

APPLICATIONS AND DIAGNOSTICS FOR DIMENSION REDUCTION
OF MULTIVARIATE DATA

by

Paul Gary Harmon

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

Doctor of Philosophy

in

Statistics

MONTANA STATE UNIVERSITY
Bozeman, Montana

August 2022

©COPYRIGHT

by

Paul Gary Harmon

2022

All Rights Reserved

DEDICATION

This thesis is dedicated to my parents, Gary and Debby Harmon, and my twin brother, Sean Harmon. Thank you for all the support and encouragement along the way.

ACKNOWLEDGEMENTS

There are many people who helped me along the path to my PhD and I greatly appreciate the support I have received along the way. First, I'd like to thank my parents and my twin brother, Sean, for supporting and encouraging me along this endeavor.

I would not have been able to be successful without the flexibility, creativity, support, and guidance provided to me by my adviser, Dr. Mark Greenwood. My committee has been an invaluable help providing feedback and helping me to direct my thinking. The Office of Planning and Analysis, including Dr. Chris Fastnow, Dr. Ian Godwin, Becca Belou, Ian Johnson, and Terry Dysart introduced me to the world of Institutional Research. Without their help I would not have pursued the nuances of the Carnegie Classifications. Thanks for the funding and support, and for a great three years!

Thanks to my coauthors that helped with Chapter 2 of this dissertation, which is published in *Statistics and Public Policy*: Sarah McKnight, Dr. Mark Greenwood, Dr. Laura Hildreth, and Dr. Ian Godwin. Thanks to Eric Loftsgaarden and my coworkers at Atrium: Dr. Christopher Barbour, Dr. Andrew Carroll, Keely Weisbeck, Priscilla Bacino, Noah Benedict, Laurie Rugemer, and Dr. Kenny Flagg all provided inspiration, feedback, and moral support along the way. Thanks to Dr. Tan Tran for his assistance with his implementation of monothetic clustering.

Thanks to Dr. Mark Dahlke and Dr. Darren Homrighausen, who mentored me at CSU and helped me find the perfect graduate program at MSU. Lastly, thanks to Jane White for sparking my interest in AP Statistics more than a decade ago.

VITA

Paul Harmon was born in Grand Junction, Colorado, in 1993. Paul graduated from Grand Junction High School in 2011 and attended Colorado State University for his undergraduate degree, studying Statistics and Economics. While at Colorado State, Paul earned the Maurice Davies Outstanding Student Award from the Colorado-Wyoming Chapter of the ASA. As part of his studies at CSU, Paul completed an Honors Thesis under the direction of Dr. Darren Homrighausen and Dr. Mark Dahlke. Paul graduated with Honors from Colorado State with a BS in Mathematics (Concentration in Statistics) and a BA in Economics in May 2015.

Paul's graduate career began at Montana State University in Bozeman, MT, in Fall 2015. After teaching Pre-Calculus and College Algebra for three semesters, Paul joined the MSU Office of Planning and Analysis to help build dynamic data visualizations for institutional KPIs and performing various statistical analyses of university-related data. As part of Paul's research at OPA, his writing project (working under Dr. Mark Greenwood) focused on explaining why MSU dropped from R1 to R2 in the Carnegie Classifications. This work directly led to the paper that would become Chapter 2 of this dissertation, which focuses on positing an alternative method for institutional classification. En route to earning his PhD, Paul was awarded his MS in Statistics from Montana State University in 2017.

In 2018, Paul joined a technology consulting startup, Atrium Insight, which focuses on building statistical modeling tools for enterprise sales and marketing organizations. While finishing his PhD, Paul has continued to work as a data science consultant, building models to help organizations identify customers, provide thought leadership on AI/ML tools in CRM, and help companies make better use of their data. Paul has spoken at several conferences and webinars as part of his time at Atrium, including presenting at the 2019 Big Sky Dreaming Salesforce user conference.

Paul has presented at numerous statistical conferences, including the Joint Statistical Meetings in 2018, 2019, 2020, as well as the 2020 Midwest Sports Analytics Meetings and 2018 Montana Active Statistics Conference. Paul has published work in *Statistics and Public Policy*.

Outside of his professional pursuits, Paul can be found chasing trout on either the Madison or Gallatin Rivers in Southwest Montana, or "collecting data" on his golf game at one of several local golf courses in Bozeman.

TABLE OF CONTENTS

1. INTRODUCTION	1
1.1 Noise And Signal Variables.....	1
1.2 Multivariate and High-Dimensional Data	2
1.2.1 Dimensionality Reduction	3
1.2.2 Principal Component Analysis.....	3
1.2.3 Factor Analysis.....	5
1.2.4 CFA and Path Analysis.....	6
1.2.5 Structural Equation Modeling	7
1.2.6 Model Fit.....	9
1.3 Mapping High-Dimensional Data	11
1.3.1 PCA for Dimension Reduction.....	12
1.3.2 Multidimensional Scaling	12
1.3.3 Classical MDS	13
1.3.4 Sammon Mapping and Non-Metric MDS.....	14
1.3.5 t-Distributed Stochastic Neighbor Embedding	15
1.3.6 Understanding Perplexity	21
1.3.7 Minimum Spanning Trees.....	23
1.3.8 Best Practices For t-SNE	24
1.4 Introduction to Clustering.....	25
1.4.1 Clustering Methods.....	25
1.4.2 Hierarchical Clustering	26
1.4.3 Monothetic Clustering.....	31
1.4.4 K-means Clustering.....	34
1.4.5 Model-Based Clustering	34
1.4.6 Picking the Number of Clusters	38
1.4.7 Elbow Method.....	38
1.4.8 Silhouette Statistic and Pseudo-F	39
1.4.9 Gap Statistic	41
1.4.10 Clustering Implementation in R.....	42
1.5 Introduction to Sparsity	42
1.5.1 Lasso	43
1.5.2 Sparse Clustering	45
1.5.3 Implementation in R.....	47
1.6 Overview of Repeatedly Used Data Sets.....	47
1.6.1 Carnegie Classification	47
1.6.2 Palmer Penguins.....	49
1.7 Dissertation Overview	50

TABLE OF CONTENTS – CONTINUED

2. AN ALTERNATIVE TO THE CARNEGIE CLASSIFICATIONS: IDENTIFYING SIMILAR DOCTORAL INSTITUTIONS WITH STRUCTURAL EQUATION MODELS AND CLUSTERING.....	52
2.1 Introduction	54
2.2 Methodology of the Carnegie Classifications	58
2.2.1 The Data	58
2.2.2 The Methodology	58
2.2.3 Problems with the Carnegie Classifications.....	62
2.3 SEM-Based classifications	63
2.4 Determination of Groups.....	69
2.4.1 Model-Based Clustering	70
2.5 Shiny Applications for Sensitivity Analysis	73
2.6 Discussion	75
2.6.1 Comparing Carnegie vs. SEM Classifications.....	75
2.6.2 Problems Addressed by the SEM Classifications	76
2.6.3 Further Research	78
3. A METHOD FOR DIAGNOSING INFLUENCE IN HIGH-DIMENSIONAL MAPPING TECHNIQUES.....	79
3.1 Introduction	81
3.2 Tools for High-Dimensional Data Visualization.....	83
3.2.1 Multidimensional Scaling	83
3.2.2 Sammon Mapping and IsoMDS	84
3.2.3 Additional Mapping Techniques.....	85
3.3 Defining “Influence” in High Dimensional Maps.....	86
3.3.1 Cook’s Distance.....	86
3.3.2 Identifying Influential Observations in Maps.....	87
3.3.3 Permutational Multivariate Analysis of Variance.....	89
3.4 Method Overview	91
3.4.1 Generation of P-Values	98
3.5 Simulation Studies	100
3.5.1 Power to Detect Differences.....	101
3.5.2 Type 1 Error Assessment: No Extreme Values.....	101
3.5.3 Single Influential Points	102
3.5.4 Dimensionality of Influential Features	109
3.6 Overview of Real Data Results	113
3.6.1 Penguin Data	113
3.6.1.1 Uncontaminated penguin data set.....	113

TABLE OF CONTENTS – CONTINUED

3.6.1.2 Contaminated penguin data set	116
3.6.2 Carnegie Classification	120
3.7 The Curious Case of Extreme Points and t-SNE	124
3.7.1 t-SNE	124
3.7.2 Simulations	127
3.7.3 A Note on Perplexity	128
3.7.4 MNIST	129
3.8 Conclusions and Extensions	133
3.8.1 A Note on Sample Size and Dimensionality	134
3.8.2 Future Work	134
4. METHODS FOR SPARSE MONOTHETIC CLUSTERING	136
4.1 Introduction	138
4.1.1 Monothetic Clustering	140
4.1.2 Motivating Example	144
4.2 Existing Methods for Sparse Clustering	145
4.2.1 Sparse K -means Clustering	147
4.2.2 Sparse Hierarchical Clustering	148
4.2.3 COSA Alternative	149
4.3 Methods	151
4.3.1 Our Proposed Sparse MC Method	151
4.3.2 Selection of Tuning Parameters	153
4.4 Simulation Studies	158
4.4.1 Selecting Tuning Parameters	158
4.4.2 Simulation 1: Use of Gap To Optimize K (With No Noise Features)	159
4.4.3 Simulation 2: Use of the Gap Statistic to Optimize s with fixed K	162
4.4.4 Simulation 3: Simultaneously Choose Tuning Parameters	165
4.4.5 Simulation 4: Comparison with Other Methods	172
4.5 Real Data Analysis	173
4.5.1 Palmer Penguins	175
4.5.2 2015 Carnegie Classification	183
4.5.3 High-Dimensional LC-MS Data	189
4.6 Glass Data	190
4.7 Discussion	194
4.7.1 Computational Gains	196
4.7.2 Future Work and Extensions	197

TABLE OF CONTENTS – CONTINUED

5. CONCLUSIONS AND EXTENSIONS	199
5.1 Conclusions	199
5.2 Extensions	201
5.2.1 One-Step Ordination and Extensions to SEM Framework.....	201
5.2.2 Influence in Other Map Settings	202
5.2.3 Monothetic Clustering Extensions.....	205
5.2.4 Sparse Implementation of t-SNE.....	206
5.2.5 Sparse Improvements to Influential Point Diagnostic.....	208
5.2.6 Applications to Natural Language Processing	208
REFERENCES CITED.....	210

LIST OF TABLES

Table	Page
2.1 Table of the variables used in the aggregate CC dataset.....	59
2.2 Table of variables used in the per-capita CC dataset.	60
2.3 SEM standardized loading estimates, standard errors, test statistics, p-values, and variance estimates. Latent variables are shown in italics.	68
2.4 SEM classifications vs. Carnegie Classifications indicate some disagreement about middle-tier institutions.	76
3.1 Table of pseudo-F statistic results for the 100 simulations when no influential points are present for three mapping methods. By construction, the method has the same detection rate and non-detection rates for $\alpha = 0.05$ due to the structure of the permutation-based tests performed. The median pseudo-F statistics for the two largest pseudo-F statistics observed (or equivalently, for the smallest two p -values which, in this case, are always < 0.05) are large for MDS and isoMDS, and are smaller but still relatively large on a pseudo-F scale for the Sammon maps.....	104
3.2 Correct identification rates for each mapping method for an aberrant observation that varies across groups with mean 0 (i.e., is not larger/smaller than the other observations). Results from 100 simulations.....	109
3.3 Correct identification rates for each mapping method for an aberrant observation that is extremely large (outlying) on all features, which are substantially lower than those generated for a point that overlaps between the observed groups.....	111
3.4 Table of the top 15 most “influential” universities in the scaled 2015 Carnegie Classification data set.	124
3.5 Identification rates for t-SNE for an aberrant observations, including the mean-0 observation that varies across groups and one that is extremely large (outlying) on all features. Note that detection rates are substantially smaller than those of the other methods explored.	127

LIST OF TABLES – CONTINUED

Table	Page
4.1 Bivariate Gap statistic rules considered for tuning for s and K	159
4.2 Percentage of correctly-identified K -values across 3 and 6-cluster simulations for each of the Gap statistic tuning rules.	162
4.3 Table showing the detection rates for signal-bearing features across the simulations, as well as the mean spurious detection rates for noise features for each of the single variable Gap tuning rules. Results based on 50 simulations.	164
4.4 Table showing the detection rates for signal-bearing features across the 100 simulations, as well as the mean spurious detection rates for noise features for each of the bivariate Gap tuning rules.	169
4.5 Mean Rand indices with standard deviations in parentheses for each of the tuning methods used. Results based on 100 simulations.....	171
4.6 Results from comparison of clustering method comparison show best performance (in terms of Rand Index) is from the Global Max-tuned Sparse MC method. All non-sparse methods were tuned using the Tibshirani 1SE rule. Results based on 100 simulations.	174
4.7 Optimal tuning parameter values across each of the bivariate Gap rules applied to the Palmer penguin data, with varying levels of noise features added.....	179
4.8 Definitions of the variables used in the Carnegie data set.	183
4.9 Table comparing the clusters generated by the sparse MC (tuned by Tibshirani KS) and the class labels generated in the 2015 Carnegie Classification. The labels are not true class labels, but inform an interesting comparison about how the sparse MC results compare to the CC groups.	187
4.10 Table showing the Rand indices, number of nonzero weights, and number of clusters for each of the methods used on the ApoE3 data set.	191

LIST OF TABLES – CONTINUED

Table	Page
4.11 Mean run times for sparse and non-sparse MC applied to the data sets used in Simulations 3. Sparse MC was able to provide results 32% faster, on average, than the non-sparse MC clustering.	197
5.1 Correct and spurious identification rates for the LDA maps and classical MDS maps with a potentially influential observation included.....	205

LIST OF FIGURES

Figure	Page
1.1 Example of path diagram on college student belonging data from Bollen (1989). The circles represent latent (un-observed) factors whereas the squares represent observed manifest variables. Error terms are excluded for clarity.	8
1.2 Maps produced from classical MDS in Panel (a), isoMDS in Panel (b), and Sammon mapping in Panel (c) are very similar on the CSF data set, but have some scaling differences and do appear to separate groups out slightly differently. HD stands for “Healthy Donor”, OIND is “Other Inflammatory Neurological Disorder”, NIND is “Non-Inflammatory Neurological Disorder”, RR-MS is “Relapsing-Remitting Multiple Sclerosis (MS)”, PP-MS is “Primary Progressive MS”, and SP-MS is “Secondary-Progressive MS”.	16
1.3 t-SNE on CSF proteomics data from Barbour et al. (2017) with data points sized by relative cost to the t-SNE algorithm (perplexity of 30) and colored based on patient diagnosis.....	20
1.4 t-SNE maps show different shapes and groupings at different levels of perplexity for CSF proteomics data, ranging from a perplexity of one in the top-left panel down to a perplexity of 100 at the bottom-right panel. Perplexity can be roughly thought of as a number of neighbors that an object might have.....	22
1.5 t-SNE cost function based on KL divergence decreases as a function of perplexity, run on the CSF data set, with 10 runs per perplexity value shown. KL cost decreases as the perplexity values increase.....	23
1.6 Algorithms for Agglomerative and Divisive Hierarchical clustering illustrate the Divisive methods, which start with a single cluster and make recursive partitions until the data are separated into N singleton clusters, and Agglomerative methods, which fuse observations, starting from N singleton clusters and culminate in a single cluster containing all the observations.	28

LIST OF FIGURES – CONTINUED

Figure	Page
1.7 Three methods for hierarchical clustering on the scaled Palmer enguin bill and flipper lengths (Horst et al., 2020). The plot compares single linkage, complete linkage, and Ward’s method, with dendrograms shown in the top panels. Resulting clusters are shown directly below the dendrograms.	30
1.8 Plot of monothetic bipartitions made to the Palmer penguins data set. Splits are made on flipper length only and bill length, shown in the tree diagram in Panel (a) and resulting horizontal/vertical splits in Panel (b) showing the resulting clusters. The clusters closely align with the species labels in the plot based on bill lengths and flipper lengths.	33
1.9 Example of K -means clustering on Palmer penguin data with three reasonably well-separated groups.....	35
1.10 Results from model-based clustering of Palmer penguin data set, including BIC results shown in Panel (a), assigned clusters in Panel (b). Multivariate cluster uncertainties in Panel (c), with larger circles representing higher estimated probabilities of belonging to another cluster.	37
1.11 Carnegie Classification as released in 2015, specifically focusing on the doctoral-granting group of institutions. The R1 category is classified as “Very High Research”, with R2 referring to “Higher research” and R3 the “Moderate research” group.	48
1.12 Palmer penguin data set with selected comparisons of the four quantitative measurements, including bill length, flipper length (Panel a), body mass, and bill depth (Panel b), and overall bill lengths (Panel c) for penguins from three species.	50
2.1 The 2015 update of the Carnegie Classifications with R1, R2, and R3 classifications and boundaries displayed.	61
2.2 Path diagram of the Carnegie Classifications, modeling two latent factors, one for aggregate production and the other for per-capita production. All path diagrams were created using OnyxR (Brandmaier, 2016).	65

LIST OF FIGURES – CONTINUED

Figure	Page
2.3 Our SEM specification, which models STEM and non-STEM production as two separate latent traits.	66
2.4 Correlations of the observed ranked variables. Some of the observed variables are highly correlated.	67
2.5 Clockwise from top left: Panel A: Plot of BIC values associated with normal mixtures. Panel B: Plot of cluster membership. Panel C: Uncertainty plot associated with each cluster. Panel D: Density plot of clusters.	72
2.6 Screenshot of the SEM-classifications application assessing Oregon State University	74
2.7 SEM classifications after adding PhDs.....	75
3.1 MDS maps for maps with four different observations held out. The lower-right Panel (“61”) provides the map with the potentially influential point removed whereas the maps for points “1”, “2”, and “60” show the maps when those points are removed, respectively. The structure when observation 61 is removed changes somewhat, as demonstrated by slight shift in relative position of observation 45 to its group and changes to “angles” within each of the groups, but only minor changes in the shape of the maps occur when observations 1, 2, or 60 are removed (despite a reversal of the y-axis when observation 60 is held-out).	93
3.2 Plot of dissimilarity matrix for map ($N = 61$) with the third observation held out. White space refers to missing observations. Purple refers to the smallest distances with green, blue, and yellow distances being larger.....	95
3.3 Permutation distributions (Panel (a)) shown for the MDS maps shown in Figure 3.1 and the Panel (b) shows permutation distribution for t-SNE maps.....	97
3.4 Plot of simulated data with no aberrant observations, the group structure in top Panel (a) and three different maps from the simulated data set in Panels (b), (c), and (d).	103

LIST OF FIGURES – CONTINUED

Figure	Page
3.5 Simulated data with unusual point (simulated with mean 0): Panel (a) includes an observation that breaks clearly-defined group structure and its impacts on the three maps in Panels (b), (c), and (d).....	105
3.6 Panel (a) shows an observation that is much larger across all features than the other observations and its impacts on the three maps in (b), (c), and (d).	106
3.7 Detection rates by Mahalanobis distances for each of the 100 simulations, using conditional density plots. As distance increases, detection rate increases rapidly for the aberrant points with mean 0 in MDS maps, and increases somewhat less clearly for Sammon and isoMDS maps. Bandwidth set to 3.	107
3.8 Detection rates by Mahalanobis Distances for each of the simulations for the larger aberrant observation. For MDS and Sammon mapping, detection rates generally increase as Mahalanobis distances increase. For isoMDS, the initial detection rates start higher than in the initial case, but appear to decrease as Mahalanobis distance increases. Results shown for 100 simulations. Bandwidth set to 3.	110
3.9 In Panel (a), data were simulated to differ across two dimensions to assess detection rates as the number of extreme values increases across features. Panel (b) shows the detection rate trends for each map type. Results shown from 100 simulations.	112
3.10 Spaghetti plot of the scaled Palmer penguin data across the four morphological variables, colored by species type.	114
3.11 Pseudo-F test statistics for penguins without any influential points added, colored by penguin species type. Panel (a) shows the pseudo-F statistics for MDS, Panel(b) for isoMDS, and Panel (c) for Sammon mapping.	115

LIST OF FIGURES – CONTINUED

Figure	Page
3.12 Spaghetti plots of the penguin with largest pseudo-F statistic from the three map methods. Panel (a) and (c) show the same penguin (an Adelie) selected as most influential (observation highlighted) in the isoMDS and Sammon maps. Panel (b) shows the Chinstrap penguin selected as most influential in the MDS-map. In both cases, the selected penguins tend to be either large or small relative to the penguins of their same species.	116
3.13 Palmer penguins data with various influential observations included. Panel (a) shows a spaghetti plot of a penguin that is large on all four measurements, Panel (b) shows the profile of a penguin with small bill measurements and large mass and flipper length, and Panel (c) shows a penguin that looks like a Chinstrap except that it has extremely large mass relative to other Chinstrap.	118
3.14 Pseudo-F statistics for penguin data contaminated with penguins of various types. Panel (a) provides the pseudo-F statistics for a penguin that is large on all features considered. Panel (b) provides pseudo-F statistics for a penguin with small bill but large mass and flipper length, and Panel (c) provides the pseudo-F statistics for a penguin with vary large mass.....	119
3.15 Map of the doctoral-granting institutions from the 2015 Carnegie Classification. R1 institutions refer to the “Very High Research” category, R2 refers to the “High Research” group, and R3 refers to the “Moderate Research” category.	121
3.16 Diagnostic results and scaled, original data values for the three most influential institutions in the 2015 CC data set. Panel (a) shows the pseudo-F statistics highlighting the most influential schools in the map, and Panel (b) shows the original scaled values for each of the institutions relative to the the other institutions in the CC data set.	123

LIST OF FIGURES – CONTINUED

Figure	Page
3.17 Two maps showing the position of an aberrant point. Panel (a) gives the t-SNE map which puts the aberrant point (shown with the orange dot in the bottom right-hand group. Panel (b) gives the MDS map that correctly places the aberrant point away from any of the groups.	128
3.18 t-SNE and MDS plots showing the original class labels without any holdout observations, and showing the map when either the observation with the largest pseudo-F statistic is removed, or when the aberrant observation (“7”) is removed. Panel (a) shows the t-SNE maps with clearer group separation with Panel (b) showing the MDS maps, which do not separate the groups very clearly. Both maps seem to be largely unaffected by the removal of each observation, possibly because there are 200 observations so no single point exerts much influence over the entire map, or because “7” resides too close to the other digits explored.	131
3.19 Pseudo F-Statistics for MDS (shown in Panel (a)) and t-SNE maps (shown in Panel (b)), each of which do not show evidence that the long “7” included is influential.	132
4.1 Plot of MC of Palmer penguins data set. Two bipartitions are made on flipper length and bill length, shown in Panel (a) and used to generate the three clusters shown in Panel (b) based on the red lines partitioning the space. The clusters are highlighted and the monothetic characterization for each group is annotated over the resulting clusters.	143
4.2 Motivating example showing two clustering solutions, where MC results are improved with only one feature used and two cluster solutions. Panel (a) shows the actual clusters, Panel (b) shows the MC using both dimensions that split on feature “V2”, and Panel (c) shows the more intuitive clustering using only feature “V1” both in the distance calculation and as the candidate for bipartitions.	146

LIST OF FIGURES – CONTINUED

Figure	Page
4.3 Heatmap showing the distance matrices created when using Euclidean distance from the sparse clustering, or equivalently taking the Euclidean distance matrix of the weighted data, in Panel (a), and the plot the vectorized distance matrices plotted against each other in Panel (b) indicate that, for Euclidean distance, the weights can be applied to the original data and input to the clustering.	152
4.4 Monothetic bipartitions for a simulated data set with $N = 80$, $K = 6$, shown in Panel (a), $Gap_N(K)$ values across potential values of K in Panel (b), and feature weights in Panel (c) shown for a scenario that is too sparse. The first 20 features are “signal” bearing whereas the other 80 features do not vary across groups. While the overly sparse solution does remove the noise-bearing features from consideration in the MC, it also removes nearly all of the signal features and reduces the features available for splits to two (with primary weight on feature V1).	156
4.5 Optimal K values across different rules for the Gap statistic across 100 simulations. Panel (a) gives the one-cluster version, which chooses a single cluster for all simulations except one (for the First and Global Max rules). Panel (b) shows 3-cluster solution, and Panel (c) shows 6-cluster solution. Note that the $K = 1$ solution was never selected when multiple groups were simulated in Panels (b) or (c).	161
4.6 Panel (a) shows the plot of Gap statistics for 50 simulations across spectrum of s values on simulated data with three clusters. Panel (b) shows the optimal s -value across 50 simulations based on each rule.	163
4.7 Optimal feature weights evaluated for each of the Gap rules for all 50 simulated realizations applied to MC with K held constant. The first 20 features (left of the red dotted line) are the signal-bearing ones, and the last 80 are noise features.	165

LIST OF FIGURES – CONTINUED

Figure	Page
4.8 Gap statistics across s , K visualized in two ways for a single simulated data set. Panel (a) shows Gap varying across K -values and colored by s . Panel (b) shows plots varying across s (left is more sparse) and colored by K	167
4.9 Panel (a) contains the optimal number of clusters and Panel (b) contains the optimal sparsity parameter values chosen by each of the rules across 100 simulations. The $K = 1$ solution is not shown in Panel (a) as it was never selected as an optimal choice for any of the rules. Note that Tibshirani SK clearly favors a sparser solution and appears to recover the correct four cluster solution more often than other rules.....	168
4.10 Weights of features from Gap-tuned sparse MC across 100 simulations for each of the rules. The first 20 features (left of the dotted red line) are “signal” bearing and vary across groups, whereas the remaining 80 features are “noise” features. The percentages of signal-bearing features with nonzero weight vs. noise features spuriously assigned nonzero weight are given in Table 4.4.	170
4.11 Gap statistic plots for the sparse and non-sparse monothetic clusterings of the Palmer penguin data show that the one-cluster solution is not optimal, indicating the presence of multiple groups, even when noise features are present. Panels (a) and (b) show the case where no noise features are present, Panels (c) and (d) give the Gap for the 10-noise feature situation, and Panels (e) and (f) provide the 100-noise feature Gap plots.	176
4.12 Number of nonzero features retained in the clustering solution for each value of s across the data sets with no noise, 10 noise features, and 100 noise features added.	177

LIST OF FIGURES – CONTINUED

Figure	Page
4.13 Feature weights for the sparse MC on the Palmer penguin data, with each version tuned for s using the Global Max rule. The version with no noise features added includes weights only for the first four features, and 14 weights are included for the version with 10 noise features added, with just the first 14 shown for the version with 100 noise features added.	180
4.14 Monothetic bipartitions of the Palmer penguins data, both in the presence of noise features and when only the original four variables are used. Bipartitions are shown for cluster solutions using the Global Max rule.	181
4.15 Rand indices comparing each of the clustering solutions to each other and to the species label, both on the original data as well as the data with 10 noise features added. The sparse MC produced a very similar clustering ($\text{Rand} = 0.99$) in the presence of noise features as did the MC with no noise features present.	182
4.16 Bivariate Gap statistic plots for the Carnegie data using sparse MC, presented with K running along the x-axis in Panel (a) and across s values in Panel (b), indicate that a multiple-cluster solution is optimal; however, the tuning rules disagree on the optimal values of s and K . Tibshirani KS chooses the $s = 3.1$ and $K = 4$	184
4.17 Monothetic bipartitions for the 2015 Carnegie Classifications ranked data set. Panel (a) shows the splits for the sparse MC using Tibshirani KS Gap rule (choosing a four-cluster solution) and Panel (b) shows the splits for the non-sparse MC, which chooses a six-cluster solution.....	185
4.18 Parallel coordinate plots showing the values for each of the eight Carnegie features for the original 2015 Carnegie Classifications, sparse MC, and non-sparse MC. Panel (a) shows the original classifications, Panel (b) shows the sparse MC clusters, and Panel (c) shows the clusters resulting from the non-sparse MC clustering.	188

LIST OF FIGURES – CONTINUED

Figure	Page
4.19 Monothetic bipartitions and feature weights for sparse and non-sparse versions on the ApoE3 data set show that a single feature drove the main split for both methods, with additional splits being based on other signal-bearing features.	192
4.20 Feature weights for the two sparse MCs of the ApoE3 data. The PMD step used in this example appears to focus specifically on Feature 238, which leads to a less optimal clustering for each of the sparse methods (including both the sparse monothetic and sparse hierarchical methods considered).	192
4.21 Plots of Gap statistics across differing values of K and s on the Glass data show strong evidence in favor of multiple clusters, but there appears to be a divergence in Gap behavior depending on the level of sparsity. Less sparse methods prefer a nine-cluster solution whereas more sparse methods prefer fewer clusters.	193
4.22 Feature weights across the chemical concentrations in glass for each of sparse monothetic clusterings based on the tuning suggestions for Tibshirani rules ($K = 2$, $s = 1.1$), Global Max ($K = 5$, $s = 4.6$), and Global Max SE rules ($K = 4$, $s = 3.1$).	194
4.23 Plots of the ratio of $\text{CaO}/(\text{CaO} + \text{K}_2\text{O})$ plotted against Na_2O concentration for all glass vessels, colored by groups from Lemberge et al. (2000) in Panel (a) and by sparse MC cluster labels from each of the optimal tuning rules in Panels (b), (c), and (d).	195
5.1 Results from analysis of influence diagnostic on LDA maps using data from simulation studies in Chapter Three, with the first and potentially-influential 61st observations held out for LDA in Panels (a) and (b), respectively, and for classical MDS in Panels (c) and (d).	204

LIST OF FIGURES – CONTINUED

Figure	Page
5.2 t-SNE maps with perplexity of 30 for noisy (with 10 noise features added) vs. sparse CC data using PMD for sparsification show some potential improved grouping based on classes (at least visually, particularly between groups R2 and R3) and could be an interesting avenue for future research.	207

ABSTRACT

Working with high-dimensional data involves various statistical challenges. This dissertation overviews a suite of tools and methods for dimension reduction, using latent-variable models, techniques for mapping high-dimensional data, clustering, and working with multivariate responses across a variety of use cases.

First, we propose and develop a method for classifying institutions of higher education and compare with the current standard for university classification: the Carnegie Classification. We present a classification tool based on Structural Equation Models that better allows for modeling of correlated indices than the PCA-based methodology that underlies the Carnegie Classification. Additionally, we create a Shiny-based web application that allows for assessment of sensitivity to changes in the underlying characteristics of each institution.

Second, we develop a novel methodology that extends the Cook's Distance diagnostic for identifying influential points in regression to a new application on high-dimensional mapping tools. We highlight a PERMANOVA-based method for calculating the difference in the shape of resulting ordinations based on inclusion/exclusion of points, similar in style to the influence diagnostic Cook's Distance for regression. We present a set simulation studies with several mapping techniques and highlight where the method works well (Classical Multidimensional Scaling) and where the methods appear to work less effectively (t-distributed Stochastic Neighbor Embedding). Additionally, we examine several real data sets and assess the efficacy of the diagnostic on those data sets.

Finally, we introduce a new method for feature selection in a specific type of divisive clustering, called monothetic clustering. Utilizing a penalized matrix decomposition to re-weight the input data to the monothetic clustering algorithm allows for reduction in noise features allows this clustering method to better make splits based on single features at a time, leading to better cluster results. We present a method for tuning both the number of clusters, K , and the degree of sparsity, s , as well as simulation studies that highlight the efficacy of noise reduction in monothetic clustering solutions.

CHAPTER ONE

INTRODUCTION

1.1 Noise And Signal Variables

A fundamental issue in statistics refers to the identification of the right data elements in order to provide meaningful information to solve a problem. In many cases, lack of data is not a problem; rather, too much data may be available. This happens in two ways: either there are more features than necessary or the number of observations is large enough to be computationally infeasible. In the former case, only some of the those features may be useful for answering a statistical question or identifying structure in data; some “noise features” may not contribute meaningful information to the problem of interest and may obscure those features that do provide worthwhile signal. In the latter case, it may be useful to reduce the dimensionality of the data to better identify meaningful, signal-bearing structure.

Nate Silver, in his book *The Signal and the Noise*, worded the distinction more succinctly, noting that “the signal is the truth. The noise is what distracts us from the truth” (2012). Our aim is to identify signal features and distinguish them from noise-bearing features across a variety of different statistical problems pertaining to multivariate data.

Numerous methods have been developed to identify signal-bearing features across different settings; however, they are not always used and many statistical learning techniques do not take them into account. For instance, in many unsupervised problems, often an entire set of data will be used to generate clusters without respect to whether or not the individual variables specifically drive meaningful separation between observations. In high-dimensional problems, it is often the case that only a select few features may drive meaningful signal—

sometimes, this can be on the order of hundreds or thousands of noise features relative to only a handful of signal-bearing ones.

This purpose of this work is to propose novel extensions to several different methodologies related to both feature selection and dimension reduction of multivariate or high-dimensional data. This chapter overviews some of the important background information to understand the existing literature and methodology that is extended in further chapters. First, to motivate the problem posed in Chapter Two, relevant methods for dimension reduction are discussed, including principal component analysis (PCA) and several latent-variable methods. Second, to provide necessary background information for a map-based influence diagnostic proposed in Chapter Three, we review methods for mapping high-dimensional data into two or three-dimensional representations. Third, we discuss methods for clustering data and outline a framework for feature selection across a broad range of clustering methods in order to review background details for sparse monothetic clustering proposed in Chapter Four. Finally, this chapter overviews a list of common data sets that are used throughout this work.

1.2 Multivariate and High-Dimensional Data

One particular element of importance to this distribution is the distinction between data that are considered “high dimensional” and data that are multivariate in nature. Typically, multivariate data consist of a series of response variables that are simultaneously analyzed. A subset of multivariate data is called high-dimensional if the number of features P is particularly large, and can include data sets where the number of features is equal to or exceeds the number of observations N , (i.e., $P \gg N$). Throughout this dissertation, P will refer to the number of features and N the number of observations in a data set.

One potential consequence of dealing with higher and higher dimensionality is that the inclusion of various additional features (including potentially noise features) can actually

lead to a degradation of statistical methods’ performance or fit, known as the curse of dimensionality (James et al., 2013). The more features are present in the data (i.e., larger P), the more difficult it can be to estimate quantities—more data may actually degrade performance/accuracy of methods. While the curse of dimensionality and the presence of noise features are not identical problems, they are related, and solving the latter may mitigate the effects of the former. Thus, one of the key themes of this dissertation is to identify methods for reducing the presence and impact of potential noise features via multiple methods and across various settings.

1.2.1 Dimensionality Reduction

A common method for dealing with multivariate data (and especially high-dimensional data) is to reduce the number of features considered via a method that either re-parameterizes the original variables or reduces the amount of multivariate responses into a more manageable set. From this theme arises the concept of dimension reduction, whereby methods are used to reduce the dimensionality of the data while also retaining as much of the information contained therein as possible. Both Chapter Two and Chapter Three of this work pertain to methods for dimension reduction. This section provides the relevant background necessary to understand our proposed novel extensions to existing methodology.

1.2.2 Principal Component Analysis

Principal component analysis (PCA) was initially developed by Pearson (1901) and re-developed independently by Hotelling (1933), and is one of the oldest statistical learning techniques for multivariate data. Consider a set of quantitative variables $\mathbf{X}_{N \times P} = [x_1, x_2, \dots, x_P]$. Methods for dimension reduction such as PCA seek to reduce the number of x_j variables by creating a new set of variables, $y_1, y_2, \dots, y_Q$, that retain some or all of the variation in the original P variables. In PCA, this is done via an eigenvalue decomposition of either the correlation matrix or covariance matrix of the x ’s. These new y variables, called

principal components (PCs), are linear combinations of the x 's; further, they are ordered so that the first new variable, y_1 , accounts for the most variation in the original variables and y_2 accounts for the second most, orthogonal to y_1 , etc. (Everitt and Hothorn, 2011).

PCA by itself returns a set of re-parameterized features which can be expressed via linear combinations of the original variables. Since the first of the new variables contain information about much of the variation in the old variables, it is possible to use only a subset of the first few y variables without losing much information about the variation in the original x 's. In this way, often the first two or three (or Q) features are used to describe the entire set of features (of dimension P) where $Q < P$.

PCA includes three pieces of information: the eigenvalues (variances), the eigenvectors (loadings), and the scores of each observation on the new variables (PCs) created. The scores are evaluated linear combinations of the original variables for each subject. The loadings are the coefficient weights by which the original variables must be multiplied to obtain the PC scores. Each of the new y features can be described as an equation, like the equation for y_1 :

$$y_1 = \alpha_{1,1}x_1 + \alpha_{1,2}x_2 + \dots + \alpha_{1,P}x_P, \quad (1.1)$$

where $x_1, \dots, x_P$ are the original P features combined with the (unstandardized) loadings $\alpha_1 = \alpha_{1,1}, \dots, \alpha_{1,P}$ that map to the PC scores. The eigenvalues define the variability associated with each component generated.

Interpretation of PCA results is an important aspect of the method. The loadings, $\alpha_1, \dots, \alpha_Q$, can be interpreted as weights applied to the original data in the direction of highest variation; observed variables with the largest weight may be interpreted as the main factors that drive high/low scores on the components to which they are related. Biplots allow for easy visualization of the scores (typically in two dimensions) but simultaneously illustrate the relative position of the original P features (Jolliffe, 2002).

1.2.3 Factor Analysis

Bollen (1989) describes latent variable models as unobserved or unmeasured variables that are free from both random and systematic measurement errors. A related (but certainly distinct) method for identifying un-observed latent variables (or factors) in multivariate data is called factor analysis (FA, overviewed in Everitt and Hothorn (2011)). Factor analysis projects a set of multivariate (or high-dimensional) data into a smaller set of unobserved latent traits in such a way that the covariance structure of the underlying observed responses are well approximated by the model (Bollen, 1989). Factor analysis comprises both exploratory factor analysis (EFA, initially developed by Spearman (1904) and later by Gorsuch (1983)) and confirmatory factor analysis (CFA, see Brown (2006)). The key difference between confirmatory and exploratory FA is how the methods are used. Typically, with CFA models, a hypothesized model is being explicitly tested whereas with EFA, potential relationships or factors are being investigated.

Jolliffe (2002) notes that motivating PCA as a special case of factor analysis is misguided because PCA and FA are distinct techniques, for several reasons. First, FA is based on an underlying, explicitly defined model linking the original x 's to the latent factors whereas PCA does not rely on a model to estimate eigenvectors or scores. Additionally, PCA concerns itself with the diagonal elements of the covariance matrix (the variances); FA is more concerned with the off-diagonal entries (i.e., the covariances), and is typically used in settings where the data are correlated.

Consider a set of P observed, also called manifest, variables, $\mathbf{X} = [x_1, \dots, x_P]$. The FA model can include a single factor (called the one-factor model) or many ($1 < Q < P$) factors, and the baseline model can be viewed as a series of regressions linking the manifest variables to the unobserved latent factors as:

$$\mathbf{X} = \Lambda f + u, \tag{1.2}$$

and

$$\text{var}(\mathbf{x}) = \Sigma = \Lambda\Lambda^T + \Psi, \quad (1.3)$$

where Λ is the $P \times Q$ matrix of loadings (regression coefficients) that link f , the Q vector of latent factors, and u is a $P \times 1$ vector of disturbance terms (errors) that are assumed to be uncorrelated with both each other and the latent factors, f .

In practice, the latent factors are assumed to have mean 0 and standard deviation 1. The population covariance matrix Σ is of dimension $P \times P$ and is estimated by the sample covariance $\mathbf{S}$. Everitt and Hothorn (2011) point out another key difference between EFA and PCA, that the factor analysis model will return the same result when applied to either the covariance matrix or the correlation matrix due to the scale invariance of the Q -factor model.

A key issue in EFA is to determine the optimal number of factors Q . In EFA, there may be marked differences in the values of the loadings between a solution with $Q = M$ factors vs. a solution with $Q = M + 1$ factors. This differs from PCA, where the values of the loadings in the first retained PC's will be the same regardless of the number of components retained. As noted by Jolliffe, with too few factors there will be too many high loadings, and with too many factors, factors may be fragmented and difficult to interpret (2002). There are multiple ways to estimate factors, including maximum likelihood and principal factors, among others (see Everitt and Hothorn (2011) for more information).

1.2.4 CFA and Path Analysis

Factor analysis models extend the idea of path analysis, by which a diagram that describes the dependencies and relationships between manifest and latent variables can be created. Path diagrams can be thought of as a graphical representation of a FA model. Standard conventions of path diagrams are to have squares and rectangles denote manifest

variables, circles and ovals denote latent variables including error terms, single-headed arrows depict causal relationships between variables where the variable at the tail of the arrow is the cause of the variable at the head of the arrow, double-headed arrows depict a feedback loop or reciprocal relationship between two variables, and curved double-headed arrows depict unanalyzed relationships between two variables (Bollen, 1989).

The `lavaan` R package vignette (see Roseel (2012)) contains an overview of several latent variable models in detail. An example of factor analysis can be shown using a data set that pertains to student attitudes on college campuses, which is a classical data set used to overview SEM and factor analysis-related techniques.

A CFA model that is often proposed for these seven variables consists of two latent factors, each with four indicators, can be represented using a path diagram. The factors describe two types of scores generated by the test, including scores for a sense of belonging and morale. Each of the latent factors is allowed to be correlated with the others, and each factor is modeled as a function of either $x_1, \dots, x_4$ for the “morale” factor or $x_4, \dots, x_7$ for the factor for “Sense of Belonging.” Figure 1.1 shows a path diagram for the two latent factors mapped to the seven manifest variables, with the random errors omitted for clarity.

1.2.5 Structural Equation Modeling

EFA and CFA in particular represent only a small subset of a wider group of latent variable models used for highly correlated data. Many of the developments on FA-type models occurred in the social sciences (psychometrics and econometrics), with the development of path analysis stemming from Spearman (1904) and, perhaps more importantly, Wright (1954). These methods and uses are overviewed thoroughly in (Bollen, 1989). Structural equation modeling (SEM) was introduced in 1921 by geneticist Sewall Wright (1921), and was more broadly popularized in the 1970s based on work done by Karl Jöreskog (1970, 1973, and 1978). CFA is a special case of a wider set of latent variable

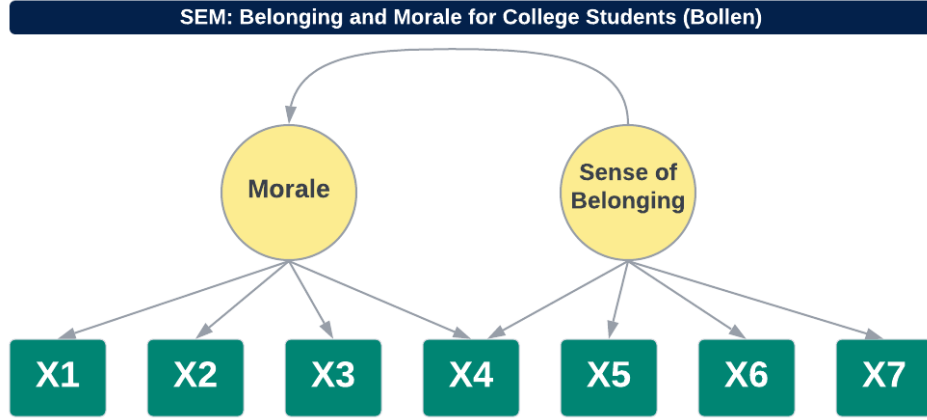


Figure 1.1: Example of path diagram on college student belonging data from Bollen (1989). The circles represent latent (unobserved) factors whereas the squares represent observed manifest variables. Error terms are excluded for clarity.

models called structural equation models (Bollen, 1989).

SEM is a statistical methodology that allows for the modeling of simultaneous equations, the use of latent or unobserved variables, and the use of variables measured with error. A typical SEM consists of two parts: the latent variable model which describes the relationships among latent variables and the measurement model which relates the latent variables to their indicators or items. The latent variable model encompasses the structural equations that summarize the relationships between latent variables, and is sometimes referred to as the “structural equation” or “causal model” (Bollen, 1989). Latent variables can be considered as either exogenous, meaning that their causes lie outside the model, or endogenous, indicating that they are determined by variables within the model. Exogenous variables are denoted with ξ and endogenous latent variables are denoted with η .

Consider an $m \times 1$ vector of latent endogenous random variables, $n \times 1$ vector of latent variables denoted ξ to represent the n latent variables. The structural equation for the latent variable model is defined as

$$\eta = \mathbf{B}\eta + \Gamma\xi + \zeta, \quad (1.4)$$

where the expected value of the latent variables $E(\zeta)$, $E(\eta)$, and $E(\xi)$ are all equal to 0 and the latent errors in the structural equations ζ are assumed to be uncorrelated with the latent exogenous variables ξ . Additionally, assumptions of constant variance (homoskedasticity) and non-autocorrelation are made for ξ .

Hildreth (2013) notes that while one can posit relationships between latent factors, they can only be tested if measures of the latent variables are collected such that these observed variables are proxies of the latent variables. The measurement model does exactly that—it includes structural equations that represent the relationship between the observed (manifest) variables and the latent traits. The terms “manifest variables”, “measures”, “indicators”, “observed variables”, and “proxies” are all used interchangeably and indicate the observed variables (Bollen, 1989). SEM can be used for identifying causal relationships in observational data; however, we do not explore that use in this dissertation. Rather, we focus on SEM as a tool for identifying and measuring the relationships of latent unobserved traits.

All SEMs in this dissertation were estimated using `lavaan` (Roseel, 2012) in the statistical software package R (R Core Team, 2022).

1.2.6 Model Fit

Unlike PCA, SEM (and EFA/CFA) allows for modeling of correlated latent factors as opposed to orthogonal ones created by PCA (or rotated PCA). SEM also includes a suite of diagnostic tools that do not exist to assess the efficacy of PCA-based methods as well as methods for estimation for SEMs when missing data are present.

Fit indices can be divided into two groups: absolute fit indices, which compare the estimated model to the saturated model and incremental fit indices, which compare the

estimated model to a baseline null model that typically assumes zero covariances among the observed manifest variables. At a high level, incremental fit metrics generally represent relative improvement compared to a “strawman” model whereas absolute fit indices more directly compare the estimated sample covariance matrix, S , to the implied covariance matrix, $\Sigma(\hat{\theta})$.

Common absolute fit indices of interest to this dissertation include the goodness of fit index (GFI, Joreskog and Sorbom (1982)) and root mean square error of approximation (RMSEA, see Steiger (1990)), among others. For GFI, typical values range from 0 to 1, and higher values indicate better absolute fit. By contrast, smaller values (i.e., below 0.05) of RMSEA indicate that a greater proportion of sample covariances are explained by the model. The Tucker Lewis index (Tucker, 1973), normed fit index (NFI, see Bentler (1980)), and comparative fit index (CFI - see Bentler (1990)) are all incremental fit indices. For each of these metrics, the values range from 0 to one with values closer to one indicating a better fit than the baseline. Additional diagnostics for identifying influential or outlying observations (as well as other assessments of residuals) in SEM were developed by Hildreth (2013).

In some cases it is necessary to examine the values of the latent factor for each observation, not unlike one might examine the predictions made from a regression model. Because latent factors are unobserved by definition, these values must be estimated. This is done by creating factor scores which can then be used in subsequent analyses (DiStefano et al., 2009). Factor scores are computed using a weighted average of the items, with a number of options available for weighting. The most common method used to estimate factor scores is Bartlett’s method (Bartlett, 1937) as it leads to unbiased estimators of the true factor scores. In subsequent analyses, we use the factor scores created using Bartlett’s method.

1.3 Mapping High-Dimensional Data

In addition to dimension reduction, it can be useful to create maps of high-dimensional data projected into a lower-dimensional space (typically two or three dimensions), which is also called ordination. In any such situation, collapsing a large number of dimensions into a two-dimensional projection will lead to some information loss, but some methods can preserve much of the important high-dimensional structure and retain that information in the resulting map in a way that is easier for an end user to identify and assess. By visually assessing the resulting (2-dimensional) ordination resulting from MDS, an observer should be able to identify structure in the observations that would not be obvious when viewing the data in their original multivariate form. Often, maps that are considered “good” will illustrate structure that is present in the high dimensional data, like common groups or large differences between data points that are different in some meaningful way. Often, the rotation, or directions of variation, can be arbitrary in maps; however, in general, points that are near each other can be construed as being similar, whereas points that are far apart should be construed as different, generally.

Many of the most common methods for doing ordination or mapping are distance based, meaning that they take into account some notion of dissimilarity or distance between objects to generate lower-dimensional projections of high-dimensional data. Moreover, they are dependent on the choice of distance used. In this work, we use Euclidean distance (so results are primarily applicable for quantitative data).

Distance-based mapping methods create maps without the notion of a model or likelihood for the observations. Model-based ordination/mapping methods, on the other hand, consider both a model/likelihood and are a potential competitor for the distance-based methods reviewed here (Stratton, 2022). Model-based ordination seeks to replace algorithm-based techniques discussed below with Bayesian mixture models or latent variable

models that allow for estimation of an underlying model to drive the resulting ordination (Hui et al., 2015). These methods can be extended to create a single-step dimension-reduction and clustering solution (Stratton, 2022). This dissertation is concerned primarily with implementation of distance-based methods for clustering and ordination; however, many of the themes explored in this dissertation are relevant in a model-based setting as well, and extensions to such a setting would be an interesting avenue for future research.

1.3.1 PCA for Dimension Reduction

As described in Section 1.2.2, PCA provides a baseline tool for creating maps of data embedded in two or more dimensions using eigen decompositions of $Var(\mathbf{X})$. To create a lower-dimensional mapping, one can take the first two PCs and plot them against each other; however, using PCA is not as flexible as those mentioned below, particularly when the data structures are non-linear or contain clusters or groups, or if non-Euclidean distances are of interest.

1.3.2 Multidimensional Scaling

A seminal tool for creating ordinations from high-dimensional data, classical multidimensional scaling (MDS, see (Borg and Groenen, 1997)) seeks to analyze similarity or dissimilarity between observations as distances in space. Borg and Groenen (1997) note that MDS has four potential uses: 1) exploratory analysis of data, 2) testing structural hypotheses, 3) as a model in psychological research, and 4) as a model for similarity judgements. Interestingly, the latter three uses are not dissimilar from the previously discussed uses for latent variable models like factor analysis and structural equation models; however, for the purposes of this dissertation, we focus on the former exploratory purpose of multidimensional scaling and related methods.

1.3.3 Classical MDS

Classical MDS was initially developed by Torgerson (1952) and is closely related to PCA and in some cases can lead to equivalent maps (Gower, 1966). Classical MDS takes as input a distance or dissimilarity matrix representing the distance between pairs of observations and returns locations for points in some selected lower dimensional space, or map. This distance does not have to be Euclidean (although it often is). Classical MDS returns maps of points that are interpreted in a Euclidean space; in general, the more dissimilar two observations are in the high-dimensional space, the farther apart they should be in the resulting lower-dimensional ordination.

Consider a multivariate data set $\mathbf{X}_{N \times P}$ where N is the number of observations and P is the number of features. For distinct pairs of points, x_i , and $x_{i'}$, define $d_{i,i'}$ as the interpoint distance or dissimilarity between the x_i and $x_{i'}$. Multidimensional scaling attempts to minimize a metric called “Stress” or a discrepancy function which is defined as

$$S_M(x_1, x_2, \dots, x_N) = \sum_{i \neq i'} (d_{i,i'} - \|z_i - z_{i'}\|)^2, \quad (1.5)$$

where $z_i, z_{i'}$ refer to the i^{th} and i'^{th} datapoints in the lower-dimensional representation and $\|z_i - z_{i'}\|$ are approximations of the original dissimilarities in the map space using Euclidean distance. The singular value decomposition (SVD) of the matrix related to $\mathbf{D}$ is used to solve this problem, further reinforcing ties to PCA, which also can use the SVD to estimate PCs (among other methods). Specifically, classical MDS can be expressed equivalently as principal component analysis when the original distances are Euclidean and variables are scaled equivalently (Everitt and Hothorn, 2008). However, classical MDS is more generally applicable because it can be applied to any distance/dissimilarity matrix.

1.3.4 Sammon Mapping and Non-Metric MDS

Various alternatives or improvements to MDS have been developed since classical MDS was proposed and developed in the 1950s. Several such variations are non-metric, meaning that rather than representing the actual distances (or parametric functions of those distances), they retain information about the ordinal ranks of the dissimilarities (Borg and Groenen, 1997). Such non-metric MDS methods rely on the calculation of a different objective function, also called a Stress metric, and relax assumptions to varying degrees about the relationships of the true high-dimensional distances and their lower-order approximations (Cox and Cox, 2008). Kruskal's non-metric MDS, also known as isoMDS, utilizes the following stress function,

$$Stress = \sqrt{\frac{\sum d_{i,i'} - ||z_i - z_{i'}||}{\sum d_{i,i'}^2}}, \quad (1.6)$$

where $d_{i,i'}$ defines the interpoint distances and $||z_i - z_{i'}||$ is the position of the points in the map that reduce Stress under the constraint that the rank orders of the points remain the same both in the ordination and the high-dimensional space (Kruskal, 1964).

Sammon mapping is a non-linear method that extends classical MDS to scale the stress metric by the pairwise distances $d_{i,i'}$, attempting to better maintain local structure by equalizing the impact of large and small distances (Sammon, 1969). This method further improves on classical MDS in various ways. The objective function for Sammon mapping is defined for the function $\phi(Z)$ which is referred to as either Sammon's Stress or Sammon's Error and takes the form

$$\phi(Z) = \frac{1}{\sum_{i,i'} d_{i,i'}} \sum_{i,i'} \frac{(d_{i,i'}^2 - ||z_i - z_{i'}||^2)}{d_{i,i'}}, \quad (1.7)$$

where z_i and $z_{i'}$ are the i^{th} and i'^{th} points in the lower-dimensional representation and $d_{i,i'}$ represents the dissimilarity between points x_i and $x_{i'}$ in the high-dimensional space.

In most implementations of classical MDS, Sammon mapping, and isoMDS, an iterative algorithm (like gradient descent) is used to optimize the STRESS metric for each non-metric mapping technique. Because both isoMDS and Sammon mapping are initialized with classical MDS, the maps may not look substantively different from maps produced by classical MDS. For instance, in Figure 1.2, the maps produced by each of the three methods on Cerebro-spinal fluid (CSF) proteomics data are nearly identical. Data on $N = 225$ and $P = 22$ from Barbour et al. (2017) contain ratios of (originally 1128) proteins identified in development of a classifier for disease status. Values are log-transformed and standardized and are colored by disease status. In all panels there is some overlap for the groups in two dimensions, but the Sammon map shows the least overlap between the groups in 2-D.

1.3.5 t-Distributed Stochastic Neighbor Embedding

A more recent method for doing high-dimensional data visualization is t-Distributed Stochastic Neighbor Embedding (t-SNE, Van Der Maaten and Hinton (2008)). t-SNE has become one of the more popular methods for visualizing high dimensional data in recent years; as of January 2022, the original paper on t-SNE has been cited more than 21,000 times on Google Scholar, in use cases ranging from biology (Amir et al., 2013) to economics (Greengard et al., 2020) to natural language processing and machine learning (Liu et al., 2018). t-SNE is an improvement on a method known as Stochastic Neighbor Embedding (SNE) proposed by Hinton and Roweis which was intended to better preserve neighborhood information in the low-dimensional representation in such a way that each high-dimensional object could be associated with more than one location in the low-dimensional space (2002). Unlike MDS and related methods, SNE (and t-SNE) is based on a probabilistic formulation: Each object x_i in the high-dimensional space has some probability of picking object $x_{i'}$ as a neighbor, and typically, objects that are closer in the high-dimensional space have higher probabilities. Similarly, points z_i and $z_{i'}$ in the lower-dimensional representation can be

Mapping Techniques on NIH Somologic Data

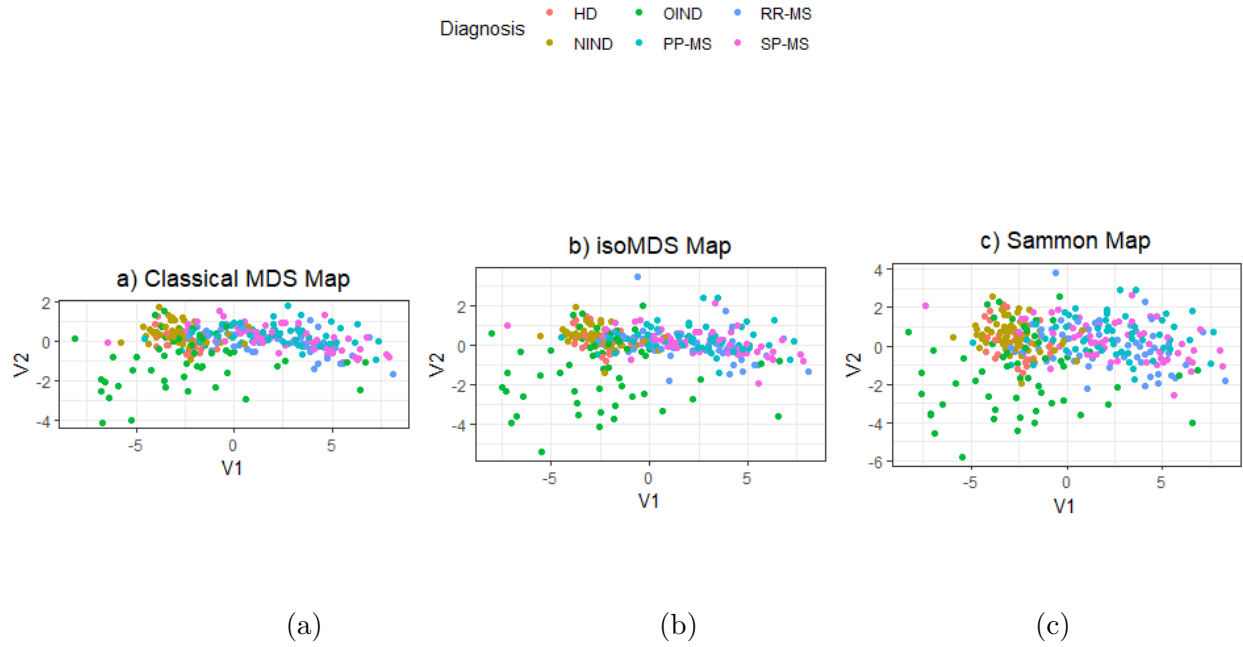


Figure 1.2: Maps produced from classical MDS in Panel (a), isoMDS in Panel (b), and Sammon mapping in Panel (c) are very similar on the CSF data set, but have some scaling differences and do appear to separate groups out slightly differently. HD stands for “Healthy Donor”, OIND is “Other Inflammatory Neurological Disorder”, NIND is “Non-Inflammatory Neurological Disorder”, RR-MS is “Relapsing-Remitting Multiple Sclerosis (MS)”, PP-MS is “Primary Progressive MS”, and SP-MS is “Secondary-Progressive MS”.

modeled with probabilities of choosing each other as a neighbor.

The first, called $p_{i,i'}$, is a conditional probability based on the Euclidean distance between pairs of points x_i and $x_{i'}$ in the high-dimensional space. In the lower-dimensional representation (the map), this similarity is defined as $q_{i,i'}$. For object i in the high-dimensional space P , this probability is symmetricized, so the probability $p_{i,i'}$ that observation i would pick i' as its neighbor is given as:

$$p_{i,i'} = \frac{\exp(-d_{i,i'}^2)}{\sum_{i \neq k} \exp(-d_{ik}^2)}. \quad (1.8)$$

The dissimilarities, $d_{i,i'}^2$, may be based on symmetric or asymmetric measures of distance, and are not necessarily restricted to be Euclidean (although that is mostly what this work will explore). However, if the distance is Euclidean, then the $d_{i,i'}^2$ takes the form

$$d_{i,i'}^2 = \frac{\|x_i - x_{i'}\|^2}{2\sigma_i^2},$$

(1.9)

with σ_i being the variance of the Normal distribution centered on high-dimensional data point x_i . In the low-dimensional representation, q , the induced probability that point z_i picks point $z_{i'}$ as its neighbor is:

$$q_{i,i'} = \frac{\exp(-\|z_i - z_{i'}\|^2)}{\sum_{k \neq i} \exp(-\|z_i - z_k\|^2)}. \quad (1.10)$$

The probability distribution used both by $p_{i,i'}$ and $q_{i,i'}$ is symmetric in t-SNE but asymmetric in SNE, and in the latter case we adopt a notation of $p_{i|i'}$ to denote the asymmetric probability of the i^{th} object choosing the i'^{th} object as a neighbor in asymmetric SNE. In SNE, this

joint probability is represented by a Gaussian distribution (Hinton and Roweis, 2002). Alternatively, in t-SNE, the distribution is represented by a heavier-tailed t-distribution with one degree of freedom (a Cauchy distribution) (Van Der Maaten and Hinton, 2008).

The sole tuning parameter for SNE (and t-SNE) is called perplexity. Perplexity can be construed as the effective number of local neighbors for each observation. SNE performs a binary search for the value of σ_i that produces a P_i with a fixed perplexity that is specified by the user as a tuning parameter. Perplexity is more formally defined as

$$Perplexity(P_i) = 2^{H(P_i)}, \quad (1.11)$$

where $H(P_i)$ is the Shannon Entropy of P_i , defined as

$$H(P_i) = -\sum p_{i,i'} \log_2(p_{i,i'}). \quad (1.12)$$

In general, the goal of the cost function is to make the representation look like the original high-dimensional data, so a good map should result when $p_{i,i'}$ and $q_{i,i'}$ look reasonably similar to each other. Since these are both joint probability distributions, the objective function of t-SNE is a symmetricized Kullbeck-Leibler divergence between the two, defined as

$$Cost = \sum_i \sum_{i'} p_{i,i'} \log\left(\frac{p_{i,i'}}{q_{i,i'}}\right), \quad (1.13)$$

with the $p_{i,i'}$ and $q_{i,i'}$ referring to the high and low-dimensional spaces, respectively. t-SNE's use of the symmetricized Kullbeck-Leibler divergence is an improvement over the original implementation of SNE, which uses the non-symmetric KL divergence in its cost function. Some recent work has focused on alternatives to the KL divergence and found that alternative divergence measures, such as Jensen-Shannon, or Chi-Squared divergence

methods, can recover certain types of structure more appropriately (Im et al., 2018). While the work presented here does not explore the impact of choice of divergence used in t-SNE, the impact of alternative divergence measures on resulting t-SNE maps across a variety of data structures and potential types of influential points could be interesting.

t-SNE’s (and SNE’s) cost function focuses on retaining the local structure of the data in the map, leveraging normal distributions in both the lower and high-dimensional similarity metrics. t-SNE instead uses the t -distribution with one degree of freedom (also known as a Cauchy distribution) to model the lower-dimensional probabilities to alleviate a problem called the “crowding problem” (Van Der Maaten and Hinton, 2008) where the 2-dimensional map cannot accommodate many close neighbors in high dimensions and preserve them in the resulting map (Van Der Maaten and Hinton, 2012). By utilizing a t -distribution for the $q_{i,i'}$, t-SNE better preserves the local structure of the data. Most of the minimization of objective functions is done via gradient descent (and some methods require simulated annealing because the algorithm can get caught in local optima). Critically, because the optimization is non-convex, and as a consequence of the gradient descent methods used to optimize it, it is common for t-SNE to generate different maps when run on the same data at the same perplexity for different random initializations. Moreover, the t-SNE maps do not appear to place points with higher cost either near the center nor the periphery of the plot consistently. Figure 1.3 shows a t-SNE map created from data on CSF measurements taken on a group of patients from Barbour et al. (2017). There appears to be no discernible pattern between relative cost and position in the ordination even though the t-SNE map appears to do a reasonable job of sorting patients into a gradient that differs to some degree based their diagnoses.

While t-SNE is a more complex method than MDS (or related techniques), it does share a connection to PCA. In the R implementation of t-SNE, the **Rtsne** package of Krijthe (2015) is subject to a pre-processing step involving PCA called “whitening” the data. To

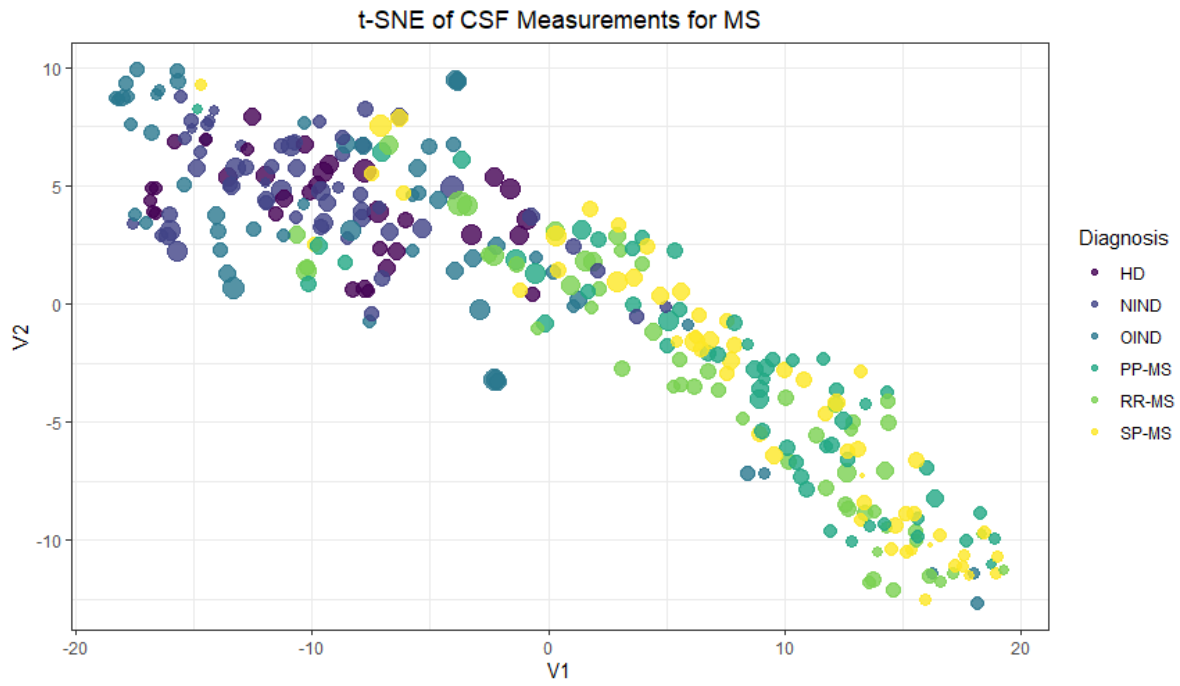


Figure 1.3: t-SNE on CSF proteomics data from Barbour et al. (2017) with data points sized by relative cost to the t-SNE algorithm (perplexity of 30) and colored based on patient diagnosis.

reduce the dimensionality prior to actually performing t-SNE PCA is performed on the initial data set (or potentially an MDS on the input distance matrix if used, although most implementations appear to do this only when the original data are available). For instance, in Van der Maaten’s initial paper, several examples reduced the initial dimensionality of the data to 30 for computational gains (2008).

Depending on the size of the initial dimensionality of the data, this step may drastically reduce the number of dimensions that t-SNE needs to deal with. However, it has some potential to impact the resulting ordination, particularly if the number of initial dimensions kept is relatively low or PCA drops important variability in the original data set. At best, the t-SNE mapping is only as good as the initial PCA is. Campbell et al. (2019) explored methods other than PCA as an intermediate dimensionality reduction step going into the t-SNE procedure. They compared PCA, Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA), Sammon mapping, and Locally Linear Embedding (LLA) as preliminary steps in t-SNE and found that resulting maps tended to form different groupings with variation in separation based on the initial step. While we did not consider assessing the impact of different intermediate dimensionality reduction techniques in this analysis, it does present an interesting route for additional research.

1.3.6 Understanding Perplexity

Because of the complex nature of how it is derived, perplexity can be a bit difficult to understand; however, because it is user defined, it is easy to set and illustrate differences across runs. Because perplexity can be interpreted as a smooth measure of the effective number of neighbors (Van Der Maaten and Hinton, 2008), it makes sense that at pathological endpoints, a perplexity that is too small will lead to a map that has too many tightly clustered groups of disconnected observations, and a perplexity value that is too high will lead to a map that does not identify distinct differences between real groups in the data.

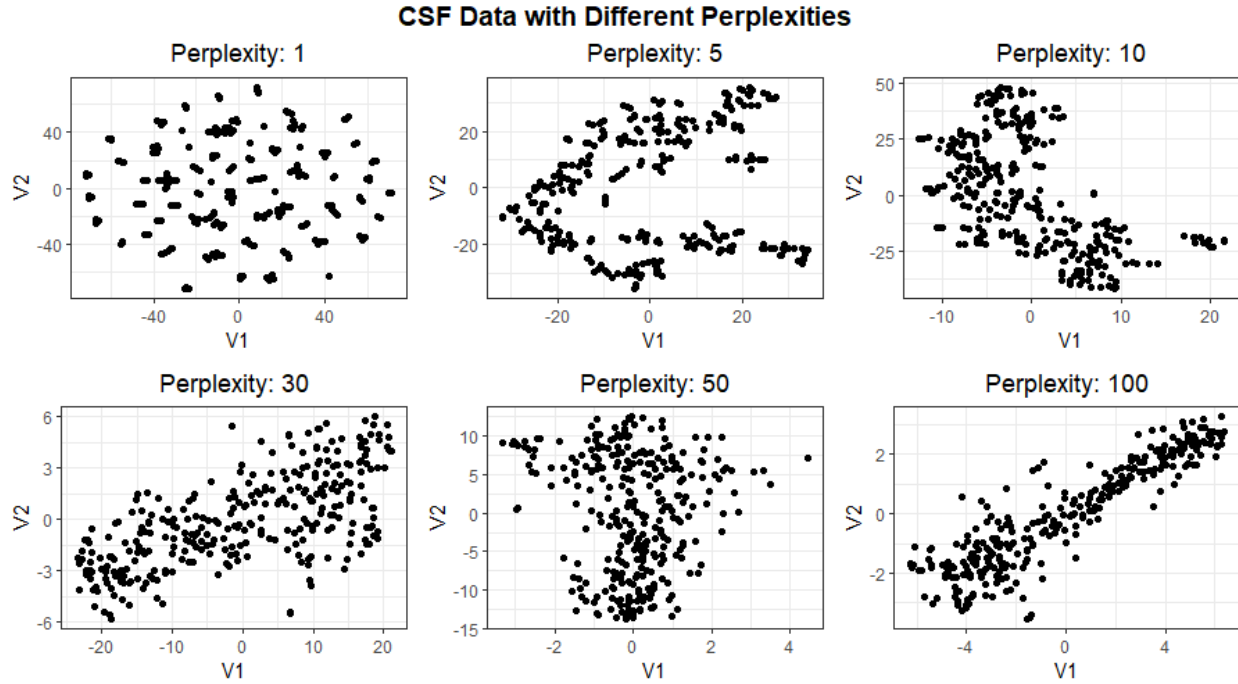


Figure 1.4: t-SNE maps show different shapes and groupings at different levels of perplexity for CSF proteomics data, ranging from a perplexity of one in the top-left panel down to a perplexity of 100 at the bottom-right panel. Perplexity can be roughly thought of as a number of neighbors that an object might have.

Van der Maaten and Hinton state that “the performance of SNE [and in turn t-SNE] is fairly robust to changes in perplexity” (2008); however, Anacker (2019) shows that this is not always the case. An example of drastic changes in the resulting ordination can be seen in the example in Figure 1.4.

Additionally, demonstrating the dependency of the resulting map the random seed, the maps can actually differ quite clearly across runs based on the same perplexity choice. In search of finding an optimal perplexity value, it is not simply sufficient to compare cost function values as each run will optimize at a different cost function, and smaller cost functions do not clearly relate to “better” maps in our explorations. Figure 1.5 shows the KL divergence as a function of perplexity, both across perplexity values and repeated 10 times per perplexity value. The KL cost increases initially, and then decreases as perplexity

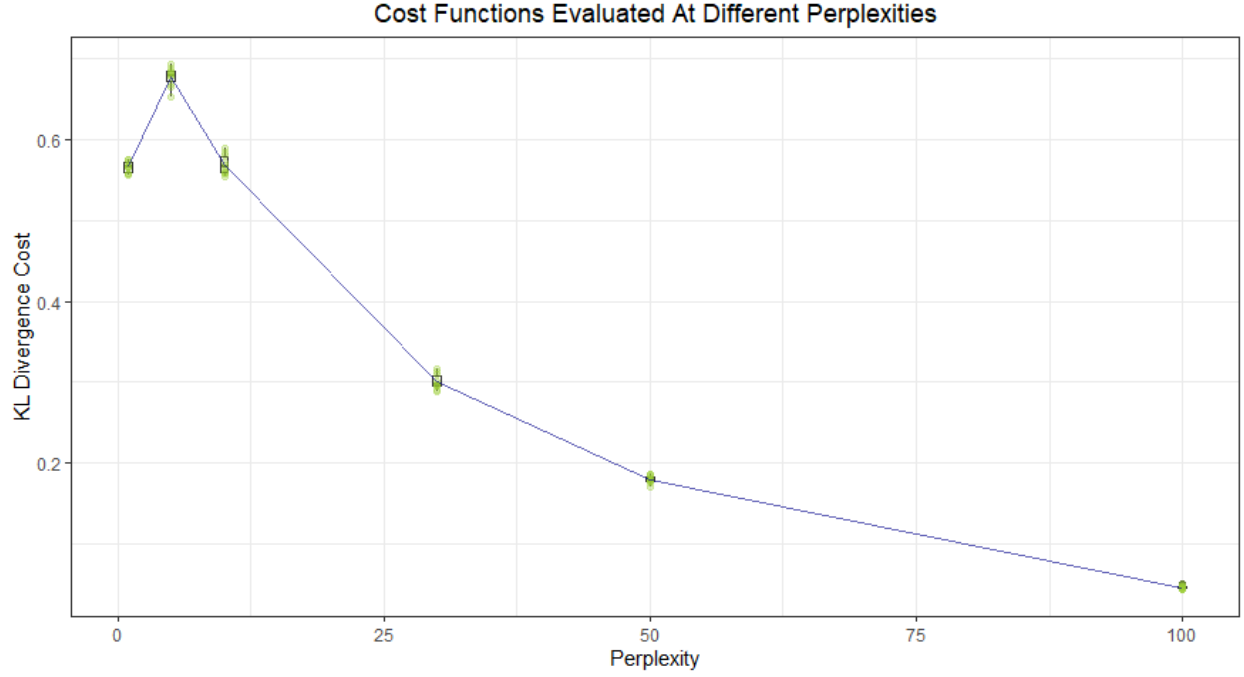


Figure 1.5: t-SNE cost function based on KL divergence decreases as a function of perplexity, run on the CSF data set, with 10 runs per perplexity value shown. KL cost decreases as the perplexity values increase.

increases, in part because the maps at higher perplexities tend to group points closer together, reducing the value of the cost function (see Anacker (2019) and Oskolkov (2019) for more detail).

Van der Maaten and Hinton provide guidelines that suggest the optimal perplexity parameter will generally be between 5 and 50 depending on the size of the data (Van Der Maaten and Hinton, 2008). Anacker (2019) notes that, after exploration of different perplexity choices, identifying an optimal value could be as simple and unsatisfying as a game of guess and check.

1.3.7 Minimum Spanning Trees

We propose adding information to maps that relates the map to nearest neighbors of observations in the original data when building t-SNE maps using a minimum spanning tree

(MST, see Everitt and Hothorn (2008)). A MST is the collection of edges between points such that the cumulative length of all the edges is minimized, and we create MSTs using the R package `ape` (Paradis and Schliep, 2019). The MST provides a “ground truth” of sorts that allows a user to check how closely the grouping of points in the lower dimensional space matches the grouping in the high-dimensional space. In some cases, t-SNE will place observations that appear to be similar in opposite positions in a map. Different points may exhibit close or far mappings that do not match the original distances across different map runs, even when the perplexity is held constant.

In theory, the MST could also inform how accurately the map is reflecting true distances. A measure of average pairwise line segment length in the map for the MST could be used to quantify how close true neighboring points are placed to each other. Analysis of mean, median, and total MST line segments found that these metrics tend to follow the same pattern as the cost function—they decrease as perplexity is increased because at larger perplexities, line segments get shorter as points get pushed closer and closer together. Thus, the MST does not help a user choose the optimal map but can help provide information on distortions in the lower dimensional space compared to the higher-dimensional starting point. The MST overlay procedure is useful for all maps created, not just those created with t-SNE, as any mapping technique has the potential to display misleading configurations of points.

1.3.8 Best Practices For t-SNE

Because of the nature of how t-SNE works, it can produce visually and sometimes substantively different maps across different runs. It is therefore important to use the maps produced by the method with great care. As such, we suggest a number of “best practices” when building t-SNE maps to both validate and check the consistency of the ordinations produced by t-SNE at a given perplexity. Those include the following:

1. Utilize Minimum Spanning Trees to display nearest neighbors from the original data,
2. Set a random seed to ensure replicability of a given result,
3. Run maps multiple times to assess consistency of resulting geometries at a given perplexity.

1.4 Introduction to Clustering

One reason to perform dimensionality reduction of multivariate or high-dimensional data is to cluster data into groups of similar observations. Clustering, which Kaufman and Rousseeuw define as “the art of finding groups in data” (1990), is a commonly used unsupervised technique, meaning that groups are not known *a priori* nor is there a “ground truth” with which to compare clusters. In more technical terms, clustering is done so that the observations within clusters are similar to each other, meaning that the dissimilarity between objects in the same cluster should be minimized compared to the between cluster dissimilarity. That is, objects in the same cluster are similar to each other; objects in different clusters are meaningfully different from each other. This section presents an overview of commonly-used methods for clustering that set the stage for our novel implementation of clustering methods discussed in Chapter Two and a novel methodology for clustering outlined in Chapter Four.

1.4.1 Clustering Methods

There are a variety of methods for clustering data (see Everitt and Hothorn (2011) as an example). Several such methods are of interest to the work presented in this dissertation. K -means clustering (and related techniques) are commonly used methods for clustering that can take as input various numeric data features. Hierarchical clustering methods take a distance or dissimilarity matrix as input. Probabilistic methods like model-based clustering methods

assume that the clusters are realizations of an underlying model that can be estimated and thus try to estimate the underlying latent model that generates the clusters. This section overviews each of the relevant methods for clustering in detail.

Irrespective of the method, the fundamental clustering problem can be posed as follows. Consider a set of observations $\mathbf{X}$ such that $x_{i,j}$ is the i^{th} observation of variable j , $j = 1, \dots, P$, where P is the number of features in the data and $i = 1, \dots, N$ with N being the number of observations in the data. The goal of a cluster analysis is to partition the data into $C_1, C_2, \dots, C_K$ where K is the total number of clusters and the k^{th} cluster is denoted by C_k . Each clustering technique solves the partitioning in a different way but will result in a choice of K as well as class/cluster labels $C_1, \dots, C_K$ for each of the observations in the data. In most cases, cluster solutions will produce a hard partitioning so an object is assigned membership in cluster C_k ; however, soft clustering (like some of the model-based methods discussed in Section 1.4.5) produces instead a probabilistic degree of membership γ_{ik} to each of the clusters (k) for every i . Observations are assigned to the cluster to which they are most likely to belong.

1.4.2 Hierarchical Clustering

Hierarchical clustering methods take as input either a distance matrix or a dissimilarity matrix. Distances are non-negative measures of heterogeneity between objects i and i' that must satisfy four requirements: 1) they must be non-negative (i.e., $d_{i,i'} \geq 0$), 2) the distance between an object and itself must be equal to 0 (i.e., $d_{i,i} = 0$), 3) they must be symmetric (i.e., $d_{i,i'} = d_{i',i}$), and 4) they must follow the triangle inequality (i.e., for objects i , i' , and h , $d_{i,i'} \leq d_{i,h} + d_{h,i'}$). Dissimilarities, on the other hand, are non-negative numbers $d_{i,i'}$ such that when objects i and i' are close together, $d_{i,i'}$ is small and inversely large when those objects are far apart (see (Kaufman and Rousseeuw, 1990)). Dissimilarities need not satisfy all the mathematical requirements of being a distance metric. Distance or dissimilarity matrices

are $N \times N$ matrices of differences of $d_{i,i'}$, denoted here as $\mathbf{D}$.

Euclidean Distance (ℓ_2 norm) is the main focus in this work, and is defined in Equation 1.14 and calculated as the square root of the sum of squared differences across features, as described by

$$d_{Euc}(i, i') = \sqrt{\sum_j (x_{i,j} - x_{i',j})^2}. \quad (1.14)$$

A hierarchical clustering solution takes this dissimilarity/distance matrix D and recursively produces partitions of the data to generate a hierarchy of splits that run from a single “cluster” containing all of the objects in the data to a set of N singleton clusters or vice-versa (Everitt and Hothorn, 2011). Two types of hierarchical clustering are possible: Agglomerative and Divisive. A graphic description summarizing the general concepts of agglomerative and divisive methods can be found in Figure 1.6. Agglomerative clustering methods start with the N singleton cluster case and successively fuses the N individuals into groups. The `hclust` function in R performs agglomerative clustering with the choice of various different linkages (discussed later). Another option is the agglomerative nesting algorithm (AGNES, see Kaufman and Rousseeuw (1990)). Divisive methods do the exact opposite—they start with a single cluster containing all the data and successively partition that cluster into multiple smaller groups. One of the main implementations of a recursive, divisive clustering algorithm is divisive analysis clustering (DIANA), also overviewed by Kaufman and Rousseeuw (1990).

In either case, the fusions/partitions are irreversible; in an agglomerative clustering case, once two objects are placed in the same cluster C_k they cannot be split apart in a subsequent step (and vice-versa for divisive clustering).

Hierarchical clustering depends on a choice of *linkage* which is critical for using the chosen distance/dissimilarity metric for defining the distance between clusters. Several

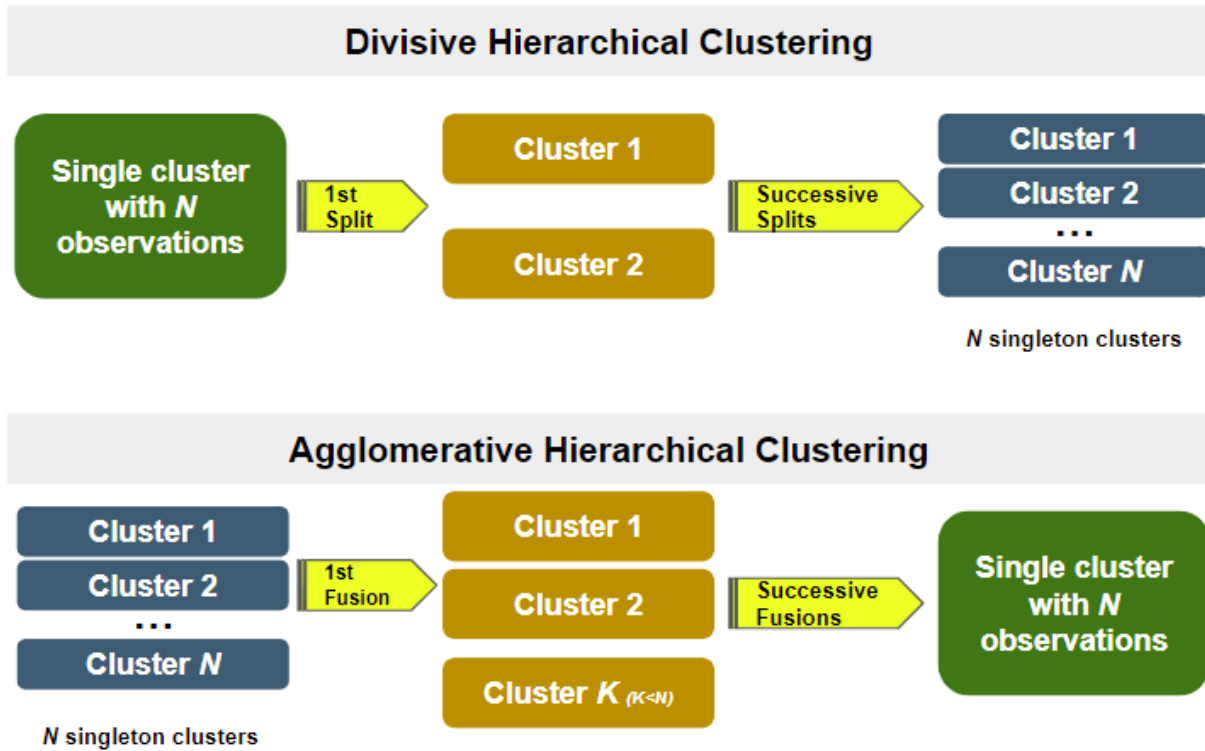


Figure 1.6: Algorithms for Agglomerative and Divisive Hierarchical clustering illustrate the Divisive methods, which start with a single cluster and make recursive partitions until the data are separated into N singleton clusters, and Agglomerative methods, which fuse observations, starting from N singleton clusters and culminate in a single cluster containing all the observations.

choices of linkages are possible, with some of the most common being complete linkage, single linkage, average linkage, and Ward’s method, which are discussed in Everitt and Hothorn (2011). Single and complete linkages use relatively simple measures of inter-group distance between groups A and B , D_{AB} , as either the minimum (for single linkage) or maximum (for complete) of distances between individual observations in two clusters.

Ward’s (1963) criterion, called inertia and denoted $I(C_k)$, for combining two clusters is based on minimizing the change in total within-cluster sums of squared distances in order to create homogeneous groups (see Härdle and Simar (2019) and Murtagh and Legendre (2014)). It poses the linkage problem in an analogous way as the within-cluster sum of squares in K -means clustering. Ward’s criterion is calculated as

$$I(C_k) = \sum_{i \in C_k} d_{Euc}^2(x_i, \bar{x}_{C_k}), \quad (1.15)$$

where $\bar{x}_{C_k}$ refers to the “center of gravity” or multivariate mean of cluster K and $d_{Euc}^2(x_i, \bar{x}_{C_k})$ is the squared total difference between each observation and the cluster mean.

Figure 1.7 provides a comparison of three linkage methods used in hierarchical clustering, showing both the dendrograms produced and the resulting clusters for three methods based on the Palmer penguin data (Horst et al., 2020). The data used in this example are scaled penguin bill and flipper lengths, based on Euclidean distances. Three clusters were chosen to align with the number of penguin species in the data (Adelie, Chinstrap, and Gentoo). The species are generally separated when split by bill length and flipper length, but each of the linkages result in different clusters; the single linkage clustering solution puts nearly all the penguins into a two clusters whereas the complete linkage and Ward’s method tend to create clusters that align more closely with the species labels (despite some overlap).

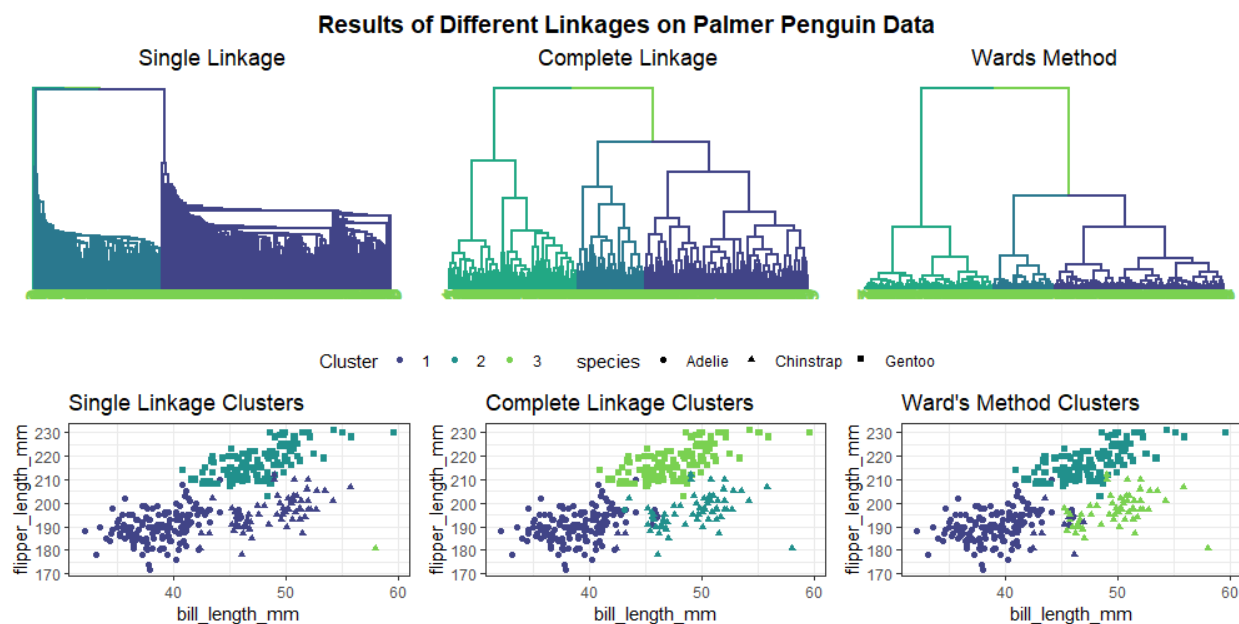


Figure 1.7: Three methods for hierarchical clustering on the scaled Palmer enguin bill and flipper lengths (Horst et al., 2020). The plot compares single linkage, complete linkage, and Ward's method, with dendrograms shown in the top panels. Resulting clusters are shown directly below the dendrograms.

1.4.3 Monothetic Clustering

Monothetic clustering (Sneath and Sokal, 1973) differs from the previously discussed “polythetic” hierarchical clustering methods in that it makes decisions based on a single property at a time; resulting clusters thus share a set of characteristics rather than simply being “similar” on average based on a dissimilarity metric that may take several features into account, leaving some subjects in the same cluster that share no characteristics in common. This “shared characteristic” property of monothetic clustering means that the resulting clusters are easier to interpret, generally, than those created by other hierarchical clustering methods. Methods for doing monothetic clustering are overviewed in Tran (2019) and are implemented in the R packages `monoClust` (Tran et al., 2022) and `Divclust` (Chavent et al., 2007).

Monothetic clustering methods first arose in the case of multivariate binary data (Williams and Lambert, 1959) and have since been known as descendant conceptual clustering methods (Michalski et al., 1981). A method for monothetic clustering, inspired partly by the classification and regression trees of Brieman et al. (1984), was developed by Chavent (1998). The algorithm performs both a hierarchy of a set of objects (a recursive clustering) as well as a monothetic characterization of each cluster (i.e., a common, interpretable characteristic (or set of characteristics) for each group).

A global criterion, called *inertia*, which is defined as the total variability around the cluster centroid, can be used as a target for a monothetic clustering algorithm that iteratively searches for optimal splits from each multivariate response (feature).

Let N be the number of objects in a set Ω . Each object is described on P features, $x_1, \dots, x_P$, which are weighted equally (although note that this need not be the case, indeed the interest of Chapter Four of this dissertation is to reweight the features in a data-driven fashion). The inertia, denoted $I(C_k)$, is a measure of homogeneity within a cluster defined

as

$$I(C_k) = \frac{1}{n_k} \sum_{i, i' \in C_k, i > i'} d^2(x_i, x_{i'}), \quad (1.16)$$

where n_k is the cardinality (i.e., the number of unique observations) in cluster C_k . Note that minimizing the within-cluster inertia of a partition (i.e., the similarity of objects within each cluster) is equivalent to maximizing the between-cluster inertia (the dis-similarity of objects in separate clusters)—this theme of monothetic clustering means that the method is similar in its goal to both K -means/medoids clustering as well as hierarchical clustering with Ward’s method as the selected linkage. When the chosen dissimilarity metric is Euclidean distance, the inertia for cluster C_k is equivalent to $I(C_k)$ in Ward’s method as noted in Equation 1.15.

A binary split, $s(C_k)$, on a given cluster C_k will result in a partition of the original cluster into two smaller clusters, C_{kL} and C_{kR} (L and R refer to ‘left’ and ‘right’ in the dendrogram produced). The inertia decrease before and after the partition is defined as

$$\Delta(s, C_k) = I(C_k) - I(C_{kL}) - I(C_{kR}), \quad (1.17)$$

and the optimal split, $s^*(C_k)$, is the one that maximizes this decrease in inertia. Then, $s^*(C_k)$ is defined as

$$s^*(C_k) = \operatorname{argmax}_s \Delta(s, C_k). \quad (1.18)$$

Figure 1.8 shows the results from a monothetic clustering on two scaled dimensions of the Palmer penguin data set, flipper length and bill length. The first split is on flipper length, with the second split on bill length as shown in the tree diagram in Panel (a) of 1.8). Panel (b) shows the resulting clusters with the splits shown by dotted lines. Each penguin assigned to a cluster shares a common characteristic: Cluster One includes all the penguins with large flipper lengths (> 206.5 mm, after un-scaling), Cluster Two penguins

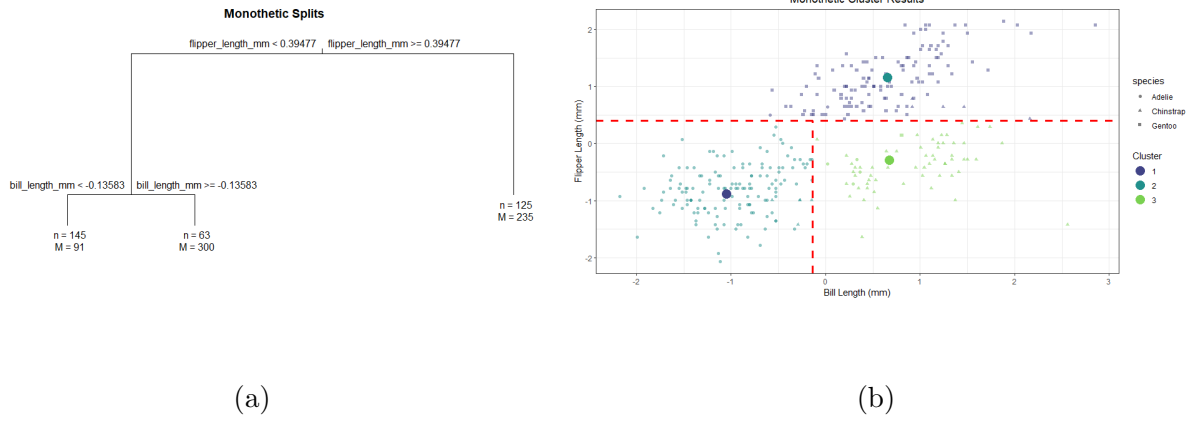


Figure 1.8: Plot of monothetic bipartitions made to the Palmer penguins data set. Splits are made on flipper length only and bill length, shown in the tree diagram in Panel (a) and resulting horizontal/vertical splits in Panel (b) showing the resulting clusters. The clusters closely align with the species labels in the plot based on bill lengths and flipper lengths.

have smaller flipper lengths and below-average bill lengths (< 43.25 mm, after un-scaling), and Cluster Three penguins have smaller flipper lengths but larger bill lengths (> 43.25 mm, after un-scaling). Note that this differs from the polythetic hierarchical clusterings in Figure 1.7, which partition the data based on both features simultaneously and thus provide harder-to-interpret groups.

Various simulation studies have been performed on the monothetic clustering method proposed by Chavent (1998) as well comparisons to other clustering methods (see Chavent et al. (2007), Chavent, Lechevallier, Vernier, and Petit (Chavent et al.), and Chavent et al. (2019)). Comparing across simulated data sets as well as six real data sets, they found that the monothetic clustering algorithm outperformed K -means and Ward's methods in making correct classifications when the number of clusters was small. Additionally, in the case of numerical data sets, monothetic clustering outperformed the other methods when data volumes were higher. Tran (2019) additionally extended the method to implementations on both circular variables and the aforementioned functional data settings, as well as proposing a hybrid hypothesis testing and criterion-based method for determining the number of clusters

to use in MC.

1.4.4 K -means Clustering

Unlike Hierarchical clustering, K -means clustering performs a *flat* clustering, meaning that clusters cannot be further partitioned into sub-clusters by the same method. The K -means algorithm for a K -group cluster analysis was presented initially by MacQueen (1967). It involves starting with an initial set of cluster centers placed arbitrarily (these are sometimes also called “centroids”), and then the cluster centroids and cluster membership are iteratively updated until convergence. These cluster centers can then be used to describe the nature of the objects in each of the clusters. K -means seeks to identify a partition of the data that minimizes the within-group sum of squares (WGSS) across all the variables considered for a given choice of K —in this sense, it is similar to Ward’s Method in hierarchical clustering as well as to Monothetic Clustering’s inertia-based framework, although those produce all of the K -cluster solutions, instead of just K .

Kaufman and Rousseeuw (1990) suggest an alternative to using cluster centroids, which is to use actual observations from the clusters as representatives, which are referred to as *exemplars* or *medoids*. This gives rise to a technique that is similar in nature to K -means, but is called K-Medoids or Partitioning around Medoids (PAM) clustering since it instead relies on assignment of observations to nearest medoids in the P -dimensional clustering space. K-Medoids is suggested to be more robust with respect to the inclusion of outliers than K -means (Kaufman and Rousseeuw, 1990) and can be applied to distance/dissimilarity matrices, $\mathbf{D}$, or directly to the original data, $\mathbf{X}$ (see Steinley (2015)).

1.4.5 Model-Based Clustering

Model-based, probabilistic clustering methods provide both a hard clustering based on an assigned cluster label, C_k , as well as measures of cluster membership that can be interpreted probabilistically. Mixture modeling, or latent-variable modeling, assumes an

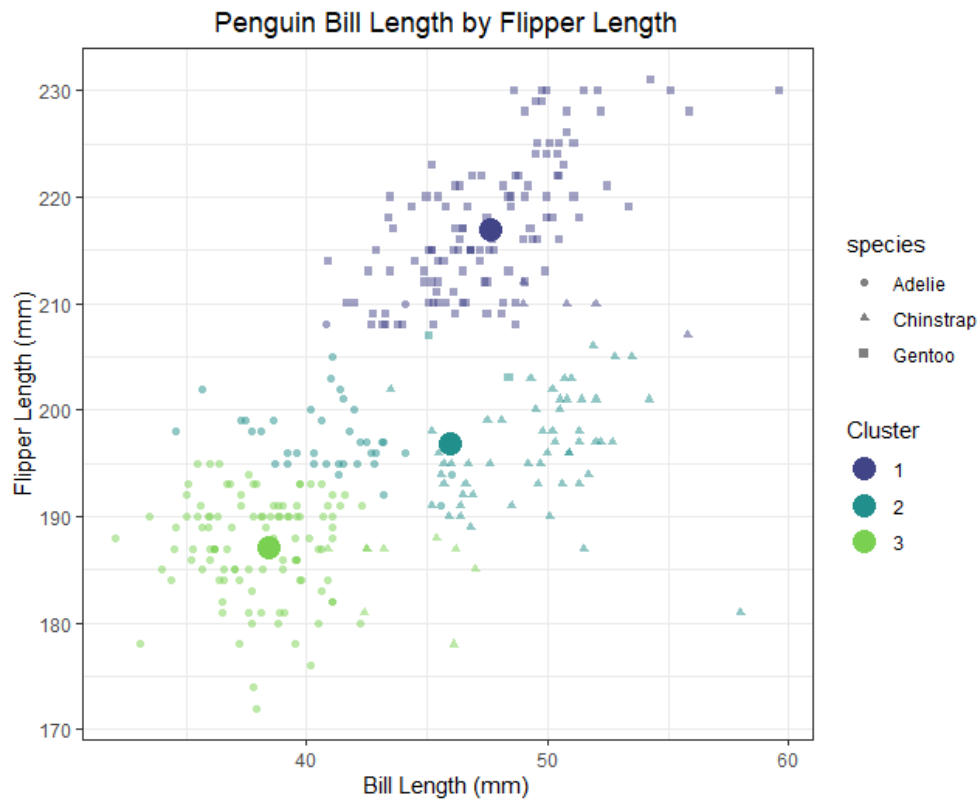


Figure 1.9: Example of K -means clustering on Palmer penguin data with three reasonably well-separated groups.

underlying model that generates the clusters and attempts to estimate the unobserved, or latent, model. The “subpopulation” can be defined as a latent categorical variable and the latent classes are described by the different components of the mixture density (Everitt and Hothorn, 2011).

In general, the basic idea for finite mixture modeling is to define the overall density of the N observations of the P -dimensional multivariate data set X_P as the product

$$f(x_1, \dots, x_P | \theta) = \prod_{j=1}^P \sum_{i=1}^K \pi_i f_i(x_j | \theta), \quad (1.19)$$

where π_i refers to the mixing proportions across K clusters and sum to 1, $f_i(x)$ is the density of the i^{th} cluster, and θ is a vector of multivariate parameters used to define the specific mixture for each of the P features. The densities are estimated based on the observations assigned to each cluster, C_k , and the method seeks to define cluster memberships of the N observed objects across the P features (Fraley and Raftery, 2002). Chapter Two focuses on the univariate application of model-based clustering, but the idea can be extended to multivariate clustering problems as well.

In both the univariate and multivariate cases, the EM algorithm is used to iteratively search for an optimal cluster allocation, and given a specific allocation, estimate the mean, variance, covariance, and mixing proportions for each of the differing K clusters. The process results in optimal choices of cluster assignments and densities of clusters for a given choice of K (a flat cluster result like K -means) and whether variances/covariances are assumed to be the same or different across the clusters. More detailed discussion is given in Banfield and Raftery (1993), Fraley and Raftery (2002), and the R package `mclust` (Fraley et al., 2012).

Because the cluster solution is based on an underlying model, information-theoretic metrics can be used to determine both the optimal number of clusters as well as the correct variance structure (i.e., different variances or the same, and various other related structures).

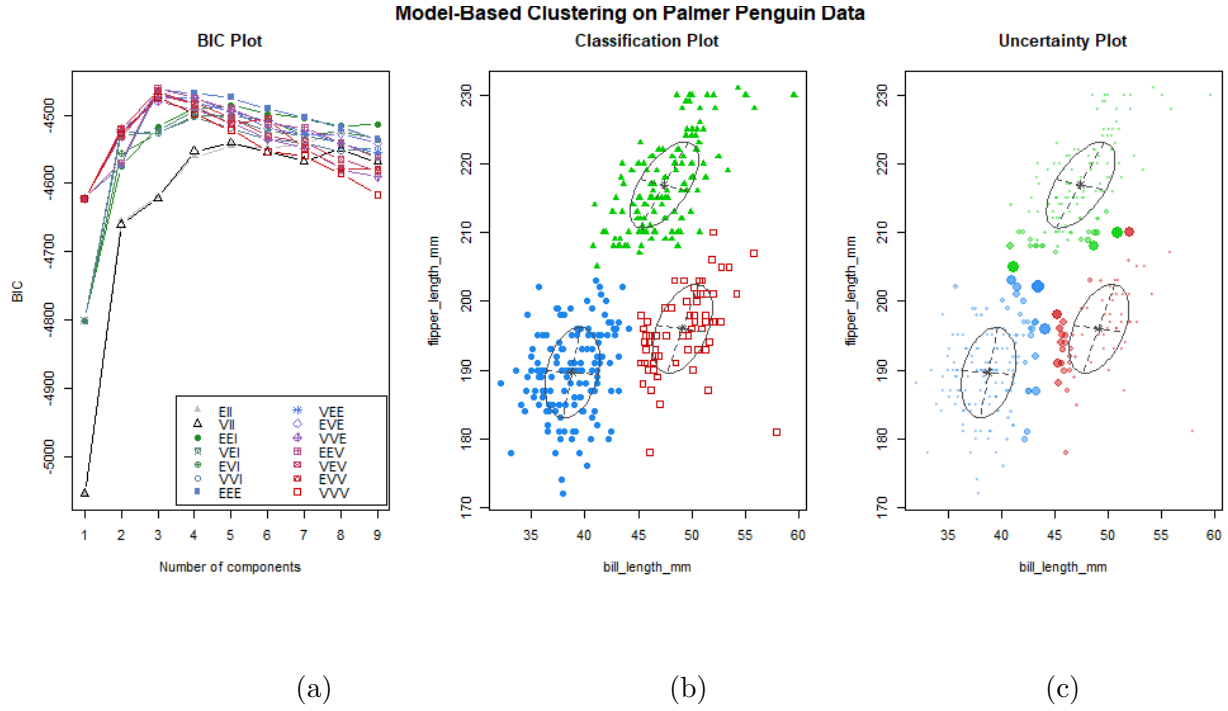


Figure 1.10: Results from model-based clustering of Palmer penguin data set, including BIC results shown in Panel (a), assigned clusters in Panel (b). Multivariate cluster uncertainties in Panel (c), with larger circles representing higher estimated probabilities of belonging to another cluster.

Metrics such as Bayesian information criterion (BIC) can be utilized—in this case, larger BIC values indicate more evidence to support a given model vs. the others. Finally, estimates of cluster uncertainty are available as output from model-based clustering, meaning that model-based clustering can provide a soft clustering solution as well as a hard clustering one.

Figure 1.10 shows the results of model-based clustering on the Palmer penguin data used earlier. Panel (a) shows the BIC selection for various different types of mixture models considered, which range in complexity from spherical with equal variance to more complex and varying ellipses with varying size, shape, and orientation (Scrucca et al., 2016). In this case, the top BIC-selected model had ellipsoidal shape, equal volume and equal orientation

for the clusters with $K = 3$; assigned clusters are visualized as in Panel (b) of Figure 1.10 and cluster uncertainty (which is higher near the cluster boundaries) is in Panel (c) of Figure 1.10.

1.4.6 Picking the Number of Clusters

In most clustering problems, methods require a choice of the number of clusters K . The number of clusters relates to the implied complexity of the data set and the interpretation of the resulting clusters. Even in the case of hierarchical clustering methods, a specific clustering is the resulting partition obtained by cutting the tree at a specific level. Generally, a solution where there are too few clusters present is too simplistic (i.e., it combines unlike observations into groups) whereas a solution that has too many clusters is too complex (i.e., it splits like observations into different groups). The topic of *cluster validation* is therefore a critical component of cluster analysis as it focuses on defining the right number of clusters K (Hennig, Christian; Meila, 2015). Indeed, in some cases, this problem can inform whether or not clustering is a worthwhile endeavor at all; if the optimal K is chosen as 1, there may not be substantive value in performing a clustering solution due to a lack of distinct groups in the data set. Cluster analysis will always produce clusters, but this discussion focuses on whether or not any clusters should be formed and, if so, how many should be identified.

Various techniques exist for trying to select the right number of clusters. The Gap Statistic (Tibshirani et al., 2001) is a method that allows for direct comparison of different numbers of clusters; as is the average Silhouette Statistic. In latent class cluster analysis or model-based clustering, information-theoretic methods, like BIC, can be used to directly assess this question, and potentially select other aspects of the clusters.

1.4.7 Elbow Method

One of the simplest techniques for determining the optimal number of clusters is simply to look at the within-cluster sum of squares (WCSS) as a function of the number of clusters,

as described by Everitt and Hothorn (2008). As the number of clusters increases, the WCSS will typically decrease; however, the marginal decrease in WCSS will often get smaller at some point with inclusion of additional clusters. Finding the “elbow” in the plot is a subjective process and there is no guarantee that such an “elbow” exists; therefore, various competitor methods for determining the optimal number of clusters have been developed, for example, as reviewed by Milligan and Cooper (1985).

1.4.8 Silhouette Statistic and Pseudo-F

Two other common methods for identifying the right value of K are the average silhouette statistic (Rousseeuw, 1987) and Caliński and Harabasz’s pseudo-F statistic (1974). Rousseeuw (1987) defines the silhouette width to quantify a measure of how “comfortable” an observation is in a given cluster. Defining $a(i)$ to be the average dissimilarity between observation i and other observations in the same cluster, $d(i)_K$ is the average dissimilarity between i and other observations in cluster K , and $b(i) = \min_K(d(i)_K)$ to be the minimum average dissimilarity from i to other clusters, then the silhouette width of an observation i is defined to be:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}. \quad (1.20)$$

Silhouette widths range between -1 and 1, where values close to 1 indicate that the heterogeneity of object i ’s cluster is smaller than its separation and object i is considered “well-classified” (Everitt and Hothorn, 2011). Vice-versa, when $s(i)$ is close to -1, object i is considered poorly classified.

A global measure of cluster solution fit can be taken by aggregating the $s(i)$ ’s to get an average silhouette statistic, $\bar{s} = \frac{\sum_{i=1}^N s(i)}{N}$. Kaufman and Rousseeuw (1990) recommend that optimal silhouette widths should be above 0.5 and that silhouette widths below 0.2 indicate a poor cluster structure. The solution with the largest average silhouette width is

a recommended choice for the number of clusters to use in the cluster solution.

Caliński and Harabasz's (1974) proposed use of a pseudo-F statistic-based criterion to choose the number of clusters to maximize the variation between clusters, $B(K)$, relative to the within-cluster variation, $W(K)$. The solution with the largest pseudo-F is a recommended choice for the number of clusters. For a given value of K , $B(K)$ is defined as

$$B(K) = \sum_{k=1}^K \sum_{q=1}^Q N_k (\bar{x}_{.qk} - \bar{x}_{.q})^2, \quad (1.21)$$

and $W(K)$ is defined as

$$W(K) = \sum_{k=1}^K \sum_{i \in C_k} \sum_{q=1}^Q (\bar{x}_{iqk} - \bar{x}_{.qk})^2, \quad (1.22)$$

where N_k is the number of observations in the k^{th} cluster, C_k , $\bar{x}_{.iq}$ is the i^{th} observation of feature q in cluster C_k , and $\bar{x}_{.qk}$ is the average value for all observations on feature q in cluster C_k , and $\bar{x}_{.q}$ is the overall grand mean of all N observations on feature q . Then, their statistic is defined as:

$$CH(K) = \frac{B(K)/(K-1)}{W(K)/(N-K)}. \quad (1.23)$$

The 1-cluster version, $CH(1)$, is not defined; similarly, the average silhouette statistic is not defined for $K = 1$. Tran (2019) and Everitt (2011) both provide comprehensive overviews of these methods and provide examples of their use. Tran (2019) shows that data sets with no true clusters (i.e., $K = 1$) tend to produce $K = 2$ solutions when using both Caliński and Harabasz's F-statistic and the average silhouette.

1.4.9 Gap Statistic

One critical issue concerning some of the other methods for determining the optimal K is to address the issue of whether clustering ought to be performed at all (i.e., if the data are best described as one homogeneous group). The Gap statistic (Tibshirani et al., 2001) provides a method for assessing the number of clusters that formalizes the “elbow method” by comparing a measure of within-cluster dispersion to its expectation under a null reference distribution, favoring cluster solutions with larger differences in the observed result compared to what would be seen if no clear structure were present. Importantly, it can be assessed for the $K = 1$ cluster alternative.

The Gap statistic, denoted $Gap_N(K)$ for a given choice of K , is defined as

$$Gap_N(K) = E_N^*[log(W_K)] - log(W_K), \quad (1.24)$$

where $E_N^*[log(W_K)]$ is the expected value of the within-cluster dispersion under a simulated reference distribution and $log(W_K)$ is the actual within-cluster dispersion from the observed clusters. The optimal $\hat{K}$ is the value of K that maximizes $Gap_N(K)$ after taking the sampling distribution into account. In cluster solutions where the clusters are well-separated, $log(W_k)$ would be expected to decrease faster than expected if no groups were present; in cases where $K \geq \hat{K}$ (i.e., an unnecessary cluster is being proposed), $log(W_K)$ should decrease more slowly than the expected rate. Thus, the Gap should be maximized when $K = \hat{K}$.

The reference distribution for the Gap statistic is calculated in one of two ways:

1. **Uniform-based:** Generate each reference feature uniformly over the range of observed values.
2. **PCA-based:** Generate the reference features from a uniform distribution over a “box” aligned with the principal components of the data, and draw features that can be back-transformed to generate a rotationally-invariant reference distribution.

Like the average silhouette statistic or elbow method, the Gap statistic is a global procedure rather than a local one. Gap is not without some drawbacks. First, it is dependent on the use of various rules to determine the “optimal” set of parameters. Those rules include selecting an optimal K such that the Gap statistic is globally maximized, but other rules exist that choose the optimal K by taking the variability of the Gap statistic into account. Additionally, Dudoit and Fridlyand (2002) compared the Gap method to their own Clest method, noting that in some cases the Gap statistic tends to favor overestimating the number of clusters in application. Yan and Ye (2007) suggest an improvement to the original Gap statistic via a weighted Gap that allows for a “multi-layer” clustering approach that better detects the optimal number of clusters and applies a “weighted Gap statistic” that weights both the observations and the features. Finally, in some cases, the Gap statistic can be computationally intensive to calculate because it requires repeated generation of reference distributions, and the clustering algorithm to be applied to each generated data set.

1.4.10 Clustering Implementation in R

Various methods for clustering data are discussed as part of this dissertation along with their respective implementations in the statistical software package R (R Core Team, 2022). Hierarchical clustering and K -means clustering can be performed in the base build of R. Model-based clustering techniques are implemented using the `mclust` package in R (Scrucca et al., 2016). Monothetic clustering is performed using the `monoClust` package (Tran et al., 2022). The Gap statistic method is implemented in the `cluster` package (Maechler et al., 2022), as are the other methods discussed for identifying K .

1.5 Introduction to Sparsity

In the presence of both noise and signal-bearing features, statistical methods can suffer from degradation of results, particularly if the signal to noise ratio is relatively low. This is

especially true in the case of high-dimensional data. *Sparsity* refers to a statistical technique by which noise-bearing features are automatically distinguished from signal-bearing ones using some sort of penalized estimation, so that “only a relatively small number of parameters play an important role” (Hastie et al., 2015). Such methods are typically simpler and thus easier to interpret because they rely only on a subset of important features. Tools for automated feature selection to obtain a sparse method exist in nearly every statistical practice; methods such as the Lasso (see Hastie et al. (2015)) are common in regression-type models and related uses of penalization exist for latent-variable models like PCA (see Shen and Huang (2008) and Witten et al. (2009)) and SEM (Jacobucci et al., 2021), as well as for clustering methods like K -means and hierarchical clustering (Witten and Tibshirani, 2010).

1.5.1 Lasso

In order to understand penalized estimation, consider the classical regression setting, where the model is assumed to be

$$y_i = \beta_0 + \sum_{j=1}^P x_{ij}\beta_j + \epsilon_i, \quad (1.25)$$

where $\beta_0, \beta_1, \dots, \beta_P$ are independent and normally-distributed unknown parameters to be estimated and ϵ_i is an error term assumed to be normally distributed with constant variance and mean 0. The standard way of estimating the parameters is least-squares estimation, which solves the following problem:

$$\text{minimize}_{\beta_0, \dots, \beta_P} \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^P x_{ij}\beta_j)^2. \quad (1.26)$$

The least-squares estimates are unbiased estimators of their respective parameters. However, there are some potential drawbacks to using least-squares. First, if the number of parameters to estimate is large, interpretation can become increasingly difficult, and

in the case where $P > N$, least-squares estimates are not unique and the model is not estimable. Moreover, although the predictions generated are unbiased, they are of relatively high variance; in some cases imposing some downward bias on the parameters (β 's) may actually decrease prediction variance. This is known as the variance-bias tradeoff.

A solution to this problem was proposed by Hoerl and Kennard (1970) to impose constraints on the process of estimation, or *regularize*, via ridge regression, which minimizes the least-squares criterion subject to a constraint on the squared ℓ_2 norm of the coefficients:

$$\min_{\beta_0, \dots, \beta_P} \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=0}^P x_{ij} \beta_j)^2 \quad \text{subject to} \quad \|\beta\|_2 \leq t, \quad (1.27)$$

where $\|\beta\|_2$ is the ℓ_2 norm of the coefficient vector β and t is a user specified parameter that, when chosen wisely, will downwardly bias the coefficients and provide more precise (low-variance) estimates (called *shrinkage*). The Lasso (Tibshirani, 1996) improves on this process of regularization by replacing the ℓ_2 norm with the ℓ_1 norm, which allows for both *regularization* as well as *feature selection*. The Lasso imposes the ℓ_1 constraint on the least squares problem as

$$\min_{\beta_0, \beta_1} \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=0}^P x_{ij} \beta_j)^2 \quad \text{subject to} \quad \|\beta\|_1 \leq t, \quad (1.28)$$

where the $\|\beta\|_1 = \sum_{j=1}^P |\beta_j|$ is the ℓ_1 norm of β and t is a specified penalty parameter. For more details on the Lasso, see Hastie et al. (2015) and Tibshirani (1996). Additional methods have developed to include the benefits of both ridge and Lasso penalties on regression models, giving rise to the elastic net framework (see Zou and Hastie (2005) and Friedman et al. (2010)) which combines the ℓ_1 (Lasso) and ℓ_2 (ridge) penalties and adds a tuning parameter to control the balance between the two.

In the case of regression models, better results can be obtained when taking advantage of the automated feature selection used by the Lasso. However, the Lasso (and related sparse

methods) need not be constrained simply to OLS regression problems; Hastie et al. (2015) overviews methods for using the ℓ_1 Lasso penalty in generalized linear models, support vector machines, and unsupervised multivariate analysis (which is of foremost interest here).

1.5.2 Sparse Clustering

In the unsupervised setting, a lasso-like penalty in clustering is desired. Consider a multivariate setting where the number of multivariate responses, or features, P , is relatively large—in some cases, potentially larger than the number of observations N . In such a situation, applying methods to cluster may cause several potential problems. First, the resulting clustering solution may not be precise as there is a possibility that some (or potentially many) of the multivariate responses may contribute noise rather than meaningful signal. Second, the computation time to calculate clusters may be infeasible, or at least potentially longer than desired. Finally, the resulting clusters may not be interpretable as they rely on a large set of features (particularly in a polythetic clustering setting).

Witten et al. (2009) propose a two-step method for “sparsifying” an input matrix to a variety of statistical methods, and then applying statistical methods to that sparse matrix. In the case of clustering, this process of “sparsification” can be carried out on the input distance/dissimilarity matrix as an initial step with a resulting reweighted dissimilarity matrix, and then a clustering method can be applied to the new reweighted dissimilarity matrix as input. The penalized matrix decomposition (PMD) proposed by Witten et al. (2009) provides a method to obtain reweighted dissimilarities.

The penalized matrix decomposition can be applied directly to a data matrix $\mathbf{X}_{N \times P}$, with $x_1, \dots, x_p$ in the columns of $\mathbf{X}$. Assume further that the columns of X have been scaled to have mean 0. Then, the *feature-specific* dissimilarity matrix can be defined for feature j as $d_{i,i',j}$. The *overall* dissimilarity/distance matrix sums over all the features is $\mathbf{U}_{N \times N} = \sum_j d_{i,i',j}$. (Note that this differs from previous notational conventions where $\mathbf{D}$

refers to the distance/dissimilarity matrix - this is done to align with the notation of Witten and Tibshirani (2010).

Then, the penalized matrix criterion of Witten and Tibshirani (2010) is framed as the solution to the optimization of a set of the j feature weights, $w_1, \dots, w_P$, applied to the sum of the overall and feature-specific distance matrices as

$$\begin{aligned} & \max_{\mathbf{U}, \mathbf{w}} \sum_j w_j \sum_{i, i'} d_{i, i', j} \mathbf{U}_{i, i'} \\ & \text{subject to } \|\mathbf{U}_{i, i'}^2\|_2 \leq 1, \|\mathbf{w}\|_2^2 \leq 1, \|\mathbf{w}\|_1 \leq s, \text{ and } w_j \geq 0, \\ & \quad \forall j, \end{aligned} \tag{1.29}$$

where s is a tuning parameter, with smaller values of s resulting in a more sparse set of weights w .

This criterion can be further simplified with some additional notation. For this section, let $\mathbf{D}_{N^2 \times P}$ be a matrix consisting of feature-wise distances/dissimilarities $d_{i, i', j}$ strung out into a vector in the j^{th} column. Then, we can re-express the sum $\sum_j w_j \sum_{i, i'} d_{i, i', j} \mathbf{U}_{i, i'} = \mathbf{u}^T \mathbf{D} \mathbf{w}$ with $\mathbf{u}$, the vectorized version of the overall dissimilarity/distance matrix $\mathbf{U}$. The resulting, more simplified, PMD criterion is

$$\begin{aligned} & \max_{\mathbf{u}, \mathbf{w}} (\mathbf{u}^T \mathbf{D} \mathbf{w}) \\ & \text{subject to } \|\mathbf{u}\|_2 \leq 1, \|\mathbf{w}\|_2^2 \leq 1, \|\mathbf{w}\|_1 \leq s, \text{ and } w_j \geq 0, \\ & \quad \forall j. \end{aligned} \tag{1.30}$$

The components $\mathbf{w}$ and $\mathbf{u}$ are iterated until convergence to get both the optimal set of weights $\hat{\mathbf{w}}$ and the vectorized distance/dissimilarity matrix $\mathbf{u}$. Either piece can be used in generating sparse inputs to statistical learning methods. In the case of hierarchical clustering (and any other distance-based clustering algorithm), sparse clustering can be obtained by simply using the obtained reweighted distance matrix as input to the clustering

algorithm. Alternatively, the weights can be applied to the original data matrix $\mathbf{X}$ to obtain a reweighted matrix to be directly input to a clustering algorithm like K -means. Another method, called clustering on subsets of attributes (COSA, Friedman and Meulman, 2004), provides a reweighted dissimilarity matrix that can be used similarly. COSA is often compared with Witten and Tibshirani’s PMD-based method for sparse clustering, and both frameworks have been shown to perform well in certain settings.

Additional implementations of sparse methods such as those listed previously exist. Sparse PCA was implemented in Witten et al. (2009) and Zou and Hastie (2005) using the penalized matrix decomposition, as well as an alternative method implemented by Shen and Huang (2008), which introduces sparsity as part of a low-rank matrix approximation using penalized estimation.

1.5.3 Implementation in R

Sparse clustering in R can be done using the `sparcl` package (Witten and Tibshirani, 2010), both for sparse K -means clustering or sparse hierarchical clustering. An additional implementation of sparse K -means clustering that contains more robust handling of outliers is in the `RSKC` package (Brodinová et al., 2019) in R.

1.6 Overview of Repeatedly Used Data Sets

Some of the data sets used throughout this dissertation are common to some or all of the chapters contained. These data sets are overviewed in this section. For data sets that are used only in one section, they are overviewed specifically in their own chapter.

1.6.1 Carnegie Classification

The Carnegie Classification of Institutions of Higher Education (CC) has sought to describe “institutional diversity in U.S. higher education for the past four and a half decades”

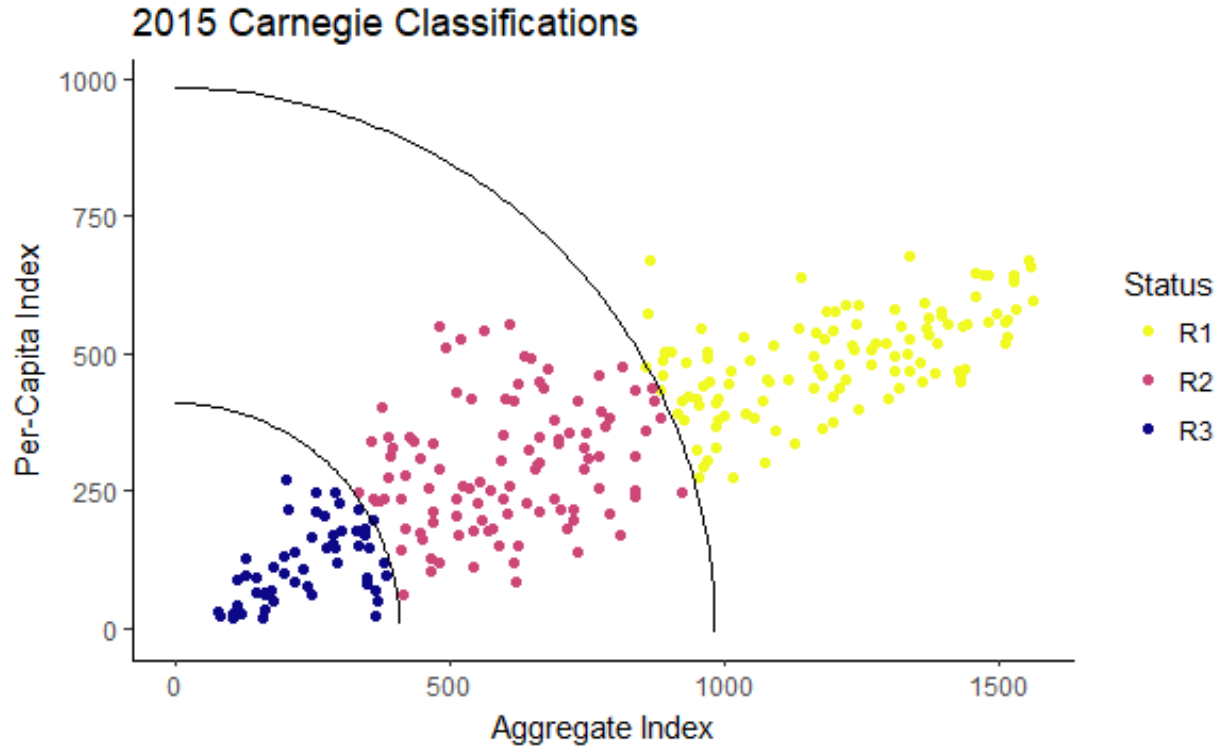


Figure 1.11: Carnegie Classification as released in 2015, specifically focusing on the doctoral-granting group of institutions. The R1 category is classified as “Very High Research”, with R2 referring to “Higher research” and R3 the “Moderate research” group.

(Carnegie Foundation for the Advancement of Teaching, 2019). Ideally, the CC is used to develop groups of peer institutions for analysis by separating similar institutions into groups.

The CC system in 2015 (see Figure 1.11) sorted $N = 276$ doctoral-granting institutions into three classes: R1: Highest research activity, R2: Higher research activity, and R3: Moderate research activity. Note that in more recent updates (2018 and 2021), the CC now identifies a threshold between the top two groups of R1: Doctoral Universities—Very High Research Activity and R2: Doctoral Universities—High Research Activity. The institutional classification provided by the CC can be construed as something akin to such institutional

rankings as the US News and World Report Best Colleges Rankings, Shanghai Academic Ranking of World Universities, the Times Higher Education World University Rankings, and others (Herzog, 2016).

This dissertation focuses primarily on the 2015 version of the CC as a primary interest in Chapter Two and as an example data set in additional chapters. The data set used throughout this dissertation includes measures of STEM and Non-STEM expenditures, counts of PhDs awarded in Social Sciences, STEM, Humanities, and Other (professional) programs, as well as two measures of faculty headcounts: tenurable faculty and non-tenurable research faculty/postdocs. The 2015 CC data set has $P = 8$ variables with clearly defined groups and observations for a set of universities, providing a data set that is of interest for cluster analysis (with comparison to CC groupings) and mapping (to compare to CC groups and with respect to individual institutions).

1.6.2 Palmer Penguins

A recent data set that has begun to be used as a common demonstration data set in R is the Palmer penguin data set. It contains body measurements taken on penguins collected from three islands in the Palmer Archipelago, Antarctica, originally collected and published by Gorman et al. (2014). Measurements of several physical characteristics were taken for $N = 344$ penguins, including bill length (mm), bill depth (mm), flipper length (mm), and body mass (grams). In total, the data comprise three species: Chinstrap, Adelie, and Gentoo penguins, each with relatively distinct profiles across those four dimensions. The Palmer penguins data set can be accessed via the palmerpenguins package in R (Horst et al., 2020). Additionally, we acknowledge the Palmer Station LTER and the US LTER Network for making the data publicly accessible. This Palmer penguin data set can be used both for mapping purposes with a relatively low number of features being collapsed into two dimensions, and provides a useful baseline set of signal-bearing features to which noise

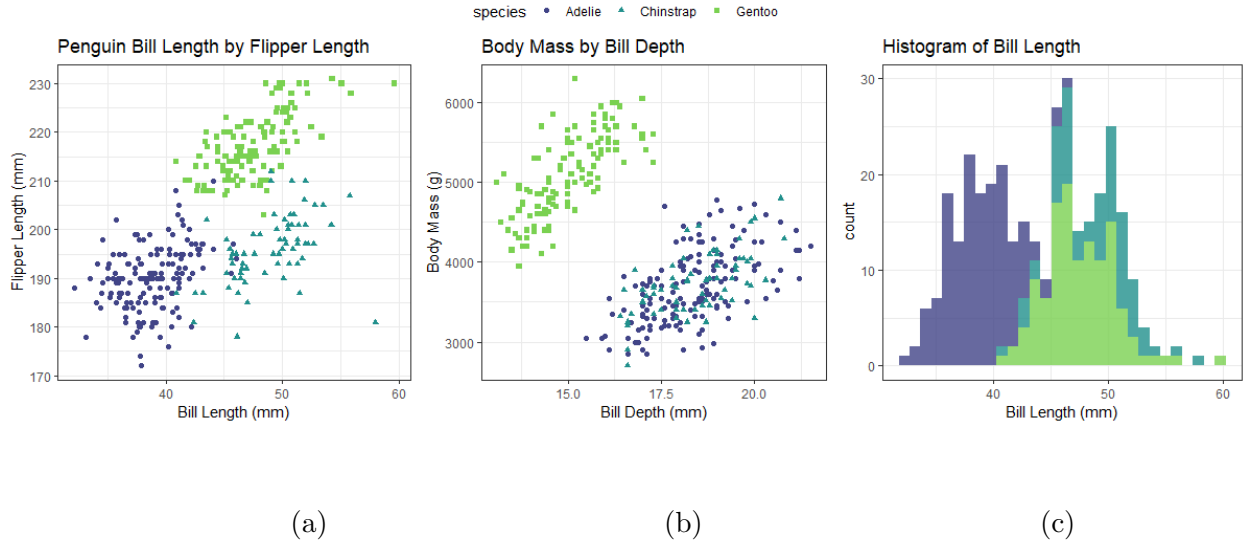


Figure 1.12: Palmer penguin data set with selected comparisons of the four quantitative measurements, including bill length, flipper length (Panel a), body mass, and bill depth (Panel b), and overall bill lengths (Panel c) for penguins from three species.

features can be simulated and added for comparisons in clustering problems, as the penguins from different species show some clear differences on some morphometric characteristics in Figure 1.12.

1.7 Dissertation Overview

This dissertation is organized in the following way. Chapter Two outlines a novel method for classification of doctoral-granting universities to be compared with the Carnegie Classification for Higher Education (CCHE). The classification method is based on a two-step process of using an SEM to estimate latent factors of doctoral productivity, and then generate clusters of like institutions using model-based clustering methods. Chapter Three introduces a diagnostic to measure the impact of potentially influential observations on methods that generate maps of multivariate data, such as classical MDS and t-SNE. The method is a novel extension of a Cook's Distance-like pseudo-F statistic for detection of influence in ordination,

and provides a tool that can be used agnostic of the mapping procedure. Chapter Four focuses on a novel method for sparse monothetic clustering using a penalized dissimilarity matrix to achieve sparsity based on monothetic splits made only on signal-bearing data elements. Methods for tuning two parameters, s —the degree of sparsity, and K —the number of clusters, are explored, including a new extension of rules used for selecting the optimal values. Finally, Chapter Five concludes with remarks and conclusions, and proposes several interesting avenues for further research related to the methods presented.

CHAPTER TWO

AN ALTERNATIVE TO THE CARNEGIE CLASSIFICATIONS: IDENTIFYING
SIMILAR DOCTORAL INSTITUTIONS WITH STRUCTURAL EQUATION MODELS
AND CLUSTERINGContribution of Authors and Co-Authors

Author: Harmon, Paul

Contributions: Primary Author responsible for the majority of writing and development of analysis and methods used.

Author: McKnight, Sarah

Contributions: Aided in analysis and design of structural equation model (SEM) and clustering solution.

Author: Hildreth, Laura

Contributions: Provided feedback on the statistical analysis (particularly for the SEM-related components) and provided some writing on the section pertaining to SEM.

Author: Godwin, Ian

Contributions: Provided feedback on methodology and background information about historical methods for institutional classification.

Author: Greenwood, Mark C.

Contributions: Provided feedback on statistical analysis and drafts of manuscripts, as well as some background information on the clustering solution utilized.

Manuscript Information

Paul Harmon, Sarah McKnight, Dr. Laura Hildreth, Dr. Ian Godwin, and
Dr. Mark Greenwood

Journal: *Statistics and Public Policy*

Status of Manuscript:

☐ Prepared for submission to a peer-reviewed journal

☐ Officially submitted to a peer-reviewed journal

☐ Accepted by a peer-reviewed journal

☒ Published in a peer-reviewed journal

Publisher: Taylor and Francis Online

Submitted: September 6, 2019

Published: October 16, 2019

Volume: 6

Issue: 1

DOI: 10.1080/2330443X.2019.1666761

Abstract

The Carnegie Classification of Institutions of Higher Education is a commonly used framework for institutional classification that classifies doctoral-granting schools into three groups based on research productivity. Despite its wide use, the Carnegie methodology involves several shortcomings, including a lack of thorough documentation, subjectively placed thresholds between institutions, and a methodology that is not completely reproducible. We describe the methodology of the 2015 and 2018 updates to the classification and propose an alternative method of classification using the same data that relies on structural equation modeling (SEM) of latent factors rather than principal component-based indices of productivity. In contrast to the Carnegie methodology, we use SEM to obtain a single factor score for each school based on latent metrics of research productivity. Classifications are then made using a univariate model-based clustering algorithm as opposed to subjective thresholding, as is done in the Carnegie methodology. Finally, we present a Shiny web application that demonstrates sensitivity of both the Carnegie Classification and SEM-based classification of a selected university and generates a table of peer institutions in line with the stated goals of the Carnegie Classification.

2.1 Introduction

Systems for institutional classification are often important to administrators, faculty, and, to a lesser extent, students at institutions of higher education. The Carnegie Classifications of Institutions of Higher Education (CC) seek to describe “institutional diversity in U.S. higher education for the past four and a half decades” (The Carnegie Classification of Institutions of Higher Education, n.d.). The nominal information (classification) provided by the CC is often either unintentionally or intentionally misrepresented as ordinal information (rankings), akin to such institutional rankings as the US News & World Report Best Colleges

Rankings, Shanghai Academic Ranking of World Universities, the Times Higher Education World University Rankings, etc. However, as Brint (Brint (2013)) states, “Classification is a method for apprehending the structure of a system; ranking is a method for stimulating competition among those at a similar level in the system.” The CC, on the other hand, was developed in 1970 by the Carnegie Commission on Higher Education to “be helpful to many individuals and organizations that are engaged in research on higher education” (McCormick and Zhao, 2005). The distinction has been reemphasized throughout the classification system’s four and a half decade history, including by the current director of the Indiana University Center for Postsecondary Research that produces the CC, Victor Borden, who stated that “the label should not be viewed as a ranking or rating” (Herzog, (Herzog, 2016)).

Since its publication in 1973, the CC has been updated seven times (1976, 1987, 1994, 2000, 2005, 2010, 2015) to account for changes in the “universe of institutions (the result of openings, closings, and mergers) in the U.S. and the institutions themselves (the result of changes in offerings and activities)” (McCormick and Zhao, 2005). In the 2005 update, instead of a single framework representing similarities and differences among institutions, the classification provided a set of independent, parallel classification frameworks through which similarities and differences can be viewed (McCormick and Zhao, 2005). Those six parallel classifications include: the Basic Classification (the traditional CC Framework), the Undergraduate Instructional Program, the Graduate Instructional Program, the Enrollment Profile, the Undergraduate Profile, and the Size & Setting classification. The 2005 update also introduced the use of a “multi-measure research index to classify [doctoral]-granting institutions” (Carnegie Foundation for the Advancement of Teaching, 2019). Doctoral-granting institutions typically focus on the Basic Classification. It is currently defined as including those higher education institution awarding 20 or more research doctoral degrees in a given year. The CC system sorts institutions into three classes: R1 - Highest research activity, R2 - Higher research activity, and R3 - Moderate research activity.

To the dismay of Clark Kerr, one of the principal architects of the classifications, who “expressed unhappiness about the amount of organizational striving the Classification had encouraged,” institutions have often sought to move up to the R1 group (McCormick and Zhao, 2005). Indeed, even among smaller R1 institutions, maintenance of that “status” can often be a priority. This is because it provides them with the opportunity to market their inclusion in the R1 category as an indicator of quality that makes them potentially more desirable to prospective faculty, funding agencies, graduate students, and (to a much lesser extent) undergraduate students. Interestingly, institutions towards the “top” of the R1 class (Johns Hopkins, Harvard, Stanford, University of Washington, University of Michigan, UC – Berkeley, etc.) are often less interested, if not uninterested, in the CC. This is because they are either assured of being in the “highest class” perpetually (assuming there are no tectonic shifts in the methodology used to determine the classes), or are more focused on their membership in the more exclusive Association of American Universities (AAU).

The impact or concern regarding the institutional classifications is more evident on either side of the boundaries between R1 and R2, and the R2 and R3 classes. In those areas, several schools have implemented explicit policy goals and timelines oriented towards ascending into the (perceived) higher class, or more quietly trying to avoid moving down to a lower class. Several institutions in the ‘Higher research activity’ (R2) group have explicitly set out policy goals directed towards improving their standings both after the 2010 update and 2015 updates to the classifications. Some, like Texas Tech University (Rangel, 2015), Temple University (Verghese, 2016), and Kansas State University (Dorhout and Cox, 2016), set explicit funding and research related goals in order to transition from R2 to R1 in the next Carnegie Classification update, while others including the University of Montana (2017) and the University of Idaho (2016) hope that similar strategic efforts result in their institutions transitioning to the R1 class in a later iteration of the classifications.

While the Carnegie Classifications, US News World Rankings, and other metrics

of institutional characteristics provide interesting and useful data for decision makers at institutions, they lack reproducibility and transparency as they are either poorly documented or proprietary. Recently, Kosar and Scott (2018b) attempted to replicate the Carnegie Classifications and illustrated several of the shortcomings in their methodology.

The purpose of this paper is to additionally discuss the Carnegie Classifications' methodology and present a possible alternative using mixture modeling on scores obtained from a Structural Equation model, or SEM (Bollen, 1989). Much of the research pertaining to processes for institutional classification deals with variable selection: Identifying the data and characteristics that most accurately describe each university (Shin et al., 2013). However, we are more concerned with the statistical methodology used to group institutions than with the data themselves. Rather than proposing that new variables be added to the dataset to improve classifications, we propose an alternate statistical methodology that utilizes the same data as were used in the Carnegie Classifications and by Kosar and Scott (2018b) and provides an easier model to interpret, a platform to assess diagnostics, and an objective way to cluster that aligns with the stated goals of the Carnegie Classification system.

The rest of the paper is organized as follows. In the second section, this paper discusses the statistical strengths and deficiencies associated with the Carnegie Classification methodology and similar Principal Component Analysis (PCA)-based alternatives. In the third section, we outline a proposed Structural Equation Model and clustering methodology that can be used as an alternative to the Carnegie method. Finally, we introduce a web-based applet that can be used to assess the sensitivity of both classification systems to changes in the characteristics of a single university, along with introducing the idea of exploring neighboring peer institutions as part of the classification system.

2.2 Methodology of the Carnegie Classifications

2.2.1 The Data

The data used in the Carnegie Classifications are published with each update. They can be obtained from the Carnegie website at <http://carnegieclassifications.iu.edu/>. The data from the 2015 update, with which this analysis is primarily concerned, contain information pertaining to 4,664 institutions within the U.S., from special focus two-year institutions, tribal colleges, baccalaureate and master’s colleges, to doctoral-granting universities. However, this analysis focuses specifically on the doctoral-granting schools ($n = 335$) since the variables used here only apply to those such institutions. Similar methods would be applicable to other tiers of the classification system as well, so this could be replicated for non-doctoral institutions using the variables pertinent to those types of institutions. Information contained in the Carnegie dataset comes from a variety of sources, including the Integrated Postsecondary Education Data System (IPEDS) and the National Science Foundation (NSF) HERD survey, and are collected in a single snapshot rather than over a period of multiple years. For the analyses that underpin the Classifications, the data are separated into two distinct sets of variables, an aggregate and a per-capita dataset. The aggregate dataset contains the seven variables shown in Table 2.1.

The per-capita dataset contains both expenditure types and the research staff size variable from the aggregate dataset, but divides them by the size of the tenured/tenure-able faculty. They are listed in Table 2.2.

2.2.2 The Methodology

The exact methodology of the Carnegie Classifications is not documented; however, Kosar and Scott (2018b) and Harmon (2017) provide some insight into the process used to construct them. Prior to any analysis, the variables are ranked from smallest to largest, and

<u>Variable Name</u>	<u>Definition</u>
STEM PhDs	Count of doctorates awarded in STEM fields, based on IPEDS classification.
Humanities PhDs	Count of doctorates awarded in Humanities fields, based on IPEDS classification.
Social Sciences PhDs	Count of doctorates awarded in Social Science Fields, based on IPEDS classification
Other PhDs	Counts of professional doctorates awarded in education (EdD or PhD), based on IPEDS classification.
STEM Exp.	Expenditures on STEM-related research, in thousands of dollars.
Non-STEM Exp.	Expenditures in non-STEM research fields, in thousands of dollars.
Research Staff Size	Non-tenurable research faculty and post-doctoral researchers, based on IPEDS classification.

Table 2.1: Table of the variables used in the aggregate CC dataset.

schools that are tied on any metric are given the minimum rank of the tied group (Harmon, 2017) and (Kosar and Scott, 2018b). Ranking is done because the values of the variables for some of the largest universities dwarf the values for the smaller institutions in the dataset, leading to many of the variable distributions being highly right-skewed. Ranking leads to a consolidation of schools that removes much of the separation between institutions, as outlined by Kosar and Scott (2018b).

The ranked datasets are put into two PCAs, which use eigenvector decompositions of each set of variables to create orthogonal axes that explain the most variation in the underlying variables (Hastie et al., 2001). The first component from the PCA of the 7

<u>Variable Name</u>	<u>Definition</u>
STEM Exp. per capita	Expenditures on STEM-related research, in thousands of dollars, divided by the headcount of tenured/tenure-track faculty.
Non-STEM Exp. per capita	Expenditures in non-STEM research fields, in thousands of dollars, divided by the headcount of tenured/tenure-track faculty.
Research Staff Size per capita	Non-tenurable research faculty and post-doctoral researchers, based on IPEDS classification, divided by the headcount of tenured/tenure-track faculty.

Table 2.2: Table of variables used in the per-capita CC dataset.

aggregate variables is used to form a single aggregate index. Similarly, a per-capita index is generated using the first principal component from the PCA of the per-capita variables. This process of dimension reduction involves some loss of information: In previous iterations of the Carnegie Classifications, these single indices explained between 68% and 72% of the variation in the underlying data and in the 2015 specifically, these were 70% and 72% for the aggregate and per-capita scales, respectively (V. Borden, personal communication, January 8, 2016).

In Figure 2.1, the per-capita scores (y-axis) are plotted against the aggregate scores (x-axis) for each school. Rather than using an objective statistical clustering method as was attempted by Kosar and Scott (2018b), the plot of scores is then partitioned approximately into thirds by hand by drawing concentric arcs that separate institutions into three groups (V. Borden, personal communication, February 2, 2017) as shown in Figure 2.1. Schools in the bottom left corner of the plot are in the “Moderate Research” category, and those

in the top-right corner are in the “Highest Research” category, with the “Higher Research” category in the middle. Because both the scores and the boundaries can change in different years, it is possible for schools to move between categories from one update to the next based either on changes in the subjective grouping in the plot or on differences in individual institutional productivity, ranks, or changes in the PCAs that then change the relative locations of institutions.

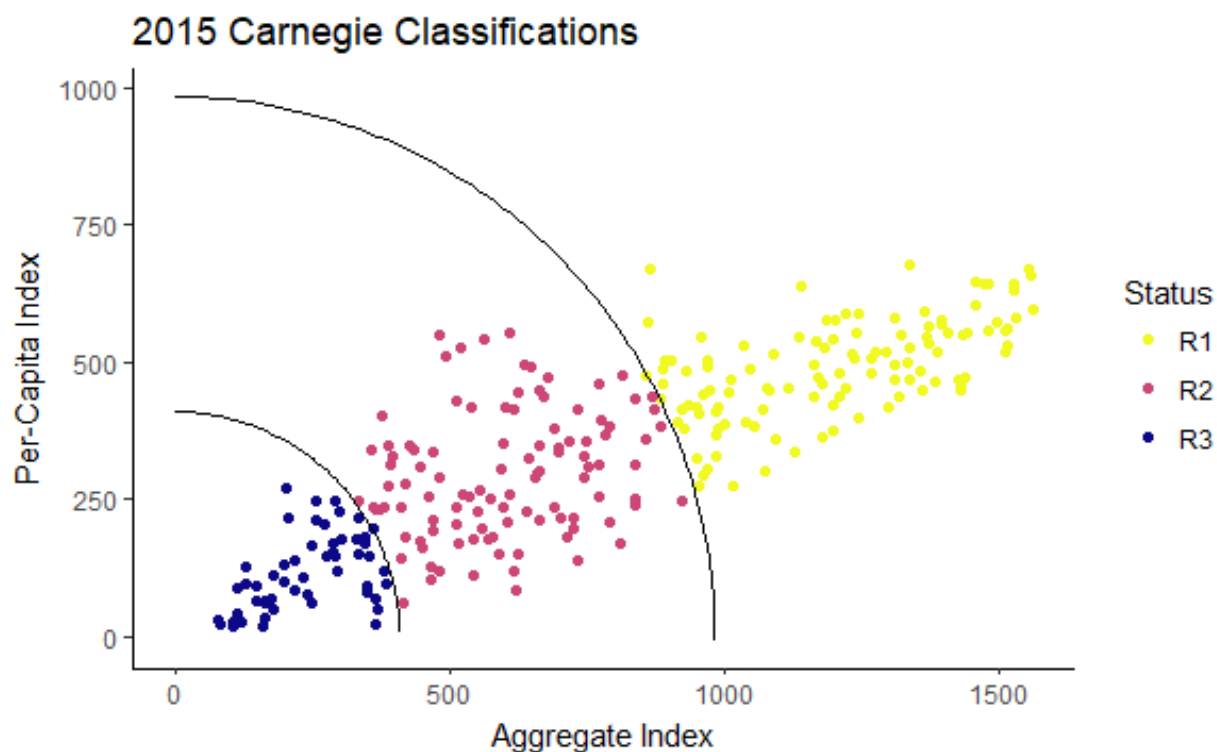


Figure 2.1: The 2015 update of the Carnegie Classifications with R1, R2, and R3 classifications and boundaries displayed.

2.2.3 Problems with the Carnegie Classifications

The Carnegie Classifications have the potential to be used by policymakers on campuses to drive institutional goals and academic development. However, universities that prioritize moving from one category to another must account for possible changes in the way that future calculations are done. Because the data used to calculate the Carnegie Classifications in any given year are based on ranked snapshots at a single time point, the weighting of a single variable can change from year to year. This is not always drastic; however, if large changes were to occur in the characteristics of many of the universities in the calculation, it is possible that the loading for a given variable could be noticeably different in the next release of the classifications. While this is not likely to be extreme, this can affect the impacts of changes in ranks on individual variables because variables may not carry the same amount of weight in each year. For instance, a school may determine based on the weights used in the 2015 calculation of the classifications that it needs to gain a certain number of doctorates in STEM and Social Science and increase STEM expenditures by a substantial amount. That school might implement those changes, only to find out that because of shifts in the underlying PCA that generates each school's score, those policy goals were not actually aligned with the most important variables in the updated 2018 classifications and thus changes that would have proven effective in 2015 to change categories may not be sufficient in 2018.

Another shortcoming of the Carnegie Classifications is that the PCA-based methodology removes information in a way that cannot be controlled. The creation of a single aggregate index from seven variables necessarily leads to information loss in the form of unexplained variation in the observed variables. However, using just the first principal component is not guaranteed to explain a consistent amount of variation in the underlying variables. It is possible that the data could be substantially less or more correlated in future years than in 2015; in such a case, the amount of variation explained by the single index has the potential to be much lower or much higher, respectively. Thus, there is not currently

a way to directly compare the results from one edition of the Carnegie Classifications to another.

Finally, the lack of transparency of the Carnegie Classifications makes their exact classifications difficult to replicate, even with some published work done on the subject (Kosar and Scott, 2018b). In particular, the arcs used to determine the boundaries can be critiqued both in the choice of their location (length of the diameter of the circles) and the origin of the circle (relative orientation of arcs to data points). By using two PCAs from similar information, the two indices must be combined in some fashion to create groups and the groups should split based on shared overall results from some or all of the original multivariate data. Our SEM-based method, on the other hand, is easily and reliably reproduced using the methodology outlined in this paper. Changes that substantively change the model can be assessed and even tested for across different iterations of the estimated model.

2.3 SEM-Based classifications

Kosar and Scott (2018b) outline two PCA-based alternatives to the current Carnegie method. Both use slightly different formulations of the data matrix as inputs to a single PCA and then build classifications off of rotations of the first two components. The rotations are designed to provide a single PCA with two components that can match the first PCs of the two PCAs currently in use in the Carnegie Classifications. This has the advantage of creating two indices from a single analysis; however, this method does not allow for easy diagnosis of model fit (other than percent of variation explained) nor will it directly lead to clusters with common characteristics because the new indices are forced to be orthogonal. Rather than using PCA, we propose using structural equation modeling (Bollen, 1989) to fit models with desired structures and then estimate factor scores which can then be used to classify institutions.

SEM is a statistical methodology that allows for the modeling of simultaneous equations, the use of latent or unobserved variables, and measured error of variables to be explicitly modeled. A structural equation model (SEM) consists of two parts: the latent variable model which describes the relationships among latent variables and the measurement model which relates the latent variables to their indicators or items. Compared to the PCA-based methods used in Carnegie and by Kosar and Scott (2018b), this method has two notable advantages. First, SEM allows for modeling of correlated latent factors as opposed to orthogonal ones created by PCA, allowing a single model to replace the more complex pair of PCAs necessary to calculate aggregate and per-capita indices. More importantly, SEM includes diagnostic tools that do not exist to assess the efficacy of PCA-based methods.

Initially, we tried to replicate the aggregate and per-capita indices as latent factors as shown in the path diagram in Figure 2.2. However, when fitting a SEM with aggregate and per capita productivity as latent factors, the model does not converge and fails to produce a positive definite matrix of residuals. This is due to the high correlation of the aggregate and per-capita factors (0.92, 0.88, and 0.96 for STEM, non-STEM, and research staff, respectively). These factors share nearly the same set of manifest variables, resulting in a model specification problem not unlike multicollinearity in a linear regression setting.

Because the Carnegie indices cannot be reliably reproduced in a latent-variable framework, we instead formulate an alternate latent variable structure. As depicted in Figure 2.3, we use a second order latent factor model. The first order latent factors are STEM productivity and non-STEM productivity. These latent factors are assumed to be measures of the second order factor for overall productivity. We chose to use STEM and non-STEM productivity as latent factors as opposed to two factors for aggregate productivity and per capita productivity for several reasons. First, it provides a single estimated factor that will be useful for comparing the institutions on a univariate scale. Second, this model provides a more intuitive set of factors to use to develop the classifications.

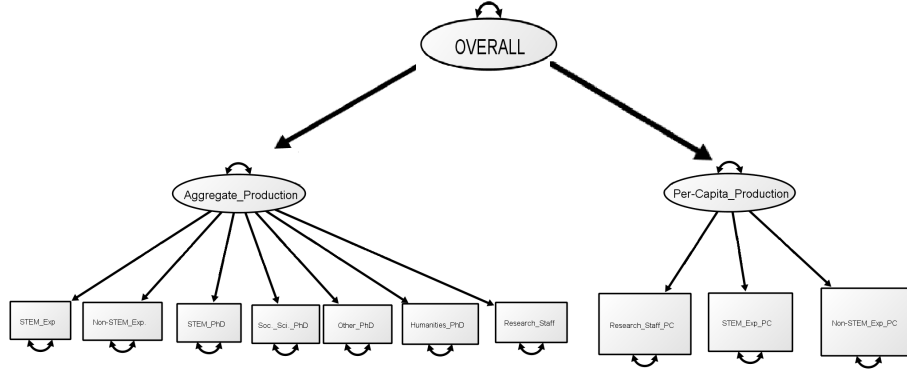


Figure 2.2: Path diagram of the Carnegie Classifications, modeling two latent factors, one for aggregate production and the other for per-capita production. All path diagrams were created using OnyxR (Brandmaier, 2016).

The second order factor models the latent STEM and non-STEM productivity factors as distinct but related concepts that can be accounted for by one underlying factor (Chen et al., 2005) of overall institutional productivity. This allows for easier interpretation of this model and allows us to obtain a single score for productivity as opposed to two scores used by the Carnegie Classifications.

The measurement model relates the STEM and non-STEM latent factors to their items. As shown in Figure 2.3, the items for STEM productivity are STEM PhDs produced, STEM expenditures, and research staff size while the items for non-STEM productivity are humanities PhDs produced, social science PhDs produced, other PhDs produced, and non-STEM research expenditures. We opted to use research staff size as a measure of STEM productivity but not non-STEM productivity as research staff are highly correlated with STEM fields and tend to be associated with labs and other physical infrastructure. As opposed to the Carnegie Classifications, we did not use per-capita measures of each variable because this does not add much independent information to the model. Instead, the number of tenured and tenure-track faculty is an item for both STEM and non-STEM productivity

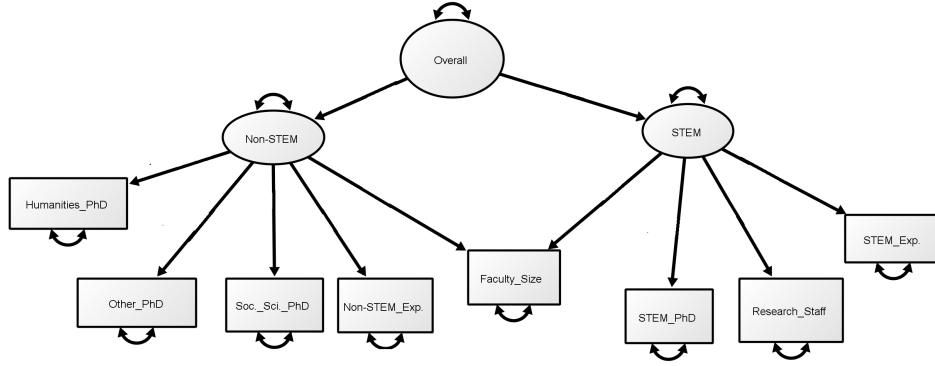


Figure 2.3: Our SEM specification, which models STEM and non-STEM production as two separate latent traits.

as the number of tenure/tenure-able faculty is clearly related to productivity in STEM and non-STEM fields (and some faculty may produce in both fields). Because of this, our model differentiates institutions by size more severely than the Carnegie methodology does.

We chose to use a specific variable for an item of a given latent factor as the choice intuitively makes sense. Further, when examining the correlation matrix of the items in Figure 2.4, the items of the latent factor for STEM productivity are highly correlated and the items for the latent factor for non-STEM productivity are also highly correlated while items of different factors are at most moderately correlated. The number of tenure and tenure-track faculty is moderately to highly correlated with items for both latent factors, with an average Pearson correlation of 0.77 for STEM factors and 0.65 for non-STEM factors. Note that none of the manifest variables were negatively correlated with each other, so variables are all oriented in the same direction, meaning larger values indicate higher productivity.

We fit the hypothesized model using R (R Core Team, 2022) using the lavaan package (Roseel, 2012). Standardized parameter estimates are displayed in Table 2.3. These results indicate that the hypothesized model fits the data moderately well (Chi-square statistic = 110.024 with 17 df, RMSEA = 0.141, CFI = 0.958); further, all the paths are supported in

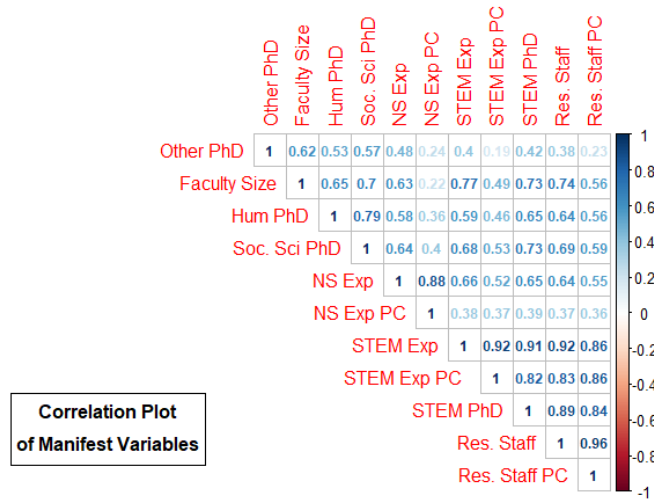


Figure 2.4: Correlations of the observed ranked variables. Some of the observed variables are highly correlated.

the models. The standardized factor loadings are all above 0.7, with the exception of number of tenure and tenure-track faculty as it cross-loads on both latent factors (factor loadings are 0.482 and 0.400 for STEM and non-STEM productivity, respectively) and number of other PhDs produced (0.639), which indicates that at least half of the variability in each of the items is explained by its associated latent factor. The largest standardized factor loadings were for the number of Humanities PhDs awarded on the non-STEM factor. For the STEM productivity variables, the standardized loadings were 0.939, 0.953, and 0.967 for the number of STEM PhDs awarded, research staff, and STEM expenditures, respectively. The path coefficients relating overall productivity to STEM and non-STEM productivity are 0.900 and 0.883 respectively. This indicates that the variability of STEM and non-STEM productivity is largely explained by overall productivity. These results are consistent with what is expected.

To compare institutions, ideally we would compare the values of the latent factor for overall productivity. Because latent factors are unobserved by definition, these values must be estimated. This is done by creating factor scores which can then be used in subsequent

	Variable	Estimate	Std.Err	Z-value	P-value	Var Est.
<i>Non-STEM</i>	Hum PhD	0.847	9.787	3.894	0.000	2612.53
	Other PhD	0.639	6.468	3.729	0.000	3835.02
	Soc. Sci. PhD	0.906	9.672	3.895	0.000	1403.25
	Non-STEM Exp.	0.729	6.978	3.913	0.000	2992.51
	Faculty Size	0.482	6.675	2.698	0.007	1908.19
<i>STEM</i>	STEM PhD	0.939	8.626	3.891	0.000	796.82
	Research Staff	0.953	8.124	4.240	0.000	629.15
	STEM Exp.	0.967	8.408	3.988	0.000	415.73
	Faculty Size	0.400	5.838	2.379	0.017	1908.19
<i>Overall</i>	<i>NonSTEM</i>	0.883	0.615	3.065	0.002	1
	<i>STEM</i>	0.900	0.634	3.260	0.001	1

Table 2.3: SEM standardized loading estimates, standard errors, test statistics, p-values, and variance estimates. Latent variables are shown in italics.

analyses (DiStefano et al., 2009). Factor scores are computed using a weighted average of the items with a number of options available for weighting. The most common method used to calculate factor scores is Bartlett’s method (Bartlett, 1937) as it leads to unbiased estimates of the true factor scores. In subsequent analyses we use the factor scores created using Bartlett’s method.

2.4 Determination of Groups

The Carnegie Classifications partitioned the schools using two indices to determine classifications for each university. Determination of the group partitions is the most subjective aspect of the Carnegie Classifications because the lines dividing groups are hand drawn each update rather than delineated in an objective fashion. In years when the data are poorly separated, a problem exacerbated by the ranking step that underlies the scores, determining an optimal place for the lines to be drawn is ambiguous at best and could cause schools to move back and forth in different classification groups without any (substantive) change in their characteristics or those around them. It is possible to consider a more objective cluster-analysis approach to identifying groups of institutions that are similar within group and different between using a method such as k-means or model-based clustering. A problem with clustering in two dimensions as would be required to use the two indices from the Carnegie Classifications or the methods of Kosar and Scott (2018) is that the clusters may not be organized along the common direction of “higher” values based on the original ranking values. In fact, some of our attempts to cluster the two-dimensional scores used in the Carnegie classifications led to clusters that were oriented from upper left (low aggregate, high per-capita) to lower right (high aggregate, low per-capita). In such a case, the groups were not tied directly to overall productivity. Unsupervised classification methods tend to identify groups in data but those groupings cannot be controlled. They may not correspond to desired groupings nor match the constrained splits used to divide

schools in the Carnegie Classifications based on more aggregate and per-capita production is better.

The SEM model described above has a notable advantage over developing two-dimensional scores because it creates a single score. This means that the schools are ordered from lowest to highest. Further, they are given a numerical value that can be used to identify groups of schools while maintaining a clear ordering to the groupings from lowest to highest on this single dimension despite using unsupervised classification.

2.4.1 Model-Based Clustering

There are dozens of algorithms for clustering data (Everitt and Hothorn, 2011), with some differences among methods becoming more apparent in higher dimensional spaces. We employ mixture modeling (also called latent variable modeling) to estimate an overall distribution based on the mixtures of these distributions as well as to identify the group memberships and quantify uncertainty in those group memberships. In this situation, identifying groups of observations in a single-dimension is particularly well-suited to considering the distribution of scores to be made up of a finite mixture of a suite of univariate normal distributions with different centers and possibly different variances. Moreover, this constitutes the key difference between the solution proposed by Kosar and Scott (2018b) and our SEM-based classifications. PCA-based solutions require clustering algorithms to operate on two (or more) dimensions as opposed to a single variable.

The basic idea for univariate (single variable) mixture modeling is to define the overall density of the n observations in the vector y as

$$f(y, \psi) = \sum_{i=1}^g \pi_i f(y_i),$$

where π_i are the mixing proportions across the g clusters and sum to 1 and $f_i(y)$ is the density of the i th cluster and ψ is a vector of parameters used to define the mixture. The

densities are assumed to be normally distributed with different means and variances that are either the same or allowed to differ across the clusters. The densities are estimated based on the observations assigned to each cluster. The challenge in this problem is to define the cluster memberships of the n observations; the EM algorithm is used to iteratively search for optimal allocations and, given a current allocation, estimate the mean, variance, and mixing rates for the different clusters. This process results in optimal choices of cluster assignments and densities of clusters for a given choice of g and whether variances are the same or differ by cluster.

We used the mixture modeling as defined in the `mclust` R package (Fraley et al., 2012) which uses the Bayesian Information Criterion (BIC) to select both the optimal number of clusters and whether the clusters are best described with the same variance or different variances. Results of this exploration are in Panel A of Figure 2.5, where the mixture model with three groups and equal variances was the optimal cluster solution based on BICs. This is fortuitous as it aligns with the number of groups used in the original Carnegie Classifications but it is possible to consider an optimal solution for a pre-defined number of clusters. The clusters themselves are displayed in Panel B of Figure 2.5, showing the non-overlapping low, medium, and high splits for the three clusters. Panel C of Figure 2.5 shows that at the boundaries of the clusters, the cluster assignments become less certain (the uncertainty is defined as 1 minus the estimated probability of the assignment of the observation to the identified cluster). It seems reasonable to assume that the classifications for the standard Carnegie results are less certain near the boundaries. Unfortunately, arbitrarily defined boundaries do not lend themselves to quantification of such uncertainty.

A critique of this approach is that we are proposing a two-step method where we first define scores on this new factor of factors and then perform clustering of these scores as opposed to directly allocating schools to different clusters from the original measurements or rankings. While a more direct approach has advantages in terms of using the original

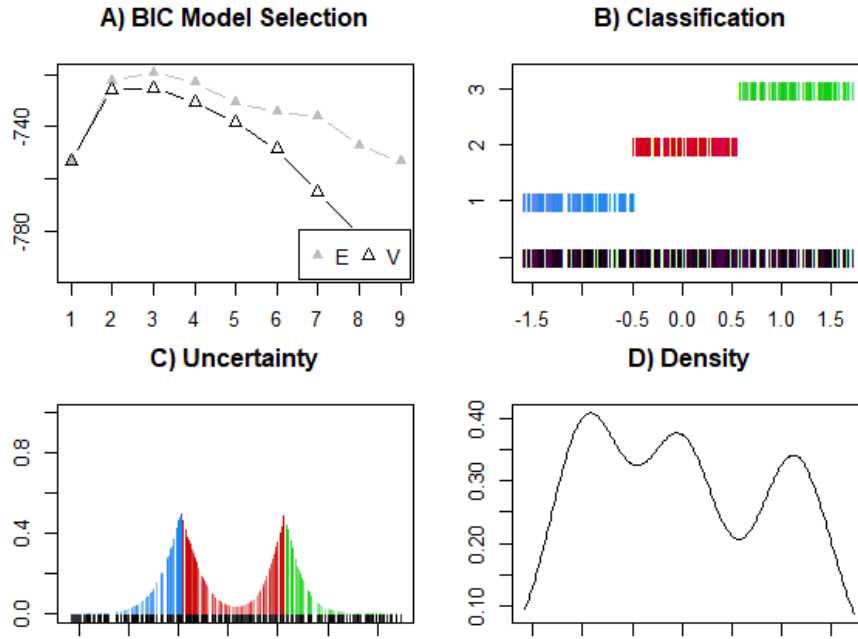


Figure 2.5: Clockwise from top left: Panel A: Plot of BIC values associated with normal mixtures. Panel B: Plot of cluster membership. Panel C: Uncertainty plot associated with each cluster. Panel D: Density plot of clusters.

information for creating groups, the unsupervised nature of the group allocation, especially in higher dimensional spaces, makes it hard to get cluster solutions that always match with expectations and desires for a school-to-group allocation model. By taking a first step to create a reasonable single dimensional scale, the mixture model is able to allocate schools into groups along this single gradient. The results that allow exploration of both the optimal number of clusters and the uncertainty of the classifications at the boundary provide tangible improvements over the arbitrariness of both choices in the standard Carnegie method. Further, it allows for group determination that is clearly better aligned with the research goals than other approaches.

2.5 Shiny Applications for Sensitivity Analysis

For any version of university ranking or classification, it is of interest to determine how sensitive both results are to changes in the underlying data, especially since institutions are implementing policies intended to change locations in Carnegie results. We independently developed a Shiny (Chang et al., 2017) application in R designed to allow the user to select a school and assess the sensitivity of that school’s classification to changes in the underlying variables for both the Carnegie and SEM-based methods and modified it to be similar in style to the one developed by Kosar and Scott (2018b). In much of the same way their application functions, the user can select a school and then use a slide bar to either increase or decrease the number of PhDs awarded in each category, research staff size, or research expenditures. Sliders can be reset to the true value by pressing the reset button. In the vein of the stated goals of the Carnegie Classifications, the application also allows for identification of a five-school cohort of similar institutions. For administrators interested in identifying a cohort of similar institutions, the application prints a list of the five closest schools to the institution of interest.

Figure 2.6 demonstrates how this application (found at: <http://ccsemclassifications>).

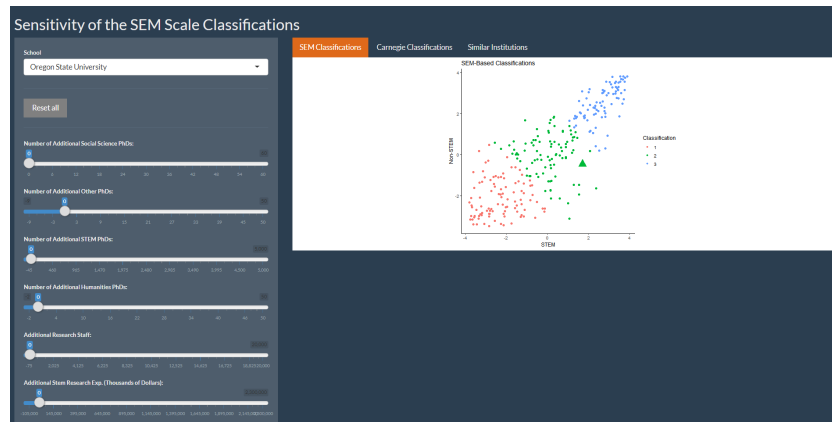


Figure 2.6: Screenshot of the SEM-classifications application assessing Oregon State University

`shinyapps.io/sem_app`). can be used by administrators and institutional researchers to assess the efficacy of a proposed policy change. As an example, we highlight Oregon State University in Figure 2.6, which is classified in the second group in the SEM-based result.

Changes can be made to either a single variable, all of them, or a selected few. The application takes the user input and re-calculates the PCA-based indices or SEM-based results on the new dataset, depending on which tab in the app is being used. It then shows where the university would be relative to other schools in that update. In Figure 2.7, we show the effect of substantive changes to the counts of PhDs awarded at the selected institution. If Oregon State had awarded an additional 10 PhDs in Social Science and Other categories, as well as 20 STEM PhDs and 10 Humanities PhDs, their score on the univariate factor of factors would be high enough to place them in the top group. The path to improvement is different for each institution, especially because moving into and out of ties can lead to substantive changes in both the SEM classifications and the Carnegie Classifications.

The application also gives a table of the selected institution’s nearest neighbors for a comparison peer set. Schools with slightly higher factor scores are considered “aspirational” peers and neighbors with slightly lower scores are considered “peer” institutions.

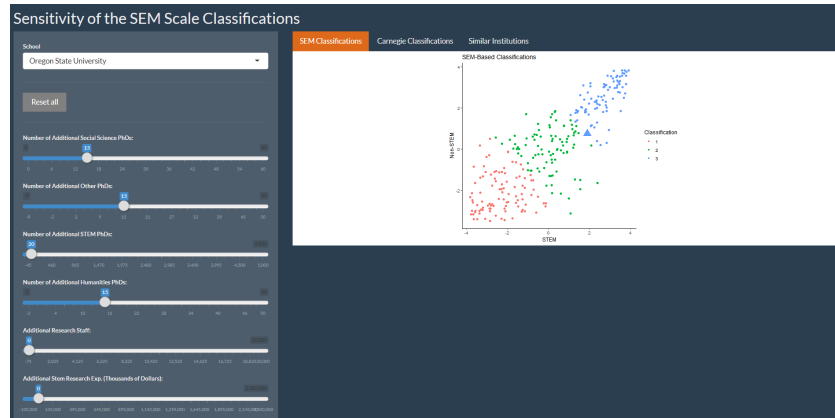


Figure 2.7: SEM classifications after adding PhDs

For administrators and decision makers at institutions, an interactive application allows users to explore and assess sensitive spots in both classification systems and inform institutional policy. Administrators can test hypothetical policy actions to determine their efficacy on both classification systems. Moreover, the applications can be used to compare the sensitivity of changes in the underlying variables across classification systems. The availability of all code and data underlying app makes the results transparent and reproducible and would make updating the results to a new dataset relatively trivial.

2.6 Discussion

2.6.1 Comparing Carnegie vs. SEM Classifications

It is similarly useful to illustrate differences in classification between the Carnegie Classifications and our SEM analogue. While they function in similar ways, the SEM classifications do not deal with institutional size in the same way that the Carnegie Classifications do; rather, the SEM classifications more harshly penalize smaller institutions. This drives separation in institutions (a goal of the analysis) but does tend to put smaller schools farther away from large ones.

Table 2.4 gives the institutions that saw the most dramatic change in classification from

the Carnegie to the SEM Scale. None of the institutions that were categorized as R1 in the 2015 Carnegie Classifications were categorized by the SEM in the lowest group; however, 31 of them were moved into the middle tier. Within the R2 schools, 46 institutions were moved into the lowest tier of the SEM classifications. All 54 R3 institutions were classified in the lowest tier of the SEM classifications. This implies that the SEM classifications tended to penalize institutions; none improved their standing relative to their Carnegie Classification. Some notable R1 institutions that are classified in the middle tier SEM classification include: Rice University, Carnegie Mellon University, University of Oregon, and Tufts University, all of which are near the borderline between R1 and R2 in the Carnegie Classifications.

	R1	R2	R3
SEM Low	0	46	54
SEM Middle	31	61	0
SEM High	84	0	0

Table 2.4: SEM classifications vs. Carnegie Classifications indicate some disagreement about middle-tier institutions.

2.6.2 Problems Addressed by the SEM Classifications

The rotated solution of Kosar and Scott (2018) is a notable improvement over using two separate PCAs to generate the scores for the Carnegie classification system that retain similar definitions while directly assessing the overall percentage of variation. However, this approach does not necessarily lead to easily defined groupings any more than the original Carnegie methodology. We employed SEM methods to first assess an equivalent approach using the same structure their target loadings and the two separate PCAs of the Carnegie classification target - but found that there is too much shared information to estimate that structure using SEM. Instead, we propose to separate STEM and non-STEM components

where possible and share information where it is not, and then create a factor of factors to map these two correlated but somewhat independent factors into a single scale that can be easily and explicitly used for classification.

SEM-based classifications do not solve every shortcoming of the Carnegie Classifications. They rely on the same data and are therefore subject to the same problems created by the use of snapshot data rather than multi-year averages. In the same way that the PCA loadings can change from one release to another, the SEM's weights relating manifest variables to latent traits can also change. However, unlike PCA, SEM has a broad set of tools for diagnosing model fit that can be utilized to determine degradation of the model over time. It is also able to test for different structures of loadings in two datasets, such as between 2015 and the upcoming 2018 data.

The SEM-based classification method allows for determining classifications based on a single factor-of-factor score rather than on hand-drawn delineations. While any automated method for determining both the number of clusters and cluster membership has the possibility of selecting either an overly complicated solution or too few groups, the mixture model resulted in a reasonable three-group solution. It is also possible to fix an optimal number of clusters if necessary; however, we did not need to do so. The ability to measure uncertainty in institution's classifications is also beneficial. For administrators at schools on either side of a partition, it is useful to know if their institution's classification is highly uncertain or if it is relatively well-classified in a certain group. Further, while one could apply bivariate clustering algorithms to the results from the Carnegie Classifications (e.g. Kosar and Scott (2018)), many of the two-variable clustering methods give results that are not well-behaved relative to the research goals.

Finally, this method of institutional classification is well documented and reproducible. It can be applied to new datasets and consistently compared. The Shiny applications created to assess the sensitivity of each classification method can be used to identify a cohort of

similar institutions, determine key differences between institutions, and predict the outcome of certain policy actions.

2.6.3 Further Research

This research addresses a statistical question, not a qualitative one. The Carnegie Classification of doctoral institutions has changed over time both in their explicit efforts to deemphasize hierarchical interpretations by institutions as well as their interest in capturing the more nuanced differences between institutions and the classes that they fall within, and it is likely that they will change in the future. In fact, the Indiana University Center for Postsecondary Research has already announced that the 2018 update “will reshape membership of the Doctoral Universities and Master’s Colleges and Universities categories... to accommodate ‘Doctor’s degree: professional practice’ within their methodology” Carnegie Foundation for the Advancement of Teaching (2019). The efficacy of including different variables into the SEM-based classification system is not something that we assessed; however, the SEM method would intuitively allow for addition of a new items and possibly a third latent factor, such as related to professional doctorates.

Other methods could be applied to the classification stage. Mixture modeling is not the only method for determining clusters with univariate data, even though it does provide a reasonable solution here. A one-step solution that combines building the latent factors and binning the factor-of-factor scores into three groups, if it were possible, would be ideal. Further work could focus on comparing different univariate clustering on the SEM scores or even focus on bivariate clustering algorithms that are constrained to provide more reasonable groups with the Carnegie indices.

CHAPTER THREE

A METHOD FOR DIAGNOSING INFLUENCE IN HIGH-DIMENSIONAL MAPPING
TECHNIQUESContribution of Authors and Co-Authors

Author: Harmon, Paul

Contributions: Primary Author responsible for the majority of writing and development of analysis and methods used.

Author: Greenwood, Mark C.

Contributions: Provided feedback and direction on statistical methods and drafts of manuscripts.

Manuscript Information

Paul Harmon and Dr. Mark Greenwood

Journal: *PLOS One*

Status of Manuscript:

- ☒ Prepared for submission to a peer-reviewed journal
- ☐ Officially submitted to a peer-reviewed journal
- ☐ Accepted by a peer-reviewed journal
- ☐ Published in a peer-reviewed journal

Abstract

It is often of interest with multivariate or high-dimensional data sets to represent the high-dimensional data in a two or three-dimensional map. Methods range from classical multidimensional scaling and related tools to more recent developments such as t-distributed Stochastic Neighbor Embedding. Across most mapping methods, it is possible for an observation to have an undue impact on the shape of the resulting map or ordination; such a point would be considered influential. This paper focuses on a method for diagnosing influential points based on the impact that each point has on the resulting map when left out singly. Utilizing Permutation-based Multivariate Analysis of Variance (PERMANOVA), we propose a pseudo-F statistic-based diagnostic that can be utilized for any type of map. We present simulation studies to assess the efficacy of the diagnostic for various mapping methods and across different data structures, and compare the effectiveness of the diagnostic across several real data sets. A specific focus on challenges related to using the diagnostic method for t-SNE maps, along with discussion on how t-SNE should be used in the presence of potentially influential points, is included as well.

3.1 Introduction

Consider the problem of visualizing (potentially high-dimensional) multivariate data. It can be useful to generate a 2 or 3-D mapping (ordination) of some higher-dimensional space, projecting the high-dimensional data into an ordination that can be directly visualized so that structure that might be hard to detect in many dimensions is made more clear in the lower-dimensional map. There are many methods that exist to accomplish this task, ranging from tools such as classical multidimensional scaling (MDS) to more modern tools like t-distributed Stochastic Neighbor Embedding (t-SNE), among others.

In a regression model, the estimated coefficients or predictions might be affected in

meaningful fashion by the inclusion or exclusion of a single point—such points are referred to as influential (Cook, 1977). A suite of tools have been developed to identify and measure the impact of influential points on the results of regression models. Mapping tools, on the other hand, have a different purpose: they are used to explore and visualize the structure of data whereas regression is concerned with inference and prediction of a chosen outcome variable. Unlike regression or classification, many tools for dimension reduction and higher-dimensional data visualization are unsupervised, meaning that they do not take advantage of some marked response variable. Instead, they capitalize on the signal present just in a set of features and produce a result driven only by the features being visualized. Despite this difference, mapping tools such as MDS are also susceptible to being influenced by aberrant data values, much in the same way that a regression model might be susceptible to the impact of influential observations.

The goal of this work is to develop a diagnostic that can be used to help identify influential points for mapping methods and potentially quantify how much they impact the resulting mapping, no matter the version of mapping that was used. Additionally, this method can be used to assess the sensitivity (or, in some cases, insensitivity) of each method to influential points and the impact that this sensitivity might have on how an end-user might want to interpret the resulting map produced. Note that while robustness to influential points might at first seem like a positive trait for an ordination method, masking distinctly different observations in a mapping may have negative consequences as well.

This paper overviews the novel statistical methodology we have developed to identify influential points that, when excluded from a data set, can meaningfully impact the shape of an ordination. We overview our methodology step by step, present results from simulation studies that show meaningful evidence of our method’s ability to detect influential observations, and highlight some mapping techniques where our method is more/less impactful.

3.2 Tools for High-Dimensional Data Visualization

3.2.1 Multidimensional Scaling

MDS was initially developed by (Torgerson, 1952), and takes as input a distance or dissimilarity matrix representing the distance between pairs of observations and returns locations for points in some selected lower dimensional space, or map. This distance does not have to be Euclidean (although it often is). MDS returns maps of points that are interpreted in a Euclidean space; in general, the more dissimilar two observations are in the high-dimensional space, the farther apart they should be in the resulting lower-dimensional ordination.

Consider a multivariate data set $\mathbf{X}_{N \times P}$, where N is the number of observations and P is the number of features. For distinct pairs of points, x_i , and $x_{i'}$, define $d_{i,i'}$ as the interpoint distance or dissimilarity between the x_i and $x_{i'}$. Generally, MDS attempts to create a map that positions map points to align with distances in the high-dimensional space by minimizing a discrepancy function which is defined as

$$S_M(x_1, x_2, \dots, x_N) = \sum_{i \neq i'} (d_{i,i'} - \|z_i - z_{i'}\|)^2, \quad (3.1)$$

where $z_i, z_{i'}$ refer to the i^{th} and i'^{th} points in the lower-dimensional representation (map) and $\|z_i - z_{i'}\|$ are approximations of dissimilarities in mapping spaces using the Euclidean distance measure. MDS is related to PCA; it relies on the same singular value decomposition as PCA and can be expressed equivalently as PCA when the original distances are Euclidean and variables are scaled equivalently (see Gower (1966) and Everitt and Hothorn (2008)). However, MDS is more generally applicable because it can be applied to any dissimilarity or distance measure and only requires the distance or dissimilarity matrix, $\mathbf{D}$.

3.2.2 Sammon Mapping and IsoMDS

Various alternatives or improvements to MDS have been developed since it was proposed and developed in the 1950s. Several such variations are non-metric, meaning that rather than representing the actual distances (or parametric functions of those distances), they retain information about the ordinal ranks of the dissimilarities (Borg and Groenen, 1997). Such non-metric MDS methods rely on the calculation of a different objective function, also called a *Stress* metric (Cox and Cox, 2008), and may relax assumptions to varying degrees about relationships between the high-dimensional distances and their lower-ordered approximations in the map. Kruskal’s non-metric MDS, also known as isoMDS, attempts to arrange the resulting z_i and $z_{i'}$ map points to minimize the following stress function,

$$Stress = \sqrt{\frac{\sum d_{i,i'} - ||z_i - z_{i'}||}{\sum d_{i,i'}^2}}, \quad (3.2)$$

where $||z_i - z_{i'}||$ is the position of the points in the map that reduce *Stress* under the constraint that the rank orders of the distances between points remain the same both in the ordination and the high-dimensional space (Kruskal, 1964).

Sammon mapping is a non-linear method that extends MDS to scale the stress metric by the pairwise distances $d_{i,i'}$, attempting to better maintain local structure by equalizing the impact of large and small distances (Sammon, 1969). This method further improves on MDS by better focusing on local structure (Ghojogh et al., 2020). The objective function for Sammon mapping is defined for the function $\phi(Z)$ which is referred to as either Sammon’s Stress or Sammon’s Error and takes the form

$$\phi(Z) = \frac{1}{\sum_{i,i'} d_{i,i'}} \sum_{i,i'} \frac{(d_{i,i'}^2 - ||z_i - z_{i'}||^2)}{d_{i,i'}}, \quad (3.3)$$

where z_i and $z_{i'}$ are the mapped points.

For MDS, Sammon mapping, and isoMDS, an iterative algorithm like Gradient Descent

is typically used to optimize the *Stress* metric for each nonmetric mapping technique (Cox and Cox, 2008). As such, the choice of parameter (z_i 's) initializations can impact the final model result.

3.2.3 Additional Mapping Techniques

This paper considers only a few of many different methods for creating maps from high-dimensional data. They range in terms of complexity, with some adding relatively minor changes to the MDS structure to those that share similarities with other branches of mathematics, such as topological data analysis. While the focus of this paper is on more classical techniques, there are various additional methods for generating lower-dimensional representations of high-dimensional data.

A more recently developed tool, Stochastic Neighbor Embedding (SNE - see Hinton and Roweis (2002)), bases maps on two measurements of similarity, $p_{i,i'}$, the similarity in the high-dimensional space, and $q_{i,i'}$, the similarity in the low-dimensional space. These metrics of similarity are based on joint probability distributions represented by Gaussian distributions, and are related to each other to create a resulting map. An extension to SNE called t-distributed Stochastic Neighbor Embedding (t-SNE, see Van Der Maaten and Hinton (2008)) improves the method by utilizing a t-distribution with 1 degree of freedom (also known as a Cauchy distribution) for the low-dimensional similarity metric, $q_{i,i'}$. More detail on t-SNE is given in Section 3.7.

Our influence diagnostic can be extended to be used on any mapping technique that produces points in lower dimensional space, but seems to be most effective for identifying influential observations for MDS and similar methods. Some of the methods may not have issues with influential points affecting the shape of their resulting ordinations due to the way that they are constructed; however, these explorations highlight some interesting behaviors of the methods as discussed in Section 3.7.

3.3 Defining “Influence” in High Dimensional Maps

3.3.1 Cook’s Distance

Belsley et al. (1980) define an influential point as a data point “which, either individually or together with several other observations, has demonstrably larger impact on the calculated values of various estimates (coefficients, standard errors, t-values, etc.) than is the case for most of the other observations.” In regression modeling, as in most statistical methods, care must be taken to properly distinguish between an outlier and a so-called influential point. An outlier is typically denoted as a point that is either much larger or smaller than similar observations on a specific scale. However, an outlier may not necessarily be influential; a truly influential point must be outlying enough relative to similar observations that it impacts the fit of the regression line or other important aspects of the model, depending on the definition of influence. The identification of influential points that can impact the results of a model is well-established.

Cook (1977) proposed a measure of influence called “Cook’s Distance” that has since become one of the most common tools for diagnosing influential points and distinguishing them from simple outliers. The idea is predicated on the concept that by leaving an out an influential observation, the results of the model should change. This process of jackknifing observations is done for all the observations in the data. Specifically, Cook’s Distance, denoted C_i for the i^{th} observed data point, is defined as

$$C_i = \frac{\sum_{j=1}^n (\hat{y}_j - \hat{y}_{j(-i)})^2}{Ps^2}, \quad (3.4)$$

where $\hat{y}_j$ is the complete-data fitted value for the j^{th} observation, $\hat{y}_{j(-i)}$ is the fitted response value for the model with the i^{th} observation omitted, s^2 defined as the mean-squared error for the model with all observations used, and P , the number of predictor variables. Cook’s Distance, C_i , asymptotically follows an F-statistic from an $F_{P,N-P}$ distribution, the method

does not typically use p -values to determine how “influential” a point is. Rather, the F-statistics themselves are typically compared to cutoff values, or observations with relatively large Cook’s Distance values are considered for investigation of influence based on “rules of thumb”. It is difficult for a single observation to exert a large degree of influence when the number of observations is large, and typically in those cases, points with large F-statistics must be increasingly abnormal.

3.3.2 Identifying Influential Observations in Maps

Similar to a regression setting, the resulting maps produced by ordination methods may be impacted by the inclusion or exclusion of points. In short, an “influential” point in a lower dimensional mapping can be defined as one that, when excluded, drives meaningful differences in the map as compared to when that point is included. An outlier, on the other hand, might look different in the high-dimensional space (or on a subset of features) from other observations, but may not meaningfully change the shape of the map when left out. Thus, an outlier may not be an influential observation in a particular mapping method. It is important to note that maps are arbitrary in location, scale, rotation and orientation, so our measure of influence needs to focus on changes to configurations of maps that are invariant to these arbitrary changes. We focus on the relative comparison of inter-point differences in the lower-dimensional mapping to assess changes in maps.

Jolliffe (2002) overviews tools for identifying influential observations in PCA, which is a dimension-reducing technique that is related to MDS. Influence functions (Critchley, 1985; Hampel, 1974; Huber, 1981) have been defined for loadings in PCA. However, most mapping techniques are not based on linear combinations of the original variables (and their estimated coefficients) as in PCA, so the influence assessment needs to be based on the mapped points.

Much of the literature around robustness of mapping results and multivariate outliers centers on dealing with the problem by modifying the dimension reduction technique to be

resistant or robust to impacts of influential points rather than identifying observations that impact the shape of the resulting map for a given method. Blouvshtein and Cohen-Or (2018) analyzed the effect of outliers on the maps produced by MDS and Sammon mapping. In general, neither method is particularly robust to the effects of multivariate outliers. They propose a method to detect outliers prior to the generation of the selected map so that those points can be removed from the data. Blouvshtein and Cohen-Or’s method involves treating the pairwise distances between points in the lower-dimensional space as edges of triangles that can be formed by every $\binom{N}{2}$ pairs of points. Generally, if one of the two points is an outlier, the edges d_1 , d_2 , and d_3 will not satisfy the triangle inequality (i.e., $d_1 + d_2 \geq d_3$), whereas inliers tend to satisfy it. They recommend a rule to examine the histogram H of b , defined as the number of edges of broken triangles created by assessing all the possible pairs of points, and generate a threshold that is intended to identify inliers vs. outliers so that outliers can be removed prior to making the final version of the map.

Two other papers, Forero and Giannakis (2012) and Kong et al. (2019), frame the problem in a slightly different way. They consider a penalized *Stress* metric for optimization, utilizing regularization to better identify outlier points. In both cases, they are able to identify outliers using a majorization-minimization procedure that iteratively produce more outlier-robust maps of the lower-dimensional embedding of interest.

Alternatively, archetypal analysis (Cutler and Breiman, 1994) can be used to identify outlying points in a multivariate space. The method approximates a convex hull that lies around the boundary of a data set based on designated “archetype” points; these extreme points have been used to identify potential multivariate outliers (Potts, 2021) in the general P -dimensional space. While this method may be effective at identifying and categorizing extreme points on the boundary of the multivariate space, it does not necessarily answer the question of whether or not the inclusion or exclusion of those points might change the shape of a resulting lower-dimensional representation. This is the problem we seek to answer with

our method.

3.3.3 Permutational Multivariate Analysis of Variance

Although the previous methods give rise to identifying extreme points in maps, they do not clearly inform the influence of a given observation on the resulting shape of the ordination. Since the goal of this research is to present a methodology for identifying and quantifying influential observations in maps, our goal is to extend the ideas that underlie Cook’s Distance from a regression framework to a mapping setting. This necessitates the use of distance matrices rather than raw observations, and requires the use of a distance-based extension of multivariate analysis of variance (MANOVA) methods to generate a partitioning of variation that can produce a statistic similar to the F-statistic used in Cook’s Distance.

Permutational multivariate analysis of variance (PERMANOVA) is a tool that allows for partitioning variation across multivariate data in a distance-based framework and assessing hypotheses using a permutation-based approach (see Anderson (2001) and Anderson (2017)). In other words, it allows for generalization of ANOVA-type tools on multivariate data sets with a given choice of distance/dissimilarity. Much like parametric multivariate extensions of ANOVA, PERMANOVA allows for testing of no difference in population means simultaneously across all P response variables, across either quantitative or categorical explanatory variables.

A critical difference between parametric MANOVA and PERMANOVA is that the latter method makes no assumption of multivariate normality. It only requires an assumption of exchangeability of permutable units (observations) under the null hypothesis and only requires the dissimilarity matrix and the design matrix (Anderson, 2001). For simplicity and its usage below, we focus on using PERMANOVA for two-group comparisons in our P -dimensional space. PERMANOVA relies on the use of permutation tests of pseudo-F test statistics to generate p -values, because the null distribution of the pseudo-F test is not

usually known. In a similar vein as ANOVA, PERMANOVA requires the calculation of among-group sum of squares (SS_A) and residual within-group sum of squares (SS_W). Using the notation of Anderson (2001), the total sum of squares (SS_T) is calculated as

$$SS_T = \sum_{i=1}^{N-1} \sum_{j=i+1}^N \frac{d_{i,j}^2}{N}. \quad (3.5)$$

Partitioning of variation can be obtained via subtraction of the residual sum of squares SS_W from the total sum of squares (as in the sums-of-squares decompositions in ANOVA), which is defined as the sum of each of the groups' squared distances between the observed data and the group centroid, as defined below for the l^{th} group as

$$SS_w = \sum_{l=1}^g W_l, \quad (3.6)$$

where $g = 2$ is total of number of groups present. The W_l is defined as

$$W_l = \sum_{i=1}^{N-1} \sum_{j=(i+1)}^N \epsilon_{ij}^{[l]} \frac{d_{i,j}^2}{n_l}, \quad (3.7)$$

with n_l defined as the the group sample size (such that $\sum_{l=1}^g n_l = N$) and where $\epsilon_{ij}^{[l]}$ is a binary indicator between the i^{th} and j^{th} samples that equals 1 when samples are in the l^{th} group and 0 otherwise.

The resulting partitioning of $SS_T - SS_W$ allows for calculation of a pseudo F-test statistic (which is a multivariate analogue for a standard F ratio). This tests the null hypothesis that there are no differences in the positions of the group centroids in the distance matrices as

$$F = \frac{SS_A}{g-1} / \frac{SS_W}{N-g}, \quad (3.8)$$

where N is the total number of observations and g (here it is always 2) is the number of

groups.

P-values are then obtained in a permutation-based method, meaning that they are directly calculated based on the number of observed permutations that are more extreme than a given result. The p -value, then, is calculated as follows, where k is the number of permutations (either a random sample of all possible permutations or, when feasible, all possible permutations as used below), k^* is the number of observed values that are greater than the observed pseudo-F statistic, f_{obs} , and the p -value generated as $\frac{k^*+1}{k+1}$ for a random sample of k permutations and is $\frac{k^*}{k}$ when generated across an exhaustive search of all possible permutations.

All PERMANOVA models were performed using the R package **vegan** (Oksanen et al., 2020). Because of the limited number of unique permutations possible below, we will focus on the calculation and interpretation of the pseudo-F statistic.

3.4 Method Overview

We propose a method here for identifying influential observations that is agnostic to the mapping method. In addition, our method follows a philosophical framework similar in flavor to Cook’s Distance for mapping methods; however, it relies on a different set of tools than its regression analogue.

We propose holding out each observation and observing those impacts, calculating hold-out maps using a jackknife resampling approach, where each of the $1, \dots, N$ observations is held out, one at a time. Rather than basing the test on a standard F distribution, as is Cook’s Distance, our method utilizes permutation-based, multivariate ANOVA (PERMANOVA) tests to calculate pseudo-F test statistics. Our method borrows from some of the Mantel test (Mantel, 1967), which is a method for comparing two distance matrices by vectorizing the pairwise dissimilarities and using the Pearson correlation of those pairwise dissimilarities to assess how similar two maps are. The correlation provides a map similarity measure that

is invariant to the arbitrary aspects of different maps made on the same observations. In Mantel’s test, permutations are used to test a null hypothesis of no relationship among two maps.

In general, we seek to identify an extreme observation which, when left out, causes large differences in the shape of the resulting ordinations and changes the relative inter-point difference in the maps. The methods rely on a few different ideas. Figure 3.1 shows four MDS plots with observations held out; the lower-right Panel labeled “61” removes the aberrant point 61. Visually, the shape of each group within the map appears slightly different with only point 61 removed, implying that the point-to-point distances in the map have changed slightly, even though the map still conveys general three-group structure. In particular, observation 45 appears to move from the edge of its group towards the center and the relative orientation across the three groups shifts. In the case of the other three observations removed, the point-to-point distances appear unchanged, even when the removal of observation 60 leads to a reversal in the second dimension’s direction that does not substantively impact the structure of the groups.

For the method, first a way to quantify inter-point distances is needed. Dissimilarity matrices contain information on point-to-point distances (or dissimilarities) and, when vectorized, the Pearson correlation of the distance matrices measures similarity of two configurations by comparing the similarities in the different resulting map distances. Because of the leave-one-out approach, each map is missing one observation and its pairwise distance in the resulting correlation exploration. A simple example is used to illustrate this. We vectorize distance matrices in the following way, where $d_{3,2}$ refers to the distance between the second and third observations in the map and $d_{3,2,[-1]}$ is the distance between observations 2 and 3 with observation 1 removed. The example distance matrix looks like the following:

<https://www.overleaf.com/project/61e4ef3266cd5f37509c3b1f>

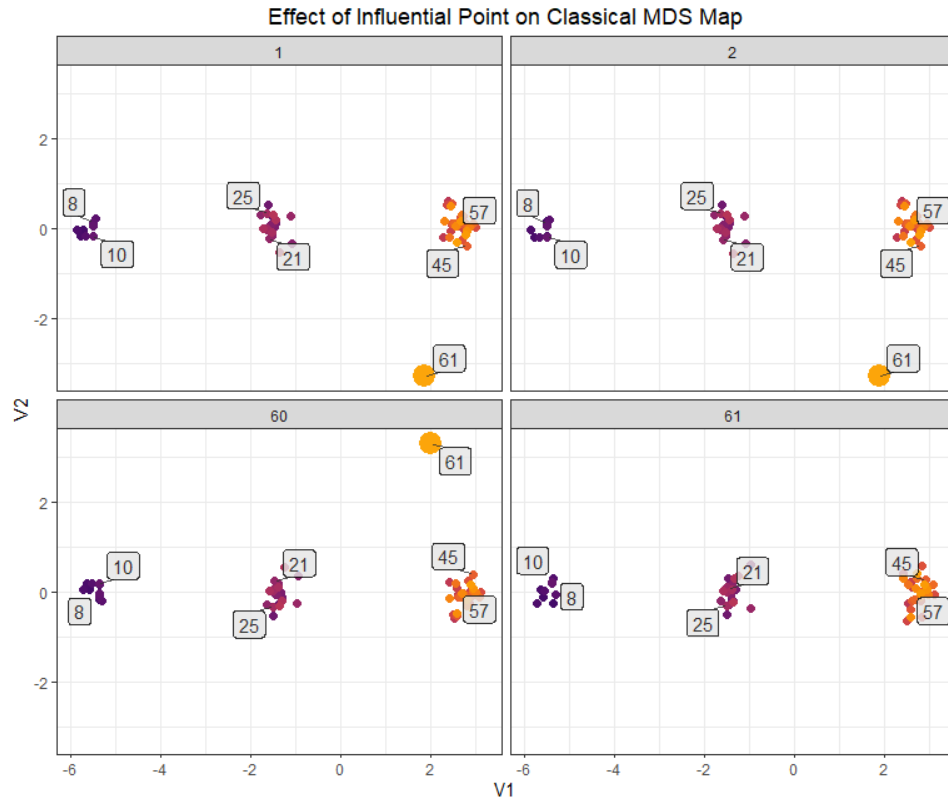


Figure 3.1: MDS maps for maps with four different observations held out. The lower-right Panel (“61”) provides the map with the potentially influential point removed whereas the maps for points “1”, “2”, and “60” show the maps when those points are removed, respectively. The structure when observation 61 is removed changes somewhat, as demonstrated by slight shift in relative position of observation 45 to its group and changes to “angles” within each of the groups, but only minor changes in the shape of the maps occur when observations 1, 2, or 60 are removed (despite a reversal of the y-axis when observation 60 is held-out).

$$\mathbf{D}(\mathbf{X}_{3 \times 3}) = \begin{bmatrix} NA & NA & NA \\ NA & 0 & d_{2,3,[-1]} \\ NA & d_{3,2,[-1]} & 0 \end{bmatrix}, \quad (3.9)$$

when the first observation is held out. The column-vector form then would be:

$$\overrightarrow{\mathbf{D}(\mathbf{X})} = \begin{bmatrix} NA \\ NA \\ NA \\ NA \\ 0 \\ d_{3,2,[-1]} \\ NA \\ d_{2,3,[-1]} \\ 0 \end{bmatrix}. \quad (3.10)$$

A correlation matrix of the vectorized distances can be computed using the pairwise complete observations to leverage all available information to estimate correlations between each of the vectorized distances in the presence of some missing data values. This is required because the maps we will compare involve withholding target observations which cause their dissimilarities to be unmeasured. Figure 3.2 visualizes the missingness pattern for the case where the third observation is held out (nonzero distances are shown across a color gradient with purple being the smallest and blue, green, yellow being the order of larger distances, respectively).

Similarities (or correlations) can then be converted to distances; one such metric for doing this is $d_{i,i'} = \sqrt{2(1 - r_{i,i'})}$ (James et al., 2013) where $r_{i,i'}$ is the Pearson correlation between i and i' , the vectorized distance matrices. Together, this generates a $N \times N$ distance

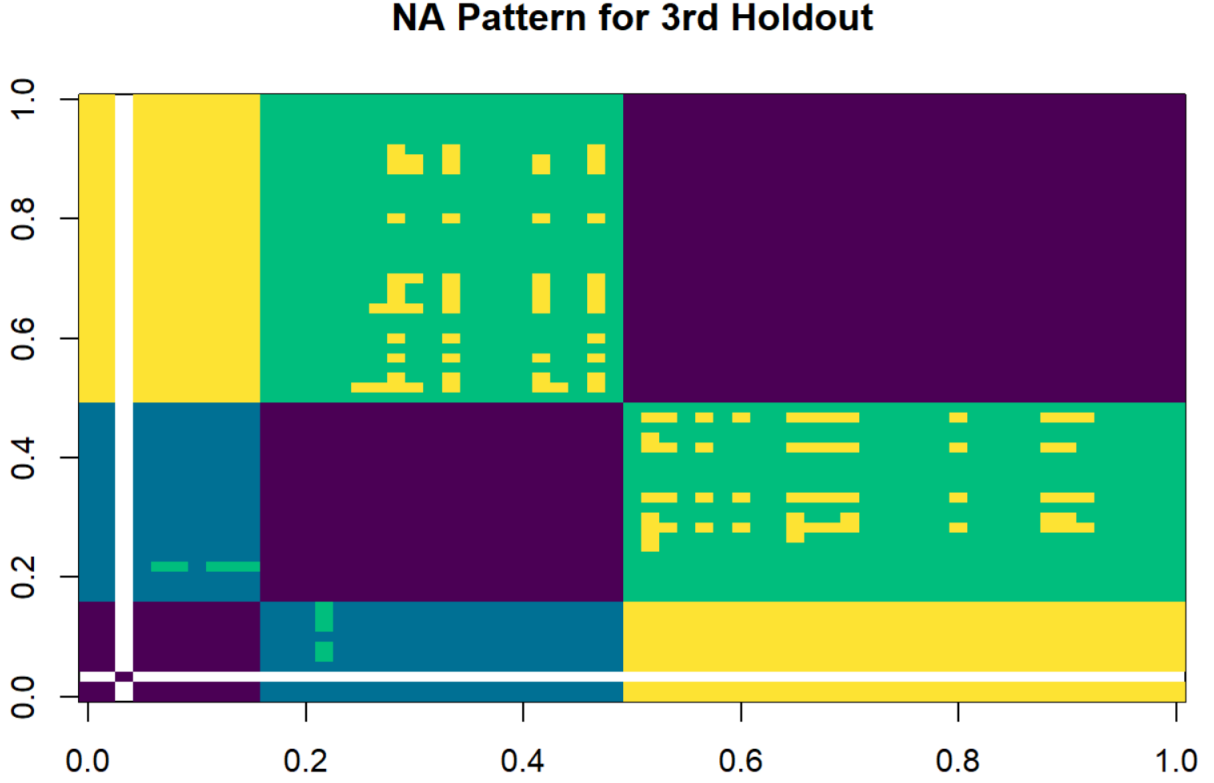


Figure 3.2: Plot of dissimilarity matrix for map ($N = 61$) with the third observation held out. White space refers to missing observations. Purple refers to the smallest distances with green, blue, and yellow distances being larger.

matrix $\mathbf{D}^*$ among the N hold-out maps.

We can then utilize PERMANOVA to assess the evidence of the differences in our resulting maps for each observation when it is removed. This step borrows from the concept of the Bonferroni Outlier test diagnostic (see Weisberg (2014)), which uses a predictor variable that is a binary indicator where a single row is one and all others are zero. The Bonferroni Outlier Test fits separate linear models for each observation, collects the N t -statistic p -values, and adjusts the p -values using a Bonferroni correction of $\min(pval * N, 1)$. We replace the t -test with a pseudo-F statistic and the t -based p -value with a permutation-based p -value.

Our method tests whether or not the configuration of the map without the held-out

point is different from maps with other single points removed. This involves the simultaneous generation of N tests, one for each of the held-out observations. However, by construction of the binary indicator variable we use in our method, each of the possible permutations are the same for each of the N tests. This yields a single permutation distribution for all of the tests that limits the smallest p -value that can be observed to $\frac{1}{N}$.

Figure 3.3 shows the permutation distribution generated for a series of MDS maps shown in Figure 3.1 as well as one for t-SNE maps built from the same data. Note that the potentially influential point has an extremely large pseudo-F statistic for the MDS map so for that map, the pseudo-F statistics are shown on a log-scale; for t-SNE, the test-statistic values are much smaller and can be shown on their original scale. It is possible to obtain negative values very close to zero in the permutation distribution when using R’s **vegan** package for PERMANOVA—and has been seen in other uses of PERMANOVA when working with non-Euclidean distances (Fletcher and Underwood, 2002). Therefore, the pseudo-F values shown in Figure 3.3 for MDS are adjusted by a small amount 0.01 to make them positive in order to allow for the log transformation; the pseudo-F statistic values for the t-SNE plot are not adjusted as they are all positive. The p -values generated from the permutation are calculated as the number of more extreme observed pseudo-F statistics than the observed value - thus, for both maps, the possible p -values are the same. Therefore, the p -values provide a relative ranking of pseudo-F statistics that matters more than the individual p -value; the most extreme points in two different maps will share the same p -value even if the test statistic values for that observation are very different relative to their respective permutation distributions.

Our influence measure approach is as follows:

1. For each observation in the data set, remove it, and generate a map in the selected number of dimensions (usually two) from the resulting $(N-1)$ observations for a selected ordination method. (For accounting purposes, it is useful to keep the $N \times 2$ structure

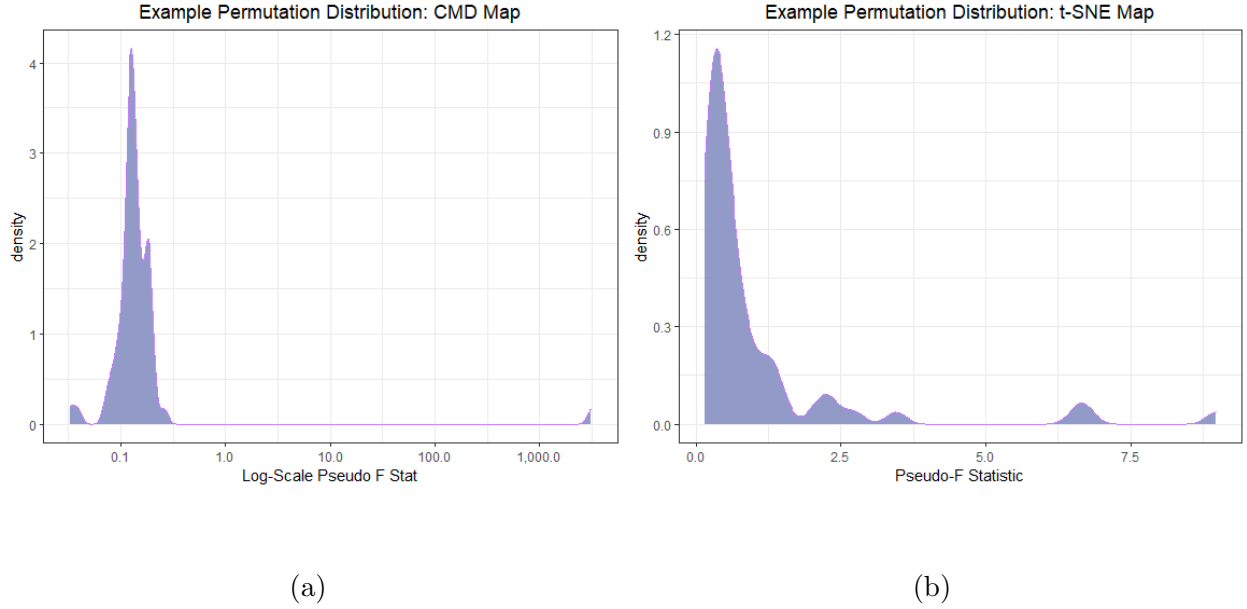


Figure 3.3: Permutation distributions (Panel (a)) shown for the MDS maps shown in Figure 3.1 and the Panel (b) shows permutation distribution for t-SNE maps.

but replace the holdout observation map locations with NA).

2. Generate a Euclidean distance matrix of the $N \times 2$ matrix from the new mapped points and vectorize it. Repeat for each of the N held-out observations.
3. Generate a pairwise-complete Pearson correlation matrix from the combination of these vectorized distances that measures similarity of distances (for pairs where information is available).
4. Convert that correlation matrix (of all the hold-out vectorized distance matrices) into a distance matrix among the maps using the $d_{i,i'} = \sqrt{2(1 - r_{i,i'})}$.
5. Generate the pseudo-F statistics for each holdout observation. This means that on a 100-observation data set, 100 individual tests are performed where each test uses an independent variable that is equal to 1 for the index of the holdout observation and 0 elsewhere. The hypotheses for this test statistic are specified as:

- H_0 : No differences in map with observation i removed vs. those without i removed, and
 - H_a : Some difference in map with observation i removed.
6. Calculate a p -value for the influence of observation i by comparing its pseudo-F statistic to the results for the same distance matrix where the 1 is shuffled across all N possible locations.

It is important to note that this method is not simply an outlier detection method; rather, observations that are simply outlying on a feature (or group of features) will not necessarily be identified as influential. For instance, consider the case of MDS maps made from just $P = 2$, which may rotate and scale into the map projection but do not reduce the dimension. In the $2D \rightarrow 2D$ case, the relative distances created from the maps will be identical for each holdout observation, meaning that our method cannot identify the unusual points as influential points, even if they are extreme on either the first or second variable. This problem decreases as the number of dimensions gets larger. As the sample size N grows, the relative ability of a point to influence others will decrease, so the ability to identify influential observations in maps where P is small and N is large may be somewhat limited, unless those points are dramatically different across the $P > 2$ features.

3.4.1 Generation of P-Values

In a typical PERMANOVA model, the number of permutations that could be generated from a covariate is potentially quite large, given that in most cases the combinations of a response and predictor can be shuffled in many ways. For instance, with $N = 100$ and a balanced design with two levels for the binary factor, we would have $\binom{100}{2} = 4950$ different arrangements; therefore, the smallest possible permutation-based p -value obtainable would be $\frac{1}{4950}$. However, because of the structure of the method presented here, the predictor

variable is always a vector with a single 1 in the held out observation row and 0's in all of the remaining rows. Thus, there are only N possible permutations for the predictor variable in each PERMANOVA model.

The p -value for each test would be directly calculated as k , the number of results that as or more extreme than a given observation's pseudo-F, meaning that a lower bound on the p -value is $1/N$. In a very small data setting, assuming a typical $\alpha = 0.05$, if $N < 39$, the smallest p -value would be larger than 0.05. Because we are considering N tests, we might consider a multiple testing correction for the p -values generated. For example, we could use the Benjamini-Hochberg False Discovery Rate adjustment (Benjamini and Hochberg, 1995). A more conservative approach, such as a Bonferroni correction to control the Type I error rate (as is used in the Bonferroni Outlier test), would require the p -value to be multiplied by the number of simultaneous tests performed, N . The Bonferroni-based approach would yield an adjusted smallest p -value of $\frac{1}{N} * N = 1$ and all of the other p -values adjusted to 1. In large data settings, it may be feasible to a false discovery rate adjustment. Instead, it may make more sense to use the p -values based on their rank regardless of their magnitude. The p -values can instead be construed as providing a sorting of the observations based on their relative extremeness that is a 1 – 1 mapping of the pseudo-F statistics.

Much in the way that Cook's Distance uses the size of the F-statistics to identify influential observations, the pseudo-F statistic or p -value can be compared based solely on relative magnitude to provide an influence diagnostic. In a setting where only a single influential point is present, we would expect to see that influential point have the smallest p -value (or, equivalently, the largest pseudo-F statistic) of those obtained. In simulations, the pseudo-F statistics of possibly influential points are often orders of magnitude higher than the other values in the permutation distribution.

3.5 Simulation Studies

We conduct a series of simulations to assess the efficacy of the method at identifying influential points in various different situations. In all cases, we simulate data from multivariate normal distributions and used Euclidean distances to generate the initial configurations. In the simulations, each iteration simulates a new data set, runs the mapping procedure, and tests the consistency of the configurations for each of the N holdout points.

We simulate observations that can be considered unusual in the mean structure, indicating that it may change between several groups formed by the rest of the data across one or many of the variables simulated. We test the impact of observations that differ in mean structure on a single feature ranging up to varying across all the features simulated. Typically, this type of aberrant value would lead to a distinct observation in the map produced, particularly in the case of MDS.

We consider the following scenarios: First, we simulate data in a scenario where all of the observations present come from one of three groups (each with the same mean across ten simulated features). While some observations may be more/less extreme than others, no obviously extreme points, nor any that break the group structure, are considered in Simulation One. This allows for assessment of spurious detections when no influential points are present. In Simulation Two, we add to the three-group data an extreme point that varies across the three group means for each of the ten features considered, breaking the clearly-defined group structure in the other observations. In Simulation Three, we instead add an observation that is very large on all ten features, but does not vary across the three-group mean structure. This could be thought of as equivalent to an observation which is exceedingly small as well.

3.5.1 Power to Detect Differences

In order to determine a notion of how extreme a potentially influential point might be, we quantify the extremeness of an observation in the multivariate or high-dimensional space. We calculate the Mahalanobis distance (Mahalanobis, 1936) for the potential influential point as compared to the original distribution of multivariate normal points of each simulated data set.

Given a vector $\vec{x}$ with P elements, the Mahalanobis distance of a given point to the data centroid, $\hat{\mu}$, is calculated as

$$D_{Mal}^2 = (\vec{x} - \hat{\mu})^t \hat{\Sigma}^{-1} (\vec{x} - \hat{\mu}), \quad (3.11)$$

with given estimated covariance matrix $\hat{\Sigma}$. The Mahalanobis distance can be calculated across each of the simulated potential influential points and compared to get a sense for how the efficacy of the influence diagnostic changes when influential points are only mildly outlying compared to very extreme points. Note that we lack an objective, single source of truth that clearly distinguishes an observation as being influential; however, we expect that as an observation is more extreme relative to its peers in a multivariate space, it is more likely to influence the shape of the resulting map. Thus, Mahalanobis Distance is used as a way to estimate the effect size of the different points to then assess the power of the method to detect influence: at larger Mahalanobis distances to the centroid, we would expect to see higher rates of detection.

3.5.2 Type 1 Error Assessment: No Extreme Values

In the most basic simulation, we consider $N = 60$ observations on $P = 10$ features, each taken from independent normal distributions with $N_k = 20$ observations from each group. Group means of one of $\mu = [20, 40, 60]$ are used, and variances of $\sigma^2 = 35$ for each of the features and covariances of 0 since the data are simulated to be uncorrelated. The data are

then standardized so that overall, for each of the j features, $\hat{\mu}_j = 0$ and $\hat{\sigma}_j^2 = 1$ across the observations for the three simulated groups. An example simulated data set we consider in the simulation study is shown in Figure 3.4, with resulting maps for MDS, isoMDS, and Sammon mapping included.

For each simulation, we assess the detection rate R as

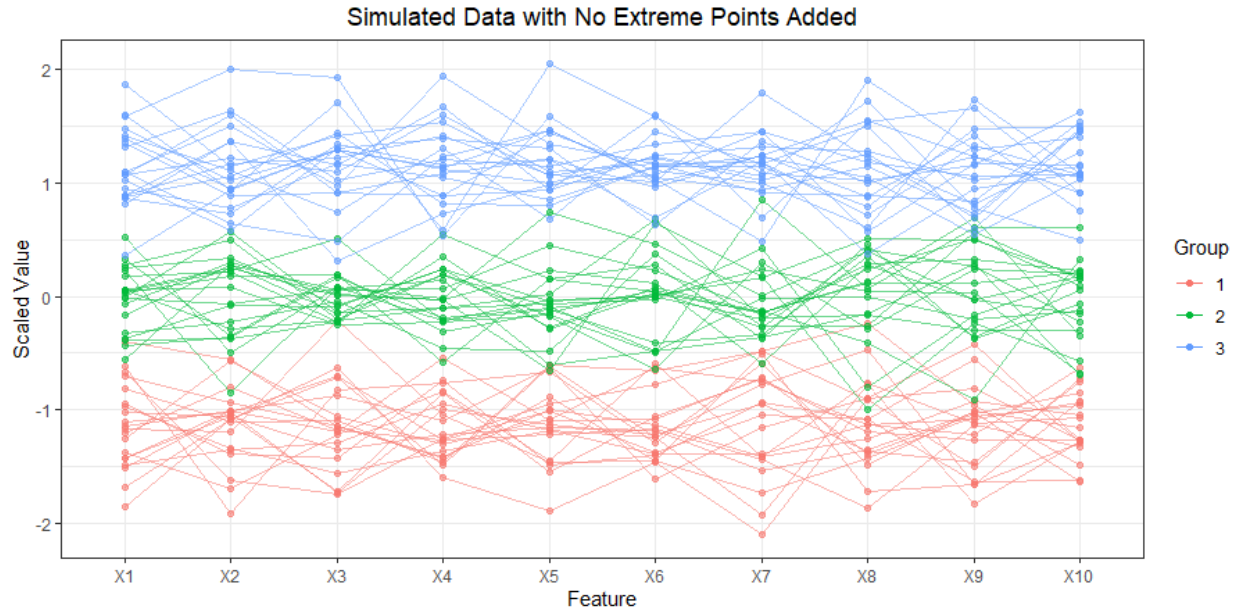
$$R_i = \sum_{n_s} \frac{\frac{\sum_{n_s} I(x_i)}{N}}{n_s},$$

where $I(x_i)$ is a binary variable evaluated for each of the $i = 1, \dots, N$ objects in each simulated map. It is defined as one if the p -value is less than the level of significance, α , and zero otherwise. The significance level for the test is typically set to 0.05, and n_s being the number of simulations run.

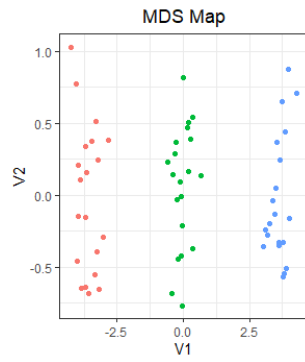
Because of the way the permutation tests work, for a large enough sample size, the most extreme points will always be less than α when the significance level is set to $\alpha = 0.05$. In the case of this simulation, two observations will always be under the significance threshold. Thus, p -values alone may not be the most useful measure for identification of potential influential points since the most extreme points will always have small p -values regardless of relative extremeness. Due to the skew in the distribution of pseudo-F values, we compare the median F-statistics in Table 3.1 for the two largest values. These results do suggest some possibility of spuriously generating large pseudo-F statistics. Note that across the simulations, the MDS maps had a median pseudo-F statistic of 11.5 vs. 3.8 for the Sammon maps for the most extreme observations.

3.5.3 Single Influential Points

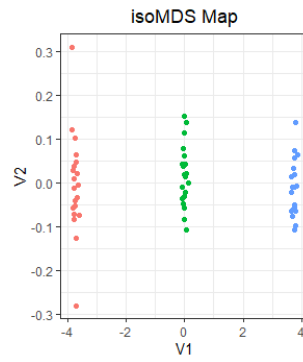
We retain the same simulation method with the three group structure from the previous simulation (with $N = 60$) but add several types of influential points in the following simulations. In the first, we consider adding a single aberrant point with mean 0 and



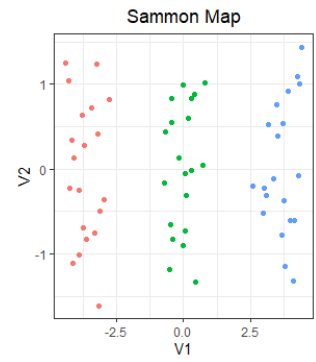
(a) Simulated Data with No Extreme Points



(b) MDS Map



(c) IsoMDS Map



(d) Sammon Map

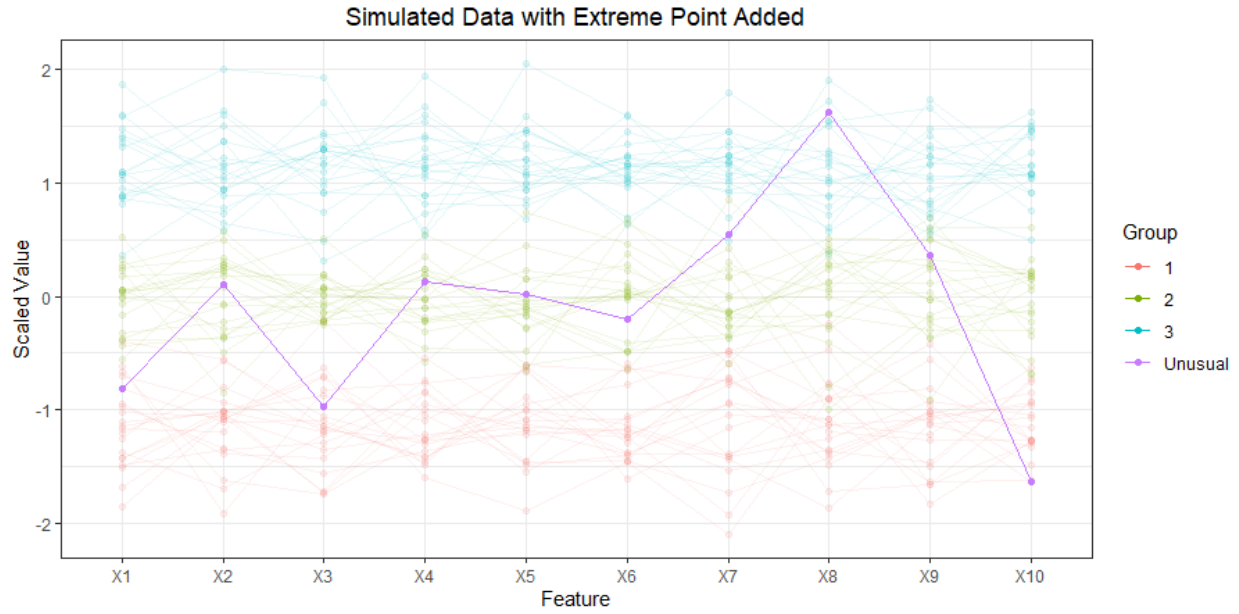
Figure 3.4: Plot of simulated data with no aberrant observations, the group structure in top Panel (a) and three different maps from the simulated data set in Panels (b), (c), and (d).

Simulation One Simulation Results		
Method	Median largest pseudo-F Stat	Median second-largest pseudo-F Stat
MDS	13.51	6.99
isoMDS	20.18	8.59
Sammon	4.09	3.36

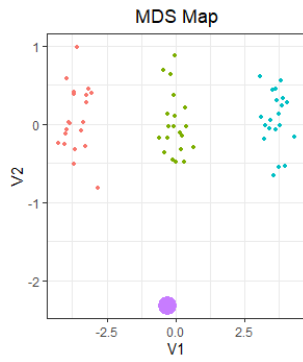
Table 3.1: Table of pseudo-F statistic results for the 100 simulations when no influential points are present for three mapping methods. By construction, the method has the same detection rate and non-detection rates for $\alpha = 0.05$ due to the structure of the permutation-based tests performed. The median pseudo-F statistics for the two largest pseudo-F statistics observed (or equivalently, for the smallest two p -values which, in this case, are always < 0.05) are large for MDS and isoMDS, and are smaller but still relatively large on a pseudo-F scale for the Sammon maps.

standard deviation 1 (after the data are standardized), which means that the point can vary across the three groups and clearly break the group structure for each of the 10 features. An example unusual observation is shown in the Panel (a) of Figure 3.5, with the Panels (b), (c), and (d) showing each of the resulting ordinations. In general, all three mapping methods place the extreme point far away from the three well-separated groups; however, the isoMDS map seems to cluster each of the groups more tightly than does the MDS or Sammon maps.

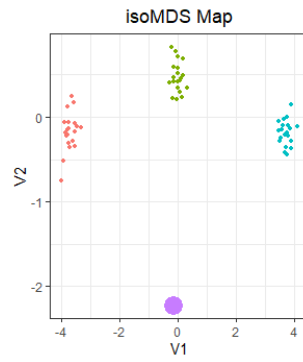
Since the goal of the method is to identify undue influence of the aberrant observation on the resulting ordination when each point is left out, visual assessment of the maps may not provide strong intuition as to whether a point will be identified as influential or not. The results of the simulation study with 100 simulations per ordination type are shown in Table 3.2, with p -values and median pseudo-F values of the aberrant, but not large, point included. The diagnostic method identifies the influential point about 90 percent of the time when assessing MDS maps, as compared to 51 percent for isoMDS and 17 percent for



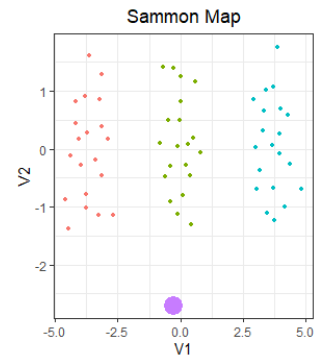
(a) Simulated Data with Mean-Zero Observation Added



(b) MDS Map

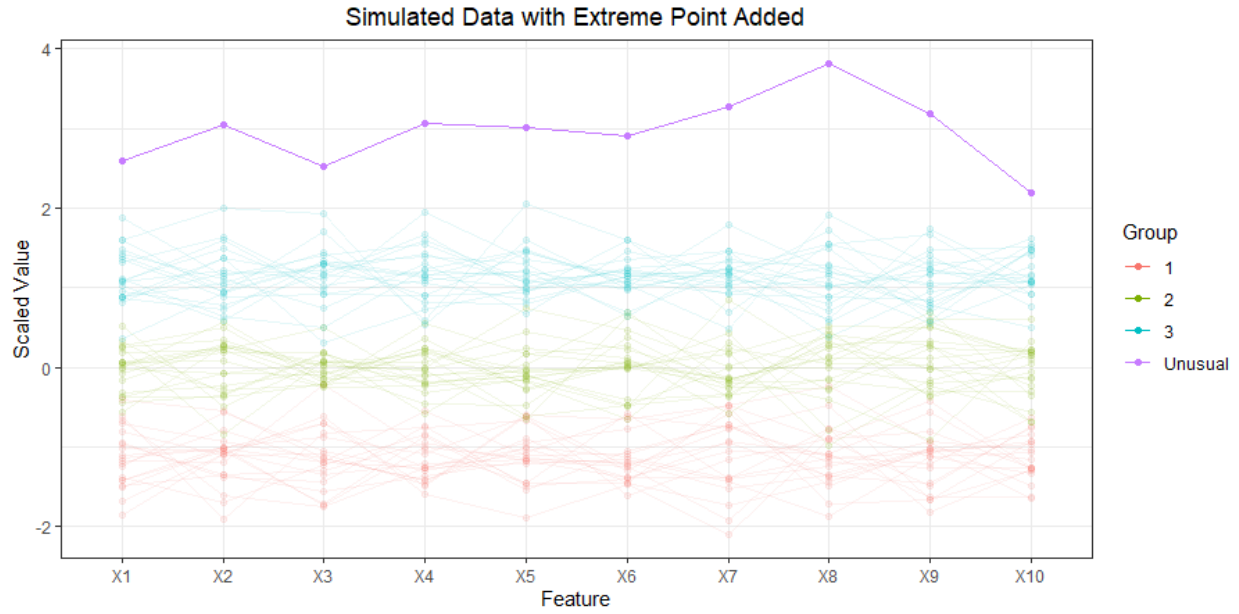


(c) IsoMDS Map



(d) Sammon Map

Figure 3.5: Simulated data with unusual point (simulated with mean 0): Panel (a) includes an observation that breaks clearly-defined group structure and its impacts on the three maps in Panels (b), (c), and (d).



(a) Simulated Data with Very Large Observation Added

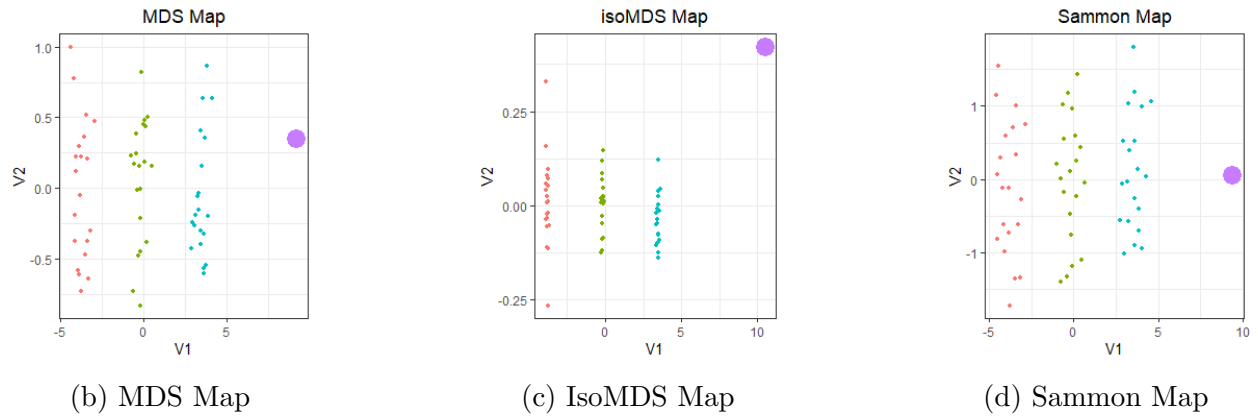


Figure 3.6: Panel (a) shows an observation that is much larger across all features than the other observations and its impacts on the three maps in (b), (c), and (d).

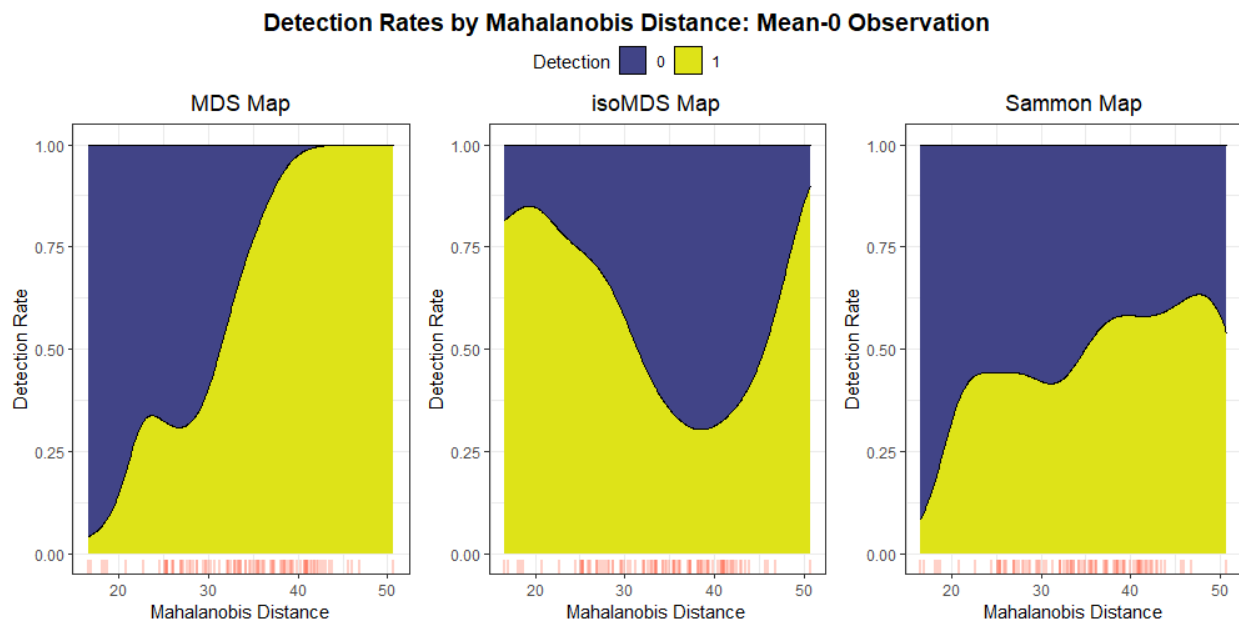


Figure 3.7: Detection rates by Mahalanobis distances for each of the 100 simulations, using conditional density plots. As distance increases, detection rate increases rapidly for the aberrant points with mean 0 in MDS maps, and increases somewhat less clearly for Sammon and isoMDS maps. Bandwidth set to 3.

Sammon mapping. Moreover, the difference in pseudo-F values generated is interesting; for MDS the median pseudo-F statistic is very large, 171.40. For isoMDS, the median pseudo-F for the potentially influential point is 8.31, which is still large on a test statistic scale, although much smaller than the MDS case, and for Sammon mapping the median pseudo-F is 2.26.

Figure 3.7 shows the detection rates by Mahalanobis distance for the 100 simulations. We would expect that as points become more extreme (i.e., points that have a higher Mahalanobis distance), they would be more likely to be flagged by the method as potentially influential points. This is certainly the case for MDS maps and appears to be similar for Sammon maps. It is less clear in the Panel (b) of Figure 3.7 where the isoMDS results do not show an increase in detection rates at higher distances; however, this may be because the variation in distances is not large enough to reveal the trend in this simulation.

It appears that generally, the MDS maps are highly sensitive to the observations that do not clearly belong to any of the groups (but are neither very small or very large), with isoMDS maps being relatively sensitive as well. By contrast, the Sammon maps are somewhat sensitive but appear to be less impacted by the presence/absence of the potentially influential point, at least in terms of preserving interpoint distances of the remaining points.

We additionally consider a second case where observations may be either very large or very small relative to the other observed data. We add a single large observation to the data in the same way as the mean-0 observation. Large observations are simulated to have a mean values equal to three standard deviations larger than the 0-mean of the rest of the standardized data, for each of the 10 features. The results for large observations are shown in Table 3.2. The influence diagnostic tool appears to treat these observations differently than it does the influential point with mean-0. An example of a “large” observation is given in the Panel (a) of Figure 3.6, with resulting maps shown in the Panels (b), (c), and (d). The MDS and Sammon maps appear to place the extremely large observation nearest to Group Three at a slight distance—the isoMDS map pushes the aberrant observation far from all three groups.

Table 3.3 contains detection rates for the extremely large observation, which are 23%, 12%, and 19% for MDS, isoMDS, and Sammon mapping methods, respectively. Interestingly, the diagnostic appears to perform the worst for isoMDS at identifying large observations as influential. Given the much smaller median pseudo-F statistic size for each of the three methods, it seems apparent that this diagnostic does not as clearly identify points that are dramatically larger as being influential in most cases, even though the Mahalanobis distances are similar to the previous simulation. This implies that the rest of the map might be less impacted by inclusion/exclusion of a single point that is extreme across all of the features.

In both settings, the power to detect the influential point increases relative to the extremeness of the observation that breaks group structure. Figure 3.8 shows an initially-

Identification Rate by Mapping Method: Mean-0 Observation				
Method	Correctly Identified Influential	Correctly Identified Non-Influential	Median pseudo-F for Candidate Influential Point	
MDS	0.90	0.982	174.40	
isoMDS	0.47	0.975	8.31	
Sammon	0.23	0.971	2.26	

Table 3.2: Correct identification rates for each mapping method for an aberrant observation that varies across groups with mean 0 (i.e., is not larger/smaller than the other observations). Results from 100 simulations.

increasing detection rate for MDS with drop in detection rates as extremeness increases. The diagnostic has increasing detection rates for Sammon mapping but decreases for isoMDS.

3.5.4 Dimensionality of Influential Features

The number of features on which an observation is anomalous is potentially an additional critical element, and can impact how the method detects influence. Per the previous simulation set up, we simulated observations from the central group (with mean $\mu = 40$) and allowed a single feature at a time to be simulated from the larger group (mean $\mu = 60$). The same variances/covariances were kept from the previous simulations, and the data were scaled so that in the final data set, the mean of the central group was 0 for all 10 features. We repeated this set up, incrementing the number of features that differed from the central group to a different group mean up to $P - 1 = 9$ features in order to see how the diagnostic would detect differences based on different points that are only unusual on a small number of features.

Examples of two simulated data sets, along with resulting detection rates from the study, are shown in Panel (a) and Panel (b) of Figure 3.9, respectively. For the methods where

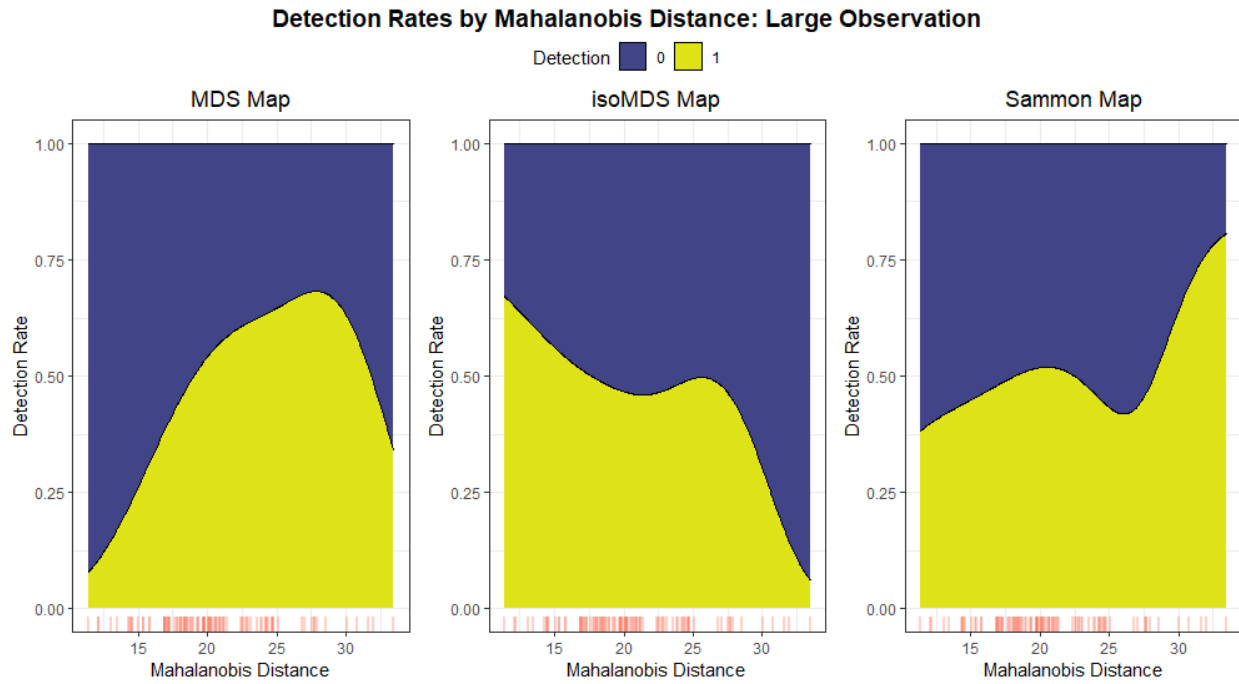
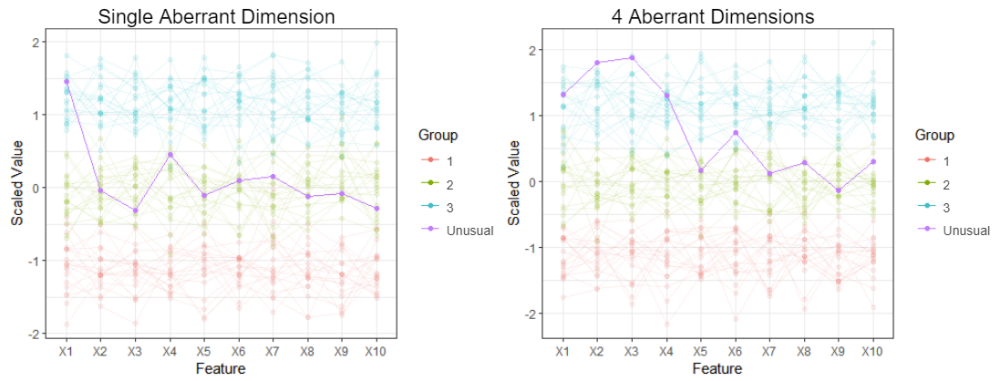


Figure 3.8: Detection rates by Mahalanobis Distances for each of the simulations for the larger aberrant observation. For MDS and Sammon mapping, detection rates generally increase as Mahalanobis distances increase. For isoMDS, the initial detection rates start higher than in the initial case, but appear to decrease as Mahalanobis distance increases. Results shown for 100 simulations. Bandwidth set to 3.

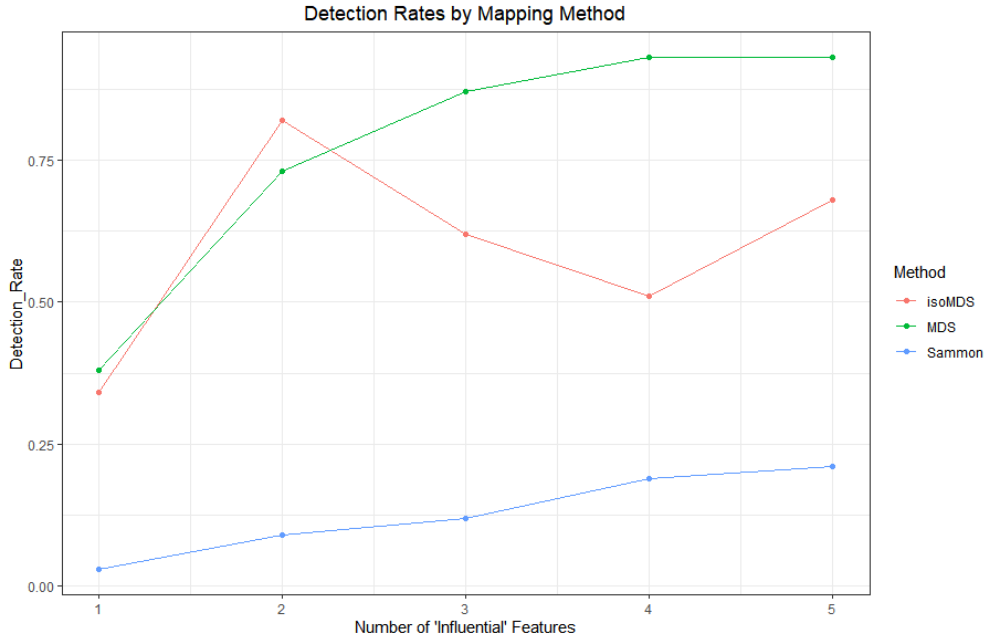
Identification Rate by Mapping Method: Large Observation				
Method	Correctly Identified Influential	Correctly Identified Non-Influential	Median pseudo-F for Candidate Influential Point	
MDS	0.23	0.971	1.97	
isoMDS	0.12	0.969	0.50	
Sammon	0.19	0.969	1.35	

Table 3.3: Correct identification rates for each mapping method for an aberrant observation that is extremely large (outlying) on all features, which are substantially lower than those generated for a point that overlaps between the observed groups.

influential points more heavily impact the shape of the resulting map, the rate of correct detections of potentially influential points increases as the observation differs from the main group on more dimensions. For the MDS maps, detection rates appear to increase up to the point where shifting the individual dimension from one group to another shifts the entire observation’s group membership—at that point, at least in a situation where the “influence” from an observation is shifting from one group to a second group, we would expect to see a nearly symmetric decrease in detection rates (since being relatively extreme in 1 dimension and “regular” in nine dimensions is essentially the opposite scenario of being extreme in nine dimensions and “regular” in one). The diagnostic appears to detect influential points for isoMDS maps at a slightly higher rate as the number of features increases, but it does so inconsistently and even decreases from three to four features. The detection rates for Sammon mapping were generally low but does increase as the number of unusual features increase.



(a) Two realizations with increasing numbers of features that differ from main group.



(b) Detection Rates shown across the increasing number of features.

Figure 3.9: In Panel (a), data were simulated to differ across two dimensions to assess detection rates as the number of extreme values increases across features. Panel (b) shows the detection rate trends for each map type. Results shown from 100 simulations.

3.6 Overview of Real Data Results

3.6.1 Penguin Data

The Palmer penguin data set (Horst et al., 2020) contains information pertaining to three different species of $N = 344$ penguins: Chinstrap, Adelie, and Gentoo. Several of the critical distinguishing features include the bill length, bill depth, body mass, and flipper length. Bill lengths and depths are measured on the culmen, which is the upper ridge on the penguins' bills. Researchers may be interested in identifying differences in the species across a suite of morphological characteristics, and potentially in seeing those differences projected into a lower-dimensional space. We might assume that penguins differ by species, but the actual groupings might be complex or overlapping. The penguins form relatively distinct groups that align with their species when measured on these biometric markers.

For the purposes of testing the mapping diagnostic, we utilized on the physical measurements of the penguins, including the bill length (in mm), bill depth (in mm), flipper length (in mm), and body mass (in grams), as shown in Figure 3.10. We consider two versions of this exploration: the original, uncontaminated data set, and one where we add artificial points to the real observations.

3.6.1.1 Uncontaminated penguin data set We first assess the performance of the influence diagnostic on the actual penguin data set, as shown in Figure 3.10, in order to identify whether any penguins appear to be potentially influential. Using MDS, isoMDS, and Sammon mapping on the standardized variables with Euclidean distance, we identified potentially influential penguins as follows: 1) Use the p -value to determine candidate influential observations, and 2) Assess the relative extremeness of each candidate influential observation using the size of the pseudo-F statistic.

For MDS, the most influential penguin is a Chinstrap (see Panel (b) in Figure 3.12),

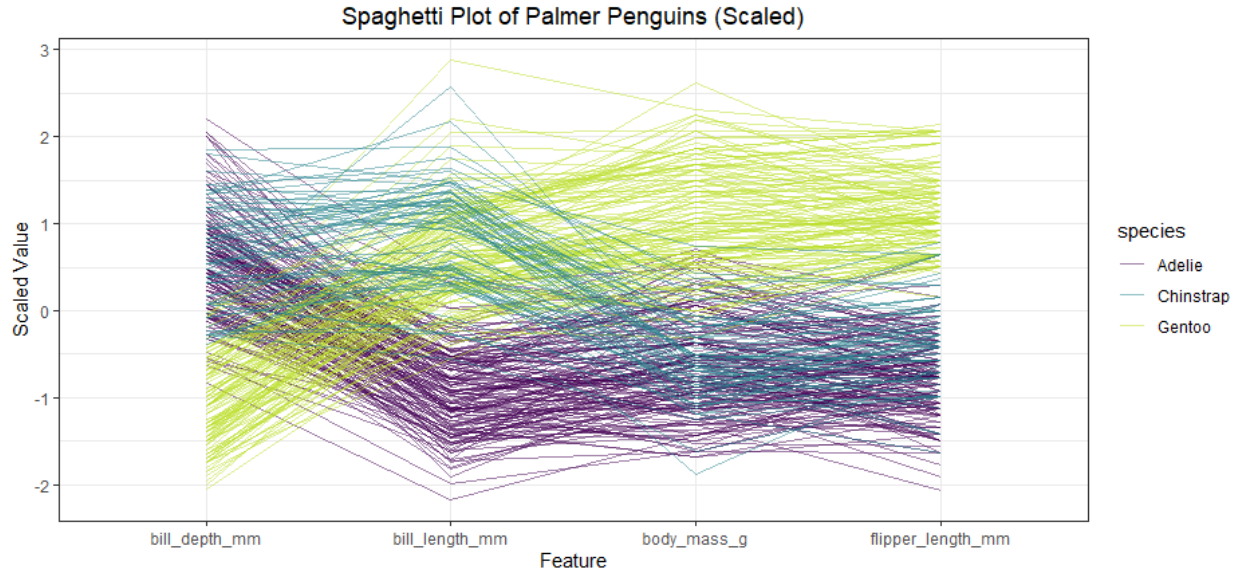
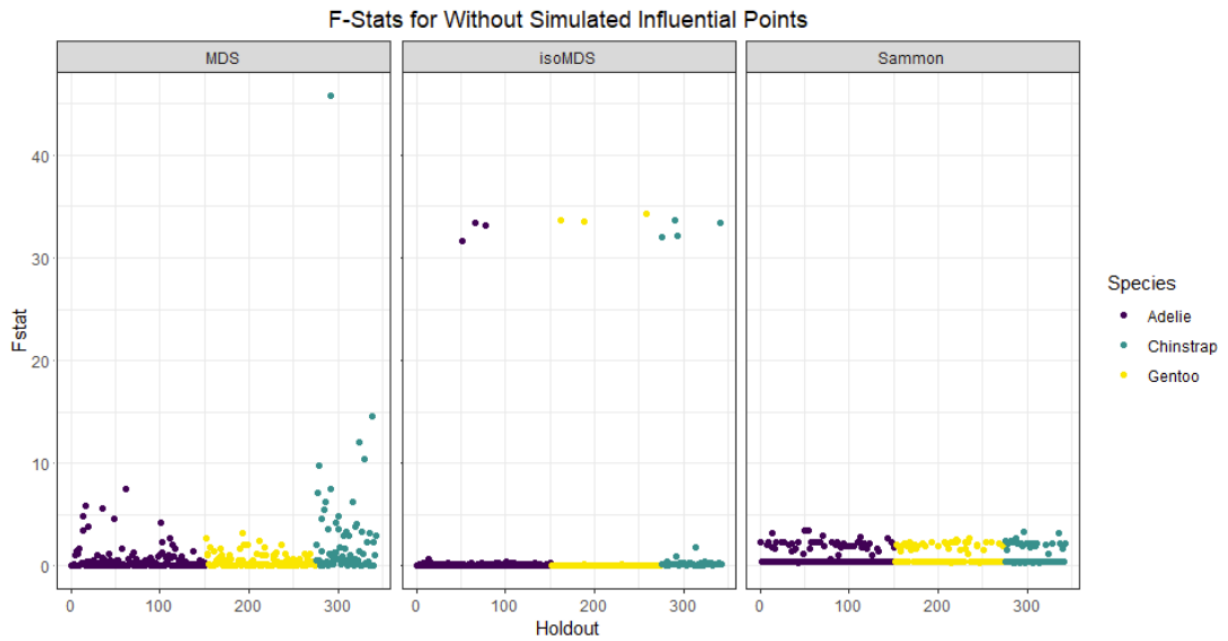


Figure 3.10: Spaghetti plot of the scaled Palmer penguin data across the four morphological variables, colored by species type.

and is highlighted in a spaghetti plot in Figure 3.12. The identified penguin appears to differ somewhat from the characteristics of other Chinstrap penguins in that it has longer bill length and shorter flipper length than other Chinstraps. The isoMDS map and Sammon map (shown in Panels (a) and (c), respectively, in Figure 3.12) pick the same penguin as having the largest pseudo-F statistic. They both choose a Gentoo penguin that appears to be small on all four metrics, at least relative to other Gentoos.

Figure 3.11 shows the pseudo F-test statistics for each of the penguins in the data set for each of the three mapping methods considered. Across the mapping methods, MDS appears to have the most sensitivity to extreme-values with isoMDS showing some fairly large test statistics for each of the three groups. Note that for MDS, the penguin species that appears to generate the most influence on the maps are Chinstrap penguins, which tend to be highly variable across flipper and bill measurements. The diagnostic for isoMDS may be better at identifying within-group influential points based on the fact that it identifies several large pseudo-F statistics for each of the three species whereas the MDS seems to choose a single



(a)

(b)

(c)

Figure 3.11: Pseudo-F test statistics for penguins without any influential points added, colored by penguin species type. Panel (a) shows the pseudo-F statistics for MDS, Panel(b) for isoMDS, and Panel (c) for Sammon mapping.

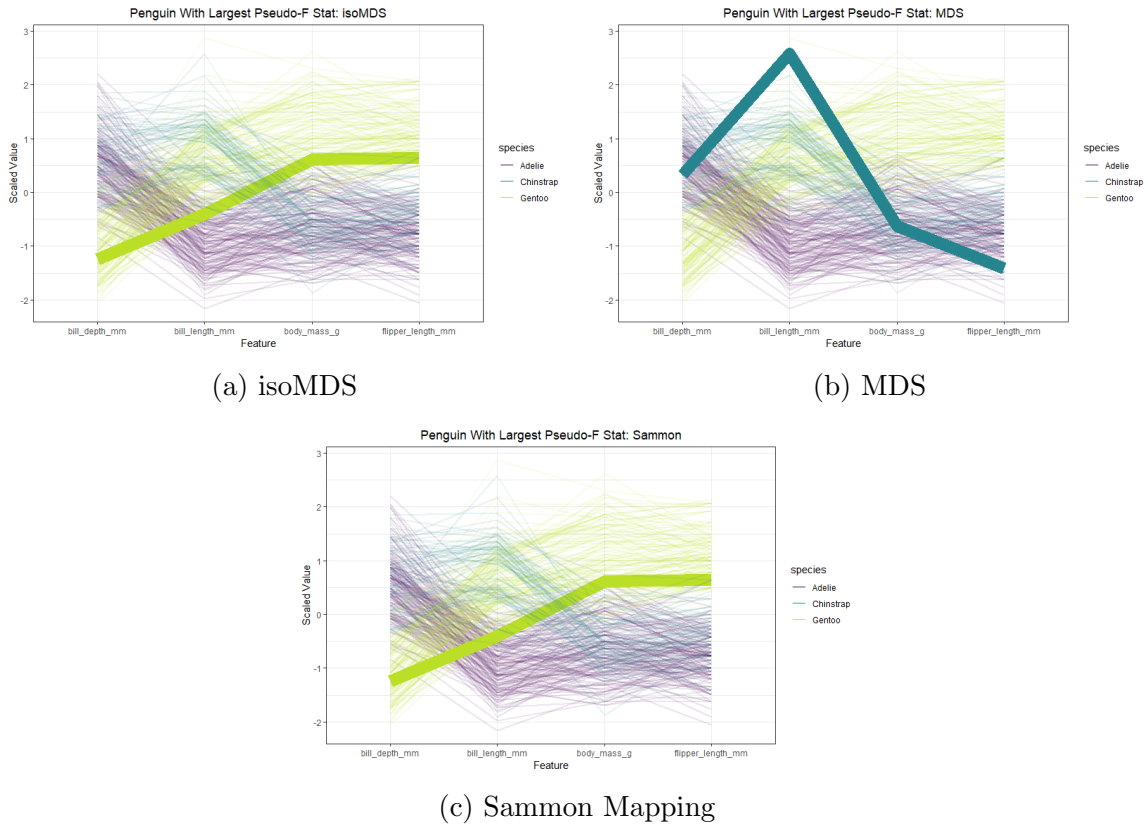


Figure 3.12: Spaghetti plots of the penguin with largest pseudo-F statistic from the three map methods. Panel (a) and (c) show the same penguin (an Adelie) selected as most influential (observation highlighted) in the isoMDS and Sammon maps. Panel (b) shows the Chinstrap penguin selected as most influential in the MDS-map. In both cases, the selected penguins tend to be either large or small relative to the penguins of their same species.

penguin with a large pseudo-F statistic and identifies some penguins with moderately large pseudo-F statistics. For Sammon mapping, the maps appear to produce consistently small pseudo-F stats, indicating that the maps are relatively stable across inclusion/removal of any of the penguins.

3.6.1.2 Contaminated penguin data set We additionally simulated various single penguins to add to the data set to test the method's ability to distinguish it from the additional penguins. We considered a handful of cases, much in the same vein as the previous simulation

studies:

- Penguin with large measurements across all 4 dimensions. Such a penguin might look similar to a Gentoo species when considering only the bill depth and bill lengths, but would have much larger mass (akin to an Adelie) and similarly larger flippers.
- Penguin with small bill (length and depth), large mass, and large flippers. Such a penguin differs from the patterns observed by all three species in the data set.
- Penguin with obviously aberrant value for mass only. In this setting, we simulate a penguin that appears similar to a Chinstrap, but with an obviously aberrant body mass measurement. This penguin is intended to simulate an incorrect single value.

Each potentially influential penguin was added to the data set and considered independently for each of the three mapping methods, and is shown in the Figure 3.13. The penguin data were scaled prior to the addition of the simulated penguin, and scaled after the addition of the new (aberrant) observation.

Resulting pseudo-F statistics are shown in Panels (a), (b), and (c) in Figure 3.14. For the first situation, where the aberrant penguin is large across all four metrics, the MDS maps appear to be impacted by its inclusion and but it is only for MDS is that penguin identified as influential. In the situation where we simulate a penguin with a small bill and large relative mass and flipper length, the structure does not appear to be different enough from the other penguins in the data set to substantially affect the shape of the isoMDS and Sammon maps, but similarly, the observation is detected as influential for MDS. Finally, in the case where we have a simulated penguin that looks like a Chinstrap with very large mass, we identify relatively large pseudo-F statistics for both the isoMDS and MDS maps (although in the case of isoMDS, it is not the largest one).

In each of the scenarios we see that in the MDS map, the aberrant penguin is clearly identified as having a large pseudo-F statistic that is much larger than the others observed. In

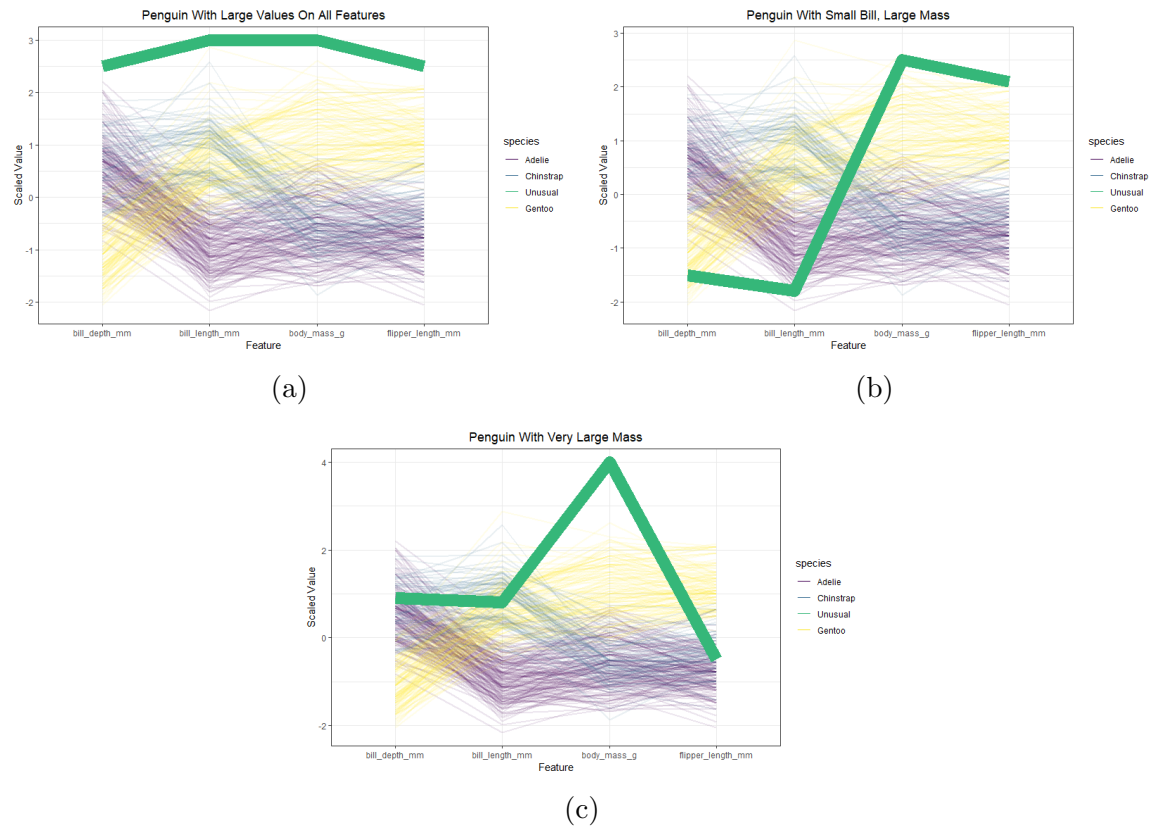
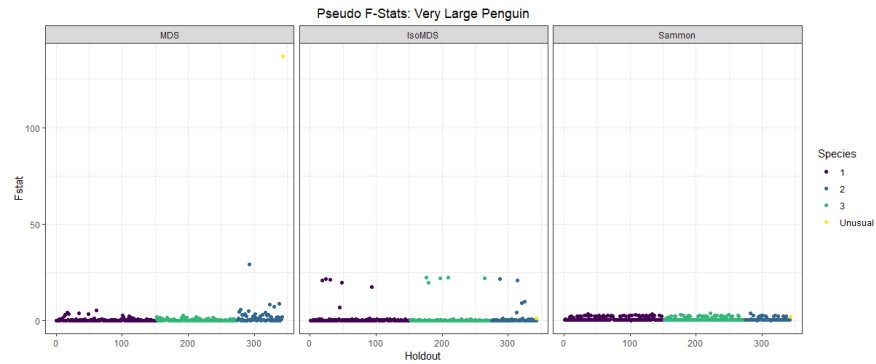
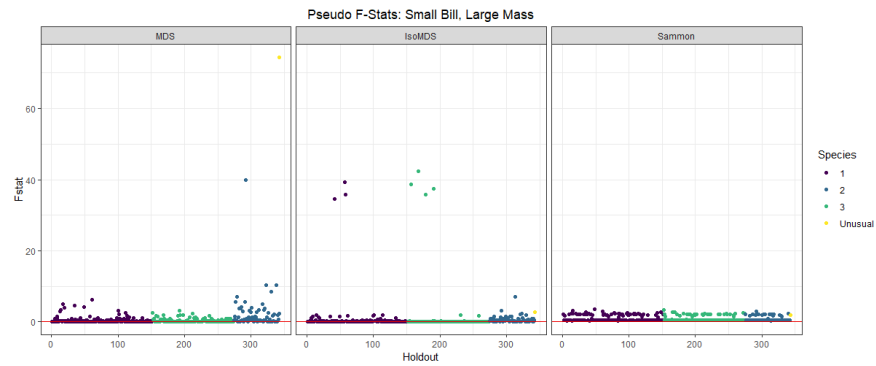


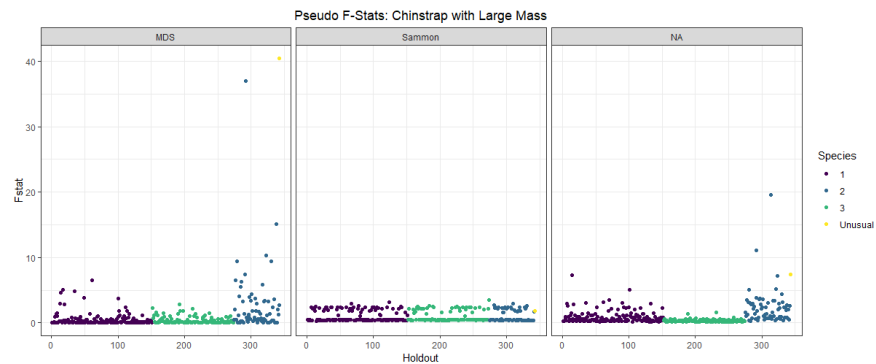
Figure 3.13: Palmer penguins data with various influential observations included. Panel (a) shows a spaghetti plot of a penguin that is large on all four measurements, Panel (b) shows the profile of a penguin with small bill measurements and large mass and flipper length, and Panel (c) shows a penguin that looks like a Chinstrap except that it has extremely large mass relative to other Chinstrap.



(a) Added simulated penguin with large measurements on all features



(b) Added simulated penguin with small bill but large relative mass and flipper length



(c) Added simulated Chinstrap with very large mass

Figure 3.14: Pseudo-F statistics for penguin data contaminated with penguins of various types. Panel (a) provides the pseudo-F statistics for a penguin that is large on all features considered. Panel (b) provides pseudo-F statistics for a penguin with small bill but large mass and flipper length, and Panel (c) provides the pseudo-F statistics for a penguin with vary large mass.

the isoMDS map, the penguin has one of the largest five such pseudo-F values. Interestingly, the isoMDS appears to be more affected by several of the “real” penguins in the Chinstrap and Adelie groups than the simulated aberrant penguin. The results for the Sammon map indicate that it does not seem to be as seriously impacted by the inclusion/exclusion of the unusual penguin.

One critical difference between the results of the simulation studies and the results for the Sammon maps on the Palmer penguin data set is that there are no large pseudo-F statistics identified here; this lack of potentially spurious detections may imply that the maps for those methods are simply too stable to be affected by the removal of a single observation when the sample size is larger. Unlike the simulation studies, where none of the observations within the simulated groups to be considered influential, it is possible that an extreme penguin could itself be different enough from the others in its group that it does substantively change the shape of the resulting MDS ordination. These results imply that different types of influential points may affect maps differently.

3.6.2 Carnegie Classification

The Carnegie Classification (CC) is a method for classifying institutions of higher education based on several metrics of institutional productivity. The 2015 version of CC was based on data ranging from doctorates awarded in STEM, Non-STEM, Humanities, Social Sciences, and Other (professional) fields, as well as research expenditures in STEM and Non-STEM fields and two proxies for institutional faculty size (tenurable faculty headcount and research faculty headcount).

The Carnegie Classification is typically based on a ranked set of institutional features due to the presence of outlier institutions on several metrics. In particular, some institutions spend a great deal of expenditures on research compared to others, so the actual CC is based on two indices generated using principal component analysis on the ranked data that are

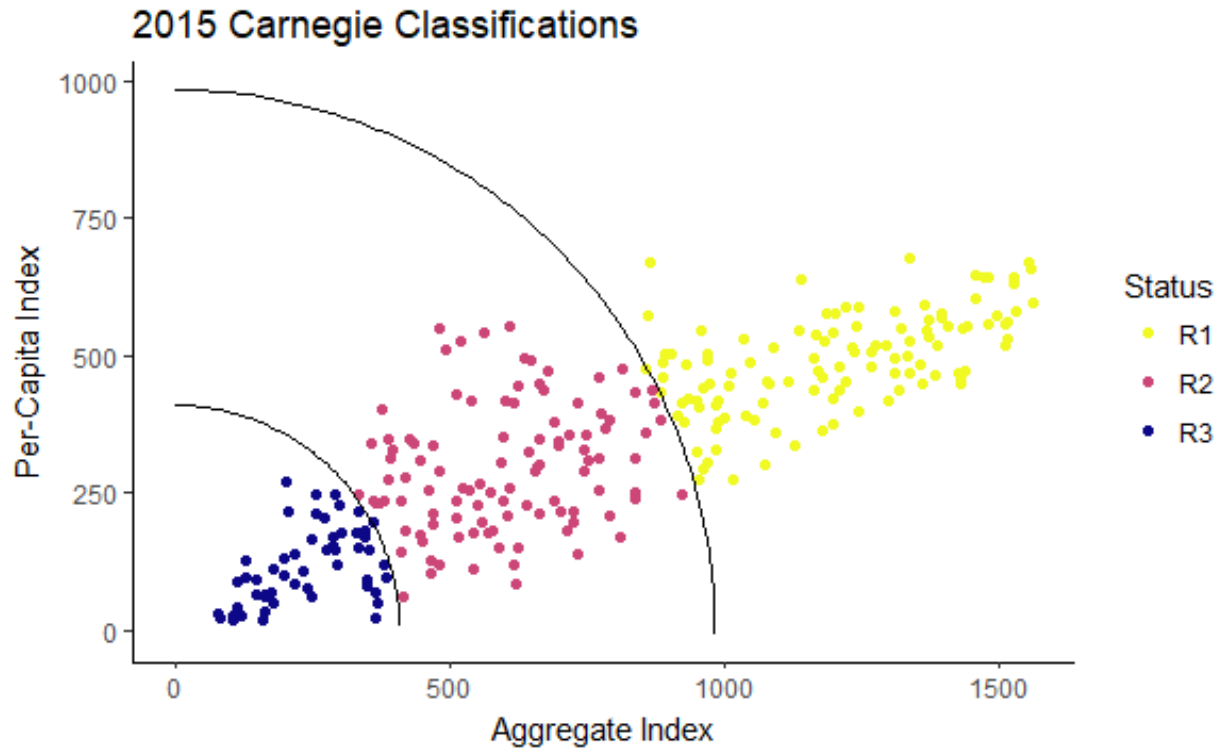


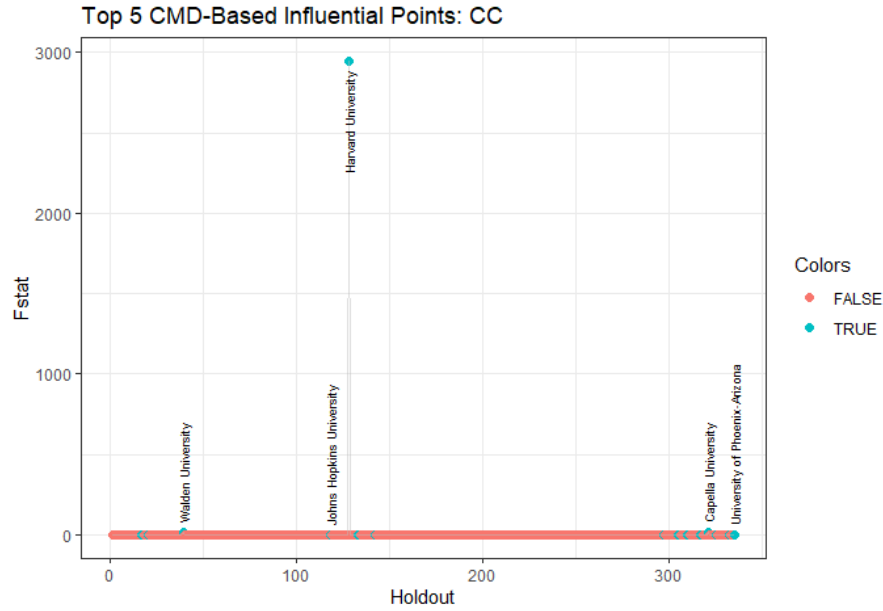
Figure 3.15: Map of the doctoral-granting institutions from the 2015 Carnegie Classification. R1 institutions refer to the “Very High Research” category, R2 refers to the “High Research” group, and R3 refers to the “Moderate Research” category.

shown in Figure 3.15 (see Harmon et al. (2019) and Kosar and Scott (2018b) for details on ranking). This ranking process has a similar effect on the results as to a log-transform (Kosar and Scott, 2018b).

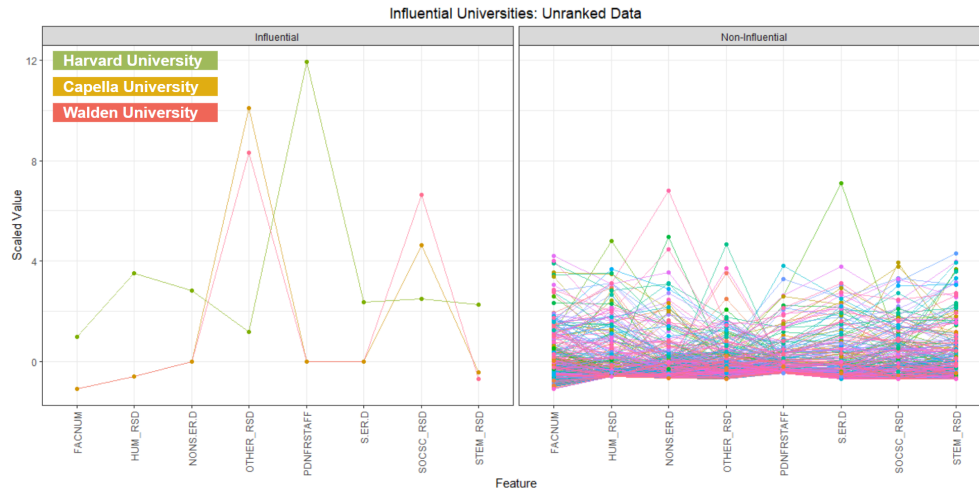
In the method used for calculation of the 2015 Carnegie Classification, the data were ranked (using the minimum method for ties, as specified in (Kosar and Scott, 2018b)). This has the effect of consolidating the data, so observations with particularly large values for features like STEM expenditures are somewhat less extreme. It is likely that fewer institutions would be detected as outlying and thus, likely, fewer would be observed as influential points in this setting. Therefore, we investigate scaled versions of the un-ranked data in part to assess whether or not some of the relative extreme points appear as influential relative to other universities in the group, using Euclidean distance and MDS.

For MDS, the institutions that are identified as influential appear to be those that differ from the average profile, particularly as it relates to STEM output. The institution with the largest pseudo-F statistic is Harvard University, followed by Capella University, Walden University, and Johns Hopkins University. Each of the top 4 schools are either institutions known to operate at large scale (Capella and Walden) or with large research endowments/funding (Harvard and Johns Hopkins). The top 15 institutions are shown in Table 3.4.

A spaghetti plot of the institutions with the largest pseudo-F statistics is given below, shown on the scaled data. For the schools with the top 10 largest pseudo-F statistics, the un-ranked values of at least one of the main features in the Carnegie Classification appears to be near the largest in the data set. For some institutions, like Harvard (shown with the largest value on PDNFRSTAFF (the scaled count of non-tenured postdoc researchers)), they are near the maximum value on most or all of the metrics included. Unsurprisingly, several of the institutions with the 10 highest pseudo-F statistics include schools that spend a great deal of money on STEM and non-STEM expenditures relative to their peers.



(a) Pseudo-F Statistics for each of the most influential schools, including Harvard, with the largest Pseudo-F statistic.



(b) Original scaled data values for the most influential schools.

Figure 3.16: Diagnostic results and scaled, original data values for the three most influential institutions in the 2015 CC data set. Panel (a) shows the pseudo-F statistics highlighting the most influential schools in the map, and Panel (b) shows the original scaled values for each of the institutions relative to the other institutions in the CC data set.

Carnegie Classification: Top-15 Influence Candidates		
Name	Pseudo F-Stat	<i>p</i> -value
Harvard University	2943.986	0.006
Capella University	13.831	0.009
Walden University	12.842	0.012
Johns Hopkins University	1.993	0.015
University of Phoenix - Arizona	0.916	0.018
Alliant International University - San Diego	0.897	0.021
University of Wisconsin - Madison	0.666	0.024
Massachusetts Institute of Technology	0.634	0.027
University of Washington - Seattle Campus	0.343	0.030
University of Michigan - Ann Arbor	0.312	0.033
Argosy University - Phoenix Online Division	0.308	0.036
Stanford University	0.301	0.039
Northcentral University	0.203	0.042
Liberty University	0.096	0.045
University of California - Berkeley	0.081	0.048

Table 3.4: Table of the top 15 most “influential” universities in the scaled 2015 Carnegie Classification data set.

3.7 The Curious Case of Extreme Points and t-SNE

3.7.1 t-SNE

One of the notable new methods for high-dimensional data visualization is t-SNE. t-SNE has been cited more than 21,000 times in use cases ranging from biology (Amir et al., 2013) to economics (Greengard et al., 2020) to natural language processing and machine learning

(Liu et al., 2018). The method generates low-dimensional representation of high-dimensional data that can be used to identify common groups, visualize data structure, and attempts to better maintain both local and global structure than previously-developed methods. t-SNE centers on calculation of a balance between two (symmetric) similarity metrics. The first, called $p_{i,i'}$, is a conditional probability based on the Euclidean distance between pairs of points x_i and $x_{i'}$ in the high-dimensional space. In the lower-dimensional representation, this similarity is defined as $q_{i,i'}$ for map points z_i and $z_{i'}$.

A good representation of the high-dimensional space should mean that $p_{i,i'}$ and $q_{i,i'}$ look reasonably similar to each other; since these are both simply joint probabilities, the objective function of t-SNE is a symmetricized Kullback-Leibler divergence between the two, defined as

$$KL = \sum_i \sum_j p_{i,i'} \log\left(\frac{p_{i,i'}}{q_{i,i'}}\right). \quad (3.12)$$

t-SNE improves on SNE, which uses a light-tailed Gaussian distribution for both the higher and lower dimensional similarity metrics. This leads to a phenomenon known as the “crowding problem” where the two-dimensional map cannot accommodate many neighbors in high dimensions, leading to an inability to preserve relative distances between points in the map (Van Der Maaten and Hinton, 2008). By utilizing a Cauchy distribution instead, t-SNE better preserves the local structure of the data and is better able to handle more neighbors in high dimensions. Most of the minimization of objective functions is done via gradient descent (and some methods require simulated annealing because the algorithm can get caught in local optima). Critically, because the optimization is non-convex, and as a consequence of the gradient descent methods used to optimize it, t-SNE has the potential to provide different maps when run on the same data at the same perplexity based on different random number seeds.

While t-SNE is a more complex method than MDS and related techniques, it does share a relationship to PCA. t-SNE is subject to a pre-processing step involving PCA. In most implementations, a process called “whitening” the data via PCA is performed on the initial data set (or input distance matrix) to reduce the dimensionality to a reasonable value (Van Der Maaten and Hinton, 2012). For instance, in Van der Maaten’s initial paper, several examples reduced the initial dimensionality of the data to 30 for computational gains (Van Der Maaten and Hinton, 2008).

Depending on the size of the initial dimensionality of the data, this step may drastically reduce the number of dimensions that t-SNE needs to deal with. However, it has some potential to impact the resulting ordination, particularly if the number of initial dimensions kept is relatively low or PCA does not do a great job of fitting the original data set. At best, the t-SNE mapping is only as good as the initial PCA is. For the simulations shown in this paper, no whitening was performed, and the original distance matrix was passed to t-SNE directly.

Campbell et al. (2019) explored different methods than PCA as an intermediate dimensionality reduction step in t-SNE. They compared PCA, Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA), Sammon Mapping, and Locally Linear Embedding (LLA) as preliminary steps in t-SNE and found that resulting maps tended to form different groupings with variation in separation based on the initial step in the process. While we did not consider assessing the impact of different intermediate dimensionality reduction techniques in this analysis, it does present an interesting route for additional research. Instead, since our influence diagnostic can be used for any mapping tool, we compared the results of t-SNE to the results obtained by MDS on several of the data sets previously considered.

3.7.2 Simulations

We consider the same simulations used for MDS and related methods: No influential points present, one that bounces between group means, and another with an observation that is relatively large on all dimensions. When no aberrant points are added, the median pseudo-F value generated for the smallest p -value is 1.67. Detection rates for the 100 simulations with potential influential points added are shown in Table 3.5. Interestingly, compared to the rates of detection for the other mapping techniques, the diagnostic does not appear to identify differences in map shapes based on inclusion/exclusion of any of the observations. Thus, it appears that t-SNE maps are not impacted by the presence of an aberrant observations.

Identification Rates for t-SNE			
Method	Correctly Identified Influential	Correctly Identified Non-Influential	Median pseudo-F for Candidate Influential Point
Mean-0 Observation	0.02	0.967	0.779
Large Observation	0.01	0.967	0.567

Table 3.5: Identification rates for t-SNE for an aberrant observations, including the mean-0 observation that varies across groups and one that is extremely large (outlying) on all features. Note that detection rates are substantially smaller than those of the other methods explored.

This is unsurprising in the t-SNE setting because the t-SNE maps we have explored tend to place the very large or very small point in the center of the map or within one of the groups of points. Generally, the relative magnitude of the outlying observation on any single dimension tends to be less important than the structural differences, at least when it comes to identifying the influence in MDS maps. The permutation distributions shown in Figure 3.3 illustrate that the range of permutation values for t-SNE maps is much smaller than that

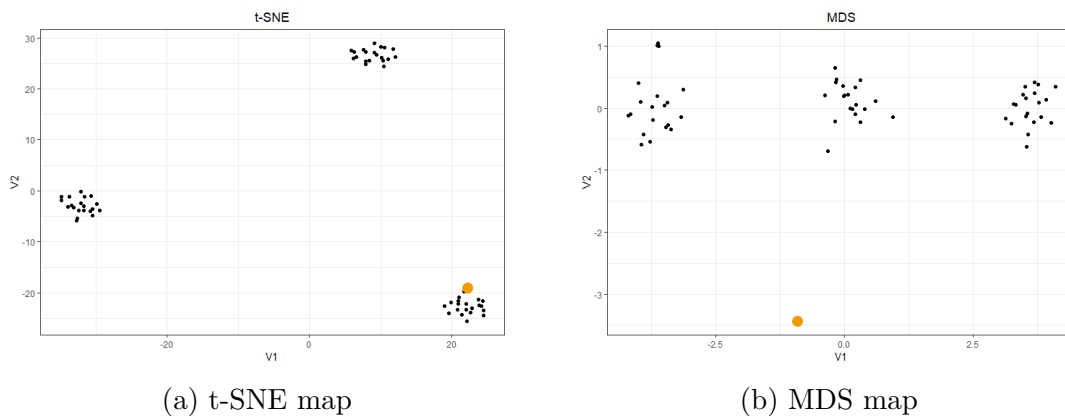


Figure 3.17: Two maps showing the position of an aberrant point. Panel (a) gives the t-SNE map which puts the aberrant point (shown with the orange dot in the bottom right-hand group). Panel (b) gives the MDS map that correctly places the aberrant point away from any of the groups.

of MDS in the presence of an influential point. Figure 3.17 shows this phenomenon a bit more clearly—in the t-SNE map in the Panel (a), the aberrant observation is placed close to the observations in the central group, whereas in the MDS map in Panel (b), the aberrant observation is pushed away from the three distinct clusters of observations.

3.7.3 A Note on Perplexity

For the simulations, we determined the value of the perplexity parameter subjectively based on the plots produced by the t-SNE runs. Van der Maaten and Hinton (2008) note that tSNE is not overly sensitive to changes in perplexity; however, if perplexity is too small, the resulting ordination will contain a large number of small groups (as perplexity can be interpreted as a rough estimate of the expected number of neighbors for a point) and if perplexity is too big, then the groups may be situated closely enough to not be distinct. In either case, a potentially influential point could be made more distinct in the map, but this is not the behavior we hope to identify. Rather, we seek to identify the impact of influential observations in well-calibrated maps, so for methods like t-SNE, we use the

tuning parameters that might be used in real applications. Here, perplexity was chosen to be 10 for the simulations (results from investigation of other perplexity values not shown here).

3.7.4 MNIST

To further investigate the performance of the diagnostic on MDS map as compared to t-SNE, we examined the MNIST data set (LeCun et al., 1998). The MNIST data set is commonly used to assess methods for multivariate or high-dimensional data, and was one of the primary data sets used in Van Der Maaten and Hinton (2008). The data consists of 60,000 images of handwritten digits, each with 28×28 pixels. When transformed into a multivariate data format, the images can be characterized as a data set with $P = 784$ dimensions (one for each pixel in the images). We selected a stratified random sample ($N = 201$) of the original MNIST data set to explore the influence diagnostic, containing a randomly-selected set of 100 “0” digits, 100 “3” digits, and a single “7”, which was selected to be considered as a potential influential point due to the likely differences in the shape of the values. Given this selection, the data set is of dimension 201×784 meaning that it has nearly 4 times as many dimensions as it has rows. Note that this wide structure is very different from the situations examined previously.

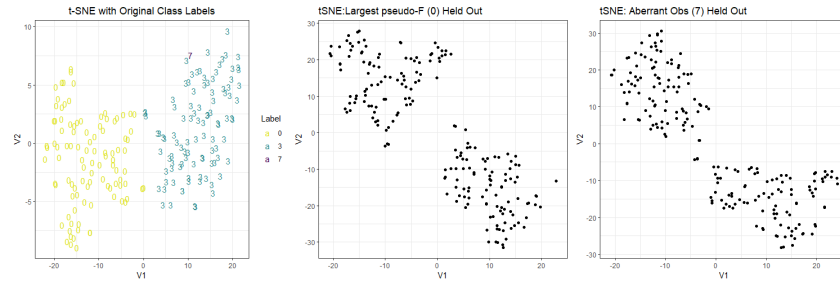
While MNIST is decidedly higher in dimension than most of the other data sets mentioned in this paper, it provides an important example of interpretable data. Typically, each digit (between zero and nine) forms into a distinct group; however, there are a number of digits that may be drawn in non-typical ways or may be written in ways that look like another number (i.e., someone might have drawn a “3” that looks similar to an “8”, etc.). Thus, by observing the specific image of a number that is identified as influential, we might gain better insight into how well the method identifies different numeric patterns.

We ran the influence diagnostic for MDS and t-SNE maps, as shown in Figure 3.18 (with

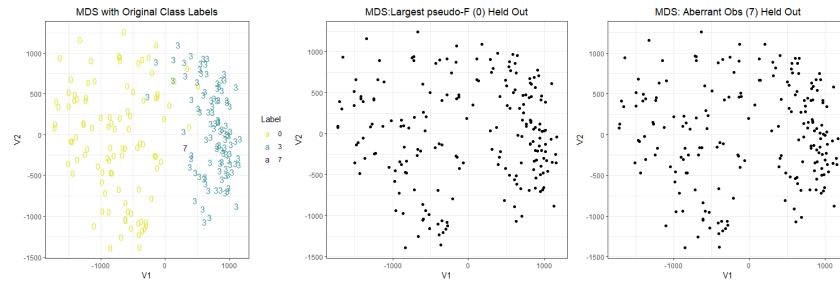
original class labels shown in the left-hand plot in Panels (a) and (b). The resulting pseudo-F test statistics for each observation are shown in Figure 3.19. Interestingly, the MDS maps seem to be unaffected by the inclusion/exclusion of the “7” observation, indicating that the MDS maps are not substantially altered by the inclusion/exclusion of the unusual observation, “7” (which is shown in the far right plot in Panel (b) in Figure 3.18).

Rather, a different observation altogether had the largest pseudo-F statistic: a “0”. The results from the holdout tests indicate that the MDS map did not appear to have been contaminated by any particularly aberrant observations; the pseudo-F test statistics were all on a relatively similar scale. Based on the shape of the MDS maps generated with each observation held out, they do not appear to differ substantively for any of the held-out values; although, it is worth noting that for the first value held out, the map is rotated 180 degrees compared to the other maps.

For the MNIST sample shown in Figure 3.18, t-SNE maps in Panel (a) were evaluated at a perplexity of 20. Based on a visual inspection of a range of perplexity values between 5 and 100, the maps appeared stable across a wide range of perplexity values (results not shown). Thus, a perplexity of 20 was chosen as a reasonably central value for map stability. The pseudo-F test statistics do not get quite as large as those for MDS, and the results suggest that t-SNE does not recognize the impact of any influential values as it appears to place the “7” in with the “3” observations. The pseudo-F statistics range between 0 and about 3. The digit with the largest F statistic identified was a “0” and the potentially influential observation, the “7”, had a very small pseudo-F statistic (and large associated p -value). Observing the shape of the maps in for each holdout observation, we see some differences in shape and rotation due to the nature of how t-SNE creates maps, but there are consistently two well-defined groups that appear to maintain their pairwise distances for the most part. Some of the challenges in identifying influential points in the t-SNE maps could be due to the random variation in map configurations for each holdout map (higher



(a) t-SNE maps with specific observations held out.



(b) MDS maps with specific observations held out.

Figure 3.18: t-SNE and MDS plots showing the original class labels without any holdout observations, and showing the map when either the observation with the largest pseudo-F statistic is removed, or when the aberrant observation (“7”) is removed. Panel (a) shows the t-SNE maps with clearer group separation with Panel (b) showing the MDS maps, which do not separate the groups very clearly. Both maps seem to be largely unaffected by the removal of each observation, possibly because there are 200 observations so no single point exerts much influence over the entire map, or because “7” resides too close to the other digits explored.

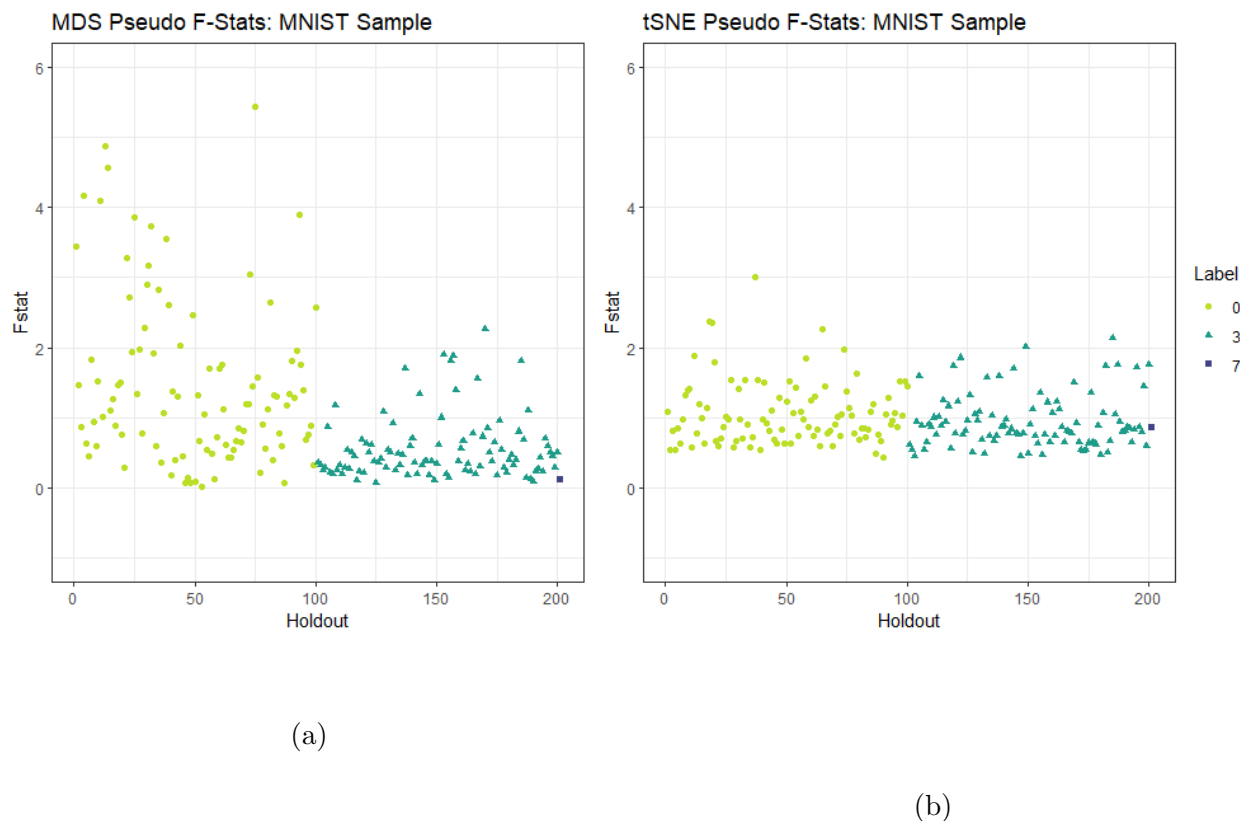


Figure 3.19: Pseudo F-Statistics for MDS (shown in Panel (a)) and t-SNE maps (shown in Panel (b)), each of which do not show evidence that the long “7” included is influential.

variability related to run-to-run map variability).

The results shown in Figure 3.19 suggest that neither method identified any substantially influential points. This may be because the “7” simply did not differ from the observed “0” and “3” digits in the data set, or that the single aberrant point was, in essence, drowned out by the larger number of well-defined, well-grouped observations in the “0” and “3” groups. This result highlights the potential risk that influential points may be harder to detect in settings where the number of observations or potentially the number of features is high (especially if the number of aberrant dimensions is low for the potential influential point). In any case, one critical takeaway is that even in the presence of no detected influential points, the range of pseudo-F statistics for t-SNE is much smaller than that of MDS.

3.8 Conclusions and Extensions

The proposed method uses a similar framework as to the tools available in regression in order to identify potentially aberrant data points that can impact the shape of maps. Critically, the choice of ordination technique can impact the efficacy and sensitivity of maps to influential points. Some methods, like MDS, are highly responsive to influential points that do not look like the rest of the data. Maps resulting from these methods will typically make such points obvious.

Alternatively, t-SNE is not particularly sensitive to single points that are highly dissimilar to others in the data set. It is often not obvious that such observations are present when viewing a t-SNE map. In many cases, the unusual data points are pushed towards the center of the resulting ordination, making them difficult to identify.

Depending on the use case and problem to be solved, this sensitivity (or lack thereof) may be beneficial or potentially problematic. In some instances, it may be necessary to create maps that are relatively stable even in the presence of influential points; the results in this paper suggest that t-SNE might be more effective for creating maps in those situations. However, care should be taken when interpreting results from such maps as there may be substantively different observations that get grouped together in the resulting ordination, like the “7” that was grouped together with the “3” values in the MNIST data.

Much like in regression, we recommend taking an iterative approach to dealing with influential points. It is common practice in regression diagnostics to identify the point with the highest Cook’s Distance, remove it, and re-fit the model to see the new dispersion of observations’ impacts on the model. A similar approach might be useful when creating high-dimensional data maps using these techniques.

We recommend identifying the observation with the largest pseudo-F statistic, removing it, and re-assessing the diagnostic’s results with the reduced set. By cleaning out some of the

most influential points, the resulting maps should be more reproducible, of higher fidelity, and provide potentially more meaningful results for the retained points.

3.8.1 A Note on Sample Size and Dimensionality

Two factors have the potential to greatly impact the computational complexity of generating maps: the dimensionality of the data set and the sample size considered. There are N distinct permutations to explore. In larger sample size scenarios, this makes the diagnostic somewhat computationally expensive. Also, the number of p -values less than a chosen cutoff value is directly related to N because of the permutation-based approach used.

In truly high dimensional data, it is possible that an observation may only differ substantively on a small number (say 1 or 2) of features in a data set with potentially thousands of dimensions. The computational cost of mapping on such a large data set can impact the speed of computing the diagnostic due to computational challenges in making each holdout map. In such a case, it may be useful to consider a method for trimming the number of features considered. Using a method such as penalized matrix decomposition (see Witten and Tibshirani (2010)) to trim noise features might reduce the complexity of the initial feature set used in the map. Additionally, a “whitening” process could be used by initializing one of the mapping methods used here with the first K principal components from a PCA, as is done in t-SNE to reduce computational complexity (Van Der Maaten and Hinton, 2008). Methods for pre-processing the data prior to influence detection is left as an avenue for future research, but might prove the key to computational feasibility on vastly more complex data sets than those considered here. Potential impacts of “whitening” on detection of influential points are unclear.

3.8.2 Future Work

There are several considerations for future work that might be worth considering. Based on the differences in t-SNE maps produced by Campbell et al. (2019), an exploration of the

impact of different intermediate dimensionality reduction step in t-SNE could be interesting. It is certainly possible that using PCA as an intermediate method for reducing the data dimensionality prior to t-SNE, or choosing a different dimension reduction technique, could either enhance or hamper the ability of this diagnostic to identify influential points.

For several of these methods, we initially examined metrics that assess the individual contribution of each feature to the cost function for specific mapping techniques. Interestingly, for t-SNE, the individual cost contributions for the points that were simulated as influential were relatively small, and observations that had high individual cost were often not obviously outlying in the maps created. Some additional research into the dynamics of individual cost contributions—and how tightly these tie to the resulting changes in ordinations when points are included/excluded from maps—could provide an interesting avenue for future work.

Additionally, examining additional mapping techniques may be worthwhile as well. Not all mapping methods are unsupervised; some methods, like Linear Discriminant Analysis (LDA, see Izenman (2008)), contain response information that can be used in driving greater separation between classes in the resulting maps. Extending this method to supervised ordination methods would require potentially re-thinking what it means for an observation to be influential given that class information is available, and the method would need to be compared to existing methods for detecting influence in supervised mapping techniques. Extensions to other unsupervised ordination techniques may be worth consideration as well; while this paper assesses a linear method for doing dimension reduction (MDS) and a non-linear methods (Sammon mapping, isoMap, t-SNE, etc.), there are a bevy of other methods that might be worth assessing in the future, including UMAP, Laplacian Eigenmaps, and others. While our method can run with any of the proposed methods, an assessment of the efficacy of the method would be needed.

CHAPTER FOUR

METHODS FOR SPARSE MONOTHETIC CLUSTERING

Contribution of Authors and Co-Authors

Author: Harmon, Paul

Contributions: Primary Author responsible for the majority of writing and development of analysis and methods used.

Author: Greenwood, Mark C.

Contributions: Provided feedback and direction on statistical methods and drafts of manuscripts.

Manuscript Information

Paul Harmon and Dr. Mark Greenwood

Journal: *Electronic Journal of Applied Statistical Analysis*

Status of Manuscript:

- ☒ Prepared for submission to a peer-reviewed journal
- ☐ Officially submitted to a peer-reviewed journal
- ☐ Accepted by a peer-reviewed journal
- ☐ Published in a peer-reviewed journal

Abstract

Monothetic clustering (MC) is a useful tool for generating interpretable clusters based on divisive bipartitions of multivariate data. When applied to multivariate or higher-dimensional data, it is possible that noise features may contaminate the MC results, necessitating the use of sparse methods to obtain improved clustering results. Additional benefits of feature selection prior to clustering include improved computational performance and potential to improve the monothetic characterization from each resulting cluster. This paper proposes a novel method for sparse monothetic clustering based on a two step process. First, the method uses a penalized matrix decomposition to reweight the distance/dissimilarity matrix (or, equivalently, the input data) and second, the application of MC to the reweighted sparse data to obtain a “sparse” monothetic clustering. Tuning for two parameters, the number of clusters, K , and the degree of sparsity, s , is discussed, and bivariate extensions to existing tuning rules are presented. The methods are demonstrated with a range of applications that illustrate both the challenges and utility of the method as compared to other methods for clustering, both sparse and non-sparse.

4.1 Introduction

Clustering is an important method for unsupervised statistical learning that allows for data-driven identification of similar observations or objects. Consider a data matrix $\mathbf{X}_{N \times P}$ with N observations or objects and P features. Various methods for clustering $\mathbf{X}$ into meaningful groups exist. Clustering has long been a primary tool for data scientists to determine groups in unsupervised data problems (see Kaufman and Rousseeuw (1990) and Everitt and Hothorn (2008)). When no clear group labels are identified *a priori*, clustering can be used to identify common characteristics between observations and to group them into similar groups based either on single characteristics or on relationships between many

characteristics.

In many cases, the data to be clustered may contain both signal-bearing features (i.e., meaningful structure related to groups), and features which may only contribute noise. Sparse clustering attempts to reduce the number of features used as input in a clustering problem in order to drive better results by focusing the clustering algorithm on just the signal-bearing features. In some cases, such as high-dimensional settings where the number of observations, N , is less than the number of data features, P , some clustering algorithms will perform poorly or break down completely (Witten and Tibshirani, 2010). However, the data need not be high-dimensional to pose potential problems for clustering algorithms; in many cases, the true number of signal-bearing features may be less than the total number of features available to be used—these remaining “noise features” can actually hamper the ability of the algorithm to segregate the data into groups that best represent the truth (Brodinová et al., 2019).

We propose a novel method for feature selection in MC using a two-step algorithm: first, selecting relevant features via a sparsification method to generate a set of weights, w , and reweighted dissimilarity matrix, u , and then to perform MC on the reweighted data. This work extends the framework of sparse hierarchical clustering proposed by Witten and Tibshirani (2010) and additionally formalized by Hastie et al. (2015) to the divisive bipartitioning method of Chavent (1998). Our sparse MC method is operationalized in the `monoClust` package (Tran et al., 2022) in the statistical software R (R Core Team, 2022). Three critical advantages can be obtained by using sparse clustering methods; the solution is more interpretable because it relies on fewer features, the cluster solution will potentially be more accurate, and removal of noise features may lead to potentially large improvements in computational performance (depending on the method). Feature selection can also improve the characterization of each cluster and lead to a more useful and interpretable result.

This paper overviews relevant background information about monothetic clustering and

the sparse clustering framework of Witten and Tibshirani (2010). We then propose our new sparse MC methodology and a novel perspective on how to choose optimal tuning parameter values for the degree of sparsity and the number of clusters. We present simulation studies to explore tuning and compare sparse MC to other methods, and evaluate the method on a set of real data sets.

4.1.1 Monothetic Clustering

Clustering requires some measure of distance, or dissimilarity, between pairs of objects. We can define $d(x_i, x_{i'})$ (or simply $d_{i,i'}$) as the measure of similarity between objects x_i and $x_{i'}$ for all of the P features. A common choice for distance is the Euclidean distance (ℓ_2 norm) across the P features, $d_{i,i'} = \sqrt{\sum_{j=1}^P (X_{i,j} - X_{i',j})^2}$. Hierarchical clustering methods take as input $\mathbf{D}$, the dissimilarity/distance matrix of all pairwise $d(x_i, x_{i'})$ across the P features. Other methods, such as K -means or K -medoids, can be applied directly to the data matrix and take the dissimilarity measure into account as part of their objective function. For hierarchical methods, a method for linkage must be defined—this is the way that distances between clusters are defined. In many cases, clustering methods are considered polythetic, meaning that the clusters can be characterized by a shared set of similar characteristics on variables, or monothetic, meaning that the clusters can be characterized by at least one well-defined common property for each cluster.

Monothetic clustering (MC, Sokal and Sneath (1963)) differs from traditional polythetic hierarchical clustering methods in that it makes cluster decisions based on a single property at a time; resulting clusters thus share a set of characteristics rather than simply being “similar” based on a dissimilarity metric that takes several features into account. This shared characteristic property of MC means that the resulting clusters are easier to interpret, generally, than those created by other clustering methods, which may generate clusters in which objects in the same cluster share no common characteristics.

MC methods first arose in the case of binary data (Williams and Lambert, 1959) and have since been known as descendant conceptual clustering methods (Michalski et al., 1981). A method for MC, inspired partly by the classification and regression trees of Brieman et al. (1984), was developed by Chavent (1998). The algorithm generates both a hierarchy of a set of objects (a recursive clustering) as well as a monothetic characterization of each cluster (i.e., a common, interpretable characteristic for observations in each group).

A global criterion, called *inertia*, $I(C_K)$, is defined as the total variability around the cluster centroid, and is used as a target for a clustering algorithm that iteratively searches for optimal splits from each multivariate response (feature). When the chosen dissimilarity metric is Euclidean distance, the *inertia* for cluster C_K is equivalent to the criterion, also called inertia, used in Ward's Method for hierarchical clustering (Ward, 1963).

Let N be the number of objects in a set Ω . Each object is described on P features $X_1, \dots, X_P$, which are weighted equally (although note that this need not be the case, indeed the interest of this work is to reweight the X 's in a data-driven fashion). The inertia is a measure of homogeneity within a cluster $k = 1, \dots, K$ defined as

$$I(C_k) = \frac{1}{n_k} \sum_{i, i' \in C_k, i > i'} d_{i, i'}^2, \quad (4.1)$$

where n_k is the cardinality (i.e., the number of unique observations) in cluster C_k . In our applications, $d_{i, i'}$ is defined as the Euclidean distance measure, but other types of distance or dissimilarity can be used.

Note that minimizing the within-cluster inertia of a partition (i.e., the similarity of objects within each cluster) is equivalent to maximizing the between-cluster inertia (the dissimilarity of objects in separate clusters)—this theme of MC means that the method is similar in its goal to K -means clustering as well as hierarchical clustering with Ward's method as the selected linkage.

Binary splits are typically displayed in a tree diagram that outlines each of the rules, as shown in Figure 4.1. A binary split $s(C_k)$ on a given cluster C_k will result in a partition of the original cluster into two smaller clusters, C_{kL} and C_{kR} (L and R refer to ‘left’ and ‘right’ in the dendrogram produced). The inertia decrease before and after the partition is defined as

$$\Delta(s, C_k) = I(C_k) - I(C_{kL}) - I(C_{kR}). \quad (4.2)$$

The optimal split, $s^*(C_k)$, is the one that maximizes this decrease in inertia and is denoted as

$$s^*(C_k) = \operatorname{argmax}_s \Delta(s, C_k). \quad (4.3)$$

Various simulation studies have been performed on the MC method proposed by Chavent (1998), as well comparisons to other clustering methods (see Chavent et al. (2007) and Chavent and Lechevallier (2006)). Comparing across simulated data sets as well as six real data sets, they found that the MC algorithm outperformed K -means and Ward’s methods in making correct classifications when the number of clusters was small. Additionally, in the case of numerical data sets, MC returned higher quality clusterings (based on a measure of cluster efficacy, proportion of inertia explained) than the other methods when data volumes were higher.

MC provides both a set of clustered groups (a “clustering”) of the data as well as a set of common characteristics for every observation in a given cluster, called a monothetic characterization. This provides a more interpretable solution than polythetic clusterings, which may group observations together that share no common characteristics. Figure 4.1 provides an example of the resulting clusters and a tree diagram of bipartitions as applied to the Palmer penguin data (Horst et al., 2020). This data set contains four measurements

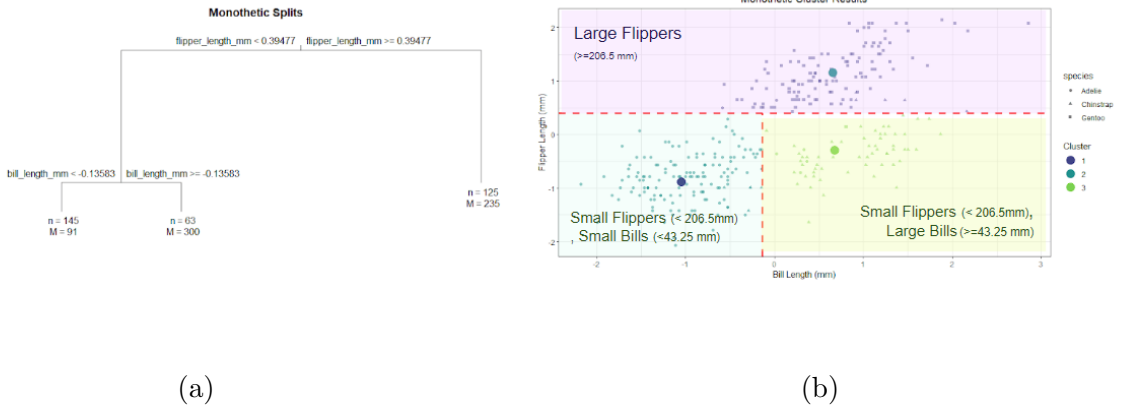


Figure 4.1: Plot of MC of Palmer penguins data set. Two bipartitions are made on flipper length and bill length, shown in Panel (a) and used to generate the three clusters shown in Panel (b) based on the red lines partitioning the space. The clusters are highlighted and the monothetic characterization for each group is annotated over the resulting clusters.

of penguin anatomy for three species (Gentoo, Adelie, and Chinstrap), including the flipper length (mm), bill length (mm), bill depth (mm), and body mass (grams). The method applied to the $P = 4$ standardized variables for the $N = 344$ penguins with Euclidean distance for $K = 3$ splits them into three clusters based on splits made on flipper length and secondary splits based on bill length. The values shown in Panel (a) of Figure 1 are scaled, whereas those reported here are on the original scale. Based on the splits, we refer to Cluster One as the “Large Flipper” group, which contains penguins with flipper lengths greater than 206.5 mm (most of which are Gentoos). Cluster Two refers to the “Small Flipper, Large Bill” group that has flipper lengths less than 206.5 mm and bill lengths greater than or equal to 43.25 mm. Finally, Cluster Three refers to the “Small Flipper, Small Bill” group with flipper lengths smaller than 206.5 mm and bill lengths that are smaller than 43.25mm. Other methods will not provide so clear an interpretation of the clusters, and objects within a cluster might not share those common characteristics, even though the clusters created may be similar.

For situations where either the number of observations or the number of features is

very large, MC can be computationally intensive because it investigates all the possible bipartitions of the features. Two R Packages allow for implementation of MC: `monoClust` (Tran et al., 2022) is used to generate the results in this paper; the `divclust` package can be used to implement MC both for quantitative and categorical variables (Fuentes and Chavent, 2015).

An objective measure of cluster efficacy is needed to assess the efficacy of the following clustering results; we use the Rand index (Rand, 1971) as a measure of similarity between two sets of group labels. Consider two potential partitions of N objects C_1 and C_2 —these may be potentially two clusterings of $K = 2$ or more, a cluster label and class label, or other partitions of some sort. The Rand index is defined as $\frac{a+b}{\binom{N}{2}}$ where a refers to the number of elements that are in the same subset in C_1 and C_2 , and b refers to the number of elements that are in different subsets in C_1 and C_2 . In their paper, Witten and Tibshirani use an alternative measure, Cluster Error Rate (Chipman and Tibshirani, 2005), to describe the quality of clusterings. CER is calculated as $CER = 1 - Rand$. For this, as well as later comparisons of cluster results, we use the Rand index, in order to quantify the similarity of the clusters produced when comparing to known class labels, or to different clustering solutions.

4.1.2 Motivating Example

Witten and Tibshirani (2010) present a motivating example to illustrate why sparse K -means is useful compared to a non-sparse K -means clustering. Simulating from a multivariate normal distribution where two groups differ only by a mean shift on the two features, they demonstrate that K -means as applied to both features will not result in good group discrimination vs. K -means applied only to the axis where the groups are shifted.

Because MC makes bipartitions based on a single feature at a time, in a bivariate case it will either split the data horizontally or vertically. Thus, the situation proposed by Witten

and Tibshirani (2010) must be modified to present some sort of challenge as a horizontal or vertical split completely separates the groups they present. Instead, we consider a tightly-bound, independent, bivariate normal distribution positioned next to a more elongated group simulated from an independent bivariate normal with a shifted mean and more variation in the vertical direction. Both groups are simulated with 500 observations; Group One has means $\mu = (0, 0)$ and variances 1 and 1 in both directions, and Group Two with $\mu = (5, 5)$ and variances 1 and 3, respectively, to make it stretch in the y -direction. The resulting data form a tight, uniform circle positioned next to a long ellipse.

As shown in Figure 4.2, the actual clusters are not completely separated, nor can they be separated with a single vertical or horizontal line. MC as applied to both features simultaneously provides a curiously unintuitive result that splits based on $V2$, assigning one cluster to $V2 > 2.08$ and $V2 \leq 2.08$. This clustering yields a Rand index of 0.86. Alternatively, by using only the first feature, $V1$, and splitting the groups into $V1 > 2.51$ and $V1 \leq 2.51$, the “sparse” MC method is able to find a more intuitive bipartition with a much higher Rand index of 0.97. In such a case, a sparse monothetic method that removes $V2$ from consideration in the clustering not only better captures the cluster membership but also more useful monothetic rule-based characterization of the data based on the more meaningful variable, $V1$.

4.2 Existing Methods for Sparse Clustering

Witten and Tibshirani (2010) introduced a framework for sparse clustering that is further expounded in Hastie et al. (2015). They define the majority of clustering problems as a function of parameters Θ and data $X_{N \times P}$ that take the form

$$\max_{(\Theta \in G)} \sum_{j=1}^P w_j f_j(X_j, \Theta), \quad (4.4)$$



Figure 4.2: Motivating example showing two clustering solutions, where MC results are improved with only one feature used and two cluster solutions. Panel (a) shows the actual clusters, Panel (b) shows the MC using both dimensions that split on feature “V2”, and Panel (c) shows the more intuitive clustering using only feature “V1” both in the distance calculation and as the candidate for bipartitions.

where $f_j(X_j, \Theta)$ is a function of the j^{th} feature X_j , and G is the parameter space to be defined for a specific clustering method. To achieve sparsity in such a clustering method, a two-step method is proposed. First, a matrix decomposition must be performed to provide either a reweighted distance/dissimilarity matrix, or a set of weights, w , that can be applied to the original P features to effectively “sparsify” the original data. Then, the reweighted distance/dissimilarity matrix can be used as a newly sparse input to methods like hierarchical clustering that take the distance matrix as input. In the case of K -means and methods that can be applied to the original data, the weights can be more directly applied.

Witten and Tibshirani first pose the sparse clustering problem in the following form:

$$\max_{(\Theta \in G)} \sum_{j=1}^P w_j f_j(X_j, \Theta) \quad \text{subject to} \quad \|w\|_2^2 \leq 1, \|w\|_1 \leq s, w_j \geq 0, \quad (4.5)$$

where w_j refers to a set of weights that are applied to each of the P features X_j and s is a tuning parameter that controls how sparse the feature set used should be. This provides a general framework wherein the first constraint, $\|w\|_2^2 \leq 1$, assures that at least one feature will be nonzero. The L1 norm, $\|w\|_1 \leq s$, provides penalization that allows weights to be shrunk all the way to 0. One of the key challenges is choosing the right value of s , which controls the right level of sparsity. Generally, smaller s values will lead to a “sparser” solution whereas larger s values will lead to more features being used.

4.2.1 Sparse K -means Clustering

In a K -means setting, this can be directly utilized to optimize the dimensionality of the space used in the K -means algorithm. A sparse K -means algorithm proposed by Witten and Tibshirani (2010) applies sparsity to clustering via estimation using a Lasso-penalized weighting of the variables, w_j , to create a weighted dissimilarity matrix, D^* . The solution below maximizes the between-cluster SS (BCSS) in K -means for a given choice of K :

$$\max_{(C_1, \dots, C_K, w)} \sum_{j=1}^J w_j \left\{ \frac{1}{n} \sum_i \sum_{i'} d_{i,i',j} - \sum_{k=1}^K \frac{1}{n_k} \sum_{i' i' \in C_k} d_{i,i',j} \right\} \text{ subject to } \|w\|_2^2 \leq 1, \|w\|_1 < s,$$

where s is a tuning parameter to control the sparsity and w_j is a set of weights applied to the P features. Essentially, this weights each feature's contribution ($d_{i,i',j}$) to the distance based on the w_j 's.

Much of the research following Witten and Tibshirani's initial paper in this area has focused on improvements to the sparse K -means implementation. Friedman and Muelman (2004) developed an alternative method for achieving sparse distance/dissimilarity matrices, called Clustering Objects on Subsets of Attributes (COSA), that is discussed in Section 4.2.3. Brodinová et al. (2019) introduced a framework for dealing with outliers in the presence of noise features by incorporating a weighting function that penalizes outlying observations, effectively adding weighting to the objects to be clustered as well as the features that were already subject to weights. Arias-Castro and Pu (2017) propose a slightly different alternative to Witten and Tibshirani's approach called Sparse Alternate Sum (SAS) clustering that is applied to K -medoids, instead applying a hill-climbing method that is more directly optimized than the alternating optimization necessary for sparse K -means.

4.2.2 Sparse Hierarchical Clustering

A direct translation of this framework cannot be applied to hierarchical clustering in the way that it is applied to K -means clustering, because hierarchical clustering does not directly optimize a criterion in the above form. This is because HC generates a dendrogram that provides a solution for all potential groupings from the single cluster solution containing all objects to the solution containing N singleton clusters. These have dendrograms have to be cut horizontally to provide a single clustering solution. Witten and Tibshirani (2010)

point out that one could attempt to build a sparse hierarchical clustering algorithm that cuts the dendrogram, calculates the BCSS that results, and optimizes based on that criterion. However, such a case would require a more obvious method for when and how to cut the resulting dendrogram, and the weights might differ across different heights in the tree.

They instead developed a method that recasts the overall (weighted) dissimilarity matrix $\mathbf{U}_{N \times N} = \sum_j d_{i,i',j}$ as the solution to the optimization:

$$\max_{w, \mathbf{U}} \left\{ \sum_{j=1}^P w_j \sum_{i, i'} d_{i, i', j} \mathbf{U}_{i, i'} \right\} \text{ subject to } \sum_{i, i'} \mathbf{U}_{i, i'}^2 \leq 1, \|\mathbf{w}\|_2^2 \leq 1, \|\mathbf{w}\|_1 \leq s, w_j \geq 0, \forall j. \quad (4.6)$$

The solution to the optimization uses components of Witten's sparse principal component proposal (2009). The algorithm is as follows. Let u be a vector of length N^2 containing all elements in $(\mathbf{U}_{i, i'})_{N \times N}$ strung out. Define $\mathbf{D}_{N^2 \times P}$ as the matrix whose j^{th} column contains the N^2 elements of the $(\mathbf{d}_{i, i', j})_{N \times N}$ dissimilarity matrix based on the j^{th} feature. Then we seek to find the solution to the following optimization:

$$\max_{w, u} (u^T \mathbf{D} w) \text{ subject to } \|\mathbf{w}\|_2^2 \leq 1, \|\mathbf{w}\|_1 \leq s, \|u\|_2^2 \leq 1, \forall j, \quad (4.7)$$

where the components w and u are iterated until convergence to get both the optimal weights $\hat{w}$ and the vectorized dissimilarity constraint, u . This criterion is the penalized matrix decomposition (PMD) that informs the sparse principal components criterion proposed in Witten et al. (2009). Both the sparse hierarchical clustering and sparse K -means presented in Section 4.2.1 are operationalized in the R package `sparcl` (Witten and Tibshirani, 2018).

4.2.3 COSA Alternative

An alternative to the reweighted distance/dissimilarity matrix provided by the PMD of Witten and Tibshirani (2009) can be generated using a method called Clustering Objects

on Subsets of Attributes (COSA, Friedman and Muelman, 2004). Motivated by problems in high-dimensional settings, COSA seeks to accomplish two tasks: clustering objects into homogeneous groups while collecting (potentially overlapping) subsets of variables for each group of objects.

Let C_k denote the index of observations in cluster k of K clusters. COSA seeks to solve the following optimization problem:

$$\underset{C^1, \dots, C_K, w}{\text{minimize}} \sum_{k \in K} a(n_k) \sum_{i, i' \in C_{(k)}} \sum_{j=1}^P (w_j d_{i, i', j} + \lambda w_j \log(w_j)), \quad (4.8)$$

over any clustering C and a set of weights $w_1, \dots, w_P \geq 0$ subject to $\sum_{j=1}^P w_j = 1$ with $\lambda \geq 0$ as a tuning parameter. The $a(n_k)$ is defined as a function of the number of elements in the k^{th} cluster, with a common choice being an equal weighting $a(n_k) = \frac{1}{n_k}$.

Minimizing the objective function of Equation 4.8 with $\lambda = 0$ results in a convex combination of attributes with smallest average within-cluster dissimilarity; Witten and Tibshirani (2010) note that this is the only way they were able to “zero-out” weights for any features with the COSA criterion. The weights will concentrate on the features with more dissimilarity, and pathologically, it is possible that a single attribute could receive weight 1 with the other features receiving weight 0.

The COSA procedure is optimized using an alternating strategy that alternates between optimizing with respect to cluster assignment C and the feature weights $w_1, \dots, w_P$. It stops when it finds a local minimum; it is not guaranteed to find the global minimum. Tuning for λ , the scale parameter, which controls how many objects in a given cluster C_k have influence on the weights, is important as results can be sensitive to the choice of λ , and is typically done by comparing clusterings across a grid of λ values (Friedman and Meulman, 2004). COSA generates a reweighted dissimilarity matrix that can be input directly into clustering algorithms; however, Witten and Tibshirani (2010) note that in practice, COSA

rarely results in a sparse set of features and instead tends to spread the weights across them all.

4.3 Methods

4.3.1 Our Proposed Sparse MC Method

Our proposed method for feature selection in MC involves using the two-step algorithm of Witten and Tibshirani and translating it specifically for use with MC. First, the PMD is generated on the original non-sparse input data matrix $\mathbf{X}_{N \times P}$. As described in Hastie et al. (2015) and Witten and Tibshirani (2010), the goal is to simultaneously generate a set of weights $w_j \geq 0 \forall j$ that control the contribution of each of the P features into a reweighted dissimilarity matrix. In the case of standard hierarchical clustering and other dissimilarity-based methods, the reweighted dissimilarity matrix is of primary interest; however, for ease of implementation with `monoClust` software, we reweight the original input data based on the w_j 's. Once the weighting vector w is obtained, a reweighted data matrix can be generated by reweighting on the original multivariate data matrix by multiplying the weights by each of the X_j features to generate $\mathbf{X}^* = [X_1^*, \dots, X_P^*] = [w_1 X_1, \dots, w_P X_P]$. Distances can then be calculated from those reweighted variables to obtain a sparse distance matrix $\mathbf{U}^* = \text{dist}(\mathbf{X}^*)$. In the case of Euclidean distance, the distance matrices obtained from the PMD, $\mathbf{U}_{N \times N} = \sum_j w_j d_{i,i',j}$, is equivalent to $\mathbf{D}^* = \sum_j \mathbf{d}_{i,i',j}^*$ where $\mathbf{d}_{i,i',j}^*$ is the feature-wise Euclidean distance matrix obtained from the reweighted data $\mathbf{X}^*$ as shown in Figure 4.3. This result allows us to use the reweighted data $\mathbf{X}^*$ as an input to monothetic clustering directly rather than passing in the sparse distance matrix $\mathbf{U}$, meaning that this first step could be extended both to distance-based methods as well as methods that work on the original dataset such as K -means, partitioning around medoids (PAM, Kaufman and Rousseeuw (1990)), or even the model-based clustering methods of Fraley and Raftery (2002).

Other methods can be used in this first step, including the COSA method, to obtain a

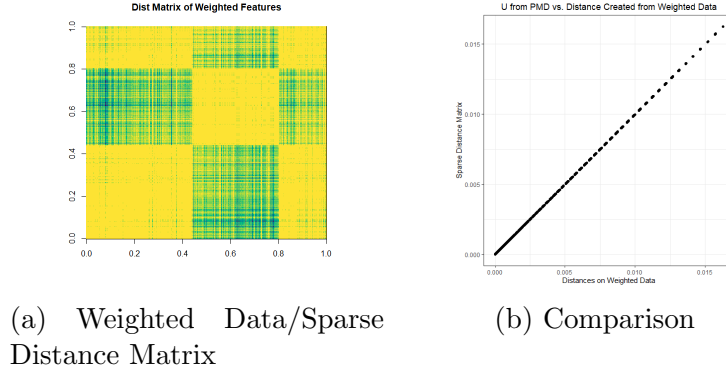


Figure 4.3: Heatmap showing the distance matrices created when using Euclidean distance from the sparse clustering, or equivalently taking the Euclidean distance matrix of the weighted data, in Panel (a), and the plot the vectorized distance matrices plotted against each other in Panel (b) indicate that, for Euclidean distance, the weights can be applied to the original data and input to the clustering.

reweighted dissimilarity matrix with an associated set of weights. Note that while the PMD has been shown to be effective in many cases (Witten and Tibshirani, 2010), other methods beyond those explored here may be useful for “sparsifying” the input data and generating weights. Additionally, while we do not consider weighting the original observations or objects to be clustered, research suggests that this might be helpful in the presence of outliers (Brodinová et al., 2019). We leave this for potential future work.

Since the R package `monoClust` (Tran et al., 2022) takes as input the original data matrix, our method essentially provides a wrapper around `monoClust` that calls it, using the reweighted variables in the clustering. If $w_j = 0$ for any X_j , that feature is not considered for bipartitioning, as well as adding no distance to the distances between observations.

We perform sparse MC on standardized (mean = 0, SD = 1) data before weighting, in order to avoid having different scales of variability input to the PMD step when sparsifying the input distance/dissimilarity matrix. Note that the number of potential bipartitions is reduced compared to non-sparse inputs to the MC algorithm; moreover, the splits are based on the scale of the reweighted data. To provide more interpretable dendrograms and results,

the resulting splits should be visualized on the variables divided by their w_j 's, and then un-standardized to return to the original units of the variables.

4.3.2 Selection of Tuning Parameters

In the case of sparse methods for MC, optimal solutions are based on two parameters:

- K : The number of clusters, and
- s : Degree of sparsity (this controls the number of features being set to zero).

Non-sparse clustering methods use all available features in generating a given clustering solution, thus the primary concern for most methods is determination of the optimal K value. Various methods exist to optimize the number of clusters K . These include the Elbow method for looking at a plot of within-cluster dispersion (see Everitt (2012)), or picking the K that provides the largest average silhouette statistic (Rousseeuw, 1987) or Caliński and Harabasz pseudo-F statistic (1974). These methods can be used to identify the optimal number of clusters as long as $K \geq 2$ but do not inform the critical question of whether clustering is reasonable or not (i.e., does $K = 1$). Tibshirani et al. (2001) define the Gap statistic as a statistic for estimating or selecting the optimal number of clusters for K -means and hierarchical clustering. The Gap statistic is defined for a given choice of K as

$$Gap_N(K) = E^*\{\log(W_K)\} - \log(W_K), \quad (4.9)$$

where W_K is a measure of observed within-cluster dispersion for a given clustering solution with K clusters, and $E^*\{\log(W_K)\}$ is the expected value of the within-cluster variability when clusters are generated from a reference distribution with no clusters present. When the distance used is Euclidean, W_K is defined as:

$$\sum_{k=1}^K \frac{1}{n_k} \left(\sum_{i,i' \in C_k} d_{i,i'}^2 \right), \quad (4.10)$$

where n_k is the number of observations in each cluster. Thus, the Gap statistic is related to the inertia calculation used to drive bipartitioning in MC.

The idea underlying the Gap statistic is that when clustering solutions are “good”, there should be a large difference, or “gap”, between the observed within-cluster dispersion that takes advantage of real structure vs. the reference distribution that is built assuming there is no cluster structure for the same number of clusters. The Gap statistic is calculated across a grid of K values in R using the `clusGap` function in the `cluster` package (Maechler et al., 2022). Various rules have been developed for tuning or “optimizing” K values associated with the “optimal” Gap statistic (and are operationalized within `clusGap`), and are discussed below:

- Global Max: Choose K value such that overall Gap statistic value is maximized Maechler et al. (2022).
- First Max: Choose the K -value of the first local maximum Maechler et al. (2022).
- Tibshirani 1SE: Choose the smallest value of K such that $Gap_N(K) \geq Gap_N(K+1) - s_{K+1}$, where s_{K+1} is the standard error of the Gap statistic defined for $K+1$. More explicitly, s_{K+1} is defined as $\sqrt{1 + \frac{1}{B}}SD(K+1)$ where B is the number of reference data sets generated and $SD(K+1)$ is the standard deviation of the within-cluster dispersion measures for the $K+1$ as initially proposed by (Tibshirani et al., 2001).
- First SE Max: Proposed by Maechler et al. (2022), choose the smallest Gap statistic value such that it is not less than the first local maximum minus its associated standard error, choosing $Gap_N(K) \geq localmax(Gap_N(K)) - SE(localmax(Gap_N(K)))$.
- Global SE: Following Dudoit and Fridlyand (2002), choose the smallest $Gap_N(K) \geq max(Gap_N(K)) - SE(max(Gap_N(K)))$.

Indeed, no method for identifying K is perfect; the Gap statistic was found by Sugar and James (2003) to fail to detect a clear four-cluster solution generated from distinct exponential distributions, and Dudoit and Fridlyand (2002) note that the Gap statistic has been found to overestimate the optimal number of clusters in some applications. Numerous improvements have been proposed to the Gap statistic, including the weighted Gap method of Yan and Ye (2007) that leverages a weighted version of the within-cluster sum of errors as the intra-cluster dispersion measure.

Gap has additionally been used to tune for s in sparse clustering implementations. In their version of sparse hierarchical clustering, Witten and Tibshirani (2010) use a modified Gap statistic based on permuted data to generate a tuning parameter for s only, with K fixed. They denote it $Gap_s(K)$, and it is defined for a given choice of K as

$$Gap_s(K) = \log(B^{sK}) - \frac{1}{A} \sum_{a=1}^A \log({}_aB^{sK}), \quad (4.11)$$

where A is the total number of permutations used to calculate the reference distribution and ${}_aB^{sK}$ is the weighted between cluster sum of squares for a given reference distribution. The solution presented by Witten and Tibshirani is reliant on knowing the true value of K as it does not account for the problem of having uncertainty in the number of clusters. Some work has been done on tackling that problem in the case of sparse K -means.

In a setting where the Gap statistic can vary across s in addition to K , the situation is more complicated. As an example, consider fixing s to be small, in this case $s = 1.1$, in order to induce a very sparse feature selection. With a small value of s , it is possible to reduce the number of features used by the MC to either a single variable, or close to one, so that all of the potential bipartitions from the MC are limited to a single feature (and the inertia it is trying to partition). We consider a simulated data set to illustrate such a situation, where data across six groups are simulated from independent normal distributions across

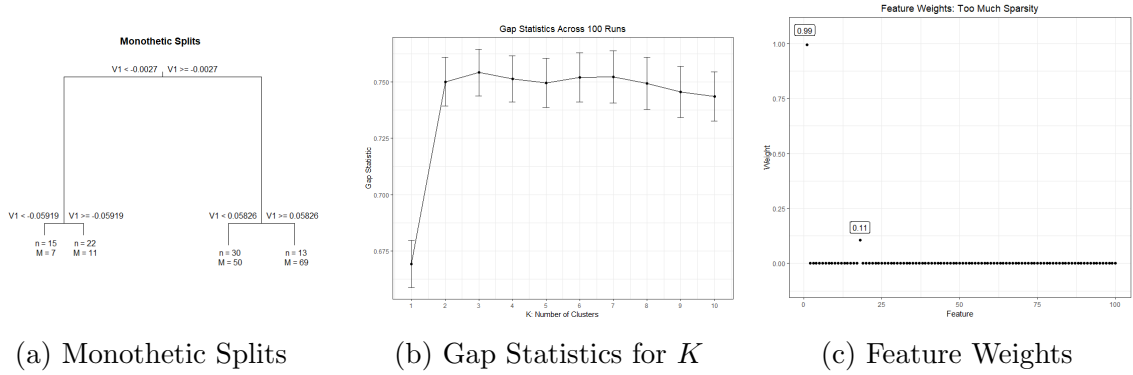


Figure 4.4: Monothetic bipartitions for a simulated data set with $N = 80$, $K = 6$, shown in Panel (a), $Gap_N(K)$ values across potential values of K in Panel (b), and feature weights in Panel (c) shown for a scenario that is too sparse. The first 20 features are “signal” bearing whereas the other 80 features do not vary across groups. While the overly sparse solution does remove the noise-bearing features from consideration in the MC, it also removes nearly all of the signal features and reduces the features available for splits to two (with primary weight on feature V1).

$P = 100$ features with only the first 20 “signal” features varying across each of the means $\mu = [42, 46, 50, 54, 58, 62]$ and the additional 80 “noise” features that do not vary by group. Each group is simulated with 20 observations, resulting in $N = 120$. Figure 4.4 illustrates such a situation—the sparsity parameter $s = 1.1$ is set to be too punitive and leads to the complete removal of all but two features; feature V1 contains $\frac{.994}{1.1} = 90\%$ of the feature weight and thus defines most of the variability present in the reweighted distance matrix, with the remaining 10% going to feature V18. $Gap_N(K)$ appears to underestimate the true number of clusters as it is optimized at $K = 3$ clusters, using the Global Max rule. Thus, when the reweighted data is input into the MC algorithm, the Gap-optimized 3-cluster solution makes 2 recursive splits using only feature V1. In this example where too much penalty is applied, K is too small, and the MC makes splits on only a single feature. Thus, real signal appears to have been removed along with the noise features.

This underscores the need to get both tuning parameters right—and because MC makes bipartitions on a single feature at a time, an overly sparse solution may prevent access to

important signal variables that aid in partitioning the full space and be more impactful for MC than in a polythetic situation. Brodinová et al. (2019) devised a method to tune both parameters simultaneously and suggest that choosing either s or K first will lead to incorrect group identification. They propose using a modified Gap statistic $Gap_{s,K}^*$ that is measured across a grid of both K and s values, and additionally weights the observations as well as the features to reduce the potential impact of outliers. This simultaneous exploration of tuning can then be presented in plots of the Gap statistic shown varying either across K with different lines used for varying values of s , or vice-versa. We find it more natural to present the x -axis of the Gap statistic plots to vary by K , but present both options. Brodinová et al. (2019) directly compare tuning based only on $Gap_N(K)$ using the Tibshirani 1SE rule to optimmizing both s and K based on the maximum value (Global Max) as part of their simultaneous tuning process (and using an observation-weighted Gap statistic, which we do not use here).

We consider using $Gap_{s,K}$ for both parameters s and K , respectively, defined as

$$Gap_{s,K}(s, K) = E^*\{\log(W_{s,K})\} - \log(W_{s,K}), \quad (4.12)$$

where $W_{s,K}$ refers to the given within-cluster dispersion for a clustering with K clusters chosen and a sparsity parameter of s . These rules must be extended to consider both comparing K values within a given choice of s^* , or across a grid of reasonable values for K grouped by values of s . Brodinová et al. (2019) tuned by choosing the optimal $Gap_{s,K}$ across a grid of s values based on a modified 1-SE rule applied for each value of K , and then chose the combination of s and K such that $Gap_{s,K}$ was maximized. We instead propose a simultaneous, holistic assessment of the resulting plots, such as those in Figure 4.8, which can be arranged so that $Gap_{s,K}$ varies across an x -axis for K , with various lines representing the s values, or vice-versa.

Additionally, we propose several novel bivariate extensions of the Gap rules that can be considered in choosing values of s and K simultaneously. Because some of the univariate Gap statistic rules consider picking either the smallest s or smallest K within a standard error of the maximum Gap statistic, we have to consider either sorting across K or s after picking an optimal value for the other parameter based on Gap. This leads to an extension of two rules per univariate rule in some cases, denoted “KS” when the Gaps are grouped by s values and sorted by K prior to picking the optimal value, and “SK” when grouped by K and sorted by s . Generally, we find that “SK” rules tend to prioritize tuning for sparsity and “KS” rules tend to prioritize tuning for number of clusters based on the way that the sorting is performed. A definition of each of the rules is given in Table 4.1. Note that these differ slightly from the two-step process favored by Brodinová et al. (2019), although the closest comparison would be the Tibshirani SK rule.

4.4 Simulation Studies

4.4.1 Selecting Tuning Parameters

In line with the methods used for simultaneously identifying K and s in Brodinová et al. (2019), we opted to use the Gap statistic for assessment of both tuning parameters. Much of the previous literature pertaining to MC methods focuses on methods for determining the optimal number of clusters that do not address the fundamental question of whether clustering should be considered at all ($K > 1$ vs. $K = 1$) (Tran, 2019).

In that vein, we first assess the viability of the Gap statistic and its tuning rules on each of the parameters independently, holding the other at either a known or reasonable value. An assessment of the Gap statistic specifically for MC does not currently exist in the literature, and has the benefit of both establishing a baseline efficacy for tuning each parameter as well as helping to inform which way of tuning is best for MC specifically.

Bivariate Gap Rule	Definition
Global Max	Chooses the combinations of s and K that generate the largest Gap statistic value observed across the range of values considered.
Tibshirani.SK	Sorts the Gap values by K within s -values (arranged in increasing order). Then, picks the optimal values of s and K according to Tibshirani's 1SE rule.
Tibshirani.KS	Sorts the Gap values by s within K -values (arranged in increasing order). Then, picks the optimal values of s and K according to Tibshirani's 1SE rule.
GlobalMaxSE.SK	Identifies the Global Max Gap statistic and sorts all combinations of s and K with Gap statistics within 1-SE of the Global Max. Optimal value is based on sorting by s within grouped K -values (arranged in increasing order).
GlobalMaxSE.KS	Identifies the global Max Gap statistic and sorts all combinations of s and K with Gap statistics within 1-SE of the Global Max. Optimal value is based on sorting by K within grouped s -values (arranged in increasing order).

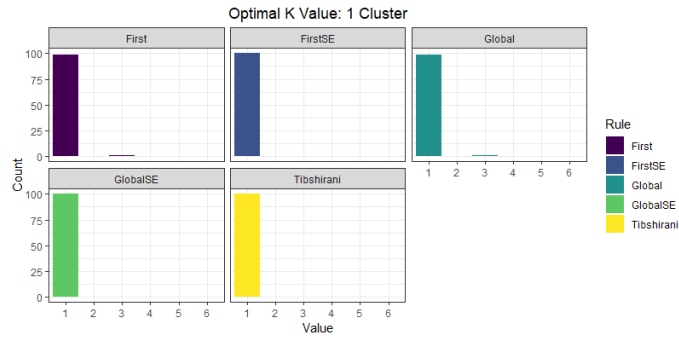
Table 4.1: Bivariate Gap statistic rules considered for tuning for s and K .

4.4.2 Simulation 1: Use of Gap To Optimize K (With No Noise Features)

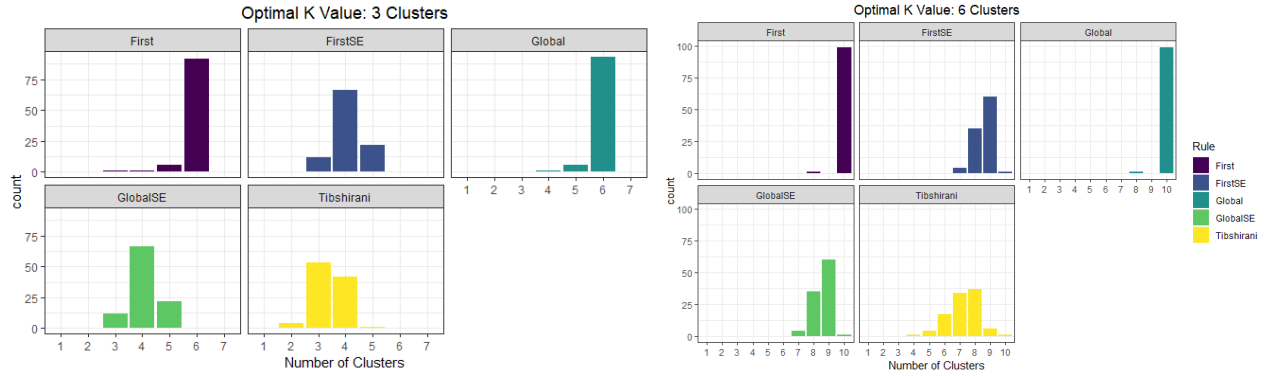
Our first simulation considers a data set with a known number of clusters in order to test the efficacy of the Gap statistic in properly tuning K for MC methods specifically. We

simulate no noise-bearing features, meaning that the groups vary across all of the features in the simulated data sets. First, a one-cluster simulation is considered, based on independent normal distributions with the same mean and variance with $N = 90$ and $P = 20$. For comparison, we examine scenarios with three and six-clusters, where the data are simulated from independent normal distributions with equal samples sizes in each group, $N_K = 30$. Each of the group means are the same across $P = 20$ features. We do this in two scenarios, one with $K = 3$ groups (resulting in $N = 90$) and the other with $K = 6$ groups (resulting in $N = 180$). In the three-group version, group means are $\mu = [42, 46, 50]$ for all 20 features. In the six-group simulations, group means are $\mu = [42, 46, 50, 54, 58, 62]$ for all 20 features. For both versions, the variance is held constant for each variable at $\sigma^2 = 21$ (and 0 for the covariances). The simulated data are then standardized before clustering.

The results are presented in Figure 4.5, which shows a bar chart of the optimal K chosen by each method, much in the same way as Brodinová et al. (2019) explored tuning parameters for their robust sparse K -means algorithm. When only one group is present (Panel (A) of Figure 4.5), the Gap statistic appears to identify this clearly, indicating that it performs reasonably well at minimally assessing the clusterability of the data with MC. The $K = 1$ value was never selected in any of the simulations where multiple groups were simulated (i.e., Panels (b) and (c) of Figure 4.5). Table 4.2 gives the percentage of correctly-identified K -values across the different rules for assessing optimal Gap values for the three and six-cluster solutions. These results appear to suggest that the efficacy of the Gap statistic decreases when tuning for K as the number of clusters increases. There also appears to be some disagreement across the rules; in both situations the Tibshirani 1SE rule appears to perform the best, while the other rules tend to overestimate the true number of clusters—sometimes substantially.



(a) One Cluster Version



(b) Three Cluster Version

(c) Six Cluster Version

Figure 4.5: Optimal K values across different rules for the Gap statistic across 100 simulations. Panel (a) gives the one-cluster version, which chooses a single cluster for all simulations except one (for the First and Global Max rules). Panel (b) shows 3-cluster solution, and Panel (c) shows 6-cluster solution. Note that the $K = 1$ solution was never selected when multiple groups were simulated in Panels (b) or (c).

Percent of Simulations Selecting Correct K Value						
True K	Global	First	Tibshirani	First	SE	Global
Value	Max	Max	1SE	Max		SE
3 Clusters	0.00	0.01	0.53	0.12		0.12
6 Clusters	0.00	0.00	0.17	0.00		0.00

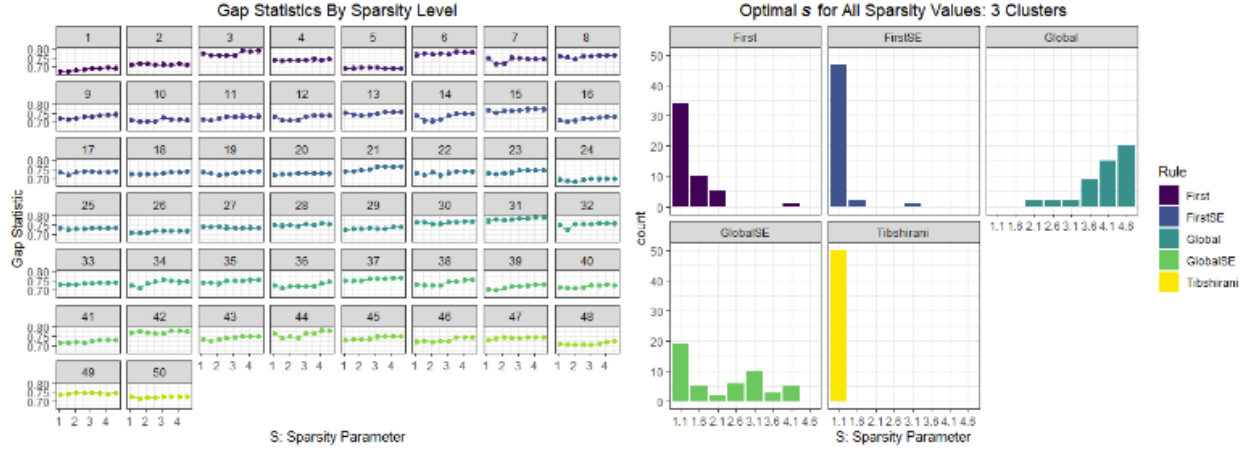
Table 4.2: Percentage of correctly-identified K -values across 3 and 6-cluster simulations for each of the Gap statistic tuning rules.

4.4.3 Simulation 2: Use of the Gap Statistic to Optimize s with fixed K

The following simulation is an extended investigation of the Gap statistic as presented in Witten and Tibshirani (2010), specifically focused on MC. In this setting, we fix the number of clusters at either three or six and only consider MC solutions with its respective number of clusters selected. Similar to the studies shown in Simulation 1, no previous literature exists to assess the effect of the PMD-based sparsification process on MC specifically; rather, most work focuses on either K -means or standard hierarchical clustering methods.

Similar to Simulation 1, we simulate data from independent multivariate normal distributions with varying means across three groups across a total of $P = 100$ features, 20 of which differ across the groups and 80 of which do not differ across groups. In the three-cluster version we allowed the means to vary across $\mu = [42, 46, 50]$ and in the six-cluster case, means vary across $\mu = [42, 46, 50, 54, 58, 62]$ with both cases having the same variance of $\sigma^2 = 21$ (and 0 for covariances). The 80 features that do not vary across groups are all simulated with the same mean. We assess the efficacy of the Gap statistic on these data across 50 simulations, each of which were standardized prior to tuning and clustering.

Panel (a) in Figure 4.6 shows the individual Gap statistic profiles across a range of s -values for each of the simulated realizations; while each simulated data set has some variability, the patterns in the Gap statistic appear to be relatively similar across runs.



(a) Gap Statistics across runs.

(b) Optimal s across Gap rules.

Figure 4.6: Panel (a) shows the plot of Gap statistics for 50 simulations across spectrum of s values on simulated data with three clusters. Panel (b) shows the optimal s -value across 50 simulations based on each rule.

Panel (b) of Figure 4.6 shows a bar plot of the optimal s value chosen by the Gap statistic when employing each of the rules defined in Section 4.3.2. Witten and Tibshirani (2010) continued simulations for sparse hierarchical clustering that showed mixed results relative to performance; our results appear to be similarly mixed and somewhat variable depending on the rule chosen. The First, Global, and Global SE rules tend to be highly variable in choosing optimal tuning parameters, whereas Tibshirani 1SE and the First SE rule of Maechler (2019) tend to favor sparser (often the sparsest possible) solutions.

Another assessment of tuning for sparsity can be obtained via examination of the feature weights obtained across each simulation. Table 4.3 provides the percentage of non-zero weights applied to signal-bearing features and noise features, and Figure 4.7 plots each of the weights for all of the 50 optimal tuning parameter values across the 50 simulations. For each of the rules, the Gap statistic performs reasonably well at identifying at least some of the signal-bearing features (to the left of the dashed line, features 1-20), and down-weighting

most of the noise-bearing features. The mean percentages of nonzero signal features given weight in Table 4.3 are relatively low because for most of the rules, a sizeable proportion of signal-bearing features are zeroed out. This is not surprising as the PMD has the ability to down-weight both the true noise variables as well as some of the signal-bearing features with less associated variability.

Average Signal/Noise Feature Identification		
Rule	Mean Percentage of Nonzero Signal Features	Mean Percentage of Spurious Noise Features
First	24.0%	1.4%
First SE	19.0%	0.4%
Global	94.3%	12.7%
Global SE	71.5%	2.6 %
Tibshirani 1SE	10.0%	0.4%

Table 4.3: Table showing the detection rates for signal-bearing features across the simulations, as well as the mean spurious detection rates for noise features for each of the single variable Gap tuning rules. Results based on 50 simulations.

The Global Max rule appears to have the most weight attributed to noise features, and it captures the highest percentage of signal-bearing features, on average, followed by the Global SE rule of Dudoit and Fridlyand. Both rules appear have more variability in the weights generated as compared with Tibshirani’s 1SE rule that assigns nearly the same weight to the same features consistently. It is important to note that across each of the simulations, most “optimal” tuning parameter values still assigned some nonzero weight to noise features; however, when viewed from the perspective that the sparsification process is essentially “cleaning up” the space of features, even the most liberal method (Global Max) appears to remove much of the influence of the noise features, even if they are less frequently

given weights of 0 and shrunk out of the distance matrix completely.

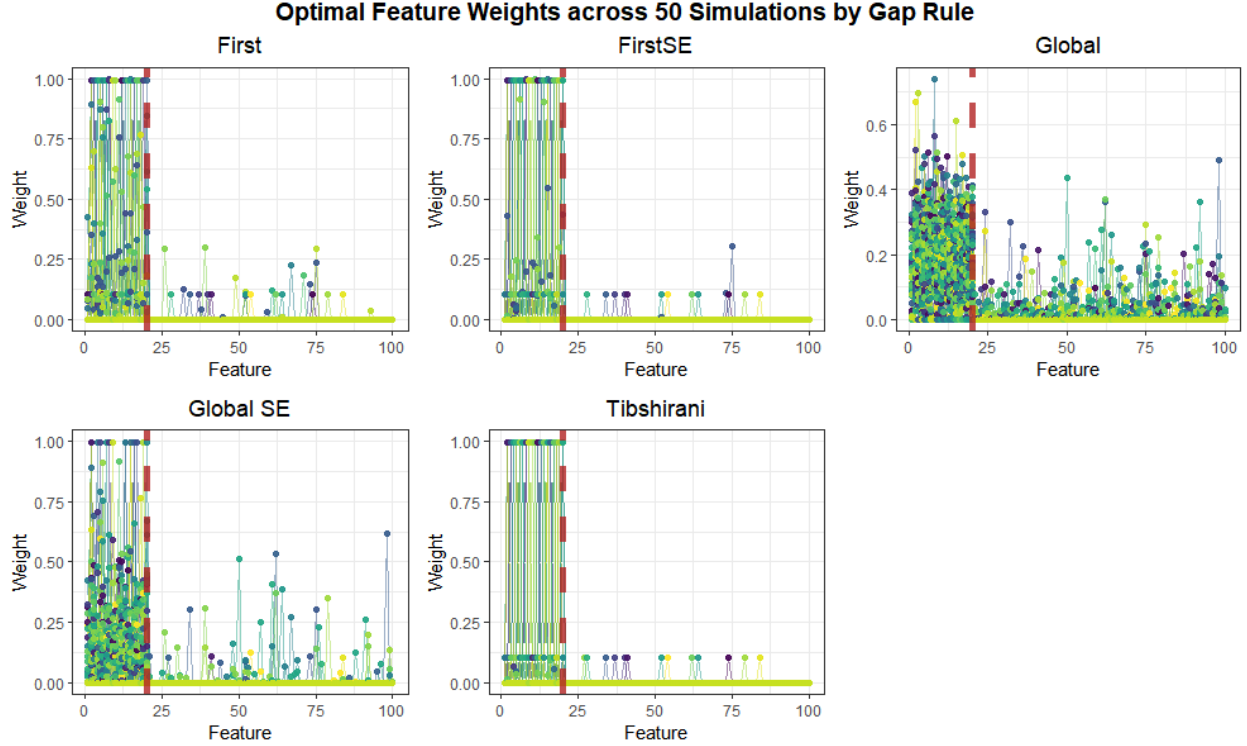


Figure 4.7: Optimal feature weights evaluated for each of the Gap rules for all 50 simulated realizations applied to MC with K held constant. The first 20 features (left of the red dotted line) are the signal-bearing ones, and the last 80 are noise features.

4.4.4 Simulation 3: Simultaneously Choose Tuning Parameters

Similar in setting to the simulations of Witten and Tibshirani (2010), we simulate 4 equally-sized groups with $N_k = 20$, $N = 80$, and $P = 100$, from independent multivariate normal distributions, of which only 20 signal features vary across the groups ($\mu = [42, 46, 50, 54]$ with $\sigma = 15$ for all variables prior to scaling), and constant means of 48 for the non-varying 80 noise features. The data sets being simulated are technically high-dimensional, given that they have dimension 80×100 .

Rather than taking the approach of trying to optimize s or K individually, we instead adopt a simultaneous approach for tuning and apply the set of bivariate extensions to the

rules overviewed in Section 4.3.2. The goals of this simulation are twofold: First, we wish to empirically compare rules for tuning the two-parameter Gap statistic to recommend a method for simultaneous tuning across s and K . Additionally, we seek to highlight that properly-tuned sparse MC will provide better grouping as compared to the MC on entire set of features. By contrast, we additionally assessed the Gap statistic when no distinct clusters exist (i.e., $K = 1$), and all rules for the Gap statistic tended to favor the $K = 1$ values (results are not shown).

Figure 4.8 shows the Gap statistic values for both tuning parameters for one of the 100 simulations, presented in the Panel (a) with the x -axis representing the K values, and in the Panel (b) varying over the s values. We find that the latter plot style better informs the initial question of whether clustering should be performed or not, because the one-cluster solution is presented as a line across all potential sparsity parameter values. In the example in Figure 4.8, it is clear that on simulated data with clear structure, there is evidence of more than a single cluster because the Gap statistic values for the one-cluster solution are clearly less than any other K value considered, for any level of sparsity.

The feature weights for the Gap-tuned sparse MC are shown in Figure 4.10. Table 4.4 provides the mean number of signal-bearing features that were assigned nonzero weight along with the mean spurious nonzero weight assigned to noise features for each of the bivariate Gap rules. The results for the bivariate case are somewhat similar to those shown in Table 4.3 for the univariate Gap tuning rules.

For each of the rules, the choice of weights appears to be reasonable; we would expect the first 20 features to have non-zero weights and the last 80 noise features should be set to 0. The Global Max typically assigns more weight to the right number of features, but it does so at the cost of also assigning more spurious weight to noise features. Additionally, clustering solutions chosen using the Global Maximum rule tend to suffer less from “honing in” on a single feature, instead rarely assigning weights near one to individual features, like

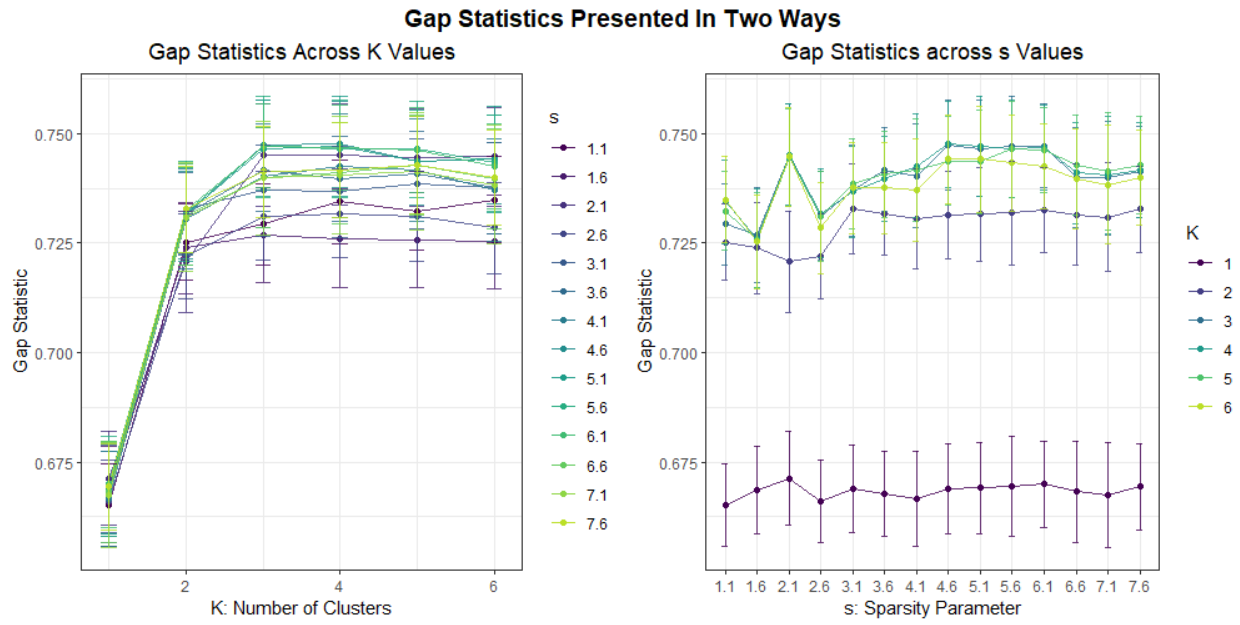
(a) Gap Plot by K .(b) Gap plot by s .

Figure 4.8: Gap statistics across s , K visualized in two ways for a single simulated data set. Panel (a) shows Gap varying across K -values and colored by s . Panel (b) shows plots varying across s (left is more sparse) and colored by K .

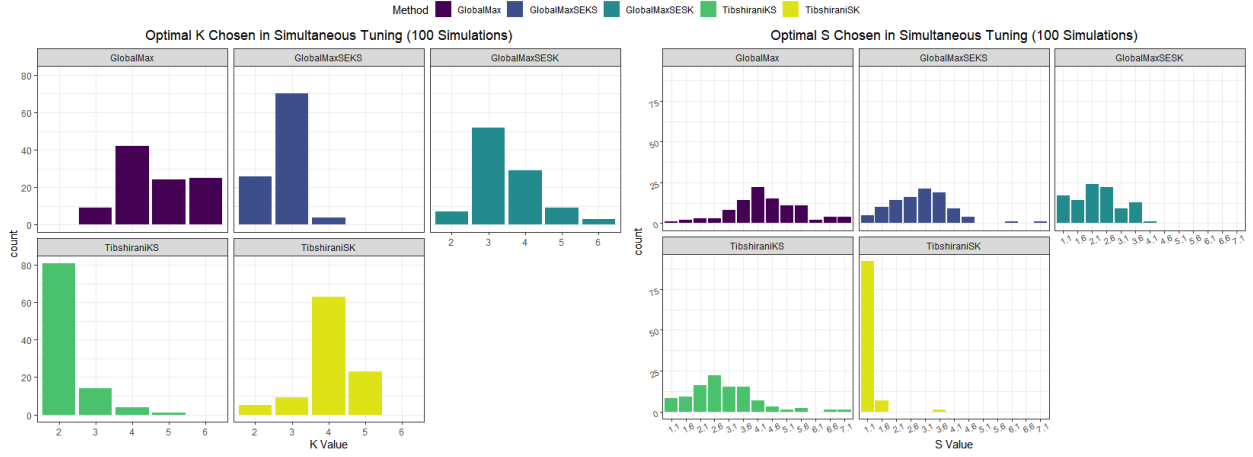
(a) Optimal K (100 runs).(b) Optimal s (100 runs).

Figure 4.9: Panel (a) contains the optimal number of clusters and Panel (b) contains the optimal sparsity parameter values chosen by each of the rules across 100 simulations. The $K = 1$ solution is not shown in Panel (a) as it was never selected as an optimal choice for any of the rules. Note that Tibshirani SK clearly favors a sparser solution and appears to recover the correct four cluster solution more often than other rules.

the other methods tend to do. Tibshirani SK and Global Max SE KS rules assign almost no spurious weights, but this may be because they tend to favor sparser solutions with most of the weight applied to only a small number of features.

Average Signal/Noise Feature Identification			
Rule	Mean Percentage of Nonzero Signal Features	Mean Percentage of Spurious Noise Features	
Tibshirani KS	11.9%	0.01%	
Tibshirani SK	57.2%	4.1%	
Global Max	89.7%	24.1%	
GlobalMaxSE KS	42.1%	0	
GlobalMaxSE SK	61.1%	2.5%	

Table 4.4: Table showing the detection rates for signal-bearing features across the 100 simulations, as well as the mean spurious detection rates for noise features for each of the bivariate Gap tuning rules.

Interestingly, the results of both simulations appear somewhat mixed; the two-dimensional use of the Gap statistic has varying success at identifying K and s depending on the rule used. None of the rules appear to clearly outperform the others. Each of the rules appeared to have relatively variable results in estimating the correct number of clusters, and the only methods that got $K = 4$ right more often than other values were the Global Max and Tibshirani SK. Interestingly, Tibshirani KS, which is intended to prioritize getting K right, selected a smaller value of $K = 2$ as optimal most frequently. In terms of sparsity, both of the Tibshirani rules both tended to be biased towards a more sparse solution (i.e., smaller s values); in particular, Tibshirani SK favored a sparsest solution much of the time. The other rules tended to favor less sparse solutions but all appeared to induce at least some sparsity as they picked more middling values of s , but were inconsistent in terms of picking

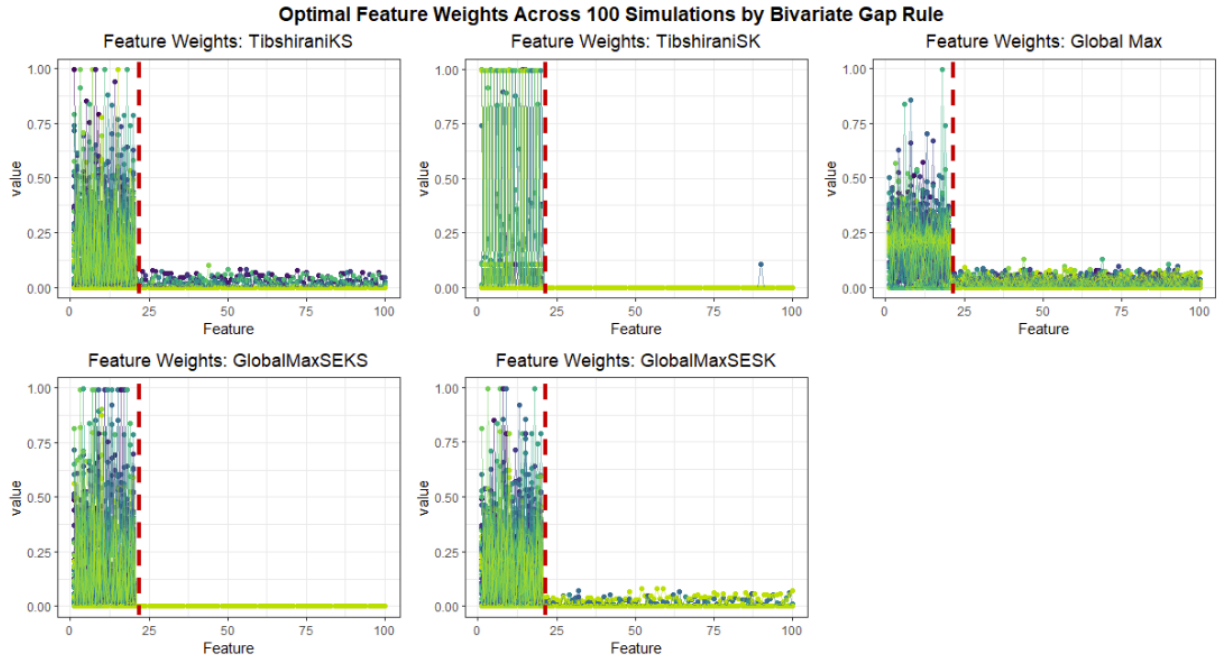


Figure 4.10: Weights of features from Gap-tuned sparse MC across 100 simulations for each of the rules. The first 20 features (left of the dotted red line) are “signal” bearing and vary across groups, whereas the remaining 80 features are “noise” features. The percentages of signal-bearing features with nonzero weight vs. noise features spuriously assigned nonzero weight are given in Table 4.4.

Rand Indices based on Bivariate Gap Rules		
Bivariate Gap Rule	Mean(Rand)	SD(Rand)
Tibshirani.SK	0.714	0.036
Tibshirani.KS	0.669	0.042
Global Max	0.817	0.028
GlobalMaxSE.SK	0.728	0.042
GlobalMaxSE.KS	0.747	0.038

Table 4.5: Mean Rand indices with standard deviations in parentheses for each of the tuning methods used. Results based on 100 simulations.

a value for s .

Finally, we assess which method appears to generate the best cluster structure. Clusterings were generated with each of the “optimal” parameters based on the rules and their Rand indices are compared to the known groups in Table 4.5. Perhaps counter-intuitively, the Global Maximum method actually appeared to result in the most accurate clusterings, despite often generating more clusters than necessary. The other methods tended to perform similarly, although it should be noted that Tibshirani KS performed the worst among the methods, at least in terms of Rand index.

These results indicate that the multivariate behavior of these rules mostly align with the univariate rules, at least for MC. The mixed performance of each of the Gap rules indicates that the none of the rules for determining the “right” tuning parameters is always optimal; however, the plots generated in Figure 4.8 appear to be the most important tool used in determining the right value of K and s in that they make a few conclusions clear. First, the one-cluster version produces smaller gaps at all levels of sparsity, indicating that the data form several distinct groups (i.e., clustering is reasonable). Second, they allow for diagnosing different behaviors of the Gap curve across K and s that might inform the tuning

process. The Gap statistic still allows for single-step tuning that can reasonably answer whether clustering is reasonable (i.e., the one vs. two or more cluster solution), so even with mixed results in efficacy, we believe that it provides a better tool for tuning than some of the other techniques that exist for choosing K . Based on this, we recommend a more holistic approach when tuning: First, create the plots across K and s and second, compare the optimal values of K and s to plots produced for each of the rules. Then, pick the values that seem most reasonable based on the results obtained, considering the results from the rules as information to assist in making a final decision on s and K .

4.4.5 Simulation 4: Comparison with Other Methods

Using the same data as Simulation 3, we assess the quality of resulting clusterings across several methods. We compare the resulting Rand index results for each of sparse MC, non-sparse MC, sparse hierarchical with complete linkage (implemented in `sparcl`) and sparse hierarchical with Ward’s method (modified from the original `sparcl` implementation), and non-Sparse hierarchical clustering using both complete linkage and Ward’s method. Finally, we compare against an “oracle” method where both tuning parameters are known *a priori*; i.e., only the signal-bearing features are used and $K = 4$.

Due to the mixed results from the tuning, we opt to show the sparse MC performance with both Tibshirani SK and Global Max, since the latter had the best performance in terms of Rand index and the former performed the best in terms of inducing sparsity while correctly returning the true value of K . The `sparcl` method is tuned using the same method, but with hierarchical clustering (using complete linkage and Ward’s method) on the weighted dissimilarity matrix; whereas we tune the other methods for K individually. The oracle method is set to the optimal $K = 4$ value to set an upper threshold on clustering performance for comparison purposes. Non-sparse methods are tuned using the Tibshirani 1SE rule (for K only).

The impact of noise features on the results of the clustering can come in two ways: 1) Mis-identification of the right number of clusters, and 2) mis-clustering observations based on use of the wrong features for bipartitions. The results of the 100 simulations are shown in Table 4.6. Sparse MC using the Global Max rule has the highest Rand index for any of the non-oracle methods and chooses, on average, roughly 37 non-zero weights, indicating that it generally tends to remove most, but not all noise features. Sparse MC with Tibshirani SK is potentially too sparse as it tends to penalize all but two to three features to zero-value weights. In general, the sparse clustering methods outperform the non-sparse methods; however, sparse hierarchical clustering with complete linkage performed substantively worse than the MC with the same rules applied. It is possible that a different set of rules would be optimal for `sparcl` with hierarchical clustering than the rules we determined for MC. Additionally, it is interesting that the sparse MC outperformed its non-sparse counterpart. Finally, note that the Global Max-tuned sparse MC solution slightly outperformed the “oracle” method that used MC with only the signal-bearing features and using the *a priori* known $K = 4$; this is likely due to the difference in how the two algorithms weight the features. The sparse method will assign different weights to the non-zero features whereas the oracle method applies the same weight of one to all features without regard to relative importance. This could reflect more clear structure in some signal-bearing features and the weights, allowing the sparse method to focus on just those features.

4.5 Real Data Analysis

To further assess both the efficacy and challenges of the sparse MC method, we applied sparse MC to various real data sets with three different structures. In one setting, we evaluate data sets that were contaminated with artificial noise features. We compare results to actual class labels in those data sets as we have natural groups to compare. Additionally, we apply the method to high-dimensional data with relatively few records.

Rand Index by Clustering Method			
Simulation Method	Rand Index	SD(Rand)	Mean # Nonzero Weights
Tibshirani SK Sparse MC	0.714	0.035	2.38
Global Max Sparse MC	0.816	0.028	37.25
Tibshirani Sparse Hierarchical Complete	0.652	0.056	2.10
Global Max Sparse Hierarchical Complete	0.680	0.041	51.02
Tibshirani Sparse Hierarchical Ward	0.706	0.395	2.73
Global Max Sparse Hierarchical Ward	0.711	0.040	58.07
Oracle MC	0.809	0.019	20
Oracle Ward	0.940	0.029	20
Oracle Complete	0.915	0.048	20
Non-Sparse MC	0.667	0.027	100
Non-Sparse HC Ward	0.666	0.0317	100
Non-Sparse HC Complete	0.687	0.0413	100

Table 4.6: Results from comparison of clustering method comparison show best performance (in terms of Rand Index) is from the Global Max-tuned Sparse MC method. All non-sparse methods were tuned using the Tibshirani 1SE rule. Results based on 100 simulations.

4.5.1 Palmer Penguins

The Palmer penguin data set refers to body measurements taken on penguins collected from three islands in the Palmer Archipelago, Antarctica, originally collected and published by Gorman et al. (2014). Measurements of several physical characteristics were taken for $N = 344$ penguins, including bill length (mm), bill depth (mm), flipper length (mm), and body mass (grams). In total, the data comprise three species: Chinstrap, Adelie, and Gentoo penguins, each with relatively distinct profiles across those four dimensions. The Palmer penguins data set can be accessed via the `palmerpenguins` package in R (Horst et al., 2020). Additionally, we acknowledge the Palmer Station LTER and the US LTER Network for making the data publicly accessible.

Researchers may be interested in taking measurements of the biological features of penguins, exploring groups of penguin morphological groups, and comparing those groups to the species of the penguins. In such a case, more features may be measured than the four mentioned above; some of those may or may not drive meaningful differences in the species segmentation. We simulated added features to replicate such a situation, where additional variables contribute only noise and drive no meaningful separation between groups. We scaled the data and added noise features from independent normal distributions with means of 0 and standard deviations of 1 to the observations to determine how well the sparse MC method would discriminate between the noise and signal-bearing features and cluster the penguins into groups.

We examine the performance of sparse MC across the various tuning rules vs. standard MC with no noise features as well as in the presence of noise at varying levels. We include 10 and 100 noise features in order to better understand how each method performs in the presence of some noise vs. substantial noise. Figure 4.11 shows the Gap statistic plots for all three cases. In all cases, the Gap statistic plots suggest that multiple groups are appropriate as opposed to a single group; Gap values are consistently higher for $K > 1$ even when

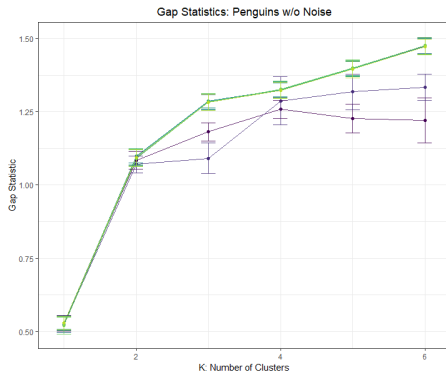
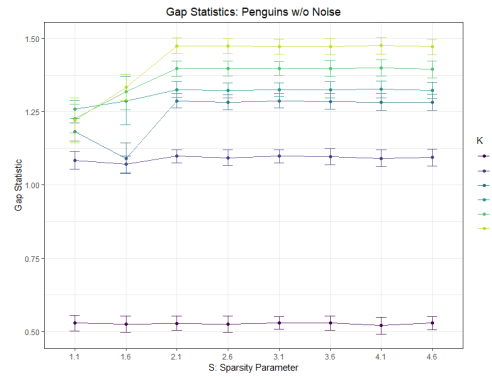
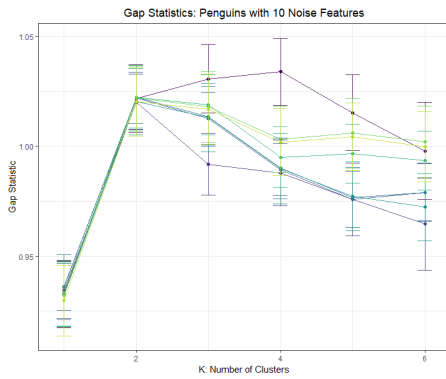
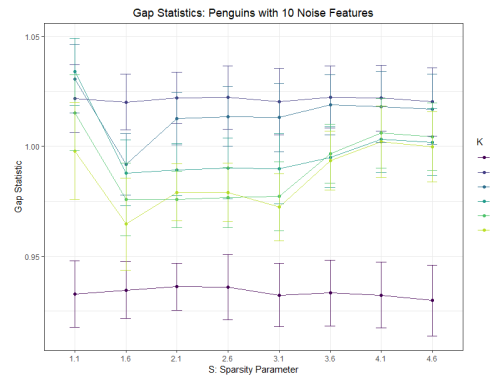
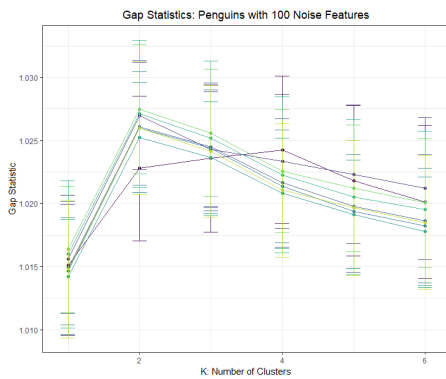
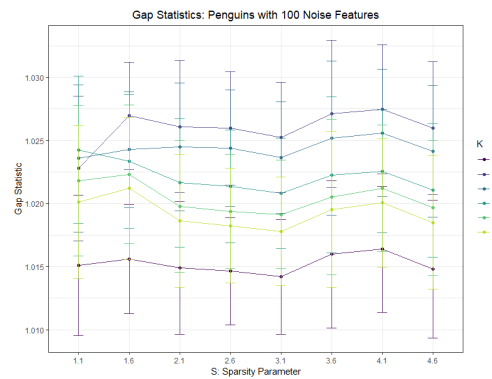
(a) K plot with no noise(b) s plot with no noise(c) K plot with 10 noise variables(d) s plot with 10 noise variables(e) K plot with 100 noise variables(f) s plot with 100 noise variables

Figure 4.11: Gap statistic plots for the sparse and non-sparse monothetic clusterings of the Palmer penguin data show that the one-cluster solution is not optimal, indicating the presence of multiple groups, even when noise features are present. Panels (a) and (b) show the case where no noise features are present, Panels (c) and (d) give the Gap for the 10-noise feature situation, and Panels (e) and (f) provide the 100-noise feature Gap plots.

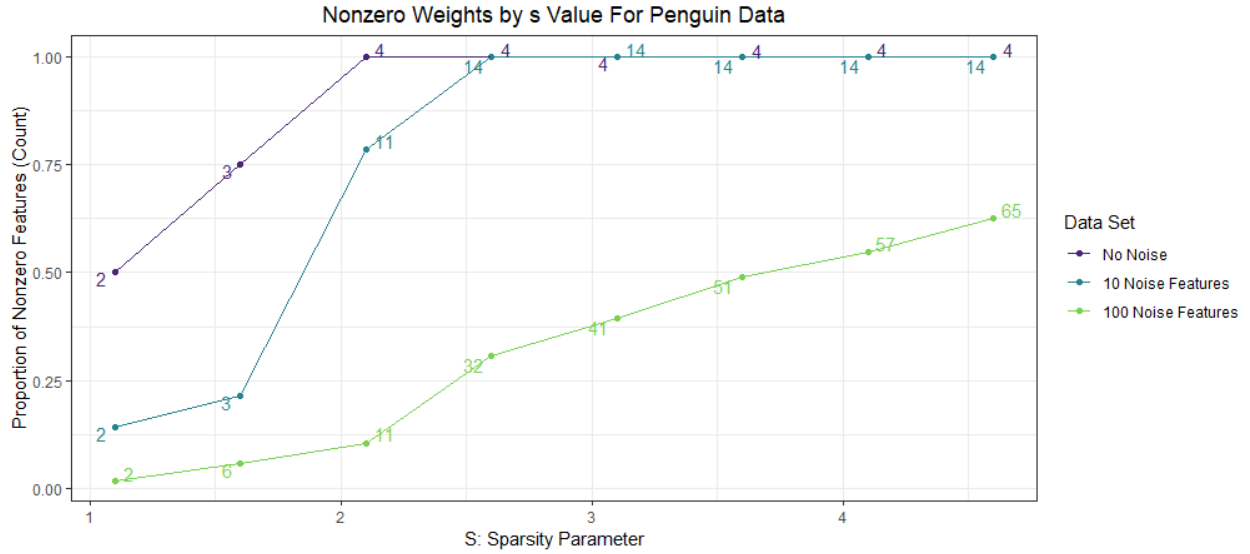


Figure 4.12: Number of nonzero features retained in the clustering solution for each value of s across the data sets with no noise, 10 noise features, and 100 noise features added.

10 or 100 noise features are added to the data at all s values. For each of the data sets, the sparsity parameter controls a slightly different amount of penalization; as the number of features shown in Figure 4.12 increases, the same s value leads to a slightly less sparse solution. For the two most sparse values of s , the number of nonzero features remains the same, two, irrespective of the number of noise features added. Thus, it might make sense that in the case of data sets with large numbers of features, a wider and finer grid of s values should be considered.

This leads to a potential difference in how the tuning rules might select values for s in the scenario where there are more noise features present. We consider the various Gap tuning rules for each version of the data, as shown in Table 4.7. Like the results from Simulation Study 3, the rules do not all agree; in general, the Tibshirani SK prefers the sparsest solution possible and Global Max (and related methods) tend to prefer more clusters. All of the rules lead to choices of s that induce sparsity in the two situations where noise features are available; only the Tibshirani SK rule leads to a choice of s that induces sparsity when no

noise variables are present. We find that at least in this situation, the Global Max SE KS and Global Max SE SK rules return the same result, which seems to either be equivalent to the Global Max rule or prefer slightly smaller values of K for a given s -value.

The resulting weights are interesting to compare as well, as shown in Figure 4.13. The key difference between the sparse MC and the non-sparse version is that the sparse method allows for differentiated weights for each of the four variables compared to constant weight for each of the four features used as splitting candidates by the non-sparse MC. Figure 4.12 shows the number of nonzero weights as a function of s for each of the data sets; as s increases, the total percentage of nonzero features increases quickly in the no-noise and 10-noise feature cases, but s must be chosen to be relatively large to include all 104 features in the 100-noise feature scenario. In each case, the sparsest value of $s = 1.1$ leads to only two nonzero features being available for monothetic splits.

With zero weights, the features will be unavailable for bipartitioning, and with nonzero weights they can be more/less likely to be included for splits. Features with high weight are much more likely to be used in splitting. In the cases where the tuning results in a solution that is potentially too sparse, only one or two variables may be available for splitting. We compare the sparse bipartitions with the non-sparse MC applied to the no-noise data set. All the methods make initial splits on flipper length as shown in Figure 4.14, and increasingly rely on flipper length in later situations, partly because the tuning preferred a sparser solution in the presence of greater noise. In the 100-noise feature scenario with non-sparse MC, we found that the MC made an initial split on the same variable, flipper length, but made all subsequent splits based on noise features (results not presented).

We additionally compare the resulting clusters to each other and to the species information for each of the penguins to determine how well the sparse MC performed in the presence of at least some noise. Ideally, the sparse MC method would look somewhat similar to the non-sparse MC when no noise is present; potentially, it might relate more

Rule	K	s	Nonzero Features
<i>No Noise Data</i>			
Tibshirani KS	2	4.1	4
Tibshirani SK	5	1.1	2
Global Max	6	3.1	4
Global Max SE KS	3	3.1	4
Global Max SE SK	3	3.1	4
<i>10 Noise Features</i>			
Tibshirani KS	3	2.1	12
Tibshirani SK	4	1.1	2
Global Max	6	2.1	12
Global Max SE KS	6	2.1	12
Global Max SE SK	6	2.1	12
<i>100 Noise Features</i>			
Tibshirani KS	4	1.1	2
Tibshirani SK	4	1.1	2
Global Max	4	1.1	2
Global Max SE KS	2	1.1	2
Global Max SE SK	2	1.1	2

Table 4.7: Optimal tuning parameter values across each of the bivariate Gap rules applied to the Palmer penguin data, with varying levels of noise features added.

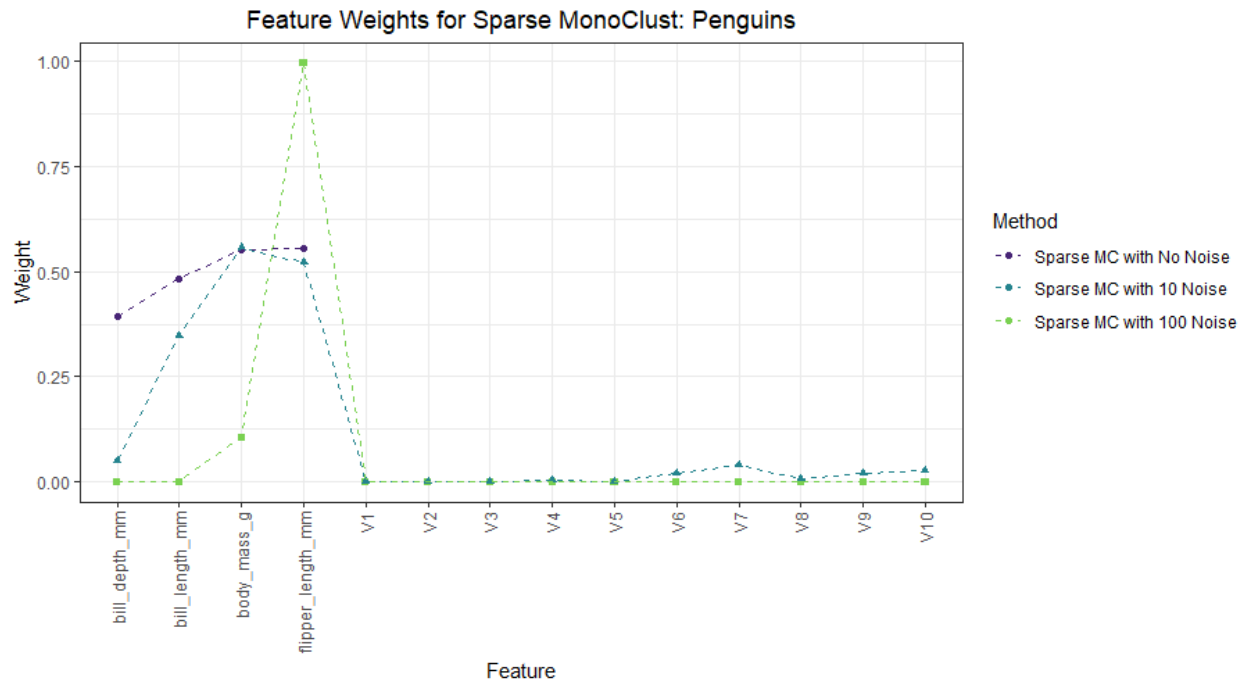
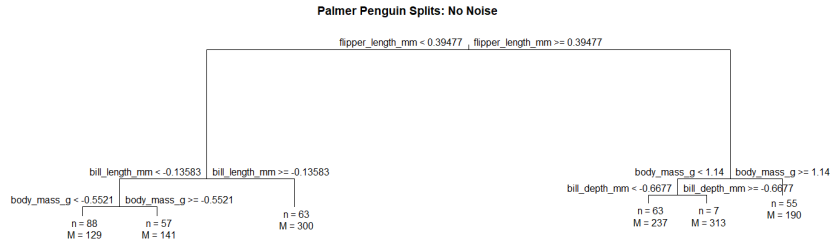
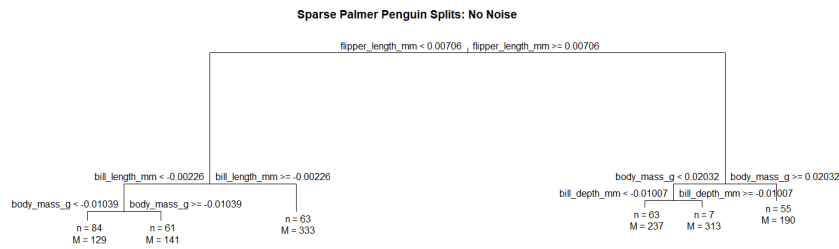


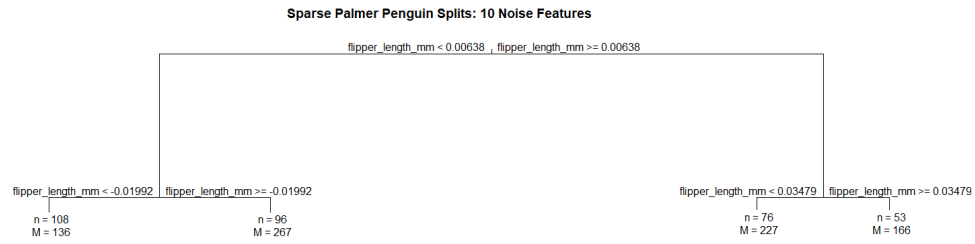
Figure 4.13: Feature weights for the sparse MC on the Palmer penguin data, with each version tuned for s using the Global Max rule. The version with no noise features added includes weights only for the first four features, and 14 weights are included for the version with 10 noise features added, with just the first 14 shown for the version with 100 noise features added.



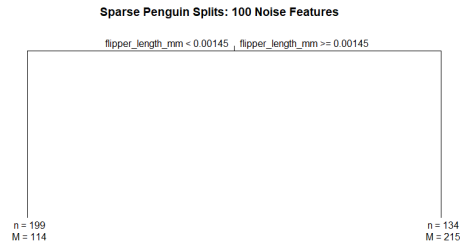
(a) MC Splits: No Noise



(b) SMC Splits: No Noise



(c) SMC Splits: 10 Noise Features



(d) SMC Splits: 100 Noise Features

Figure 4.14: Monothetic bipartitions of the Palmer penguins data, both in the presence of noise features and when only the original four variables are used. Bipartitions are shown for cluster solutions using the Global Max rule.

Rand Indices for Penguin Clustering Comparisons

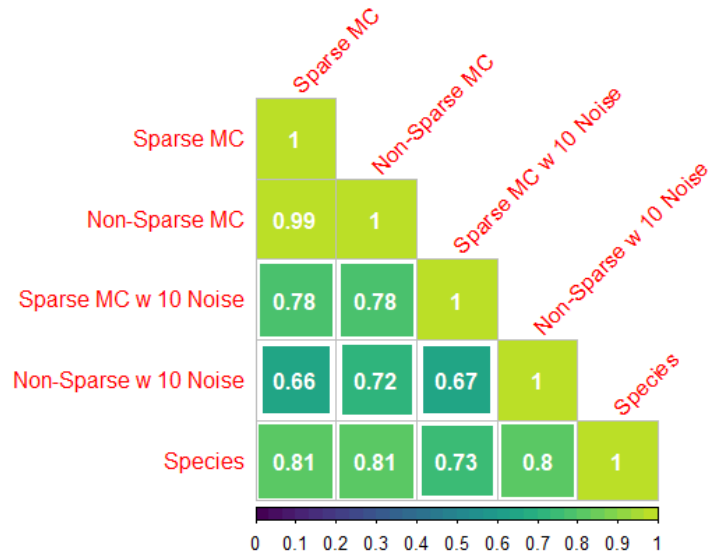


Figure 4.15: Rand indices comparing each of the clustering solutions to each other and to the species label, both on the original data as well as the data with 10 noise features added. The sparse MC produced a very similar clustering (Rand = 0.99) in the presence of noise features as did the MC with no noise features present.

closely to the species information as well. Figure 4.15 gives a comparison based on the Rand index between each of the cluster solutions and the species type to determine how similar the cluster results are to each other and to the species labels (which may not necessarily represent the true groups present). The sparse MC without noise is nearly identical to the non-sparse MC result. With 10 noise features present, the sparse MC is more similar to the non-sparse MC without noise than the non-sparse MC applied to the noisy data; this indicates at least some evidence that the sparsification process provides efficacy in mitigating the impacts of some noise on the clusters generated.

Carnegie Variables	
PDNFRSTAFF	Headcount of non-tenurable research staff
FACNUM	Headcount of tenured/tenure-track faculty
S.ER.D	Stem Expenditures (in thousands of dollars)
NONS.ER.D	Non-Stem Expenditures (in thousands of dollars)
STEM_RSD	PhDs awarded in STEM-related fields
HUM_RSD	PhDs awarded in Humanities-related fields
SOCSC_RSD	PhDs awarded in Social Sciences fields
OTHER_RSD	PhDs awarded in other fields

Table 4.8: Definitions of the variables used in the Carnegie data set.

4.5.2 2015 Carnegie Classification

The 2015 Carnegie Classification (CC) data set contains eight variables related to productivity of doctoral-granting institutions (see Harmon et al. (2019) and Kosar and Scott (2018a)). The variables and their definitions are included in Table 4.8. Typically, because some of the institutions are substantially outlying on some of the variables (particularly STEM expenditures (S.ER.D)), we rank the institutions first and work with the ranks rather than the original values). The data are ranked (with tied institutions assigned the average rank) and subsequently standardized as an initial step, and we apply sparse MC and non-sparse MC with Euclidean distance in order to assess the interpretations and splits based on the clusters generated.

Figure 4.16 shows the Gap statistic plots for the CC data set, which show a preference for multiple clusters across all levels of sparsity. Gap-tuning rules differ slightly on the right number of clusters; Tibshirani KS selects a four-cluster solution whereas the Global Max and related rules tend to favor the most clusters examined, six, and they favor slightly less sparse

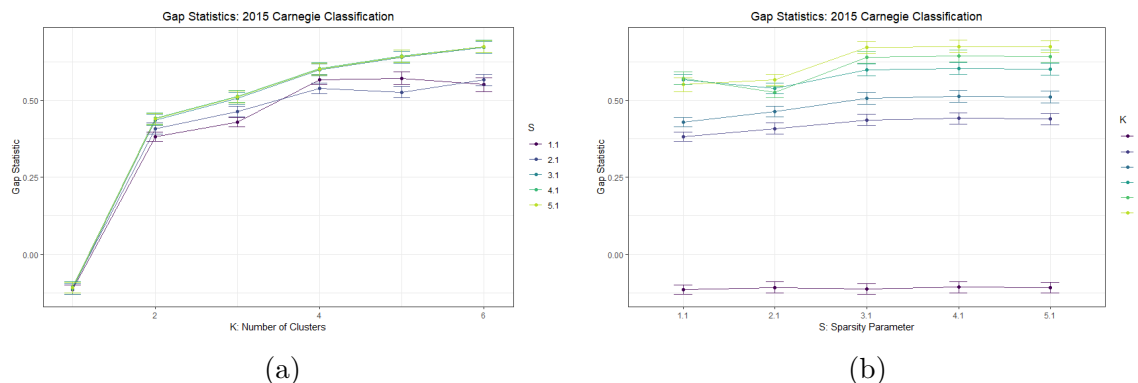


Figure 4.16: Bivariate Gap statistic plots for the Carnegie data using sparse MC, presented with K running along the x-axis in Panel (a) and across s values in Panel (b), indicate that a multiple-cluster solution is optimal; however, the tuning rules disagree on the optimal values of s and K . Tibshirani KS chooses the $s = 3.1$ and $K = 4$.

solutions. For the purposes of this example, we use the Tibshirani KS rule as it did not choose the maximum number of clusters considered. Additionally, the non-sparse version of MC tuned only for K and leveraging Tibshirani's 1-SE rule also chooses a six-cluster solution.

The resulting splits are shown in the dendrograms in Figure 4.17. Interestingly, the first splits are made on different (although related) features, STEM-related expenditures (S.ER.D) for the sparse MC, and STEM PhDs awarded (STEM_RSD) for the non-sparse version. The second splits appear to mostly focus on faculty or postdoc headcounts; in the sparse version, the method splits on Non-Tenurable Research Staff (PDFNRSTAFF) for one additional bipartition, with Social Science PhDs awarded (SOCSC_RSD) for the other. The non-sparse MC focuses on Tenured Faculty headcount (FACNUM) as well as Non-Tenurable Research staff (PDFNRSTAFF) for its second splits, and then splits based on different types of PhDs awarded in additional recursive splits.

The resulting clusters provided by both methods, as well as the original 2015 classifications, are compared in parallel coordinate plots in Figure 4.18. The sparse MC splits based on STEM expenditures, Research Staff headcount, and Social Science PhD

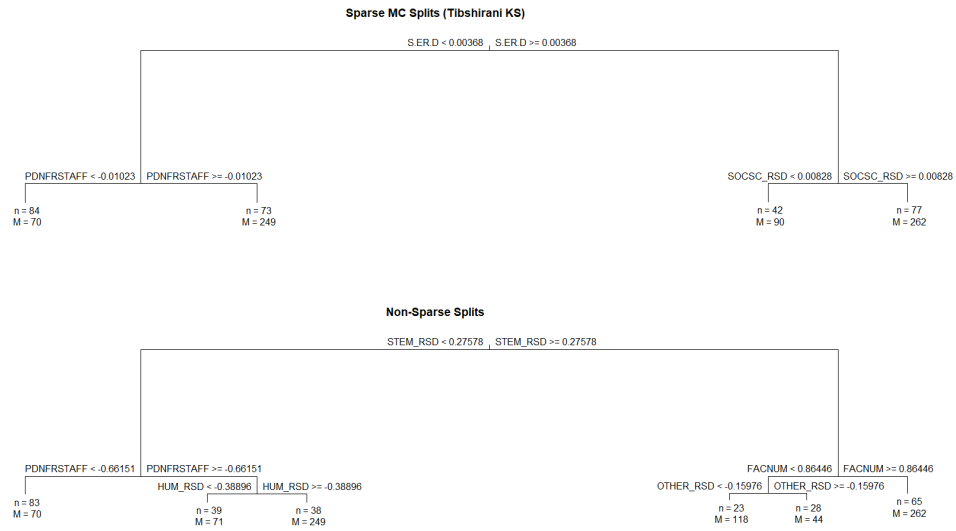


Figure 4.17: Monothetic bipartitions for the 2015 Carnegie Classifications ranked data set. Panel (a) shows the splits for the sparse MC using Tibshirani KS Gap rule (choosing a four-cluster solution) and Panel (b) shows the splits for the non-sparse MC, which chooses a six-cluster solution.

count, and the clusters are all clearly sorted based on those features. The non-sparse MC similarly sorts institutions based on features used in the bipartitioning; however, because there are six clusters selected, the groups are less clear in Figure 4.18. Table 4.9 shows the comparisons of the clustering solutions vs. the actual CC groups (categorized as R1, R2, and R3), along with cluster names that reflect both the monothetic characterization of each cluster as well as their majority CC group. These comparisons are somewhat subjective because the CC labels themselves are the result of a process akin to cluster analysis, but when paired with the monothetic split diagrams, they provide an opportunity for interesting discussion. In the four-cluster solution from the sparse MC, Cluster 1 mostly contains R2 and R3 universities, which aligns with the splits that would assign them to the schools with both the smallest STEM expenditures as well as below-average postdoctoral researchers. On the other hand, Clusters 2 and 3 appear to contain most of the mid-tier R2 institutions, and all of Cluster 4 aligns with R1 institutions, characterized by high STEM expenditures and awarding the most Social Science PhDs. In the non-sparse MC clusters obtained, the top and bottom clusters are similar to the sparse versions (Clusters 1 and 6)—but the middle clusters are further split based on Humanities and Other/Professional PhDs awarded. In this case, the preference for a four-cluster solution from the sparse MC provides a slightly easier-to-interpret set of clusters than in the non-sparse MC case.

The CC's method for deriving the Basic Classification (levels 15, 16, and 17) are mostly sorted based on size, but can vary based on different combinations of attributes. Because of the complexity of the methodology, schools in the same group can actually be very different from each other on certain attributes. In this sense, the interpretations of the resulting groups from the MC and sparse MC-based methods are clearer than that of the original CC groups, because the PCA-based classifications lack a single monothetic characterization for institutions in each group.

Carnegie Classification vs. Cluster Membership				
Cluster	Cluster Name	CC R1	CC R2	CC R3
<i>Sparse MC Clusters (with $s = 3.1$, $K = 4$)</i>				
1	Less Research	0	32	52
2	Middle Tier: More Postdocs	12	59	2
3	Borderline: STEM Over Social	26	16	0
4	Most Research	77	0	0
<i>Non-sparse MC Clusters</i>				
1	Mostly Small: Low STEM	0	31	52
2	Mostly Large: High STEM	64	1	0
3	Middle Tier: Fewer Humanities	0	37	2
4	Middle Tier: Humanities Focus	10	28	0
5	Borderline: Less Professional/Other	18	5	0
6	Borderline: More Professional/Other	23	5	0

Table 4.9: Table comparing the clusters generated by the sparse MC (tuned by Tibshirani KS) and the class labels generated in the 2015 Carnegie Classification. The labels are not true class labels, but inform an interesting comparison about how the sparse MC results compare to the CC groups.

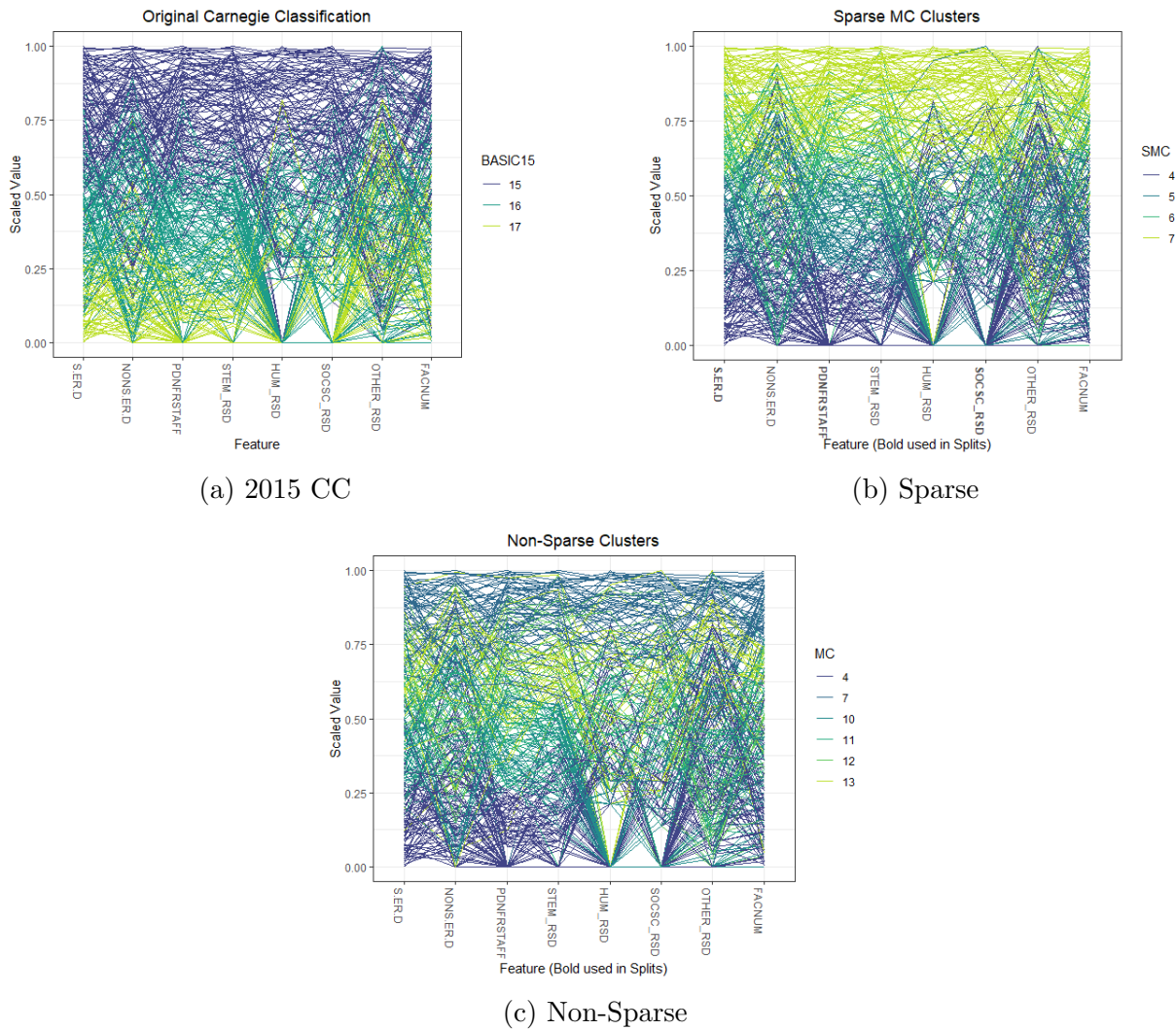


Figure 4.18: Parallel coordinate plots showing the values for each of the eight Carnegie features for the original 2015 Carnegie Classifications, sparse MC, and non-sparse MC. Panel (a) shows the original classifications, Panel (b) shows the sparse MC clusters, and Panel (c) shows the clusters resulting from the non-sparse MC clustering.

4.5.3 High-Dimensional LC-MS Data

It may be of interest to generate sparse clusterings of data where the number of objects is much smaller than the number of features. An example of this was used in an example comparing COSA to sparse hierarchical clustering in Kampert et al. (2016). The data refer to $P = 1550$ attributes of liquid chromatography-mass spectrometry (LC-MS) measurements of plasma lipids from two types of mice: transgenic and wildtype (Damian et al., 2007), which we are calling the ApoE3 data. Each mouse was sampled twice; however, one transgenic mouse did not survive the experiment, leading to measurements of 10 wildtype and 9 transgenic mice and a data dimension of 38×1550 with no missing values in the data (data were standardized and Euclidean distances used). In Kampert’s comparison between COSA and sparse hierarchical clustering using `sparcl`, the latter method performed surprisingly poorly in reconstructing consistent clusters that matched species information (Kampert et al., 2016).

We assessed the same data set and compared sparse MC to sparse hierarchical clustering and non-sparse clustering methods with both complete and Ward’s method linkages selected, optimally tuning each method independently, and analyzing the data directly as overviewed in Kampert et al. (2016). Tuning via the Tibhsirani SK rule (in order to prioritize inducing sparsity) as well as the Global Maximum Rule (which was optimal in Simulation 4), using both the Global Max and Tibshirani SK Gap rules for the sparse clustering methods, and Gap-based tuning using Tibshirani’s 1SE rule was used to tune for K in the non-sparse methods. Since the data were originally used to compare COSA to existing sparse clustering methods, we additionally include COSA as implemented in Kampert et al. (2016) (with complete linkage and Ward’s method selected in the resulting hierarchical clustering, and Gap-tuned only to select optimal values of K). Direct implementation of COSA with MC, along with additional tuning considerations, is left for future research.

Table 4.10 shows the Rand index for each of the clustering methods shown vs. the species

of the mouse, assuming that the species labels are the only true labels for comparison. It is possible that there are several sub-groups based on the species (or the first/last measurement per mouse) but Kampert et al. (2016) focused on whether clusters found were predominantly transgenic or wildtype, which is related to using species as an assessment target.

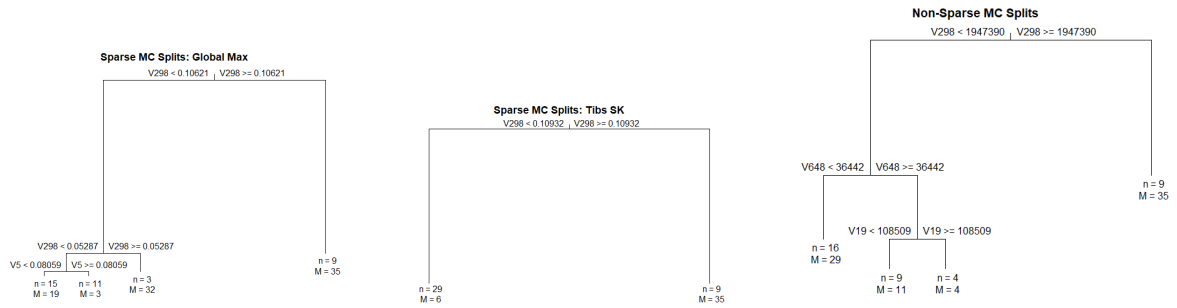
Although there are slight differences due to the independent tuning done for each sparse clustering method, all of the sparse methods tended to choose similar tuning values for each clustering method, resulting in similar penalization of features for each Gap rule. The non-sparse MC had clusters with closer ties to the species than the sparse implementations; other versions were all similarly low on the Rand index. This may be due to the generally poorer performance of the PMD in sparsifying the data, and is reflected in the feature weights for the sparse MCs shown in Figure 4.20. In both cases, the majority of the weight is on Feature 238, with some weight being spread over other features in the slightly less sparse version tuned by the Global Max rule. Even at less sparse choices of the tuning parameter s , we found that the PMD tended to focus mostly on Feature 238 both for sparse MC and sparse hierarchical clustering implementations. By contrast, COSA did not fully shrink any of the features; this behavior of “honing in” on the single feature exhibited by the PMD was one of the key reasons why the sparse hierarchical clustering result was less desirable than COSA with hierarchical clustering on the same data set (with the same linkage) in Kampert et al. (2016).

4.6 Glass Data

Data pertaining to $P = 13$ different measurements on chemical concentrations of different elements for $N = 180$ archaeological glass vessels from various groups were originally presented in Janssens et al. (1998) and Lemberge et al. (2000) and a high-dimensional spectroscopy version of these data were used in an example by Brodinová et al. (2019). We examined the thirteen variable version (standardized with Euclidean distance) in order

Rand Indices vs. Species on ApoE3 Data			
Simulation Method	Rand Index	Nonzero Weights	Number of Clusters
Tibshirani SK Sparse MC	0.511	34	2
Global Max Sparse MC	0.492	4	4
Tibshirani Sparse Hierarchical Complete	0.488	4	2
Global Max Sparse Hierarchical Complete	0.496	34	6
Tibshirani Sparse Hierarchical Ward	0.488	4	2
Global Max Sparse Hierarchical Ward	0.559	34	6
COSA Complete	0.88	1550	5
COSA Ward	0.85	1550	5
Non-Sparse MC	0.736	1550	4
Non-Sparse HC Ward	0.489	1550	2
Non-Sparse HC Complete	0.489	1550	2

Table 4.10: Table showing the Rand indices, number of nonzero weights, and number of clusters for each of the methods used on the ApoE3 data set.



(a) Global Max Sparse Mono- (b) Tibshirani SK Sparse (c) Non-Sparse Monotheitic Splits
theitic Splits Monotheitic Splits

Figure 4.19: Monotheitic bipartitions and feature weights for sparse and non-sparse versions on the ApoE3 data set show that a single feature drove the main split for both methods, with additional splits being based on other signal-bearing features.

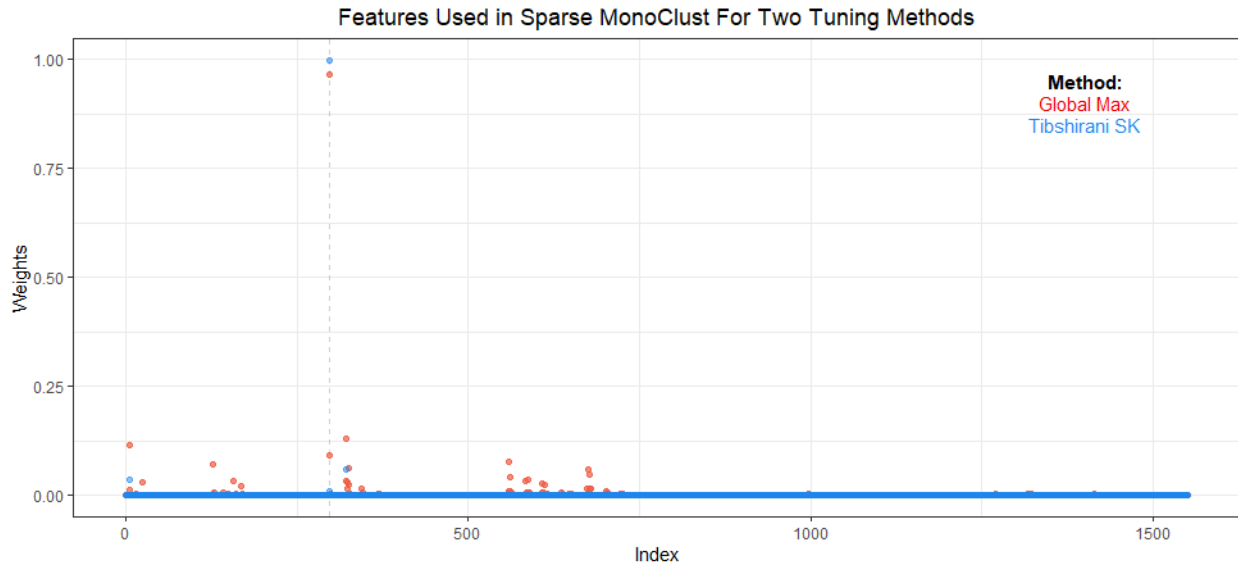


Figure 4.20: Feature weights for the two sparse MCs of the ApoE3 data. The PMD step used in this example appears to focus specifically on Feature 238, which leads to a less optimal clustering for each of the sparse methods (including both the sparse monotheitic and sparse hierarchical methods considered).

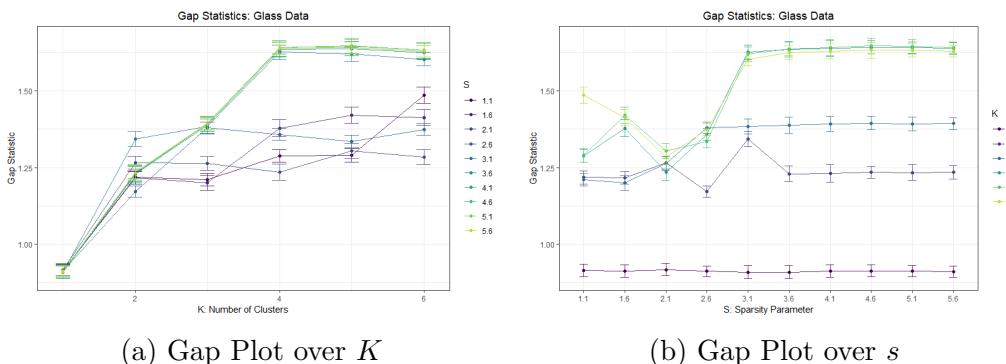


Figure 4.21: Plots of Gap statistics across differing values of K and s on the Glass data show strong evidence in favor of multiple clusters, but there appears to be a divergence in Gap behavior depending on the level of sparsity. Less sparse methods prefer a nine-cluster solution whereas more sparse methods prefer fewer clusters.

to examine the performance of the method and compare to the groups generated by the sparse MC.

We ran the method and generated Gap statistic plots across a grid of 10 potential values of K and 10 potential s -values. Interestingly, there is a bit of a divergence between the “less sparse” values of s and the more sparse ones, and this is reflected in the optimal choices from the tuning rules. The Tibshirani-based rules both select $s = 1.1$ and $K = 2$, the Global Max rule selects $s = 4.6$ and $K = 5$, and the Global Max SE rules both select $s = 3.1$ and $K = 4$. We evaluated cluster solutions for each of the tuning rules.

The feature weights from the sparsest clustering, the Tibshirani-based version, result in nonzero weights only for the “P205” and “Al₂O₃” concentrations, with the majority of the weight on the former. The other methods chose values of s that did not lead to any chemical features being set to 0, but reweighted them with most weight still applied to “P205” and “Al₂O₃” features.

Lemberge et al. (2000) analyzed four distinct groups of glass types: Sodic, Potassic, Calcic, and Potassic, each of which are well-separated in a plot of the ratio of $\text{CaO}/(\text{CaO} + \text{K}_2\text{O})$ plotted against Na_2O concentration for all glass vessels present. We

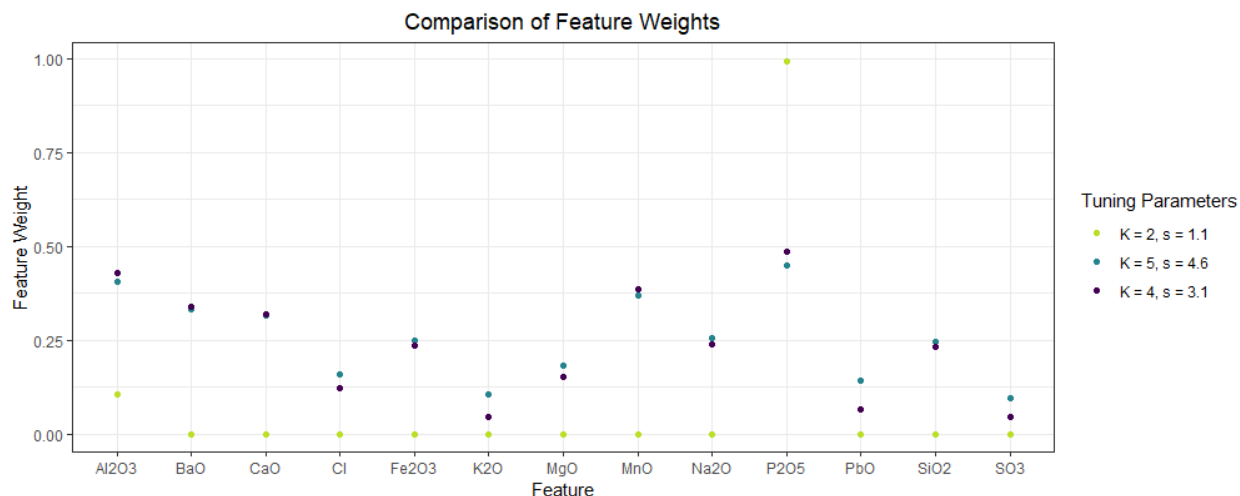


Figure 4.22: Feature weights across the chemical concentrations in glass for each of sparse monothetic clusterings based on the tuning suggestions for Tibshirani rules ($K = 2$, $s = 1.1$), Global Max ($K = 5$, $s = 4.6$), and Global Max SE rules ($K = 4$, $s = 3.1$).

recreate that plot in Figure 4.23 in order to compare the results of each sparse MC clustering solution based on the $P = 13$ features. The Tibshirani-based method chooses only two clusters and separates the Potassic from the rest of the glass types, in Panel (b) of Figure 4.23. Panels (c) and (d) show that the less sparse versions tend to separate out the Calcic and Potasso-Calcic glass objects, but both separate the Sodid group into two separate groups, each after making initial splits on different features (P2O5 for the sparse version, and Al2O3 for the less sparse versions). Note that both variables are relatively strongly correlated ($r = 0.84$) so subsequent splits were based on different features for both of the less sparse MC solutions.

4.7 Discussion

We have proposed a method for imposing sparsity in MC using a two-step algorithm: first, selecting relevant features via a sparsification method to generate a set of weights w and reweighted dissimilarity matrix u , and then to perform MC on those reweighted data.

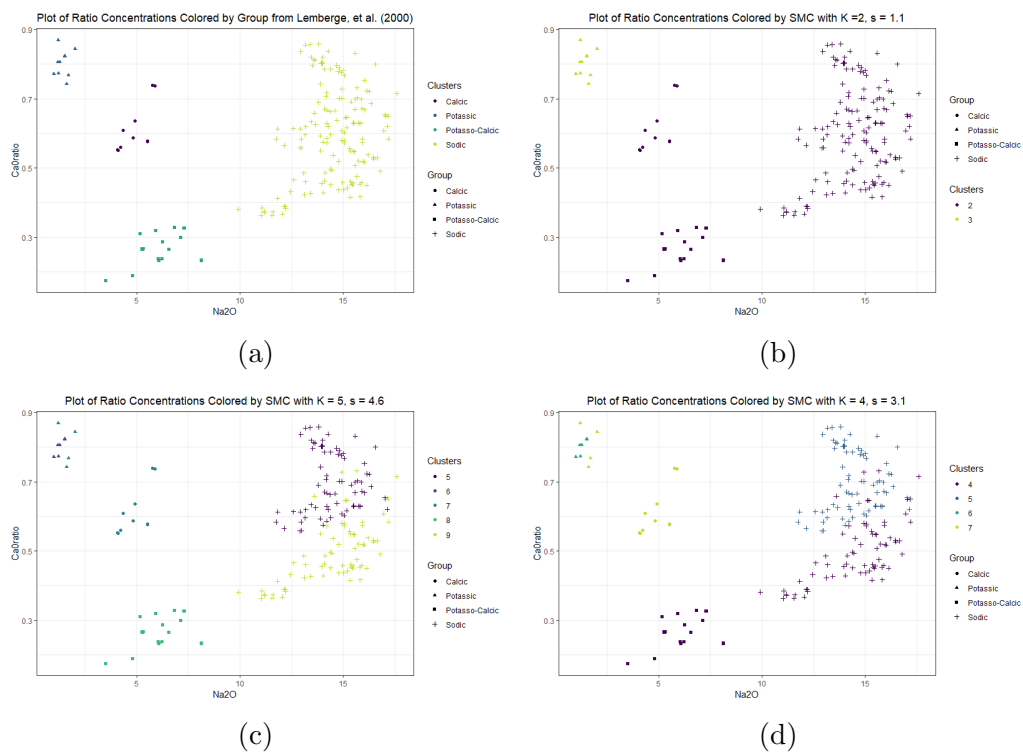


Figure 4.23: Plots of the ratio of $\text{CaO}/(\text{CaO} + \text{K}_2\text{O})$ plotted against Na_2O concentration for all glass vessels, colored by groups from Lemberge et al. (2000) in Panel (a) and by sparse MC cluster labels from each of the optimal tuning rules in Panels (b), (c), and (d).

This extends the framework of sparse hierarchical clustering to the divisive bipartitioning method of Chavent (1998). Results indicate that, at least in some cases, the bipartitions made on sparse input data will be more interpretable and lead to better clustering results.

Because this method relies on Witten and Tibshirani's penalized matrix decomposition to reweight the input data matrix, the same computational constraints are a potential consideration. If either N or P are very large, this method may perform slowly; indeed, by comparison to sparse hierarchical clustering proposed by Witten and Tibshirani, sparse MC may be more problematic because MC is typically slower than traditional hierarchical clustering.

4.7.1 Computational Gains

Inducing sparsity prior to using MC allows for potential gains in computational efficiency, an additional benefit in addition to providing a more interpretable set of bipartitions and potentially more accurate clustering. Without considering the impact of the calculation of the distance matrix, the computational complexity of MC on numeric data is $o(KPN(\log(N) + P))$ where K is the number of clusters of the finest partition, P the number of features, and N the number of objects (Chavent et al., 2007). By contrast, agglomerative hierarchical clustering methods with Ward's method can be implemented with complexity $o(N^2)$ (see McQuitty (1966); Chavent et al. (2007)).

The removal of q features used from P to $P^* = P - q$ features reduces the computational complexity of sparse MC by

$$o(KPN(\log(N) + P) - K(P - q)N(\log(N) + (P - q))).$$

It can be shown that this results in a reduction in computation time by $o(KNq(\log(N) - 2P + q))$. In Table 4.11, which gives the results from a comparison of run times for sparse vs. non-sparse MC on the 100 simulated data sets used in Simulation 4, the sparse MC was able to

Run Times for Sparse vs. Non-Sparse MC		
Method	Mean Time (seconds)	SD Time
Sparse MC	1.10	0.14
Non-Sparse MC	1.63	0.19

Table 4.11: Mean run times for sparse and non-sparse MC applied to the data sets used in Simulations 3. Sparse MC was able to provide results 32% faster, on average, than the non-sparse MC clustering.

outperform the non-sparse MC by 32%, on average, on a laptop with 16 GB of memory. This illustrates that by reducing the number of features input to the MC algorithm, the number of bipartitions necessary to explore at each split is potentially substantively reduced. For large data sets with many features, the more sparsification is induced, the more benefits for computing speed.

Note that at least when comparing sparse MC to other forms of sparse hierarchical clustering, the time to calculate the $N \times N$ dissimilarity matrix should be considered; however, it is worth noting that our implementation calculates this twice—both in the step to calculate the weights and sparse reweighted dissimilarity/distance matrix, and again in the actual calculation of the MC (on the new dissimilarity/distance matrix).

4.7.2 Future Work and Extensions

This method currently considers only quantitative variables. Tran (2019) points out that categorical features with many categories, or mixtures of data that include both categorical and quantitative features, can lead to computational difficulty in searching for best splits. Fuentes and Chavent (2015) attempt to tackle this problem with Divclust method on non-sparse data; however, obtaining the reweighted sparse data matrix to use as input is not a trivial problem. It does, however, represent a useful avenue for direct extensions to this work, perhaps extending work related to the Grouped Lasso for sparse

clustering of mixed categorical and quantitative data (Choi and Park, 2014).

The results of the ApoE3 data analysis also indicate that the PMD is not always the most appropriate method for “sparsifying” a data set. The COSA method, and potentially other methods that borrow from the sparse implementations of PCA, could be interesting alternatives to consider comparing to the performance of the PMD used in this paper. It may be the case that some methods for sparsification lend themselves better to different data structures and that the choice of sparsification algorithm could be considered a parameter to test in future implementations. Future work could therefore consider implementing a COSA-based version of sparse MC and comparing different sparsification methods with MC in various situations, and additional investigation of tuning strategies for COSA might be necessary.

Additionally, it would be useful to identify a framework for a one-step solution to both induce sparsity and generate clusters simultaneously in monothetic clustering. The two-step framework of Witten and Tibshirani provides a useful tool for extending sparsity to various methods, but a more natural way to select features and generate clusters may be found in Bayesian methods such as mixture modeling. Extensions of MC to such a framework, which might be able to both induce sparsity and potentially drive bipartitioning, could be an interesting path for future work.

CHAPTER FIVE

CONCLUSIONS AND EXTENSIONS

5.1 Conclusions

Methods for dealing with multivariate data in both supervised settings and unsupervised ones will continue to be an area of importance as the fields of statistics, data science, and computer science converge. As more data are generated (and at an increasing rate), methods that are more judicious in identifying and using features will become increasingly important.

Chapter Two discussed the Carnegie Classification (CC) and proposed an alternative method to generate institutional classifications based on the same data, but using structural equation modeling and objective model-based clustering to determine both the number of groups and groups themselves. This framework provides a platform for a discussion about the limitations of the CC methodology and provides a novel approach to classify institutions; however, improvements to the substantive quality of the groupings could be made by making enhancements to the data as well as the algorithm. Our method does not explicitly match the CC groupings, and in some cases, it may put too much weight on aggregate production (particularly in the case of schools with high per-capita production like the California Institute of Technology, which our method considers in the second group). Additionally, the CC continues to evolve and change, with potential new data being introduced in future releases. Our method would need to expand to take any such features into account either as new manifest variables mapped to the current latent factors of production, or to create new latent factors to be added to the model.

Chapter Three focused on a diagnostic tool for identifying influential points in mapping tools, primarily MDS and similar methods. The method can be used quite flexibly — we simply need a distance matrix and maps, meaning that our method could also be

implemented on multi-step variable creation methods like those underlying the CC or factor analysis, where a “map” of two factors could be created from multiple input features. For researchers using any such methods, our diagnostic either adds an entirely new way to think about influential points, or provides another tool for diagnosing influence. We identified a key drawback associated with the use of the diagnostic is that it detects influence at substantially different rates depending on the mapping techniques considered. For instance, t-SNE maps appear to place dramatically different singleton points near the center of the map or within groups of points rather than placing them clearly far from other observations. Thus, t-SNE maps are less impacted by obviously aberrant objects and may not accurately reflect their location in the map in the same way that MDS and similar methods might, and our method does not detect such points as having large pseudo-F statistics. Additionally, irrespective of the mapping technique used, some other limitations of the method are important to consider. For one, the method relies on permutation-based techniques, meaning that the p-values are limited and can be difficult to interpret—the most extreme observation will have the same p-value for two different methods, even if one has a relatively small pseudo-F statistic compared to another. We recommend using the pseudo-F values to quantify “extremeness” over simply using the p-values. On large data sets, the efficacy of the diagnostic may be limited; the computational complexity of the PERMANOVA-based procedure may cause this method to perform slowly, and the individual contribution of a single influential point may become harder to detect in the presence of many observations.

Chapter Four developed an implementation of sparse monothetic clustering (MC) that allows for improvement of results based on feature selection prior to recursive bipartitioning. This novel approach combined the penalized matrix decomposition framework of Witten and Tibshirani (2010) and the regression tree-inspired clustering algorithm proposed by Chavent (1999) and implemented by Tran (2019). Imposing sparsity on MC leads to various benefits, including computational improvements as the MC has fewer potential bipartitions

it needs to search across, and potentially better cluster reconstruction in the face of noise features in multivariate data sets. Additionally, by splitting on signal-bearing features, the monothetic characterization (and thus the interpretation) of the resulting clusters can be improved. We discussed various challenges associated with the method, including tuning for two parameters, s , the degree of sparsity, and K , the number of clusters. In certain cases, the PMD favored a solution that was too sparse, giving the MC algorithm only one or two nonzero features to consider for splits. In those cases, a COSA-based sparse MC algorithm, or potentially something different, might be worth consideration. An important aspect of our implementation of sparse MC is that data sets used for analysis may not always respond well to either the PMD step or the MC step—data sets where this method works well need to both contain strong signal features in the presence of noise features, and also contain clear clusters that can be well-partitioned via monothetic splits.

5.2 Extensions

5.2.1 One-Step Ordination and Extensions to SEM Framework

Our proposed alternative to the CC in Chapter Two could be extended to frameworks that involve a more thorough characterization of each institution, potentially along the lines of using IPEDS data as suggested by Zong (2021). Although our model only includes two latent factors that combine to form the factor-of-factors score, inclusion of additional features beyond the scope of the Carnegie data could provide a potentially interesting avenue for research. Additionally, sparse methods for SEM may be worthwhile in this setting because IPEDS considers a much wider set of variables than does the original CC. Implementations of sparse SEM have been developed by Jacobucci et al. (2021) and could be considered in this setting as well as on the original eight CC variables. An alternate way to obtain a “sparse” SEM would be to use a Bayesian implementations of SEM like that of Merkle and Rosseel (2018), and choose prior parameter values to induce a sparse set of weights.

Methods for identifying latent factors like those overviewed in the SEM alternative to the Carnegie Classification could be extended to other settings beyond classification of higher education or institutional research. In sports statistics, for instance, there may be several interesting extensions: for instance, this framework could be used to quantify metrics that pertain to both offense (one latent factor) and defense (another latent factor)—for NBA basketball players. A single score could be created in a similar manor by generating an overall factor-of-factors score that combines contribution to both offense and defense, and used to generate a score for each player per game. Those latent factor scores—or score distributions across games in a season or entire career—could be then used either directly in a clustering algorithm or as part of a multi-step clustering involving functional estimates of player score densities as overviewed using QBR for NFL quarterbacks by Davis and Lopez (2015) and extended by Anderson (2017).

Additionally, while the two-step process we generated is sufficient to generate scores and clusters, it would be desirable to utilize a method that combines both the dimension reduction/latent variable model and the group identification. Stratton (2022) overviews a Bayesian framework using infinite mixture modeling in an ecological framework. This one-step process could be viewed as an extension to the model-based clustering used, which explores mixture modeling for the clusters; instead of simply using algorithm-based techniques the SEM we used to generate latent productivity variables, we could instead estimate an underlying mixture model to drive the resulting ordination (Hui et al., 2015).

5.2.2 Influence in Other Map Settings

The PERMANOVA-based influence diagnostic could be explored in various other mapping techniques, both unsupervised as well as supervised. Unsupervised mapping tools like UMAP, Laplacian Eigenmaps, isoMap, etc. that are often used as competitors for MDS and t-SNE could be assessed to evaluate how well potentially influential points

affect the shape of the resulting ordination. In the related world of supervised mapping techniques, methods such as linear discriminant analysis (LDA, see Izenman (2008)), quadratic discriminant analysis (QDA, see James et al. (2013)), and other similar methods that all can be used to create maps using both a set of input features as well as an observed categorical response variable. Generally, because these methods use the observed class information in the generation of the maps, they can visualize more meaningful separation between groups. As a proof of concept, we applied the PERMANOVA-based influence diagnostic to maps created using LDA on the same simulated data sets as those explored in Chapter Three. In this case, the potential influential point is marked as a member of a single group, but is simulated to have observed values that vary across the group structures. Figure 5.1 shows the LDA maps created and assessed.

We conducted an initial simulation study to compare the efficacy of the influence diagnostic on LDA maps and compared them to classical MDS maps. The data were simulated with three groups where the potentially influential point was assigned to the middle group, but its values were allowed to vary across the group structures. The results from 10 simulations are summarized in Table 5.1. The method appears to detect the influential point about 70% of the time for LDA maps compared to 80% for the classical MDS, with a nearly identical spurious detection rate. The pseudo-F statistics are dramatically different, however, as the candidate influential point has a huge value for classical MDS vs. a mean value of 8.74 for LDA, which is still quite large on a pseudo-F scale. This may be because the potentially influential point does not appear to be pushed as far away from the groups in the LDA maps as in the classical MDS maps, as seen in Figure 5.1.

Further exploration of existing influence diagnostics for LDA and related methods (like partial least squares discriminant analysis) would be necessary to formalize this approach and determine both how well our method would work and how novel our diagnostic would be in this setting, but a Cook’s Distance-flavored diagnostic could be a useful tool given the

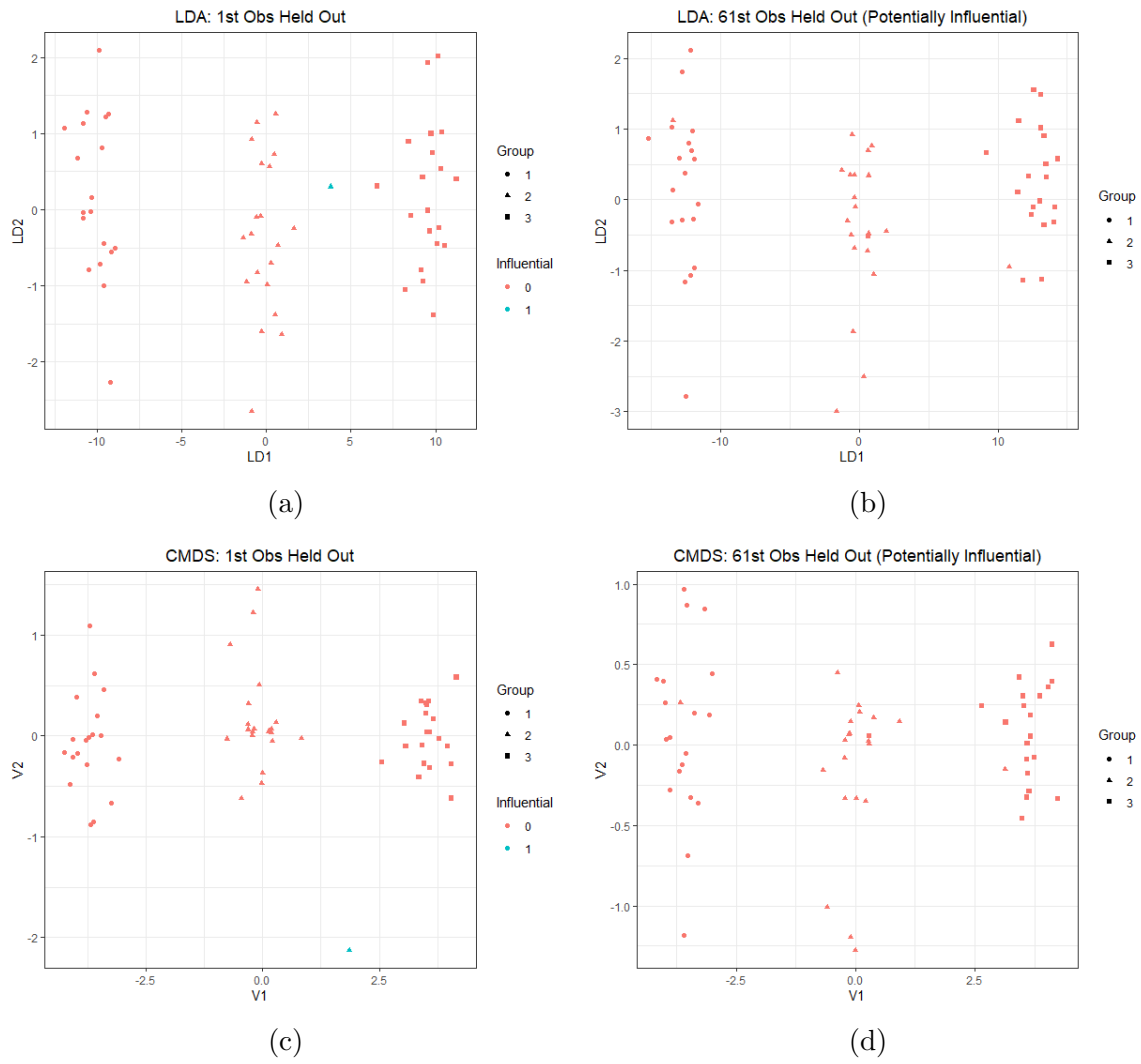


Figure 5.1: Results from analysis of influence diagnostic on LDA maps using data from simulation studies in Chapter Three, with the first and potentially-influential 61st observations held out for LDA in Panels (a) and (b), respectively, and for classical MDS in Panels (c) and (d).

Identification Rate by Mapping Method			
Method	Correctly Identified Influential	Correctly Identified Non-Influential	Mean pseduo-F for Candidate Influential Point
LDA	0.70	0.978	8.74
Classical MDS	0.80	0.98	467.32

Table 5.1: Correct and spurious identification rates for the LDA maps and classical MDS maps with a potentially influential observation included.

similarities in how supervised maps are created and used. It is possible that in a supervised setting where input data points do not align with the class label, these types of mapping techniques may actually be more substantively impacted than in the unsupervised settings, and be a reason for poor group discrimination due to the influence of a single observation.

5.2.3 Monothetic Clustering Extensions

The performance of the Gap statistic as specifically applied to monothetic clustering was mixed; while it seemed to perform a reasonable job in identifying the right level of sparsity, it tended to favor more clusters than necessary. This aligns with the findings of Tran (2019) where the Caliński and Harabasz’s pseudo-F statsitic proved to be liberal in identifying the proper number clusters, and required a two-step hybrid approach to answer the fundamental one vs. two (or more) clusterability question. Each of the methods for tuning monothetic clustering solutions (both sparse and non-sparse) represents a series of pros and cons; no single method appears to outperform the others. Additional research focused specifically on identifying an optimal tuning method may provide a worthwhile opportunity for future research.

A few potential tweaks to the Gap statistic could be worth investigating. Weighted

measures of the Gap statistic, such as the weighted Gap statistic proposed by Yan and Ye (2007) or the observation-weighted robust version employed by Brodinová et al. (2019) might help to mitigate the impacts of outlying observations on the recursive bipartitioning performed, and thus lead to more precise tuning for K .

This work primarily focused on the implementation of monothetic clustering in the **MonoClust** package (Tran, 2019); however Chavent’s **Divclust** package provides a potentially broader set of tools for implementing categorical features in addition to quantitative ones (2015). The PMD framework for generating a sparse distance matrix would potentially need to be altered in order to perform feature selection for categorical variables; however, this may be possible utilizing the PCA-based method that allows **Divclust** to recursively bipartition categorical or mixed categorical-quantitative data.

5.2.4 Sparse Implementation of t-SNE

The framework explored for generation of sparse monothetic clustering can be applied to any distance-based method, not just clustering. The two-step process used for monothetic clustering could be used to develop a sparse method for t-SNE (and other distance-based mapping procedures) involving the use of the penalized matrix decomposition first and the input of the sparse distance/dissimilarity matrix as the second step. This idea is explored as an extension to COSA by inputting the dissimilarity matrix from COSA into classical MDS maps (Kampert, et al 2017), but could easily be extended to other mapping techniques and using different methods for obtaining feature selection in the first step. Alternatively, one could replace the PCA initialization step in the t-SNE algorithm with sparse PCA as a competitor method for obtaining sparse t-SNE maps; however, this would be a less direct way of generating t-SNE maps from sparse data.

Figure 5.2 shows an example of two t-SNE maps created from a contaminated version of the Carnegie Classification data where 10 noise features have been added to the original

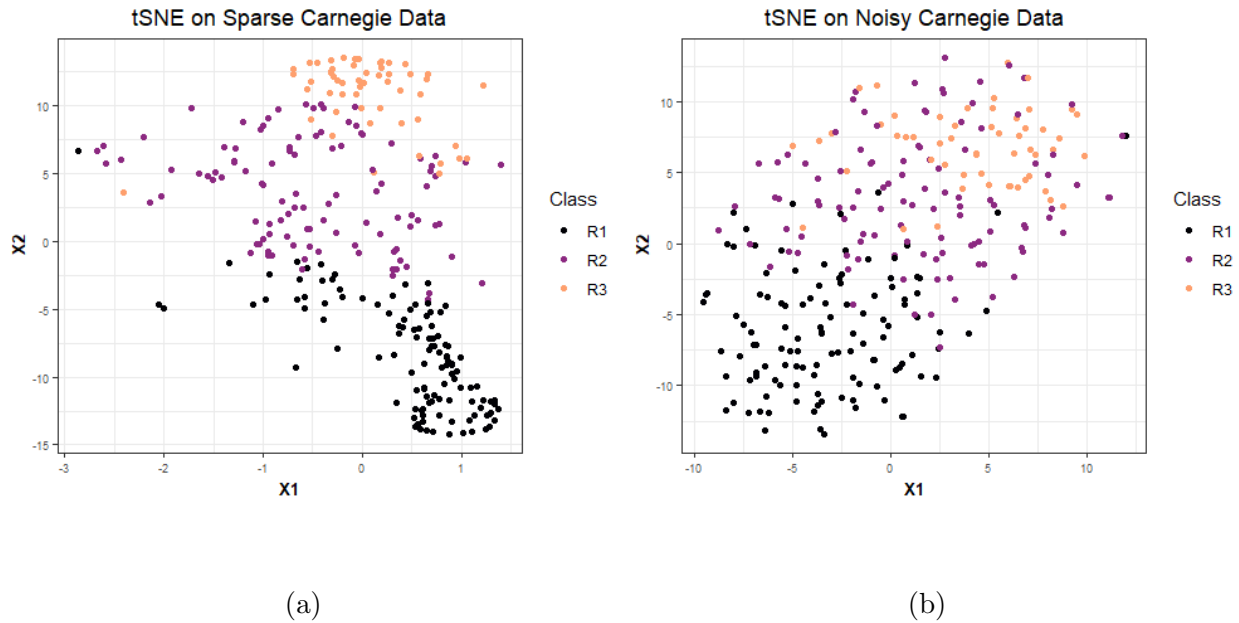


Figure 5.2: t-SNE maps with perplexity of 30 for noisy (with 10 noise features added) vs. sparse CC data using PMD for sparsification show some potential improved grouping based on classes (at least visually, particularly between groups R2 and R3) and could be an interesting avenue for future research.

8 CC variables. Panel (a) of the figure shows the sparse version where many of the noise features have been removed using the PMD and the sparse distance matrix is input to t-SNE; Panel (b) shows the t-SNE fit to the CC data with all the noise features added.

On the face of things, the sparse t-SNE map appears to better organize the plot so that the Carnegie-defined groups are close together and relatively well-separated; the noisy version is not so clear. However, a more objective measure of quality is necessary to determine the efficacy of sparse t-SNE maps. Since t-SNE's cost decreases as a function of perplexity, cost is not a suitable measure for this comparison. Determination of a suitable metric to determine the quality of a t-SNE map, and simulation studies that show the efficacy of sparse t-SNE in a repeatable, objective fashion, could lead to worthwhile improvements to t-SNE in the future.

5.2.5 Sparse Improvements to Influential Point Diagnostic

Given that the penalized matrix decomposition, along with other methods for sparsifying matrices like that of COSA, can potentially reduce noise in a distance matrix, it might be interesting to attempt to merge the PERMANOVA-based influence diagnostic with the first-step sparsification methods presented in Chapter Four.

Such an exploration could focus on situations where the distances are contaminated by noise variables that might make seeing real differences between objects more difficult. Consider a scenario where one is tasked with creating maps from a $P = 100$ dimensional data set, but only 20 of those features actually drive substantive structure. Then, an influential point may be different in some meaningful sense on those 20 features; removing the 80 noise-bearing features could certainly increase the PERMANOVA-based influence diagnostic's ability to detect such an influential point. Efficacy could be assessed through gains in power, but the impact of potentially inflated false detection rates would need to be investigated because the PMD itself could be impacted by influential points.

5.2.6 Applications to Natural Language Processing

Some interesting extensions of the work presented in this discussion could be used as part of a Natural Language Processing (NLP, see Loukanova, 2021) workflow. NLP involves the processing of unstructured text data to identify common patterns in text, including keywords, common themes, and an attempt to quantify semantic meaning. Typically, the unit of analysis considered is called a “document”, which is a collection of words that may range from a few sentences to something as long as an entire novel. A document-term matrix is a $n \times w$ matrix where n refers to the number of distinct documents and w refers to the individual words/phrases that occur in each document, with each element of the matrix referring to the frequency of occurrence of each word per document. These document-term matrices are always multivariate and may be high-dimensional if the number

of words/phrases w is large enough.

It is often of interest to visualize documents or identify common documents based on frequency of words or themes extracted from methods like from methods such as Term Frequency-Inverse Document Frequency (TF-IDF, Ramos, 2003). Mapping methods like classical MDS, t-SNE, and others will often be used for this purpose. Based on the results overviewed in this work, our proposed diagnostic could feasibly be used to identify aberrant or influential documents in a corpus, if they exist, based on MDS maps created from the document-term matrix. Additionally, although t-SNE has been adopted as a common method for visualizing documents in this way, our findings imply that such maps may not be useful in identifying obviously different documents unless they form a distinct group. If t-SNE maps are desired, we recommend overlaying a MST (built from the original document-term matrix) as a simple way to enhance the quality and consistency of those visualizations.

Additionally, clustering methods may be applied to identify common groups of documents. While pre-processing is often used as an initial step to remove irrelevant stopwords, etc. from the text, an additional filtering via the use of sparse clustering methods could provide an additional layer of simplification to improve the results of a clustering. Sparse monothetic clustering could be used in this (quantitative) setting in order to generate distinct groups based on counts of single words or phrases, leading to a more interpretable grouping of documents from a corpus.

REFERENCES CITED

- Amir, E., K. Davis, M. Tadmor, E. Simonds, J. Levine, S. Bendall, D. Shenfeld, S. Krishnaswamy, G. Nolan, and D. Pe'er (2013). viSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. *Nature Biotechnology* 31.
- Anacker, T. (2019). *To t-SNE or not to t-SNE: An Examination of t-SNE's Perplexities*. MS Writing Project, Montana State University.
- Anderson, M. J. (2001). A new method for non-parametric Multivariate Analysis of Variance. *Austral Ecology* 26(1), 32–46.
- Anderson, M. J. (2017). Permutational Multivariate Analysis of Variance (PERMANOVA). *Wiley StatsRef: Statistics Reference Online*, 1–15.
- Arias-Castro, E. and X. Pu (2017). A simple approach to sparse clustering. *Computational Statistics and Data Analysis* 105, 217–228.
- Banfield, J. and A. Raftery (1993). Model-Based Gaussian and Non-Gaussian Clustering. *Biometrics* 49(3), 803–821.
- Barbour, C., P. Kosa, M. Komori, M. Tanigawa, R. Masvekar, T. Wu, K. Johnson, P. Douvaras, V. Fossati, R. Herbst, Y. Wang, K. Tan, M. Greenwood, and B. Bielekova (2017). Molecular-based diagnosis of multiple sclerosis and its progressive stage. *Annals of Neurology* 82(5), 795–812.
- Bartlett, M. (1937). The Statistical Conception of Mental Factors. *British Journal of Psychology. General Section* 28(1), 97–104.
- Belsley, D., E. Kuh, and R. E. Welsch (1980). *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. John Wiley and Sons.
- Benjamini, Y. and Y. Hochberg (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)* 57(1), 289–300.
- Bentler, P.M., B. D. (1980). Significance tests and goodness of fit in the analysis of covariance structures. *Psychological Bulletin* 83, 586–606.
- Bentler, P. (1990). Comparative fit indexes in structural models. *Psychological Bulletin* 107.
- Blouvshtein, L. and D. Cohen-Or (2018). Outlier detection for robust multi-dimensional scaling. *arXiv*, 1–10.
- Bollen, K. (1989). *Structural Equations with Latent Variables*. Wiley.

- Borg, I. and P. Groenen (1997). *Modern Multidimensional Scaling Theory and Applications* (1st ed.). New York: Springer New York.
- Brieman, L., J. Friedman, C. Stone, and R. Olshen (1984). *Classification and Regression Trees* (1 ed.). Chapman and Hall/CRC.
- Brint, S. (2013). A Priori and Empirical Approaches to the Classification of Higher Education Institutions: The United States Case. *50*, 96–114.
- Brodinová, Š., P. Filzmoser, T. Ortner, C. Breiteneder, and M. Rohm (2019). Robust and sparse k-means clustering for high-dimensional data. *Advances in Data Analysis and Classification* *13*(4), 905–932.
- Brown, T. (2006). *Confirmatory Factor Analysis for Applied Research*. The Guilford Press.
- Calinski, T. and J. Harabasz (1974). A Dendrite MEthod for Cluster Analysis. *Communications in Statistics*, 1–27.
- Campbell, A., K. Caudle, and R. Hoover (2019). Examining Intermediate Data Reduction Algorithms for use with t-SNE. *Proceedings of the 2019 3rd International Conference on Compute and Data Analysis*.
- Carnegie Foundation for the Advancement of Teaching (2019). The Carnegie Classification of Institutions of Higher Education.
- Chang, W., J. Cheng, J. Allaire, Y. Xie, and J. McPherson (2017). *shiny: Web Application Framework for R*. R package version 1.0.5.
- Chavent, M. (1998). A Monothetic Clustering Method. *Pattern Recognition Letters* *19*(11), 989–996.
- Chavent, M., R. Genuer, and J. Saracco (2019). Combining clustering of variables and feature selection using random forests. *Communications in Statistics: Simulation and Computation* *0*(0), 1–20.
- Chavent, M. and Y. Lechevallier (2006). Empirical comparison of a monothetic divisive clustering method with the ward and the k-means clustering methods. In V. Batagelj, H.-H. Bock, A. Ferligoj, and A. Žiberna (Eds.), *Data Science and Classification*, Berlin, Heidelberg, pp. 83–90. Springer Berlin Heidelberg.
- Chavent, M., Y. Lechevallier, and O. Briant (2007). Divclus-t: A monothetic divisive hierarchical clustering method. *Computational Statistics Data Analysis* *52*(2), 687–701.
- Chavent, M., Y. Lechevallier, F. Vernier, and K. Petit. Monothetic Divisive Clustering With Geographical Constraints. *Compstat*, 67–76.
- Chen, F., K. Sousa, and S. West (2005). Testing Measurement Invariance of Second-Order Factor Models. *Structural Equation Modeling* *12*(3), 471–492.

- Chipman, H. and R. Tibshirani (2005). Hybrid hierarchical clustering with applications to microarray data. *Biostatistics* 7(2), 286–301.
- Choi, S. and J. Park (2014). Nonparametric additive model with grouped lasso and maximizing area under the ROC curve. *Computational Statistics & Data Analysis* 77(C), 313–325.
- Cook, R. D. (1977). Detection of influential observation in linear regression. *Technometrics* 19(1), 15–18.
- Cox, M. A. and T. F. Cox (2008). Multidimensional Scaling. In *Handbook of Data Visualization*, pp. 315–348. New York: Springer.
- Critchley, F. (1985). Influence in principal components analysis. *Biometrika* 72(3), 627–636.
- Cutler, A. and L. Breiman (1994). Archetypal analysis. *Technometrics* 36(4), 338–347.
- Damian, D., M. Oresics, E. Verheij, J. Meulman, J. Friedman, A. Adourian, N. Morel, A. Smilde, and J. Van der Greef (2007). Applications of a new subspace clustering algorithm (cosa) in medical systems biology. *Metabolomics* 3, 69–77.
- Davis, N. and M. Lopez (2015). The 10 Types of NFL Quarterbacks. *FiveThirtyEight*. Available at: <https://fivethirtyeight.com/features/the-10-types-of-nfl-quarterbacks/>.
- DiStefano, C., M. Zhu, and D. Mindrila (2009). Understanding and Using Factor Scores: Considerations for the Applied Researcher. *Practical Assessment, Research and Evaluation* 14(20).
- Dorhout, P. and K. Cox (2016). K-State upgraded to "Highest Research Activity" in Carnegie Classification.
- Dudoit, S. and J. Fridlyand (2002). A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome biology* 3(7), 1–21.
- Everitt, B. (2012). *Cluster Analysis*. Wiley.
- Everitt, B. and T. Hothorn (2008). *An Introduction to applied multivariate analysis*. New York: Springer New York.
- Everitt, B. and T. Hothorn (2011). *An Introduction to applied multivariate analysis with R*. Springer.
- Fletcher, D. and A. Underwood (2002). How to cope with negative estimates of components of variance in ecological field studies. *Journal of Experimental Marine Biology and Ecology* 273(1), 89–95.

- Forero, P. A. and G. B. Giannakis (2012). Sparsity-exploiting robust multidimensional scaling. *IEEE Transactions on Signal Processing* 60(8), 4118–4134.
- Fraley, C. and A. Raftery (2002). Model-based Clustering, Discriminant Analysis and Density Estimation. *Journal of the American Statistical Association* 97, 611–631.
- Fraley, C., A. Raftery, T. Murphy, and L. Scrucca (2012). *mclust Version 4 for R: Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation*.
- Friedman, J. H., T. Hastie, and R. Tibshirani (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33(1), 1–22.
- Friedman, J. H. and J. J. Meulman (2004). Clustering objects on subsets of attributes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 66(4), 815–849.
- Fuentes, M. and M. Chavent (2015). *divclust: Divisive hierarchical clustering*. R package version 0.3.
- Ghojogh, B., A. Ghodsi, F. Karray, and M. Crowley (2020). Multidimensional Scaling, Sammon Mapping, and Isomap: Tutorial and Survey.
- Gorman, K. B., T. D. Williams, and W. R. Fraser (2014). Ecological sexual dimorphism and environmental variability within a community of Antarctic penguins (Genus *Pygoscelis*). *PLoS ONE* 9(3).
- Gorsuch, R. (1983). *Factor Analysis*. Lawrence Erlbaum Associates.
- Gower, J. (1966). Some Distance Properties of Latent Root and Vector Methods Used in Multivariate Analysis Published. *Biometrika* 53(3), 325–338.
- Greengard, P., Y. Liu, S. Steinerberger, and A. Tsyvinski (2020). Factor Clustering with t-SNE. pp. Available at SSRN: <https://ssrn.com/abstract=3696027>.
- Hampel, F. R. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association* 69(346), 383–393.
- Härdle, W. K. and L. Simar (2019). *Applied Multivariate Statistical Analysis*.
- Harmon, P. (2017, May). *Demystifying the Carnegie Classifications*. MS Writing Project.
- Harmon, P., S. McKnight, L. Hildreth, I. Godwin, and M. Greenwood (2019). An alternative to the carnegie classifications: Identifying similar doctoral institutions with structural equation models and clustering. *Statistics and Public Policy* 6(1), 87–97.
- Hastie, T., R. Tibshirani, and J. Friedman (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. New York, NY, USA: Springer New York Inc.

- Hastie, T., R. Tibshirani, and M. Wainwright (2015). *Statistical Learning With Sparsity* (1 ed.). CRC Press.
- Hennig, Christian; Meila, M. (2015). Cluster Analysis: An Overview. In *Handbook of Cluster Analysis* (CRC Press ed.), pp. 1–20. New York.
- Herzog, K. (2016). UW-Milwaukee elevated to elite status as research university. *Milwaukee Journal Sentinel*.
- Hildreth, L. (2013). *Residual Analysis for Structural Equation Modeling*. PhD Thesis, Iowa State University.
- Hinton, G. and S. Roweis (2002). Stochastic Neighbor Embedding. *Advances in neural information processing systems* 15, 833–840.
- Hoerl, A. E. and R. W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* 12(1), 55–67.
- Horst, A. M., A. P. Hill, and K. B. Gorman (2020). *palmerpenguins: Palmer Archipelago (Antarctica) penguin data*. R package version 0.1.0.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* 24(6), 417–441.
- Huber, P. (1981). *Robust Statistics*. John Wiley and Sons.
- Hui, F. K., S. Taskinen, S. Pledger, S. D. Foster, and D. I. Warton (2015). Model-based approaches to unconstrained ordination. *Methods in Ecology and Evolution* 6(4), 399–411.
- Im, D., N. Verma, and K. Branson (2018). Stochastic Neighbor Embedding under f-divergences. *arXiv*.
- Izenman, A. (2008). *Linear Discriminant Analysis*, pp. 237–280. New York, NY: Springer New York.
- Jacobucci, R., K. Grimm, A. M. Brandmaier, S. Serang, R. A. Kievit, F. Scharf, X. Li, and A. Ye (2021). regsem: Regularized structural equation modeling. R package version 1.8.0.
- James, G., D. Witten, T. Hastie, and R. Tibshirani (2013). *An Introduction to Statistical Learning*. New York: Springer New York.
- Janssens, K., I. Deraedt, O. Schalm, and J. Veeckman (1998). Composition of 15th–17th century archaeological glass vessels excavated in Antwerp, Belgium. pp. 253–267. Springer Verlag.
- Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer Series in Statistics.

- Joreskog, K. (1970). A general method for analysis of covariance structures. *Biometrika* 57, 239–251.
- Joreskog, K. (1973). A general method for estimating a linear structural equation system. *Structural Equation Models in the Social Sciences*, 85–112.
- Joreskog, K. (1978). Structural analysis of covariance and correlation matrices. *Psychometrika* 36, 109–133.
- Joreskog, K. and D. Sorbom (1982). Recent development in structural equation modeling. *Journal of Marketing Research* 19, 404–416.
- Kampert, M., J. Meulman, and J. Friedman (2016). rcosa: A software package for clustering objects on subsets of attributes. *Journal of Classification*.
- Kaufman, L. and P. Rousseeuw (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley Series in Probability and Mathematical Statistics.
- Kong, L., C. Qi, and H.-d. Qi (2019). Classical Multidimensional Scaling : A Subspace. 67(14), 1–15.
- Kosar, R. and D. Scott (2018a). Examining the Carnegie Classification Methodology for Research Universities. *Statistics and Public Policy* 5(1), 1–12.
- Kosar, R. and D. W. Scott (2018b). Examining the Carnegie Classification Methodology for Research Universities. *Statistics and Public Policy* 5(1), 1–12.
- Krijthe, J. H. (2015). *Rtsne: T-Distributed Stochastic Neighbor Embedding using Barnes-Hut Implementation*. R package version 0.16.
- Kruskal, J. B. (1964). Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis. *Psychometrika* 29(1), 1–27.
- LeCun, Y., C. Cortes, Y. Bengio, and P. Haffner (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 2278–2324.
- Lemberge, P., I. De Raedt, K. H. Janssens, F. Wei, and P. J. Van Espen (2000). Quantitative analysis of 16–17th century archaeological glass vessels using PLS regression of EPXMA and μ -XRF data. *Journal of Chemometrics* 14(5-6), 751–763.
- Liu, S., P.-T. Bremer, J. Thiagarajan, V. Srikumar, B. Wang, Y. Livnat, and V. Pascucci (2018). Visual exploration of semantic relationships in neural word embeddings. *IEEE Transactions on Visualization and Computer Graphics* 24(1), 553–562.
- MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, 281–297.

- Maechler, M., P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik (2019). *cluster: Cluster Analysis Basics and Extensions*. R package version 2.1.0 — For new features, see the ‘Changelog’ file (in the package source).
- Maechler, M., P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik (2022). *cluster: Cluster Analysis Basics and Extensions*.
- Mahalanobis, P. (1936). On the Generalized Distance in Statistics. *Journal of Genetics* 41, 159–193.
- Mantel, N. (1967). Ranking Procedures for Arbitrarily Restricted Observation. *International Biometric Society* 23(1), 65–78.
- McCormick, A. and C. Zhao (2005). Rethinking and Reframing the Carnegie Classification. *Change: The Magazine of Higher Learning* 37(5), 51–57.
- McQuitty, L. (1966). Single and multiple hierarchical classification by reciprocal pairs and rank order types. *Educational and Psychological Measurement* 26(2), 253–265.
- Merkle, E. C. and Y. Rosseel (2018). blavaan: Bayesian structural equation models via parameter expansion. *Journal of Statistical Software* 85(4), 1–30.
- Michalski, R., E. Diday, and R. Stepp (1981). A recent advance in data analysis: Clustering objects into classes characterized by conjunctive concepts. In L. Kanal and A. Rosenfield (Eds.), *Progress in Pattern Recognition*, pp. 33–56.
- Milligan, G. and M. Cooper (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika* 50(2), 159–179.
- Murtagh, F. and P. Legendre (2014). Ward’s hierarchical agglomerative clustering method: which algorithms implement ward’s criterion? *Journal of Classification* 31, 274–295.
- Oksanen, J., F. Blanchet, M. Friendly, R. Kindt, P. Legendre, D. McGlinn, P. R. Minchin, R. O’Hara, G. Simpson, P. Solymos, M. Stevens, E. Szoecs, and H. Wagner (2020). *vegan: Community Ecology Package*. R package version 2.5-7.
- Oskolkov, N. (2019). How to tune hyperparameters of t-SNE. Access at: <https://towardsdatascience.com/how-to-tune-hyperparameters-of-tsne-7c0596a18868>.
- Paradis, E. and K. Schliep (2019). ape 5.0: an environment for modern phylogenetics and evolutionary analyses in R. *Bioinformatics* 35, 526–528.
- Potts, C. (2021). *Adapting Archetypal Analysis to Scientific Imaging Applications*. Phd thesis, Montana State University.
- R Core Team (2022). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

- Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* 66(336), 846–850.
- Rangel, E. (2015). Tier one: Texas tech making progress towards prestigious designation. Available at: <http://www.lubbockonline.com/education/2015-03-07/tier-1-texas-tech-making-progress-toward-prestigious-designation>.
- Roseel, Y. (2012). lavaan: An R Package for Structural Equation Modeling. *The Journal of Statistical Software* 48(2), 1–36.
- Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20(C), 53–65.
- Sammon, J. W. (1969). Nonlinear Mapping Structure Analysis. *IEEE Transactions on Computers* C-18(5), 401–409.
- Scrucca, L., M. Fop, T. B. Murphy, and A. E. Raftery (2016). mclust 5: Clustering, Classification and Density Estimation using Gaussian Finite Mixture Models. *The R Journal* 8, 289–317.
- Shen, H. and J. Z. Huang (2008). Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis* 99, 1015–1034.
- Shin, J., R. Toutkoushian, and U. Teichler (2013). *University Rankings: Theoretical Basis, Methodology and Impacts on Global Higher Education*. Springer.
- Silver, N. (2012). *The Signal and the Noise: Why So Many Predictions Fail - but Some Don't*. New York: Penguin Press.
- Sokal, R. and P. Sneath (1963). *Principles of Numerical Taxonomy*. W.H. Freeman & Co.
- Spearman, C. (1904). “General intelligence” objectively determined and measured. *The American Journal of Psychology* 15, 201–292.
- Steiger, J. (1990). Structural model evaluation and modification: An internal estimation approach. *Multivariate Behavioral Research* 125, 173–180.
- Steinley, D. (2015). K-Medoids and Other Criteria for Crisp Clustering. In C. Hennig, M. Meila, F. Murtagh, and R. Rocci (Eds.), *Handbook of Cluster Analysis*, pp. 55–66. New York: CRC Press.
- Stratton, C. (2022). *Bayesian Hierarchical Latent Variable Models For Ecological Data Types*. Phd thesis, Montana State University.
- Sugar, C. A. and G. M. James (2003). Finding the number of clusters in a dataset: An information-theoretic approach. *Journal of the American Statistical Association* 98(463), 750–763.

- Tibshirani, R. (1996). Ridge shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society* 5(1), 267–288.
- Tibshirani, R., G. Walther, and T. Hastie (2001). Estimating the number of clusters in a data set via the Gap statistic. *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 63(2), 411–423.
- Torgerson, W. S. (1952). Multidimensional Scaling: I. Theory and Method. *Psychometrika* 17, 401–419.
- Tran, T. (2019). *Monothetic Cluster Analysis with Extensions to Circular and Functional Data*. PhD Thesis, Montana State University.
- Tran, T., M. Greenwood, and B. McGuire (2022). monoClust: Perform Monothetic Clustering with Extensions to Circular Data. R Package Version 1.2.1.
- Tucker, L.R., L. C. (1973). A reliability coefficient for maximum likelihood factor analysis. *Psychometrika*.
- University of Idaho (2016). University of Idaho Strategic Plan 2016-2025. Technical report. Available at: <https://www.uidaho.edu/-/media/UIIdaho-Responsive/Files/provost/StrategicPlan/UI-Strategic-Plan.ashx?la=en&hash=92419E8A855E321BCC6095DA187C2B2AA487F439>.
- University of Montana (2017). University of Montana Strategic Vision. Technical report. Available at: http://www.umt.edu/strategy/strategic-vision/UM_Strategic_Vision_1_1.pdf.
- Van Der Maaten, L. and G. Hinton (2008). Visualizing Data Using t-SNE. *Journal of Machine Learning Research* 9, 2431–2456.
- Van Der Maaten, L. and G. Hinton (2012). Visualizing non-metric similarities in multiple maps. *Machine Learning* 87(1), 33–55.
- Verghese, A.P. Jelesiewicz, E. (2016). Temple University reaches height of Carnegie Research classification. *Temple Now*.
- Ward, J. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association* 58, 236–244.
- Weisberg, S. (2014). *Applied Linear Regression* (4th ed ed.). Wiley series in probability and statistics. Somerset: Wiley.
- Williams, W. and J. Lambert (1959). Multivariate methods in plant ecology. *Journal of Ecology* 47, 83–101.
- Witten, D. M. and R. Tibshirani (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association* 105(490), 713–726.

- Witten, D. M. and R. Tibshirani (2018). *sparcl: Perform Sparse Hierarchical Clustering and Sparse K-Means Clustering*. R package version 1.0.4.
- Witten, D. M., R. Tibshirani, and T. Hastie (2009). A penalized matrix decomposition with applications to sparse principal components and canonical correlation analysis. *Biostatistics* 10, 515–534.
- Wright, S. (1921). Correlation and causation. *Journal of Agricultural Research* 20, 557–585.
- Yan, M. and K. Ye (2007). Determining the number of clusters using the weighted Gap statistic. *Biometrics* 63(4), 1031–1037.
- Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 67(2), 301–320.