top publications in data mining.
`http://academic.research.microsoft.com/RankList?entitytype=1&topDomainID=2&subDomainID=7&last=0&start=1&end=100`, Accessed: 2016-10-10.

Monroe, B. and P. Schrodt (2008). Introduction to the special issue: the statistical analysis of political text. *Political Analysis 16*(4), 351–355.

Mosler, K. (2013). Depth statistics. In C. Becker, R. Fried, and S. Kuhnt (Eds.), *Robustness and complex data structures*, pp. 17–34. Springer.

Mueen, A., E. Keogh, and N. Young (2011). Logical-shapelets: an expressive primitive for time series classification. In *Proc. of the 17th ACM SIGKDD*, pp. 1154–1162. ACM.

Munro, R. and S. Chawla (2004). An integrated approach to mining data streams. *In Technical Report, University of Sydney. School of Information Technologies*.

Nikulin, V. (2015). Strong consistency of the prototype based clustering in probabilistic space. *Journal of Machine Learning Research 16*, 775–785.

Niu, Z., D. Chasman, A. J. Eisfeld, Y. Kawaoka, and S. Roy (2016). Multi-task consensus clustering of genome-wide transcriptomes from related biological conditions. *Bioinformatics 32*(10), 1509–1517.

Nowlin, M. (2015). Modeling issue definitions using quantitative text analysis. *Policy Studies Journal*.

Nykamp, D. Q. (2016, 01). Introduction to local extrema of functions of two variables. Web. URL: http://mathinsight.org/local_extrema_introduction_two_variables. Accessed: 2017-01-23.

OpenSecrets.org (2011, 04). Bp firing up political machine one year after start of oil spill. `http://www.opensecrets.org/news/2011/04/bp-firing-up-political-machine-one/`, Accessed: 2016-02-12.

OpenSecrets.org (2013). Energy/natural resources: Long-term contribution trends. Web. URL: http://www.opensecrets.org/industries/totals.php. Accessed: 2014-03-16.

Pang, B. and L. Lee (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval 2*(1-2), 1–135.

Pang, B., L. Lee, and S. Vaithyanathan (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pp. 79–86. Association for Computational Linguistics.

Phoha, S., N. Jacobson, D. Friedlander, and R. Brooks (2003). Sensor network based localization and target tracking through hybridization in the operational domains of beamforming and dynamic space-time clustering. In *Proc. GLOBECOM'03*, Volume 5, pp. 2952–2956. IEEE.

Pokotylo, O., P. Mozharovskyi, and R. Dyckerhoff (2016). Depth and depth-based classification with r-package ddalpha. *arXiv preprint arXiv:1608.04109*.

Pollard, D. (1981). Strong consistency of k-means clustering. *Annals of Statistics 9*(1), 135–140.

Quinn, K., B. Monroe, M. Colaresi, M. Crespin, and D. Radev (2010). How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science 54*(1), 209–228.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association 66*(336), 846–850.

Rashidi, P. and D. J. Cook (2010). Mining sensor streams for discovering human activity patterns over time. *Proc. 2010 IEEE ICDM*, 431–440.

Ratanamahatana, C., E. Keogh, A. J. Bagnall, and S. Lonardi (2005). A novel bit level time series representation with implication of similarity search and clustering. In *Advances in Knowledge Discovery and Data Mining*, pp. 771–777. Springer.

Rothenberg, L. S. and M. S. Sanders (2000). Severing the electoral connection: Shirking in the contemporary congress. *American Journal of Political Science*, 316–325.

Rousseeuw, P. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Computational and Applied Mathematics 20*, 53–65.

San Jose Mercury News (2015, 03). Google's eric schmidt says h-1b visa changes would help economy. `http://www.mercurynews.com/business/ci_27739294/googles-eric-schmidt-says-h-1b-visa-changes`, Accessed: 2016-02-12.

Schaeffer, E. D., J. M. Testa, Y. R. Gel, and V. Lyubchich (2016). On information criteria for dynamic spatio-temporal clustering. In A. Banerjee, W. Ding, J. Dy, V. Lyubchich, A. Rhines, I. Ebert-Uphoff, C. Monteleoni, and D. Nychka (Eds.), *Proc. of the 6th Int. Workshop on Climate Informatics: CI 2016*, pp. 5–8. NCAR Technical Note NCAR/TN-529+PROC.

Scheel, I., E. Ferkingstad, A. Frigessi, O. Haug, M. Hinnerichsen, and E. Meze-Hausken (2013). A Bayesian hierarchical model with spatial variable selection: the effect of weather on insurance claims. *Journal of the Royal Statistical Society, Ser. C 62*(1), 85–100.

Schickler, E. (2001). *Disjointed pluralism: Institutional innovation and the development of the US Congress.* Princeton University Press.

Shellman, S. (2008). Coding disaggregated intrastate conflict: machine processing the behavior of substate actors over time and space. *Political Analysis 16*(4), 464–477.

Sherman, M. (2011). *Spatial statistics and spatio-temporal data: covariance functions and directional properties.* John Wiley & Sons.

Silva, D. F., V. De Souza, and G. E. Batista (2013). Time series classification using compression distance of recurrence plots. In *Data Mining (ICDM), IEEE 13th International Conference on*, pp. 687–696.

Sinclair, B. (1983). *Majority Leadership in the United States House.* Johns Hopkins University Press.

Sinclair, B. (2016). *Unorthodox lawmaking: New legislative processes in the US Congress.* CQ Press.

Sireci, S. G. and K. F. Geisinger (1992). Analyzing test content using cluster analysis and multidimensional scaling. *Applied Psychological Measurement 16*(1), 17–31.

Skocpol, T. and V. Williamson (2012). *The Tea Party and the remaking of Republican conservatism.* Oxford University Press.

Soliman, M., V. Lyubchich, Y. R. Gel, D. Naser, and S. Esterby (2015). *Machine Learning and Data Mining Approaches to Climate Science*, Chapter Evaluating the impact of climate change on dynamics of house insurance claims, pp. 175–183. Switzerland: Springer.

Stein, M. L. (2005). Space–time covariance functions. *Journal of the American Statistical Association 100*(469), 310–321.

Steinbach, M., G. Karypis, V. Kumar, et al. (2000). A comparison of document clustering techniques. In *KDD workshop on text mining*, Volume 400, pp. 525–526. Boston.

Stramp, N. and J. Wilkerson (2015). Legislative explorer: Data-driven discovery of lawmaking. *PS: Political Science & Politics 48*(01), 115–119.

Talbert, J. C., B. D. Jones, and F. R. Baumgartner (1995). Nonlegislative hearings and policy change in congress. *American Journal of Political Science*, 383–405.

The Wall Street Journal (2014, 10). Americans hate congress, but like their own representatives. `http://blogs.wsj.com/numbers/americans-hate-congress-but-like-their-own-representatives`, Accessed: 2016-02-11.

The Washington Post (2011, 11). Congress' approval problem in one chart. `https://www.washingtonpost.com/blogs/the-fix/post/congress-approval-problem-in-one-chart/2011/11/15/gIQAkHmtON_blog.html`, Accessed: 2016-02-11.

Urbancic, T., C. Trifu, J. Long, and R. Young (1992). Space-time correlations ofb values with stress release. *Pure and Applied Geophysics 139*(3-4), 449–462.

U.S. Environmental Protection Agency (2012, 12). Toxic contaminants in the Chesapeake Bay and its watershed–extent and severity of occurrence and potential biological effects–technical report. Web. URL: http://executiveorder.chesapeakebay.net/ChesBayToxic_finaldraft_11513b.pdf. Accessed: 2017-01-23.

U.S. Government Printing Office (2014, 3). Congressional hearings — senate committee on energy and natural resources. `http://www.gpo.gov/fdsys/browse/committee.action?chamber=senate&committee=energy`.

Van Aelst, S. and P. Rousseeuw (2009). Minimum volume ellipsoid. *Wiley Interdiscip. Rev. Comput. Stat. 1*(1), 71–82.

Vines, R. (1985). European rainfall patterns. *Journal of Climatology 5*(6), 607–616.

Vinh, N. X., J. Epps, and J. Bailey (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research 11*(Oct), 2837–2854.

Von Luxburg, U. (2010). *Clustering Stability.* Now Publishers Inc.

Wagner, S. and D. Wagner (2007). *Comparing clusterings: an overview.* Universität Karlsruhe, Fakultät für Informatik Karlsruhe.

Wang, J. (2010). Consistent selection of the number of clusters via crossvalidation. *Biometrika 97*(4), 893–904.

Wei, L. and E. Keogh (2006). Semi-supervised time series classification. In *Proc. KDD'06*, pp. 748–753. ACM.

Werner, P. and F. Gerstengarbe (1997). Proposal for the development of climate scenarios. *Climate Research 8*(3), 171–182.

Womack, A., J. Gill, and G. Casella (2014). Product partitioned dirichlet process prior models for identifying substantive clusters and fitted subclusters in social science data. Washington University Technical Paper.

Yano, T., N. A. Smith, and J. D. Wilkerson (2012). Textual predictors of bill survival in congressional committees. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 793–802. Association for Computational Linguistics.

Yano, T., N. A. Smith, and J. D. Wilkerson (2014). Textual predictors of bill survival in congressional committees. In *Proceedings of the LWA 2014 Workshops*.

Ye, L. and E. Keogh (2009). Time series shapelets: a new primitive for data mining. In *KDD*, pp. 947–956.

Zakaria, J., A. Mueen, and E. Keogh (2012). Clustering time series using unsupervised-shapelets. In *ICDM*, pp. 785–794.

Zuo, Y. and R. Serfling (2000). General notions of statistical depth function. *Ann. Stat.*, 461–482.

BIOGRAPHICAL SKETCH

Xin Huang was born in Jinan, Shandong, China, on June 1, 1990, the son of Yan Huang and Ge Gao. He went to Shandong Experimental High School and received his Bachelor of Science in Applied Mathematics from Shandong Normal University at Jinan in June 2013. In August 2013, he joined the PhD program in Statistics at The University of Texas at Dallas. Under the supervision of Professor Yulia R. Gel, he conducts research on statistical foundations of data science, in particular, robust analysis of non-parametric space-time clustering. He received his Master of Science in Statistics in December 2015 and his PhD in Statistics in August 2018 from The University of Texas at Dallas.

Xin Huang has published three papers and has one paper under revision during his PhD study. In 2015 he received the Best Student Presentation Award from Wiley at the 25th conference of The International Environmetrics Society (TIES) at Al Ain, United Arab Emirates. In 2016 he was awarded the Julia Williams Van Ness Merit Scholarship (the only graduate awardee) from School of Natural Science and Mathematics at The University of Texas at Dallas.

In his spare time, he loves basketball, soccer, and painting. He took Kobe Bryant as his role model whose spirit (Mamba Mentality) motivates and inspires him all the way in both study and life. After completing his PhD, he will be joining JP Morgan & Chase as a Quantitative Researcher in New York.

CURRICULUM VITAE

Xin Huang

hxgy610@gmail.com

Education

Ph.D. in Statistics. GPA: 3.90 / 4.0 **2013 – Expected 2018**
The University of Texas at Dallas Richardson, TX

B.S. in Mathematics and Financial Mathematics. GPA: 3.74 / 4.0 **2009 – 2013**
Shandong Normal University Jinan, China

Work Experience

JPMorgan Chase & Co. **New York, NY**
Summer Associate in Quantitative Research May 2017 – August 2017

- Deep learning in time series missing value imputation with Python.
  - Applied Long Short-Term Memory (LSTM) recurrent neural network, collaborative filtering algorithm, lasso regression, and SVM to impute missing values in a time series with multiple correlated time series as predictors.
  - Designed an imputation framework to automatically select the best model based on different time period of the data.

Procter & Gamble (P&G) **Cincinnati, OH**
PhD Summer Internship in R&D Quantitative Sciences May 2016 – August 2016

- Deep learning in time series classification: predicting shaving pressures of consumers.
  - Employed Multi-Channels Deep Convolutional Neural Networks (MC-DCNN) model in Python to classify time series of shaving pressures on the face using multi-dimensional time series of motion sensor data as features.
  - Implemented XGBoost model with grid search in Python to classify time series of shaving pressures on the face using the Fourier transform of motion sensor data as features, comparing with the MC-DCNN model.
- Machine learning in image analysis: developing an interactive web application for fungus detection on package materials.
  - Applied dlib in Python to detect the object of interest in images and utilized JMP to build a labeled dataset.
  - Implemented logistic regression in R to classify pixels of images using L, a, b color scale as features.
  - Developed an interactive web application of the above detection process using R Shiny and improved the efficiency of fungus detection in product quality monitoring and experimental studies for company.

Travelers Insurance **Hartford, CT**
Advanced Analytics Internship in R&D Program June 2015 – August 2015

- Design an automatic monitoring procedure for Highway Loss Data Institute (HLDI) vehicle characteristic data using SAS.
  - Used advanced data visualization techniques such as geographic heat maps and multi-panel distribution charts to explore geospatial patterns of vehicle classes within 700 million records and automated all the process in SAS Macros.
  - Improved the efficiency of future HLDI data exploration for company by using the created SAS Macros.

The University of Texas at Dallas **Richardson, TX**
Teaching Assistant for Integral Calculus August 2013 – Present

  - Led problem sessions every week, demonstrating mathematical problem-solving skills to students.

Computer Skills

Programming Language: Python, R, C++, Java, SQL, SAS, JMP, MATLAB.

Operating System: UNIX / Linux, Windows, OS X

Academic Projects

Unsupervised Machine Learning in Spatial-Temporal Data: Detecting Dynamic Trend-Based Clusters

Current Research with Advisor – Dr. Yulia R. Gel August 2015 – Present

- Proposed a new density-based clustering algorithm named CRAD which is based on a new neighbor searching function with a robust data depth as the dissimilarity measure and further extended CRAD to time series clustering.
- Developed a new data-driven procedure called Downhill Riding (DR) for optimal selection of tuning parameters in dynamic clustering algorithms such as DBSCAN, OPTICS, and TRUST using the notion of clustering stability probe.
- Applied dynamic clustering algorithms with the new DR procedure on environmental data – clustering climate stations in Germany based on temperature records – and political data – clustering senators in U.S. congress based on their rhetoric.

Application of Text Mining, Machine Learning Document Classification in Identifying the Speaker of Unmarked Presidential Campaign Speeches Using Python October 2015

- Built clean structured corpora from Obama and Romney campaign speeches and created a term-document matrix using TF-IDF weighting scheme.
- Used the k-nearest neighbor algorithm to build a model based on the term-document matrix and predicted the speaker of unmarked presidential campaign speeches.

Publications

- **Xin Huang**, Yulia R. Gel, "CRAD: Clustering with Robust Autocuts and Depth," *Proc. 17th IEEE International Conference on Data Mining (ICDM), 2017.*
- **Xin Huang**, Iliyan Iliev, Vyacheslav Lyubchich, Yulia R. Gel, "Riding down the Bay: Space-time clustering of ecological trends," *Environmetrics,* e2455, 2017.
- **Xin Huang**, Iliyan Iliev, Alexander Brenning, Yulia R. Gel, "Space-Time Clustering with Stability Probe while Riding Downhill," *Proc. 22nd ACM SIGKDD workshop on Mining and Learning from Time Series (MiLeTS),* 2016.
- **Xin Huang**, Vyacheslav Lyubchich, Alexander Brenning, Yulia R. Gel, "Analysis of Dynamic Trend-Based Clustering on Central Germany Precipitation," *Proc. 5th International Workshop on Climate Informatics,* 2015.

Relevant Coursework

Statistical Inference, Linear Statistical Model, Advanced Probability & Stochastic Process, Advanced Statistical Method (General and generalized linear model, Model selection, Multicollinearity, Influential data analysis, Categorical data and dummy variables), Machine Learning (Logistic regression, SVM, Neural network, Random forests, PCA, K-means), Decision Theory and Bayesian Inference, Time Series Modeling & Filtering, Algorithms: Design and Analysis, Introduction to Databases.

Academic Honors

Julia Williams Van Ness Merit Scholarship (only graduate awardee) 2016

School of Natural Science and Mathematics - The University of Texas at Dallas

The Best Student Presentation Award from Wiley 2015

The 25th Conference of The International Environmetrics Society (TIES) at Al Ain, United Arab Emirates

Graduate Tuition Scholarship and Teaching Assistantships 2013 – 2018

The University of Texas at Dallas

National Certificate of Utility Model Patent (An improvement for the refill of gel-ink pens) 2012

The State Intellectual Property Office of the People's Republic of China

Excellent Academic Scholarship 2009 – 2012

Shandong Normal University