NEW DATA MINING TECHNIQUES FOR SOCIAL AND HEALTHCARE SCIENCES

A Dissertation

By

KISUK SUNG

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Erick Moreno-Centeno
Committee Members,	Yu Ding
	David Matarrita-Cascante
	Justin Yates
Head of Department,	César O. Malavé

August 2016

Major Subject: Industrial Engineering

ABSTRACT

Data mining is an analytic process for discovering systematic relationships between variables and for finding patterns in data. Using those findings, data mining can create predictive models (e.g., target variable forecasting, label classification) or identify different groups within data (e.g., clustering). The principal objective of this dissertation is to develop data mining algorithms that outperform conventional data mining techniques on social and healthcare sciences. Toward this objective, this dissertation develops two data mining techniques, each of which addresses the limitations of a conventional data mining technique when applied in these contexts.

The first part (Part I) of this dissertation addresses the problem of identifying important factors that promote or hinder population growth. When addressing this problem, previous studies included variables (input factors) without considering the statistical dependence among the included input factors; therefore, most previous studies exhibit multicollinearity between the input variables. We propose a novel methodology that, even in the presence of multicollinearity among input factors, is able to (1) identify significant factors affecting population growth and (2) rank these factors according to their level of influence on population growth. In order to measure the level of influence of each input factor on population growth, the proposed method combines decision tree clustering and Cohen's d index. We applied the proposed method to a real county-level United States dataset and determined the level of influence of an extensive list of input factors on population growth. Among other findings, we show that poverty ratio is a highly important factor for population growth while no previous study found poverty ratio to be a significant factor due to its high linear relationship with other input factors.

The second part (Part II) of this dissertation proposes a classification method for *imbalanced data* — data where the majority class has significantly more instances than the minority class. The specific problem addressed is that conventional classification methods have poor minority-class detection performance in imbalanced dataset since they tend to classify the vast majority of the test instances as majority instances. To address this problem, we developed a guided undersampling method that combines two instance-selecting techniques — ensemble outlier filtering and normalized-cut sampling — in order to obtain a clean and well-represented subset of the original training instances. Our proposed imbalanced-data classification method uses the guided undersampling method to select the training data and then applies support vector machines on the sampled data in order to construct the classification model (i.e., decide the final class boundary). Our computational results show that the proposed imbalanced-data classification method outperforms several state-of-the-art imbalanced-data classification methods, including cost-sensitive, sampling, and synthetic data generation approaches on eleven open datasets, most of them related to healthcare sciences.

Dedicated to My Family

ACKNOWLEDGMENTS

It was a long “journey”. First, I would never have been able to finish my Ph.D. journey without the guidance of my advisor, Dr. Erick Moreno-Centeno. I really appreciate all his supports of ideas and funding that make my Ph.D. pursuit possible. His enthusiasm for the academic research always gave me motivation for finding a fundamental question and providing a solution for the research during tough times in the Ph.D. pursuit.

I would like to give my sincere appreciation to Dr. Yu Ding for serving on my committee. He provided his expertise to develop and refine the direction of this dissertation research. I will be grateful to Dr. Ding for accepting the committee member of my dissertation with pleasure. I am also especially grateful to Dr. David Matarrita-Cascante for serving on my committee and for encouraging me to explore new viewpoint about the dissertation research. I also express my sincere appreciation to Dr. Justin Yates for serving on my committee and helpful discussion about the dissertation research.

The members of Dr. Moreno’s lab contributed immensely to my personal and professional life in Texas A&M. I thank to Adolfo R. Escobedo and Yaping Wang for the good advice and collaboration. Also, special thanks to my best friends, Dr. Sangwhan Moon, Dr. Su Inn Park and Youngkwon Cha for time spent in College Station.

I would like to thank my family for their love and encouragement. I give my best sincere appreciation to my wife, Jinnie Park, for supporting and encouraging me during the hard time in the Ph.D. pursuit. Also, my lovely son and daughter, Aidan and Ella, were my greatest source of joy and strength during the Ph.D. Last, I owe my deepest gratitude to my parents and parents-in-law. Their endless love always supported all the time during this Ph.D. pursuit. I cannot express my gratitude in words. Their faithful supports and encouragements have totally overwhelmed me and I thank them from the bottom of my heart.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1 INTRODUCTION

Data mining is an analytic process for discovering systematic relationships between variables and for finding patterns in data. Using those findings, data mining can create predictive models (e.g., target variable forecasting, label classification) or identify different groups within data (e.g., clustering). Although data mining is already well-established and widely used in many fields including computer vision, natural language processing, and bioinformatics, data mining techiniques were not as widely used in the social and healthcare sciences until recently. Indeed, there is a growing interest to develop data mining techniques specifically tailored for the unique discovery problems arising in many fields such as the social sciences (Attewell et al., 2015).

In the social sciences, a very important problem is that of identifying the factors that promote or hinder population growth; data mining tools are ideal for addressing this problem. Identification of such factors is important for the effective public policy development plan and the allocation of infrastructure investments that align with the future population growth. To understand and explain population growth in terms of its underlying factors (i.e., economic, social, infrastructural, or amenity factors), population researchers have used statistical models such as linear regression analyses (Carlino and Mills 1987; Clark and Murphy 1996; Beeson et al. 2001; Chi and Voss 2010; Chi and Marcouiller 2011; Iceland et al. 2013). However, these studies sometimes showed inconsistent results between one another due to the presence of multicollinearity — a near-linear relationship between two or more input factors. Specifically, these previous studies included input factors without considering the statistical dependence among the included input factors.

In the healthcare sciences, a very important problem is that of determining the acceptance/rejection of cancer treatment plans; data mining tools are ideal for addressing this problem. For example, proposed radiation therapy (RT) plans need to be reviewed by RT experts to determine whether these RT plans are acceptable. This review process involves a laborious manual evaluation and a large amount of human resources. Thus, an automated system to classify the proposed RT plans as acceptable or erroneous can be useful in reducing the overload of RT experts and eliminating human errors. However, an RT-plan classification system developed using conventional classification methods would have poor erroneous-case detection performance. This is because (1) among the RT plans, erroneous cases are very rare and (2) conventional classification methods are designed to minimize the number of misclassified cases over the training data, and thus they would tend to predict the vast majority (if not all) of the test set cases as acceptable cases.

The principal objective of this dissertation is to develop data mining algorithms that outperform conventional data mining techniques on social and healthcare sciences. Toward this objective, this dissertation develops two data mining techniques, each of which addresses the limitations of a conventional data mining technique when applied in these contexts. First, we propose a novel data mining methodology that can identify significant input factors affecting a given target variable, even in the presence of multicollinearity. Moreover, the proposed method can rank these input factors according to their influence on the target variable. Then, we apply our proposed method to a real dataset in demographic research — identification of significant factors promoting or hindering population growth (Part I). Second, we develop a classification method for *imbalanced data* — data where the majority class has significantly more instances than the minority class. Then, we apply our proposed imbalanced-data classification method to eleven open datasets, most of them related to healthcare sciences (Part II).

Part I of this dissertation addresses the problem of identifying important factors that promote or hinder population growth. When addressing this problem, previous studies included variables (input factors) without considering the statistical dependence among the included input factors; therefore, most previous studies exhibit multicollinearity between the input variables. As most of these studies are based on regression analysis, their results are usually not consistent due to this multicollinearity. Moreover, these previous studies did not provide the level of influence (importance) of each input factor on population growth. Thus, we propose a novel methodology that, even in the presence of multicollinearity among input factors, is able to (1) identify significant factors affecting population growth and (2) rank these factors according to their level of influence on population growth. In order to measure the level of influence of each input factor on population growth, the proposed method combines decision tree clustering and Cohen's d index. Specifically, the proposed method first employs decision tree clustering to group communities into several clusters so that each cluster has similar values in the target variable (i.e., population growth) and also has similar values in each input factor. This clustering allows us to find the clusters with the highest and lowest population growth while also ensuring that the constituents within each cluster have similar characteristics. Then, Cohen's *d* index is used to measure the level of difference of each input factor between the clusters with the highest and lowest population growth, and thus identify the level of influence of each input factor on population growth. Even in the presence of multicollinearity, the final output of the proposed model is not affected by the correlation between input factors because decision tree clustering is not affected by the correlation between input factors and because the level of influence of the input factors on the target variable is measured independently for each input factor.

Part II of the dissertation proposes a classification method for *imbalanced data* — data where the majority class has significantly more instances than the minority class. The specific problem addressed is that conventional classification methods have poor minority-class detection performance in imbalanced dataset since they tend to classify the vast majority of the test instances as majority instances. To address this problem, we develop a guided undersampling method that combines two instance-selecting techniques — ensemble outlier filtering and normalized-cut sampling — in order to obtain a clean and well-represented subset of the original training instances. Specifically, the ensemble filtering technique aims to remove the outlier instances from both the majority and minority training data while normalized-cut sampling method aims to obtain a sample of majority instances that is spread out over the majority class region. Our proposed imbalanced-data classification method uses the guided undersampling method to select the training data and then applies support vector machines on the sampled data in order to construct the classification model (i.e., decide the final class boundary).

The remainder of this dissertation is organized as follows. Section 2 addresses the problem of identifying important factors for population growth (Part I). This section describes the details of the previous population growth models that have been used to identify factors predicting population growth. It also presents the proposed methodology for measuring the level of influence of each input factor on population growth. Then, it provides results on a real county-level dataset. These results show that the proposed method identifies significant factors for population growth even in the presence of multicollinearity and ranks the factors according to their level of influence on population growth. Section 3 addresses the imbalanced-data classification problem (Part II). This section presents a detailed literature review associated with the underlying data mining techniques of the proposed method. It also describes our proposed imbalanced-data classification method and shows that the proposed method achieves better performance than several state-of-the-art imbalanced classification methods. Section 4 summarizes the contributions of this dissertation.

2 PART I: A NEW METHODOLOGY FOR MEASURING THE LEVEL OF INFLUENCE OF EACH INPUT FACTOR ON POPULATION GROWTH

2.1 Introduction

Communities often face significant economic and social challenges that must be understood and overcome to ensure a stable and sustainable setting for their inhabitants and the physical environment where they reside. As communities constantly change, understanding the factors that promote such change and the consequences of such change is critical. For instance, in the case of communities with an initially low population density experiencing *boomtown* scenarios, examples of such factors and consequences would be physical infrastructure failing to meet the expansion demand, public policy inhibiting/limiting growth, poor social integration, and involvement in community affairs (Graber, 1974; Gilmore, 1976; Hunter and Smith, 2002; Smith et al., 2001). Without an appropriate understanding of the causes of community change, the resulting local experiences can be detrimental to the local living conditions; that, on occasion, can lead to the collapse of the community.

In order to develop integrated models capable of relating economic, policy, and geographic factors together to identify factors predicting population growth, previous studies have typically used statistical regression analyses such as ordinary least squares models or two–stage least squares lagged adjustment models (see, for example, Carlino and Mills

1987; Clark and Murphy 1996; Beeson et al. 2001; Chi and Voss 2010; Chi and Marcouiller 2011; Iceland et al. 2013). While highly important, these methodologies contain certain weaknesses. First, these statistical approaches do not determine the level of influence (importance) that each input factor has on population growth. In other words, these studies focused on identifying which input factors are better at predicting population growth, but did not rank the input factors according to their level of influence on population growth. This is because a low p-value (e.g., < 0.05) indicates that we can reject the null hypothesis (i.e., the coefficient of the corresponding input factor is equal to zero) but does not indicate the level of influence of the factor on population growth. Second, multicollinearity, which refers to a linear relationship between two or more input factors, may impact the usefulness of regression analysis (Greene, 2012; Chatterjee and Hadi, 2006; Montgomery et al., 2012). Since most previous studies selected input variables without considering their statistical dependence from each other (except the studies which introduced statistical techniques to avoid multicollinearity — see, for example, Deller et al. 2001; Chi and Voss 2010; Chi and Marcouiller 2011; Iceland et al. 2013), most previous studies exhibit multicollinearity between input variables and thus, this multicollinearity impacts the consistency of the results obtained using regression analysis.

To overcome the issues explained above, we develop a comprehensive data mining analysis of population growth. In this study, the proposed method employs population growth as our target variable. First, the proposed method uses decision tree clustering to group communities into several clusters so that each cluster has similar values in the target variable (i.e., population growth) and also has similar values in each input factor. This clustering allows us to find the clusters with the highest and lowest population growth and ensures that the constituents within each cluster have similar characteristics. Second, Cohen's *d* index is used to identify the level of influence that each input factor has on population growth by measuring the level of difference of each input factor between the clusters with the highest and lowest population growth. Even in the presence of multicollinearity, the final output of the proposed model is not affected by the correlation between input factors because decision tree clustering is not affected by the correlation between input factors and because the level of influence of the input factors on the target variable is measured independently for each input factor.

The remainder of this chapter is organized as follows: Section 2.2 summarizes previous models of community growth, focusing on population growth models. Section 2.3 introduces the data employed in this study and shows that there is a high level of multicollinearity among the input variables. Section 2.4 presents the proposed method and Section 2.5 presents the results. Section 2.6 compares the results of our proposed method to those of previous studies, and shows how multicollinearity impacts the consistency of the results obtained using regression analysis. Section 2.7 summarizes the main findings, and Section 2.8 addresses the limitations of the proposed method and suggests further research directions.

2.2 Literature Review

Community researchers using secondary data draw usually from two approaches to explain community growth. The first approach, characteristic of very early studies of community growth, focused on understanding community growth from an economic or a demographic perspective independently. Using this approach, academics with particular training were studying community growth based mainly on their area of expertise. Typically, economists were measuring economic growth through economic data while demographers and sociologists were examining community growth as measured by demographic data (see, for example, Pearl and Reed 1920; Pritchett 1891).

The second approach is more comprehensive and uses different types of information (e.g., demographic, economic, environmental, and policy variables) to explain community growth. As research advanced, researchers examining economic growth realized that demographic factors (e.g., population density, percentage of minorities present, educational attainment of the population), environmental factors (e.g., climate, topography, natural amenities), and policy factors (e.g., taxes, subsidies, regulations) needed to be included as input factors in their models in addition to economic factors (see, for example, Carlino and Mills 1987; Clark and Murphy 1996; Quigley 1998; Deller et al. 2001). Similarly, studies examining population growth also noted the importance of combining different types of explanatory factors such as economic factors (e.g., income, labor mobility), and cultural and environmental factors (e.g., personal preferences on community and residential characteristics) as predictors of population growth besides demographic factors (see, for example, Leslie and Richardson 1961; Sjaastad 1962; Golant 1971; Zelinsky 1971; Speare 1974; Fuguitt and Zuiches 1975; Greenwood 1975; Carlino and Mills 1987; Clark and Murphy 1996; Brown et al. 1997; McGranahan 1999; Deller et al. 2001; Beeson et al. 2001; Rupasingha and Goetz 2004; Brown 2002).

In the early and mid–1980s, studies used simple statistical tools such as correlation coefficient analyses (see, for example, Leslie and Richardson 1961) or ordinary least squares regressions (see, for example, Speare 1974; Greenwood 1975) when they conducted their statistical analyses. However, by the end of the 1980s, Carlino and Mills (1987) published their seminal article that moved community growth research forward. A trend was set to understand community growth by exploring economic and social factors simultaneously using a two–stage least squares lagged adjustment regression model. Research published during this period contained economic and population models at the same time. Such models were characterized by the inclusion of large sets of input variables/-factors (economic, social, political, infrastructural, biophysical, and geospatial) that were used as predictors for the two models present (economic and population). The two–stage least squares lagged adjustment regression models became the predominant statistical tool of studies during the 1990s (see, for example, Clark and Murphy 1996; Beeson et al. 2001; Rupasingha and Goetz 2004).

Recently, some studies have focused on improving community research models by overcoming the issue of multicollinearity. The issue of multicollinearity arises when there is a near-linear relationship among two or more input variables, and this multicollinearity leads to inaccurate estimates or low statistical significance values. The favored two–stage least squares lagged adjusted regressions of the 1990s were very vulnerable to multicollinearity. Previous studies that used regression analyses selected several input variables in the same type of category (i.e., high school degree ratio and college degree ratio in the education category) without considering the statistical dependence from other variables and thus, are most likely exposed to the risk of multicollinearity. When multicollinearity is predominant, (1) small changes in the data produce wide swings in the parameter estimates; (2) coefficients may have very high standard errors and low significance levels even though they are jointly significant and the R^2 for the regression is quite high; and, (3) coefficients may have the *wrong* sign or implausible magnitude in a regression analysis (Greene, 2012). Therefore, to overcome this multicollinearity issue, some researchers used only a subset of the input variables for calculating the level of significance (see, for example, Chi and Voss 2010; Iceland et al. 2013). Alternatively, Chi and Marcouiller (2011) and Deller et al. (2001) overcame this problem by merging the input variables into several category variables using *Principal Factor Analysis* (PFA) and *Principal Component Analysis* (PCA) respectively.

Table 1 shows the list of significant factors for population growth determined by previous regression-based studies. One observation, as mentioned in Section 2.1, is that previous studies did not provide the level of influence of each input variable on population growth. Another important observation of this table is that the results of previous population growth studies are not consistent with each other. For example, the variable College

Table 1: List of significant factors for population growth determined by previous regression-based studies

	Carlino	Clark	Beeson	McGranahan	McGranahan	Chi	Chi	Significance
Variable	et al.	et al.	et al.	(1999)	et al.	et al.	et al.	Ratio
	(1987)	(1996)	(2001)		(2002)	(2010)	(2011)	
Median Income ($)	O	O						100%
College Ratio (%)		O			X		O	67%
Temperature Gap ($^\circ F$)		O		O	O			100%
Poverty Ratio (%)		X			X			0%
July Humidity (%)				O	O			100%
Asian Ratio (%)								100%
January Temperature ($^\circ F$)				O	O			100%
WaterArea Ratio (%)			O	O	O			100%
Crime Rate (per 1000)		X						0%
Highway (TH$)	O	X					X	33%
Black Ratio (%)	X	O					O	67%
Population Density					O	O	O	100%
January Sun (hour)		O		O	O			100%
Local Net (%)	X	X						0%
Employment Rate (%)	O	O						100%
Hispanic Ratio (%)							O	100%

Notes: *O* denotes the variables which were determined as significant for population growth by the corresponding regression analysis. *X* denotes the variables which were determined as non-significant for population growth by the corresponding regression analysis. Unmarked cells indicate that the corresponding study did not include the corresponding variable. Significant ratio represents the ratio between the number of studies which determined the variable as significant and the number of all studies which included the variable in their regression model. A description of the list of variables appears in Section 2.3.

Ratio in Table 1 was determined as a significant factor for population growth by two previous regression analyses (Clark and Murphy 1996; Chi and Marcouiller 2011), while it was determined as a non-significant factor for population growth by one previous regression analysis (McGranahan and Beale, 2002). Similarly, Table 1 shows that the variables Highway Expenditure and Black Ratio were determined as significant factors for population growth by one and two previous studies, respectively; while they were determined as non-significant factors for population growth by two and one previous regression analyses, respectively. This observation shows that results of previous studies are not consistent. We attribute these inconsistent results to the multicollinearity between the input variables included in the respective models. Section 2.6.2 demonstrates how multicollinearity impacts the consistency of regression analysis results.

2.3 Data Description

Section 2.3.1 gives a brief description of the target variable (i.e., population growth) and the input variables (factors) used in this study. In Section 2.3.2, using the input variables introduced in Section 2.3.1, we measure the level of multicollinearity among the input variables and show that some of the variables have a high level of multicollinearity with other input variables.

2.3.1 List of Variables

This study employs county-level United States dataset as the population growth data. We chose the county level because a variety of categories of population data are available at the county level but not at the sub-county level. The data used in this study consists of 3,108 counties in the United States (Alaska and Hawaii are excluded). This study also excludes counties that cannot be tracked because of administrative or name changes between 2000–2010. The target variable (population growth) as well as most of the input variables were taken from the *USA counties*$^{\text{TM}}$ dataset of the U.S. Census Bureau (2011). The only exception were the natural amenity variables, which were taken from county-level natural amenity data provided by the U.S. Department of Agriculture (2011).

The *USA counties*$^{\text{TM}}$ dataset is part of a series of products featuring county-level

Table 2: Description of target and input variables obtained from county-level US dataset

Target Variable

Category	Variable	Description	County Average	Year
Target	Population Growth	Percent change of county population from 2000 to 2010	5.26%	2010

Input Variable

Category	Variable	Description	County Average	Year
Income	Median Income	Median household income of the county	$ 36,274	2000
	Poverty Ratio	People in poverty in the county given as a proportion	13.31%	2000
Policy	Federal Expenditure	Amount of money that the federal government expended	TH$ 484,422	2000
	Local Net	Local government budget balance given as a proportion	1.10%	1997 – 2002*
	Highway	Local government highway expenditure	TH$ 12,557	1997 – 2002*
Race	Black Ratio	Proportion of Black persons in the county	8.84%	2000
	Asian Ratio	Proportion of Asian persons in the county	0.77%	2000
	Hispanic Ratio	Proportion of Hispanic persons in the county	6.19%	2000
Amenity	WaterArea Ratio	Proportion of water area in the county	4.63%	2000
	January Temperature	Average temperature of January of the county	$32.90^{\circ}F$	1941 – 1970
	January Sun	Average sunny hours of January of the county	151.57 *hours*	1941 – 1970
	Temperature Gap	Temperature regression residual gap between January and July	$0.00^{\circ}F$	1941 – 1970
	July Humidity	Average July humidity of the county	56.13%	1941 – 1970
	Urban Influence Code	Urban–Rural classification	- (categorical variable)	1941 – 1970
	Topography Code	Topographic classification of land formation	- (categorical variable)	1941 – 1970
Other	Crime Rate	Number of violent crimes per 1000 persons in the county	0.24	2000
	College Ratio	Proportion of bachelor's degree of the county	16.50%	2000
	Employment Rate	Proportion of employed people to population(15 and over)	65.94%	2000
	Population Density	Population per square mile of the county	245.10 persons/mile^2	2000

*The data of $Local\ Net$ and $Highway$ is from the weighted averages of 1997 and 2002 because of unavailability of year 2000.

data and contains a collection of data from most major departments and agencies such as the U.S. Census Bureau, the Federal Bureau of Investigation, the Internal Revenue Service, etc. The topics of the data vary from demographic to economic and governmental data. The use of credible, governmental data ensures consistency with the data of other papers.

In this study, the target variable, population growth, is defined as the percent change of county population from 2000 to 2010. For input variables, many previous studies tried to include as many input variables as possible (see, for example, Carlino and Mills 1987; Clark and Murphy 1996; Beeson et al. 2001; Deller et al. 2001). To avoid including an unnecessarily large number of input variables, we classified input variables into five categories (*Income*, *Policy*, *Race*, *Natural Amenity*, and *Others*) and attempted to minimize the number of input variables while ensuring each category was adequately represented; this was achieved by selecting the variables that are commonly used ones in past studies. Table 2 gives a description of each input–variable category followed by the explicit list of variables within the category. A detailed description of the input variables appears in Appendix A.

2.3.2 Multicollinearity among the Input Variables

This section measures the level of multicollinearity among the input variables used in this study and shows that some of the variables have a high level of multicollinearity with other input variables. The linear relationship between the input variables can be measured by the so-called $Variance\ Inflation\ Factor$ (VIF). Let X_i be the input variable i and R_i^2 be the coefficient of determination when X_i is regressed against all other input variables. Then, VIF of the input variable i is

$$VIF_i = \frac{1}{1-R_i^2}, i = 1, ..., p \tag{2.1}$$

where p is the number of input variables (Chatterjee and Hadi, 2006). VIF of input variable X_i measures the amount by which the variance of X_i's regression coefficient is increased due to the multicollinearity between input variables. If input variable X_i has a strong linear relationship with other input variables, then its VIF would be large since R_i^2 would be close to 1. Pan and Jackson (2008) suggested that a VIF in excess of 4 is an indication that multicollinearity may cause problems in regression analysis.

Table 3 gives the VIF of each input variable described in Section 2.3.1 except for

Table 3: Variance Inflation Factor (VIF) index of input variables. The VIF of input variable is large if the input variable has a strong linear relationship with other input variables (which refers to high level of multicollinearity). The variables are sorted in decreasing order of VIF.

Variable	Variance Inflation Factor (VIF)
Federal Expenditure (TH$)	**5.65**
Highway (TH$)	**5.21**
Median Income ($)	**5.00**
Poverty Ratio (%)	**4.97**
College Ratio (%)	2.55
Black Ratio (%)	2.15
January Temperature ($^\circ F$)	2.03
Hispanic Ratio (%)	2.00
Asian Ratio (%)	1.94
July Humidity (%)	1.78
January Sun (hours)	1.58
Crime Rate (per 1000)	1.38
Employment Rate (%)	1.37
Temperature Gap ($^\circ F$)	1.32
Population Density (per mile ^{2})	1.29
WaterArea Ratio (%)	1.15
Local Net (%)	1.03

Notes: Bold text indicates that corresponding input factor has a high VIF index (VIF $\geq$ 4.0). TH$ means thousand $.

the nominal variables (urban influence code and topography code). The factors whose VIF exceeded 4 are median income (5.00), poverty ratio (4.97), federal expenditure (5.65), and highway expenditure (5.21). VIF values of all other input variables are less than 3. Therefore, the four above-mentioned input variables have a high linear relation with other input variables, and there exists multicollinearity between the input variables. Two extreme examples are (1) poverty ratio, which has a high negative correlation with median income (-0.79) and (2) highway expenditure, which has a high positive correlation with federal expenditure (0.89).

2.4 Method of Analysis

The proposed approach groups counties into clusters and uses a decision tree method to find the classification rules of each cluster. Then, Cohen's *d* index is used to measure the level of influence that each input factor has population growth (target variable). This approach (decision tree combined with Cohen's *d* index) not only avoids the multicollinearity issue but allows us to rank the input factors according to their level of influence on the population growth. The reminder of this section provides a detailed description and the implementation of the method.

2.4.1 Decision Tree: Clustering of Counties

Clustering analysis partitions the data records (counties in our case) into clusters, such that closely related data records are in the same cluster and disparate data records are in different clusters. Among clustering analysis approaches, our proposed method uses $Classification\ And\ Regression\ Tree$ (CART), first introduced by Breiman et al. (1984). The proposed method uses the CART algorithm instead of other clustering methods because its results are easy to interpret. Another advantage of CART is that it can use nominal variables as input variables without the computational burden incurred by other methods in the presence of nominal input factors (Loh, 2011).

Conceptually, the CART algorithm partitions the data records in the input variable space as follows. The algorithm starts with one node (the *root node*) containing all of the data records. Then, it chooses a splitting input variable, for example x_1, and a splitting point, s, and partitions the data records into two *interior nodes* (also called *child nodes*) such that one node contains all the data records satisfying $x_1(i) \geq s$ where $x_1(i)$ is the value of input variable x_1 of data record i and the other node contains all the data records satisfying $x_1(i) < s$. The splitting input variable and splitting point are chosen so that the sum of the heterogeneities with respect to the target variable on the child nodes is minimized — the technical details are given below. The splitting procedure is applied recursively until a pre-specified stopping criteria is met. In other words, each interior node is further split into child nodes unless the stopping criteria has been met. A *leaf node*, a node that cannot be further split, contains the data records comprising a cluster created by the model. Note that there is a unique path from the root node to each leaf node (cluster). Moreover, from these paths, one can easily extract the classification rules needed to determine which data records belong to each cluster.

The splitting procedure in the CART algorithm chooses the splitting variable j and splitting point s in order to minimize the following function (Hastie et al., 2009):

$$\min_{j,s}\Big[\sum_{\text{record } i:\, x_j(i)\geq s} (y(i)-c_1)^2 + \sum_{\text{record } i:\, x_j(i)<s} (y(i)-c_2)^2\Big], \tag{2.2}$$

where $y(i)$ is the response of data record i, $x_j(i)$ is the value of input variable j of data record i, $c_1 = \text{average}(y(i)|x_j(i) \geq s)$, and $c_2 = \text{average}(y(i)|x_j(i) < s)$.

A practical implementation of any decision tree model (including CART) needs to consider when to stop partitioning the dataset (the stopping rule). Otherwise, the CART algorithm would terminate with only one data record on each leaf node. While previous research proposed several stopping rules, our proposed method uses the stopping rule proposed by Duda et al. (2001), as the clusters obtained should be of comparable sizes for clustering analysis to give meaningful results. The details of the stopping rule by Duda et al. (2001) are as follows: if an interior node contains more records than a pre-specified number (500 in our case), then the node is split, and the splitting point *s* is chosen so that both child nodes have at least a pre-specified number of records (250 in our case). Thus, the proposed stopping rule guarantees that all clusters will contain between 250 and 500 counties. These thresholds (250 and 500) are chosen because this study expects to partition the counties into roughly 10 clusters. (This said, we tried several other stopping rules, and the clusters obtained had very similar characteristics to those found using this rule.)

One may be tempted to conclude that the input factors chosen as splitting variables by the CART algorithm are the variables with the highest influence on the target variable. However, this would be incorrect as each splitting variable is only a high influence variable when restricted to the data records in the node that it partitioned. Therefore, in order to find the factors that contribute significantly to the value of the target variable, this study uses Cohen's *d* index after the clustering process; this procedure is described next.

2.4.2 Cohen's d: Level of Influence of Each Input Factor

As argued in Section 2.2, previous population growth studies did not rank the input factors according to their level of influence on population growth. Therefore, here we propose Cohen's *d* index, first introduced by Cohen (1962) and Cohen (1988), as a measure for the level of influence that each input factor has on population growth (target variable).

Specifically, after grouping the counties into clusters using the CART algorithm, the proposed method measures Cohen's d index between the high and low population growth clusters for each input variable in order to measure its respective level of influence on population growth and rank the input variables according to Cohen's d index. Below, we give a brief description of Cohen's d.

Given two groups and a variable/factor of interest, Cohen (1962) introduced the concept of effect size with the aim of quantifying the difference of the variable/factor between the groups in a units-free measure. Cohen (1962)'s effect size aims to measure the difference in means, in units of standard deviation, and provide a standardized size of difference, which is independent of the variable's specific metric. An important assumption of Cohen's effect size is that both groups are normally distributed and have equal variance. Given two groups and a variable of interest, Cohen defined effect size as follows (Cohen, 1962):

$$\text{Effect size} = \frac{\mu_1 - \mu_2}{\sigma}, \tag{2.3}$$

where $\mu_1 - \mu_2$ is the difference in means between group 1 and group 2, and σ is the (common) standard deviation.

Using the same aforementioned assumption, Cohen (1988) formalized effect size based on Cohen (1962) as follows. Given two groups and a variable of interest, Cohen (1988) defined effect size of a variable, referred to as Cohen's d, as

$$d = \frac{\bar{x}_1 - \bar{x}_2}{s}, \tag{2.4}$$

for the directional case, and as

$$d = \frac{\mid \bar{x}_1 - \bar{x}_2 \mid}{s}, \tag{2.5}$$

for the non-directional case, where $\bar{x}_1 - \bar{x}_2$ is the difference in sample means between group 1 and group 2, and s, the pooled standard deviation, is defined as

$$s = \sqrt{\frac{(n_1 - 1){s_1}^2 + (n_2 - 1){s_2}^2}{n_1 + n_2 - 2}}, \tag{2.6}$$

where the group sizes and sample standard deviations of group 1 and group 2 are denoted as n_1, n_2, s_1, and s_2.

Cohen (1988) provided an intuitive interpretation for the value of d based on the (inverse) relation of d to the percent of overlap between the two groups' distributions. Specifically, assuming that the groups being compared are normally distributed with the same variance, a Cohen's d index of 0.0 indicates that the distribution areas of the two groups have a 0% of non-overlap (i.e., the population distributions of both groups are perfectly superimposed). Similarly, a Cohen's d indices of 0.2, 0.5, and 0.8 indicate a 14.7%, 33.0%, and 47.4% of non-overlap in the two distribution areas, respectively. In terms of the percent of non-overlap of two distribution, Cohen (1988) suggested the guideline of defining the term for size of d as *small* when $0.2 \leq d < 0.5$ ($14.7\% - 33.0\%$ of non-overlap), *medium* when $0.5 \leq d < 0.8$ ($33.0\% - 47.4\%$ of non-overlap), and *large* when $d \geq 0.8$ (over 47.4% of non-overlap) for a common conventional frame of reference.

2.4.3 Process of the Proposed Method

The following steps describe how the proposed method combines CART and Cohen's *d* to determine the level of influence of each input variable/factor on the target variable.

1. Use the CART algorithm to cluster the counties into several clusters.
2. Take the counties in the two clusters with the highest average target variable value and create a *group*. Thus, this group will contain counties with both a high average target value and relatively homogeneous input-variable values. Similarly, take the counties in the two clusters with the lowest average target value and create a group. These groups are referred to as a top group and a bottom group respectively.
3. For each input variable, calculate the Cohen's *d* index between the top and bottom groups.
4. Rank the variables according to Cohen's *d* index; those with the highest (respectively the lowest) index are the variables/factors with the highest (respectively the lowest) influence on the target variable.

Figure 1 illustrates the process of the proposed method, decision tree combined with Cohen's *d* index. Note that Cohen's *d*, the proposed index for measuring the level of influence of the variable between the groups, is measured independently for each variable. Therefore, when Cohen's *d* measures the level of influence of each input variable on population growth, the correlation between input variables does not affect the calculation.

The general idea behind the above procedure is as follows. The top and bottom

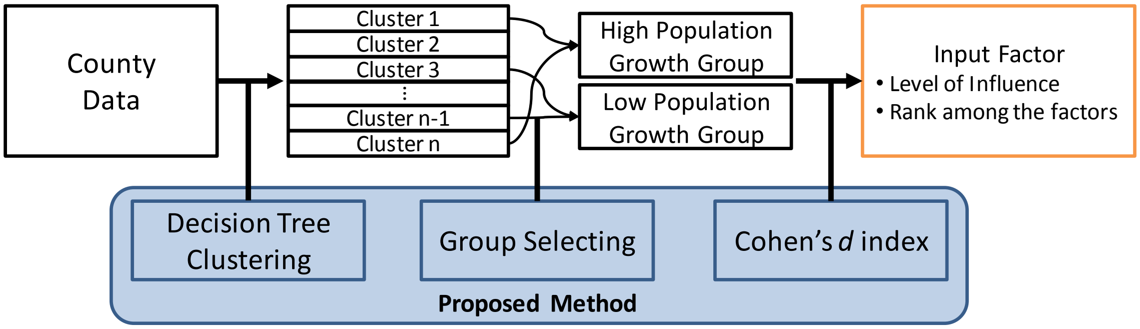


Figure 1: Process of the proposed method, decision tree combined with Cohen's *d*

groups contain counties with relatively homogeneous input-variable values (this is a property of the clustering obtained with CART algorithm). Moreover, the top and bottom groups contain counties with high and low target variable values respectively. Since the proposed procedure uses Cohen's *d* to find the input variables on which these two groups differ significantly regardless of correlations between the input variables, it is reasonable to infer that the input variables/factors with the highest (lowest) Cohen's *d* index are those with the highest (lowest) influence on the target variable.

Note that an alternative procedure to CART clustering to find the counties in the top (bottom) group is to include the individual counties with the highest (lowest) target variable value. However, using CART clustering to find the top and bottom groups is a better procedure because the groups clustered by CART will be homogeneous not only in the target variable values, but also in the input variable values.

2.5 Results

This section presents the results obtained by applying the proposed method to the real county-level United States dataset described in Section 2.3. Using the CART algorithm and population growth as the target variable, we partitioned the 3,108 counties into 10 clusters as shown in Figure 2. As previously mentioned, the decision tree model creates a clustering that is easy to interpret. Specifically, given any county, it is straightforward to determine the cluster it belongs to: start at the root node and follow the splitting rules satisfied by the given county until a leaf node/cluster is reached. For example, Figure 2 shows that cluster 10, which is the cluster whose counties have the highest average population growth (21.57%), contains the counties with a median income greater than or equal to $\$42,447.5$ and a January average temperature greater than or equal to $31.25°F$. On the other hand, cluster 2, the cluster with the highest average population decrease (-6.25%), contains the counties with a median income less than $\$42,447.5$, a population density less than 8.25 persons per square mile, and a temperature residual gap between January and July greater than or equal to $-0.13°F$.

For each cluster, Table 4 gives the cluster characteristics and the average value (taken over all the counties in the respective cluster) of each variable. From the table, it is interesting to note that even though there is a difference of input factor values between the highest population growth cluster and the lowest population growth cluster, the ordering of the clusters according to their population growth is not identical to the ordering of the clusters according to any other input factor. This suggests that the relationship between

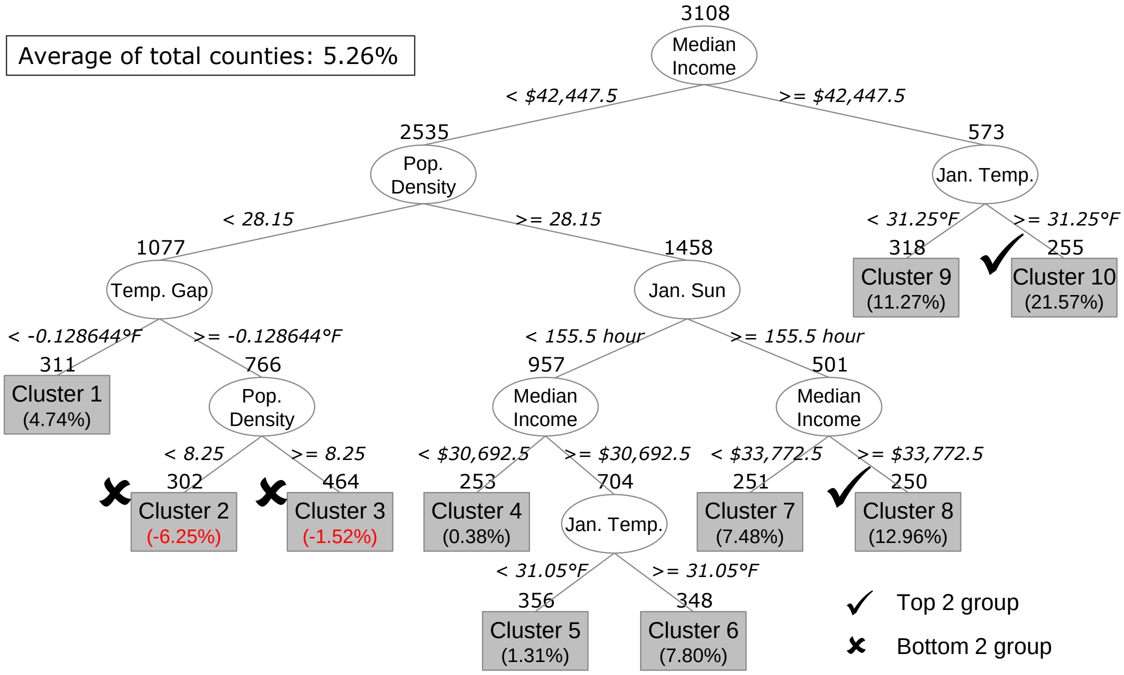


Figure 2: Decision tree obtained by applying CART to the county-level United States dataset. The number above each *node* denotes the number of counties inside the node; the label on an interior node is the variable chosen to partition the data records in this node into its two child nodes; the splitting rule and splitting point are given on the edge from the parent node to each child node; the number inside each *leaf node* denotes the target variable's average value of the data records in the cluster.

population growth and each input factor among clusters is not linear. This supports the Cohen's *d* approach, which measures the difference between the high population growth clusters and the low population growth clusters for each input factor, instead of estimating the input factor coefficients in the linear regression analysis.

Table 5 gives Cohen's *d* for each input factor when using population growth as the target variable. The variables are sorted in decreasing order of the influence (Cohen's *d*) on population growth. Table 5 also gives the characteristics of the top (cluster 10 and 8) and bottom (cluster 3 and 2) groups. According to the size of *d* described in Section 2.4.2,

Table 4: Characteristics of the clusters obtained by applying CART to the county-level United States dataset. The clusters are sorted in decreasing order of average population growth.

		Target Variable	Input Variables							
Cluster ID	Number of Counties	Population Growth (%)	Median Income ($)	Poverty Ratio (%)	Federal Expenditure (TH$)	Local Net (%)	Highway (TH$)	Black Ratio (%)	Asian Ratio (%)	Hispanic Ratio (%)
10	255	21.57	51,198	8.01	1,423,880	0.98	35,974	9.80	2.61	7.70
8	250	12.96	37,643	12.13	968,993	0.73	18,752	13.08	1.12	8.54
9	318	11.27	50,266	6.72	839,383	-0.38	29,640	2.51	1.33	4.23
6	348	7.80	35,296	13.17	569,065	0.33	10,182	10.92	0.76	3.05
7	251	7.48	30,478	17.67	297,118	1.37	4,707	22.08	0.56	8.89
1	311	4.74	32,280	15.02	94,430	1.88	3,784	5.79	0.38	7.80
5	356	1.31	37,644	10.45	500,026	0.32	15,389	2.83	0.64	2.33
4	253	0.38	26,775	19.96	419,669	0.88	9,983	15.62	0.35	2.59
3	464	-1.52	31,673	15.27	77,067	1.61	3,042	9.70	0.26	6.79
2	302	-6.25	31,277	15.31	37,748	3.23	1,404	1.01	0.22	11.39
Average	310.8	5.26	36,274	13.31	484,422	1.10	12,557	8.84	0.77	6.19

	Input Variables								
Cluster ID	Crime Rate (per 1000)	College Ratio (%)	Employment Rate (%)	Population Density (per mile^2)	WaterArea Ratio (%)	January Temp. ($^\circ F$)	Temp. Gap ($^\circ F$)	January Sun (hours)	July Humidity (%)
10	0.31	25.94	67.38	944.90	8.56	40.37	-1.63	153.98	61.22
8	0.39	18.12	67.79	237.45	6.83	40.21	0.73	175.71	60.47
9	0.17	24.23	75.79	349.93	7.54	23.30	-0.47	141.99	54.58
6	0.28	14.88	64.58	351.41	4.25	38.78	-0.40	132.64	62.50
7	0.44	13.38	60.66	134.41	4.11	44.38	0.57	175.73	63.55
1	0.23	15.58	63.53	11.37	4.79	28.32	-4.36	143.43	43.81
5	0.17	16.17	67.26	224.75	6.71	23.15	-0.91	123.64	59.34
4	0.26	11.00	51.40	399.72	2.73	39.17	-0.18	132.17	64.24
3	0.17	12.75	63.86	17.95	1.51	32.04	2.65	163.44	53.50
2	0.14	15.25	75.10	3.82	1.09	26.62	2.90	180.72	42.48
Average	0.24	16.50	65.94	245.10	4.63	32.90	0.00	151.57	56.13

Notes: TH$ means thousand $.

it is observed that the input factors which have *large* Cohen's d indices $(d \geq 0.8)$ are median income, college ratio, temperature gap, poverty ratio, July humidity, Asian ratio, and January temperature. That is, if we use overall county average as the threshold values, we expect counties that have a medium income greater than $\$36,274$, a college degree ratio greater than 16.5%, a temperature gap less than $0.00^\circ F$, a poverty ratio less than 13.31%, a July humidity greater than 56.13%, an Asian ratio greater than 0.77%, and a January tem-

Table 5: Cohen's *d* index of each input variable when using the county-level United States dataset. The variables are sorted in decreasing order of Cohen's *d*.

Variable	Bottom Group	Overall Population	Top Group	Cohen's *d*
Number of Counties	766	3,108	505	-
Population Growth (%)	-3.38	5.26	17.31	-
Median Income ($)	**31,517**	**36,274**	**44,488**	**1.86**
College Ratio (%)	**13.74**	**16.50**	**22.07**	**1.27**
Temperature Gap ($^\circ F$)	**2.75**	**0.00**	**-0.46**	**1.01**
Poverty Ratio (%)	**15.29**	**13.31**	**10.05**	**1.00**
July Humidity (%)	**49.15**	**56.13**	**60.85**	**0.91**
Asian Ratio (%)	**0.24**	**0.77**	**1.88**	**0.86**
January Temperature ($^\circ F$)	**29.90**	**32.90**	**40.29**	**0.86**
WaterArea Ratio (%)	1.34	4.36	7.70	0.68
Crime Rate (per 1000)	0.16	0.24	0.35	0.67
Federal Expenditure (TH$)	61,565	484,422	1,198,689	0.62
Highway (TH$)	2,396	12,557	27,448	0.54
Black Ratio (%)	6.27	8.84	11.42	0.38
Population Density (per mile ^{2})	12.38	245.10	594.67	0.29
January Sun (hours)	170.25	151.57	164.73	0.19
Local Net (%)	2.25	1.10	0.86	0.18
Employment Rate (%)	68.29	65.94	67.58	0.04
Hispanic Ratio (%)	8.60	6.19	8.11	0.04

Notes: Bold text indicates that corresponding input factor has a large Cohen's *d* index ($d \geq 0.8$). TH$ means thousand $.

perature greater than 32.90°F to be more likely to have a high positive population growth. It is counterintuitive that counties with high July humidity have a high positive population growth; however, this is because the top group (high population growth group) has more counties in the southern states of the United States such as Georgia or Florida than the bottom group (low population growth group). This geographical difference between the top and bottom group is a possible reason for the positive effect of high July humidity on population growth.

2.6 Discussions

Section 2.6.1 compares the results of our proposed method to those of previous studies. Then, we discuss the reasons of differences between the results of our proposed method and those of regression analysis. Section 2.6.2 shows how multicollinearity impacts the consistency of the results obtained by using regression analysis while our proposed method is not affected by the existence of multicollinearity.

2.6.1 Results Comparison with Previous Studies

This chapter aims to identify the level of influence that each input factor has on population growth regardless of the presence of multicollinearity by using a decision tree approach combined with Cohen's d index. The Cohen's d index analysis of every input factor on population growth confirmed that income–related factors, such as median income and poverty ratio, have a high influence on population growth. In addition, other factors with high influence on population growth include college ratio, natural amenity (temperature gap, July humidity, January temperature), and Asian ratio factors. This finding aligns with those reported earlier in the literature. Specifically, Carlino and Mills (1987) and Clark and Murphy (1996) found that median income is a significant factor for population growth; Clark and Murphy (1996) and Chi and Marcouiller (2011) found that college ratio is a significant factor for population growth; and Clark and Murphy (1996), McGranahan (1999) and McGranahan and Beale (2002) found that natural amenities are significant factors for population growth.

It is interesting that no other study that included poverty ratio as input factor found poverty ratio to be a significant factor of population growth because of the limitation of the regression analyses used in previous studies (see, for example, Clark and Murphy 1996; McGranahan and Beale 2002). For instance, in the study of McGranahan and Beale (2002), they found that poverty ratio was not a significant variable and attributed the exclusion of poverty ratio from the significant factors to the high negative correlation of poverty ratio with high school completion rate. Similarly, in the study of Clark and Murphy (1996), the poverty ratio was not included in the significant variable list. Next, we elaborate on this observation.

2.6.2 Inconsistency of the Results Obtained by Using Regression Analysis

This section aims to explain why our method found poverty ratio to have a large influence on population growth, while all previous studies found poverty ratio was not a significant factor for population growth. In addition, this section shows how multicollinearity impacts the consistency of the results obtained by using regression analysis. Finally, this section shows that the existence of multicollinearity does not affect the final output of our proposed method.

First, we give a brief review of regression analysis. The standard linear regression model in the case of a single target variable is an ordinary least square (OLS) and can be expressed in matrix notation as follows:

$$Y = X\beta + \varepsilon, \tag{2.7}$$

where Y is a $n \times 1$ vector of n observations of the target variable, X is a $n \times p$ matrix of input variables, β is a $p \times 1$ vector of coefficients, and ε is a $n \times 1$ vector of n error terms.

Past studies used regression analysis to identify factors that promote community growth such as economic growth or population growth (see, for example, Carlino and Mills 1987; Clark and Murphy 1996; Beeson et al. 2001; Chi and Voss 2010; Chi and Marcouiller 2011; Iceland et al. 2013). From regression analysis, the model selects important factors for community growth by their significant levels from the regression analysis. Also, the model measures the direction of the effect of those factors by the sign of the corresponding coefficient. For example, the positive coefficient of input variable X_i means that the target variable increases as the input variable X_i increases. Thus, the direction of the effect of input factor to target variable is positive when the coefficient of the input factor is positive. Similarly, the negative coefficient of input variable X_i means that the target variable decreases as the input variable X_i. increases. Thus, the direction of the effect of input factor to target variable is negative when the coefficient of the input factor is negative.

As mentioned in Section 2.3.2, poverty ratio has a high level of multicollinearity (VIF of poverty ratio = 4.97), and has a especially high negative correlation with *median income* (-0.79). We conjecture that this is why our results conflict with those on previous studies. To support this conjecture we formulated, two ordinary least square models with a different set of input variables.

Table 6 compares the results of regression analysis and our method for the variable poverty ratio when using two different input variable datasets. The first dataset, *dataset original*, includes the same input variables as the input variables used in Section 2.5 except nominal variables (urban influence code and topographic code). The second dataset,

Table 6: Comparison of results between our method (decision tree combined with Cohen's *d* index) and regression analysis for the variable poverty ratio when using two different input variable sets.

	Proposed (Decision tree + Cohen's d)				**Regression**	
	Average Poverty Ratio (%)			**Cohen's d of Poverty Ratio Variable**	**Standardized Coefficient of Poverty Ratio**	**p-value of Poverty Ratio Variable**
	Bottom Group	**Overall Population**	**Top Group**			
original	15.29	13.31	10.05	1.00	0.08	0.01
$(original - median\ income)$	15.20	13.31	9.27	1.22	-0.35	0.00

dataset $(original - median\ income)$, includes the same input variables as *dataset original* except for the variable median income which was excluded. In both regressions, the *p*-value of the variable poverty ratio is less than 0.05 (0.01 for the first dataset and 0.00 for the second dataset). Thus, the variable poverty ratio is deemed as a statistically significant factor (at confidence level 0.05) for population growth in both datasets. However, the results of the two regressions are inconsistent. In the first regression, the standard coefficient of poverty ratio is 0.08, while the standard coefficient of poverty ratio in the second regression is -0.35. We attribute this opposite sign of coefficient of poverty ratio to the multicollinearity effect, in particular, the high correlation between median income and poverty ratio.

In contrast, the results obtained with the proposed method are consistent. Specifically, in both datasets, the Cohen's *d* index of the variable poverty ratio is always large ($d \geq 0.8$) (1.00 for the first dataset and 1.22 for the second dataset). Also, the average poverty ratio in the bottom group (the two clusters with the lowest population growth) is higher than the average of poverty ratio in the top group (the two clusters with the highest population growth) in both of models. This observation indicates that, in both datasets, the top group (the two clusters with the highest population growth) is *less poor* than the bottom group (the two clusters with the lowest population growth).

2.7 Conclusions

The results obtained with the proposed method complement the population growth this literature in several ways. First, even in the presence of multicollinearity, the final output of the proposed model is not affected by the correlation between input factors because decision tree clustering is not affected by the correlation between input factors and because the level of influence of the input factors on the target variable is measured independently for each input factor. That is, when the proposed method calculates the level of influence that each input variable has on the target variable, the correlation between this input variable and other input variables does not affect the calculation. In contrast, previous studies that used regression analysis included input variables without considering the statistical dependence among the included input variables and thus, most previous studies exhibit multicollinearity between the input variables used. Consequently, their results are usually not consistent due to this multicollinearity.

Second, the proposed method not only identifies significant factors for population growth, but also allows us to measure the level of influence that each input factor has on the target variable. The proposed method uses data–clustering approach to group counties into several clusters so that we can find the clusters with the highest and lowest population growth while also ensuring that the constituents within each cluster have similar characteristics. Using these clusters with high and low population growth, the proposed method quantifies the level of influence that each input variable has on population growth by measuring the Cohen's *d* of each input variable between the clusters with high and low population growth. Using a real county-level United States dataset, the proposed method found that the factors with the highest influence on population growth, listed in decreasing level of influence, are: median income, college ratio, temperature gap, poverty ratio, July humidity, Asian ratio, and January temperature. In contrast, previous studies identified these input factors as significant factors for population growth but did not rank the input factors according to their level of influence (importance) on population growth.

2.8 Future Work

The method proposed in this chapter identified significant factors affecting population growth and successfully provided the level of influence that each input factor has on population growth. However, this study is limited as it focused on one region (United States), one time period (2000-2010), and one scale (county). We acknowledge that the statistical significance and the level of influence of the factors could vary from one region to another, from one time period to another, and from one scale to another. Thus, generalizing these results to other regions, other time periods, and other scales could be a valuable direction of future research.

3 PART II: GUIDED UNDERSAMPLING FOR IMBALANCED DATA CLASSIFICATION

3.1 Introduction

In an *imbalanced* dataset, the majority class has significantly more instances than the minority class. Conventional classification methods perform poorly when the dataset is imbalanced. This is because conventional classification methods are designed to minimize the number of misclassified instances over the training data, and thus they tend to predict the vast majority of the test set instances as majority instances (Chawla et al., 2004).

There are three approaches to address the imbalanced-data classification problem: (1) the cost-sensitive approach, (2) the sampling approach, and (3) the synthetic data generation approach. The cost-sensitive classification methods assign, during the training process, a lower misclassification cost (penalty) to the majority-class instances than to the minority-class instances (throughout this part, the instances with the majority class label are denoted as majority instances; similarly, the instances with the minority class label are denoted as minority instances). However, determining the optimal cost ratio to use within a cost-sensitive framework is challenging because the classification performance is very sensitive to the cost ratio used (Byon et al., 2010; Pourhabib et al., 2015).

The sampling classification methods balance the number of instances between the classes by oversampling the minority instances (i.e., resampling the minority instances with replication) or undersampling the majority instances. Despite the easy implementation and the generally improved performance, oversampling techniques might make the classification model over-fit to the replicated instances (Mease et al., 2007) while undersampling techniques might not construct an accurate classification model due to the loss of majority instances (Kubat and Matwin, 1997; Estabrooks et al., 2004).

The synthetic data generation classification methods balance the number of instances between the classes by creating artificial minority instances. However, as argued by Pourhabib et al. (2015), "the current literature does not present a consensus concerning the effectiveness of data synthesizing mechanisms." Moreover, using artificial data that are not among the original observations always makes researchers less comfortable with, as compared to the cost-sensitive and sampling approaches.

We propose a new imbalanced data classification method in this chapter, which uses a guided undersampling method (GUM) for sampling the training data and then applies support vector machines (SVM) on the sampled data in order to construct the classification model (i.e., the class boundary); the resulting method is referred to as GU–SVM hereinafter. GUM is an undersampling method for imbalanced-data classification, which sequentially applies two instance-selecting techniques to obtain a clean and well-represented subset of the original training instances: (1) first, GUM uses a modification of the ensemble-based outlier-filtering technique (Verbaeten and Van Assche, 2003) to remove both minority-class and majority-class outliers; (2) next, GUM applies a new cluster-based undersampling technique to obtain a sample of majority instances that is spread out over the majority class region. This new undersampling method, which we call *normalized-cut sampling*, uses Shi and Malik (2000)'s normalized-cut clustering to group the majority instances into a pre-specified number of approximately balanced clusters, and then forms a well-represented majority-class subset which comprises the medoid of each cluster.

In terms of the three approaches mentioned above, GU–SVM falls into the broad category of sampling approach, but it is a novel sampling method outperforming alternatives. In particular, our proposed method makes the following contributions: (1) Our ensemble outlier-filtering technique in GUM is specifically designed to work on imbalanced training data, and thus it achieves good outlier detection performances on *both* the majority and the minority data. In particular, the minority-class outlier identification and removal strategy in GUM is a unique procedure since, to the best of our knowledge, no previous imbalanced-data classification method has attempted to remove minority instances due to their scarcity. Yet, as Section 3.1 argues, removing minority outliers (in addition to majority outliers) is instrumental to obtaining a good classification model. (2) The normalized-cut sampling in GUM aims to obtain a subset of the majority instances that represents the majority class region. To obtain such a subset, the normalized-cut sampling is designed so that the selected instances are spread out over the majority region by grouping the majority instances into approximately evenly divided majority-class clusters and selecting one instance from each cluster. To the best of our knowledge, no previous undersampling-based classification method has attempted to obtain such a "spread-out" subset.

The remainder of the chapter is organized as follows. Section 3.2 presents the literature review of imbalanced-data classification and the ensemble-based outlier-filtering technique. Section 3.3 gives the details of the two instance-selecting techniques that compose the guided undersampling method and describes the GU–SVM. Section 3.4 describes the datasets and computational experiments on which GU–SVM outperforms several state-of-the-art imbalanced-data classification methods. Section 3.5 analyzes the relatively low performance of GU–SVM in two of the datasets. Finally, Section 3.6 summarizes our research undertaking and shares our thoughts on a further research direction (based on the analysis in Section 3.5).

3.2 Literature Review

3.2.1 Imbalanced-data Classification

Conventional classification methods are designed to minimize the number of misclassified instances over the training data, and thus they tend to classify the vast majority of the test set instances to the majority class in the imbalanced-data classification problem. Previous classification methods that dealt with this imbalanced-data classification problem can be categorized into three approaches: the cost-sensitive approach, the sampling approach, and the synthetic data generation approach. The following subsections summarize the methods within each category.

Cost-sensitive Approach

The cost-sensitive approach assigns asymmetric costs to the misclassified majority and minority instances during the training process. Specifically, the misclassification cost of the minority class C_{minor}, referred to as the $false\ negative\ cost$, is set higher than the misclassification cost of the majority class C_{major}, referred to as the $false\ positive\ cost$. Even though this approach has two parameters C_{major} and C_{minor}, in practice only their ratio matters, and thus only this ratio parameter C_{minor}/C_{major}, referred to as the $cost\ ratio$, is added to the base classifier.

Domingos (1999), Maloof (2003), and Ling et al. (2004) used decision-tree classifiers with asymmetric misclassification costs, namely a cost ratio greater than 1, to improve classification performance in imbalanced-data classification problems. SVM algorithms are also well suited for using the cost-sensitive approach. For instance, Lin et al. (2002), Bach et al. (2006), and Masnadi-Shirazi and Vasconcelos (2010) introduced different cost parameters, C_{major} and C_{minor}, where each parameter is the coefficient of the slack variables for the majority and minority instance, respectively, in the SVM objective function. The main drawback of the cost-sensitive approach is that the classification performance is very sensitive to the cost ratio used (Byon et al., 2010; Pourhabib et al., 2015); therefore, determining the optimal cost ratio to use within the classifiers is very challenging.

Sampling Approach

The sampling approach balances the number of instances between the classes by either oversampling the minority instances or undersampling the majority instances, and uses only the sampled data during the training process. Oversampling techniques increase the number of minority instances by resampling the minority instances with replication

(Estabrooks et al., 2004; Byon et al., 2010). Since oversampling techniques simply append replicated instances to the original dataset, they do not present a solution to the fundamental issue, the lack of minority instances; moreover, the constructed classifier might over-fit to the replicated instances (Mease et al., 2007). Undersampling techniques decrease the number of majority instances by removing a pre-specified number of instances from the majority class. Mease et al. (2007) used random-undersampling of the majority instances without replacement, while Japkowicz (2000) used weighted no-replacement undersampling with more weight on the majority instances near the minority instances. Due to the loss of majority instances, undersampling techniques might not construct an accurate classification model (Kubat and Matwin, 1997; Estabrooks et al., 2004). Some researchers have used both oversampling and undersampling simultaneously (Weiss and Provost, 2001; Estabrooks et al., 2004; Byon et al., 2010). Figure 3 illustrates the oversampling and undersampling approaches.

To address the aforementioned drawback of undersampling methods, some studies undersample the majority instances using clustering techniques as a guide. For example, Yen and Lee (2009) aimed to undersample more majority instances in the regions where majority instances dominate minority instances. To achieve this, they divide all the training instances (ignoring their class label) into a pre-specified number of clusters, then randomly undersample the majority instances from each cluster considering the proportion of majority instances in the cluster. Specifically, they undersample the majority instances so that more majority instances will be sampled in the clusters with higher majority-instance pro-

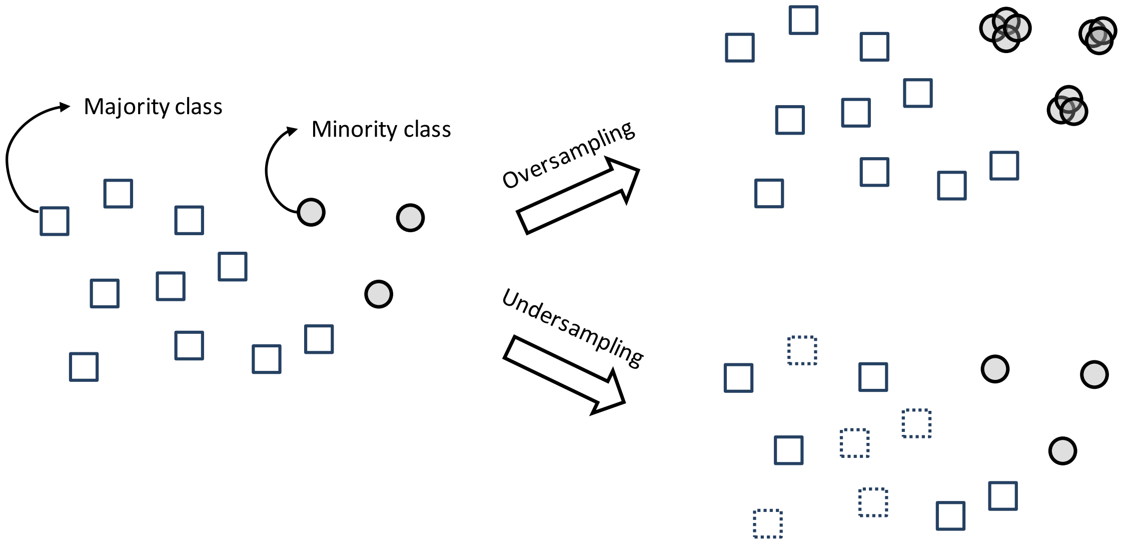


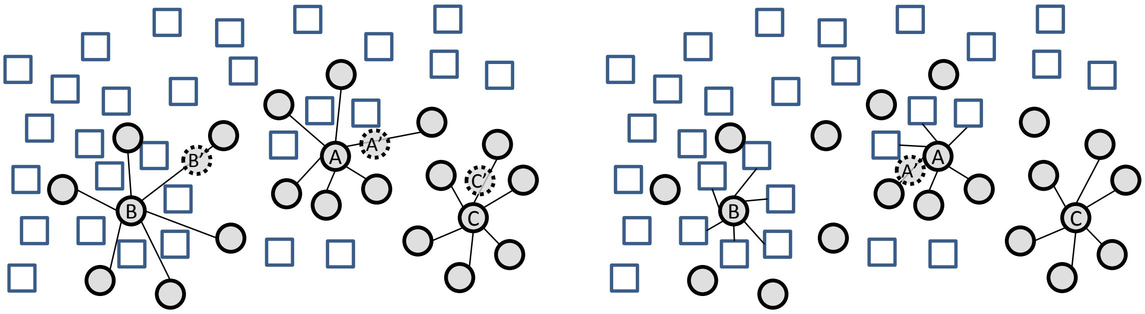
Figure 3: Oversampling approach increases the number of minority instances by resampling the minority instances with replication. Undersampling approach decreases the number of majority instances by removing a pre-specified number of instances from the majority class.

portion than in other clusters. Wang and Shi (2014) designed an undersampling method where the majority instances near the class boundary are more likely to be selected as sampled data. To achieve this, they select the majority instances near the class boundary as the seeds for the clusters in the initialization step of the majority-instance clustering. Then, they form the majority-class subset which comprises the the center of each cluster obtained after the majority-instance clustering process. Despite the improved performance, these cluster-based undersampling methods might not construct an accurate classification model as the selected instances obtained from aforementioned methods are not spread out over the majority class region, and thus might misrepresent the majority class.

Synthetic Data Generation Approach

As stated in this section, oversampling approaches sample several times each of the existing minority instances, and thus do not directly address the fundamental issue of lack of minority instances. To address this, synthetic data generation methods generate artificial minority instances (i.e., generate minority instances different to those in the original dataset).

The synthetic minority over-sampling technique (SMOTE), proposed by Chawla et al. (2002), generates a synthetic data instance for each minority instance by creating an instance located between the selected minority instance and one of its minority-class neighbors. Figure 4(a) illustrates the SMOTE method; for more details, we refer the reader to the original paper. Han et al. (2005)'s borderline SMOTE (BSMOTE) method finds the minority instances located near the class boundary (referred to as *borderline* instances). Then, for each borderline instance, it generates possible many synthetic data instances using the SMOTE data generating process. Figure 4(b) illustrates the BSMOTE method; for more details, we refer the reader to the original paper. Pourhabib et al. (2015)'s absent data generator (ADG) relies on two criteria for creating the synthetic minority instances: (i) new data should be close to the boundary between the majority-minority classes; and (ii) new data should not be too far off from the existing minority points. ADG builds the criteria into a kernel Fisher discriminant analysis formulation as two constraints. In execution, ADG alternates between a *generation* phase and a *revision* phase, in which the *generation* phase creates synthetic minority instances on the minority-class side of the



(a) SMOTE method

(b) Borderline SMOTE method

Figure 4: (a) Illustration of SMOTE approach. The synthetic instances, A', B', and C', are created based on the existing minority instances, A, B, and C. (b) Illustration of Borderline–SMOTE approach. Synthetic instance A' is generated since half of the neighbors of A are majority instances. There is no synthetic instance generated from B since all of B's neighbors are majority instances. There is no synthetic instance generated from C since all of C's neighbors are minority instances.

decision boundary, and the *revision* phase rebuilds the class boundary with the updated training data.

Previous synthetic data generation studies show that the performance of the synthetic-data-generation methods is better than that of sampling methods. However, using artificial data that are not among the original observations always makes researchers less comfortable, so the pursuit to find more effective methods in the categories of cost-sensitive approach or sampling approach has never ceased. The bottom line is that the jury is still out about which of the approaches will prevail, so that these approaches will almost surely co-exist for the foreseeable future.

3.2.2 Outlier-filtering Technique

Outlier-filtering methods remove unrepresentative training instances (i.e., outlier instances) of a given class so that the classification model's performance is not affected by the outliers. While the importance of outlier removal is well known, to the best of our knowledge, there are very few previous studies that used outlier filtering for imbalanced-data classification. Thus, most of this section reviews general purpose outlier-filtering techniques.

The undersampling procedure in some undersampling methods can also be regarded as an outlier-filtering method because its aim is to remove majority instances which are believed (by the designers of each respective method) to hinder the performance of the classification model. However, these undersampling procedures were not specifically designed to identify and remove outliers; and these undersampling procedures put much less effort to remove minority outliers than to remove majority outliers. For example, Kubat and Matwin (1997)'s *k*-nearest neighbors-based undersampling procedure aims to remove the majority instances that are either too close or too far from the minority class. Similarly, Japkowicz (2000) used weighted no-replacement undersampling with more weight on the majority instances near the minority instances.

Early outlier-filtering techniques used a single classifier to identify outlier instances on the training data. For example, in his seminal paper, Wilson (1972) identified and removed the training instances that were not correctly classified by a *k*-NN classifier (*k* was set to three in his experiment). Similarly, John (1995) used a decision-tree classifier (C4.5) to identify and remove outlier instances; specifically after building the classifier, the instances with different class-label from the class predicted on each tree leaf were deemed as outliers.

The drawback of the single-classifier outlier-filtering methods is that they implicitly assume that the classifier being used is the most appropriate for the data, which is obviously not always the case. Because ensemble outlier-filtering methods remove this assumption, they are more robust than single-classifier methods (Brodley and Friedl, 1999). As it has been proved that constructing a diverse set of base classifiers in an ensemble is theoretically important to improve classification performance (see, for example, Krogh and Vedelsby, 1995; Brown et al., 2005), ensemble-based outlier-filtering techniques use either different base classification algorithms (Brodley and Friedl, 1999) or different subsets of the training set to train multiple classifiers (Verbaeten and Van Assche, 2003). Given a number of single classifiers in ensemble N_{ensem}, Brodley and Friedl (1999) create an ensemble classifier using N_{ensem} single classifiers (each with a different classification algorithm), and deemed as outliers the training instances that were not classified to their original class by the ensemble. Verbaeten and Van Assche (2003) built an ensemble classifier by partitioning the data into N_{ensem} sets and train N_{ensem} decision-tree classifiers (each on a different subset). Then, training instances not classified by the ensemble classifier to their original class are deemed outliers.

As noted previously, these ensemble outlier-filtering techniques were not specifically designed for imbalanced datasets. Therefore, we argue that these ensemble-based outlier-filtering techniques will have poor outlier-detection performance on imbalanced datasets since: traditional classification methods will perform poorly to detect minority instances; so, the base classifiers in these ensembles will perform poorly to detect minority instances; thus, most of the minority instances will be incorrectly classified as majority instances by most of the base classifiers; and consequently, most of the minority instances will be deemed as outliers. Based on these reasons, replacing the base classifier used in these ensemble methods with imbalanced-data classification methods may improve the outlier detection performance. However, this is not possible in Verbaeten and Van Assche (2003)'s approach because when building the partition, there will be extremely low or none minority instances on each subset. In Section 3.3.1, we propose a manner to overcome this limitation and simultaneously obtaining a balanced partition (i.e., each subset is balanced).

3.3 Proposed Method

In this section, we propose an undersampling method for imbalanced-data classification, which aims to obtain a clean and well-represented subset of the original training instances. Our proposed undersampling method sequentially applies two instance-selecting techniques — ensemble outlier filtering and normalized-cut sampling. We label this undersampling method Guided Undersampling Method (GUM). Then, we propose an imbalanced-data classification method, which uses GUM for sampling the training data and then applies support vector machines (SVM) on the sampled data in order to construct the classification model. We label this imbalanced-data classification method Guided Undersampling – Support Vector Machines (GU–SVM). We first give the details of the two instance-selecting techniques that compose GUM in Sections 3.3.1 and 3.3.2, respectively, and then describe GU–SVM in Section 3.3.3.

3.3.1 Ensemble Outlier-filtering Technique for Imbalanced Data

As argued in Section 3.2.2, traditional ensemble outlier-filtering techniques have poor outlier-detection performance on imbalanced datasets because they were not designed for this challenging datasets. To address this shortcoming, we propose to modify the ensemble outlier-filtering technique so that the training sets of each single classifier in the ensemble are balanced.

The main process of our ensemble outlier-filtering technique is based on Verbaeten and Van Assche (2003)'s ensemble outlier-filtering method, which employs a pre-specified number of single classifiers N_{ensem}, each trained on a different subset of the training set. Verbaeten and Van Assche (2003)'s technique works as follows:

1. Partition the original training set into N_{ensem} equally-sized subsets. Note that even if the initial training set is balanced, this partitioning process may create unbalanced partitions.
2. Train N_{ensem} C4.5 decision-tree classifiers, each on a different partition of the training set.
3. Predict the class of every instance in the original training set using the majority voting scheme (classifying an instance to the class which receives at least $N_{ensem}/2$

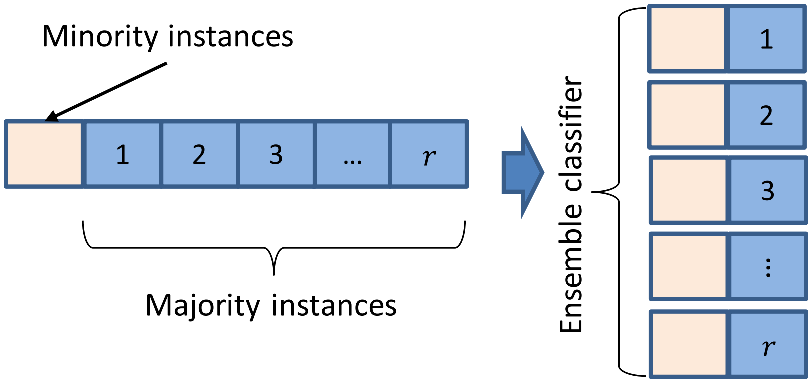


Figure 5: Dataset partitioning scheme of our proposed ensemble outlier-filtering with imbalance ratio r.

votes from the single classifiers).

4. Remove the *outlier instances* from the original training set. The outlier instances are those whose predicted class differs from their true class.

Our modified ensemble outlier filter for imbalanced data makes changes to Steps 1 and 2 of Verbaeten and Van Assche (2003)'s technique, so that the new procedure guarantees a balanced training set for each training partition. Specifically, given an original training data $D_{train} = D_{majority} \cup D_{minority}$ with imbalance ratio r (i.e., r is the number of majority instances divided by the number of minority instances), our ensemble outlier-filtering technique for imbalanced data works as follows (Steps 3 and 4 are unchanged and thus are not given here):

1. Partition $D_{majority}$ into r equally-sized subsets and build r distinct training subsets; each composed of $D_{minority}$ and one of the partitions of $D_{majority}$. Note that the each training subset is balanced (i.e., has an imbalance ratio of 1). Figure 5 illustrates the

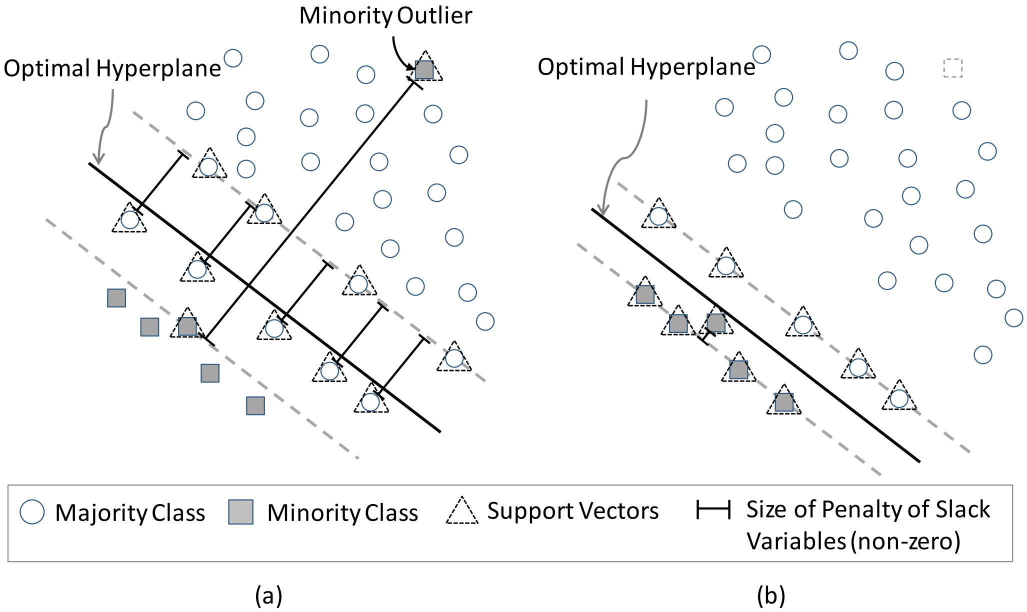


Figure 6: Removing minority outliers is particularly important to get an accurate classification boundary.

dataset partitioning scheme of our ensemble outlier-filtering technique.

2. Train r SVM classifiers, each classifier on a different training subset.

The objective of the ensemble outlier filtering process is to enable the construction of a classifier without a bias from outlier instances. Note that our outlier filtering process removes outliers not only from the majority class but also from the minority class. Removing minority instances seems counterintuitive due to their scarcity. However, we claim that removing minority outliers is very important to build an accurate classifier — perhaps even more important than removing majority outliers (whose adverse effects are dampened by the presence of all the other numerous non-outlier majority instances).

We support this claim via the example given in Figure 6. Figure 6(a) plots the original imbalanced data and the class boundary constructed by a cost-sensitive SVM (C_{minor} :

$C_{major} = 5 : 1$) with the original imbalanced data, while Figure 6(b) shows the data plot of the minority-outlier filtered data and the class boundary constructed by the cost-sensitive SVM with the minority-outlier filtered data. Figure 6(a) shows that, for this particular dataset, the minority outlier biases the decision boundary toward to the majority instances, causing some majority instances to be misclassified. It is important to note that, due to the higher penalty used on the misclassified minority instance, this single outlier is able to "pull" the boundary in spite of causing five majority instances to be misclassified. Figure 6(b) shows that, for this particular dataset, the decision boundary constructed by the cost-sensitive SVM with the minority-outlier filtered data correctly separates the majority instances and the minority instances. Therefore, removing minority outliers is important to obtain a good classification model.

3.3.2 Normalized-cut Sampling

As argued in Section 3.2.1, traditional undersampling methods might not construct an accurate classification model due to the loss of majority instances. Specifically, sampled majority instances, which are not spread out over the majority class region, may lead to an inaccurate decision boundary on the regions where no instances are selected. Thus, here we propose a new cluster-based undersampling method — normalized-cut sampling — which undersamples the majority instances so that selected instances are spread out over the majority class region, and thus are representatives of the majority class as a whole. For this purpose, the proposed normalized-cut sampling method first runs recursively Shi and Malik (2000)'s normalized-cut clustering to group the majority instances into a pre-specified number of approximately balanced clusters, and then forms the majority-class sample by including the medoid (the instance with the smallest average distance to all the instances in the given cluster) of each cluster. Below, we describe normalized-cut clustering and then describe how it is used in our normalized-cut sampling method.

Given a set of n-dimensional points (instances in our case), Shi and Malik (2000)'s normalized-cut clustering aims to bipartition the data instances so that 1) the instances within each cluster are as similar as possible while instances on distinct clusters are as different as possible, and 2) both subsets contain approximately the same number of instances. Normalized-cut clustering works as follows. The data instances are represented with an undirected complete graph $G = (V, E)$, where V is the set of data instances and E contains an edge between each pair of data instances. The weight on the edge between instances i and j, w_{ij}, is a measure of the similarity between i and j. Shi and Malik (2000) defined the w_{ij} as:

$$w_{ij} = e^{-\|x_i - x_j\|^2}, \tag{3.1}$$

where x_i and x_j are n-dimensional data of instance i and j, respectively. With this representation, the clustering problem is formulated as a graph partitioning problem which partitions V into the two subsets S and $\bar{S}$ that minimize the following function:

$$Ncut(S, \bar{S}) = \frac{cut(S, \bar{S})}{assoc(S, V)} + \frac{cut(S, \bar{S})}{assoc(\bar{S}, V)}, \tag{3.2}$$

where $cut(S, \bar{S})$ is the total sum of the edge-weights (similarities) between the two partitions S and $\bar{S}$ (i.e., $cut(S, \bar{S}) = \sum_{i \in S, j \in \bar{S}} w_{ij}$), $assoc(S, V)$ is the total sum of the edge-weights from nodes in S to all nodes in the graph G (i.e., $assoc(S, V) = \sum_{i \in S, j \in V} w_{ij}$), and

$assoc(\bar{S},V)$ is is the total sum of the edge-weights from nodes in $\bar{S}$ to all nodes in the graph G. Note that any partition where one of the subsets, say S, only contains a small set of isolated nodes in the graph will have a large $Ncut$ value since $assoc(S,V)$ will be very small. Thus, normalized-cut clustering method creates a bipartition with roughly the same number of instances in both clusters, S and $\bar{S}$.

Since optimizing Equation (3.2) is NP-hard, Shi and Malik (2000) proposed an heuristic whose rough outline is as follows (for more details, we refer the reader to the original paper). First, they reformulated the normalized cut problem as a minimization problem of the Rayleigh quotient with integrality constraints on the variables. Then, they showed that the continuous relaxation of the problem can be solved by finding an eigenvector of the matrix associated with the similarities between the nodes.

Next, we give the details of our normalized-cut sampling method. Given a training data $D_{train} = D_{majority} \cup D_{minority}$, our normalized-cut sampling method undersamples the majority instances so that the number of sampled majority instances is the same as the number of minority instances. We denote the number of minority instances by $K = \mid D_{minority} \mid$, and our algorithm is described as follows:

1. Construct $G_1 = (V,E)$ by assigning all $majority$ instances to the node set V and all node pairs in V to the edge set E. Use Equation (3.1) to assign the edge-weight (similarity) between nodes i and j, w_{ij}, just like in Shi and Malik (2000).
2. For $k = 1, \cdots, K-1$ do

   2.a. Bipartition the graph G_k using Shi and Malik (2000)'s normalized-cut clustering.

2.b. Construct a new undirected complete graph G_{k+1} which only includes the instances in the cluster with the largest number of instances (among the $k+1$ clusters that have been found so far).

3. Form a subset which comprises the medoid (the instance whose average distance to all the instances in the same cluster is minimal) of each cluster and return this subset as the sampled majority instances.

After Step 2, we have K approximately balanced clusters because at each iteration we split one of the given clusters into two clusters but we keep split the larger remaining cluster until they are roughly the size of the minority set. Then Step 3 simply takes one data point per cluster and forms the sample majority subset.

Note that our normalized-cut sampling method has a different objective from that of other cluster-based undersampling methods. Specifically, our normalized-cut sampling uses clustering in order to select a spread-out sample of majority instances to form a well-represented subset of the majority class (well represented in the sense that it covers all of the majority class region). By contrast, the cluster-based undersampling methods introduced in Section 3.2.1 use clustering to sample/select more majority instances in the regions where majority instances dominate minority instances (Yen and Lee, 2009) or sample so that the majority instances near the class boundary are more likely to be selected (Wang and Shi, 2014).

The main idea/motivation of our normalized-cut sampling method is similar to that of the stratified sampling method used in the context of statistical survey design. Stratified sampling works as follows (Thompson, 2012): (1) Partition the population into subgroups (strata) before sampling so that the instances within a subgroup share similar characteristics; (2) randomly sample the instances from each subgroup so that the proportion of each subgroup in the sampled data is equal to the proportion of each subgroup of the whole population. Stratified sampling methods are widely used in statistical survey design when the researcher aims to sample a representative subset of the whole population since stratified sampling ensures that each subgroup is equally represented between in the population and in the sample.

Despite the similarity in idea/motivation, the specific grouping/clustering techniques used in stratified sampling and normalized-cut sampling are different. Specifically, one hand, stratified sampling usually divides the original instances by thresholds of a limited number of known variables (e.g., gender or age); and these thresholds are set "by hand" by the researcher and this creating subgroups have some meaning and importance to human observers. While, on the other hand, normalized-cut sampling determines the clusters via solving an optimization problem whose objective is that the distances between the instances within a subgroup are as small as possible and the distances between the instances on distinct subgroups are as large as possible. Thus, the sampled instances using normalized-cut sampling are more representative of the population than those obtained using stratified sampling (at the loss of "meaningfulness" to a human observer).

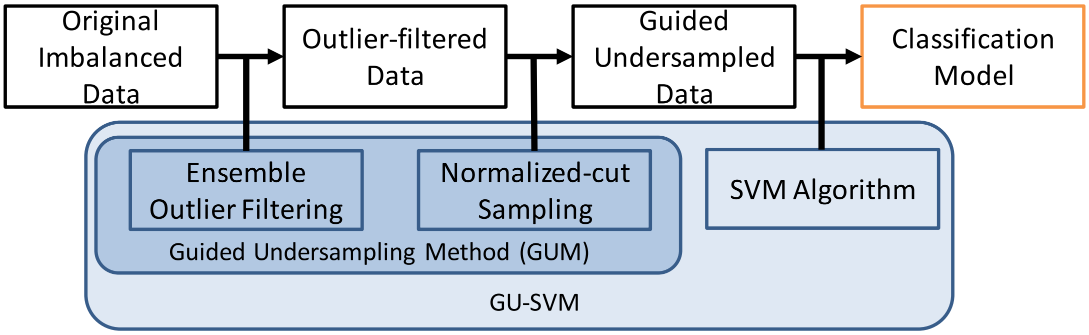


Figure 7: Flowchart of the main procedures within GU–SVM

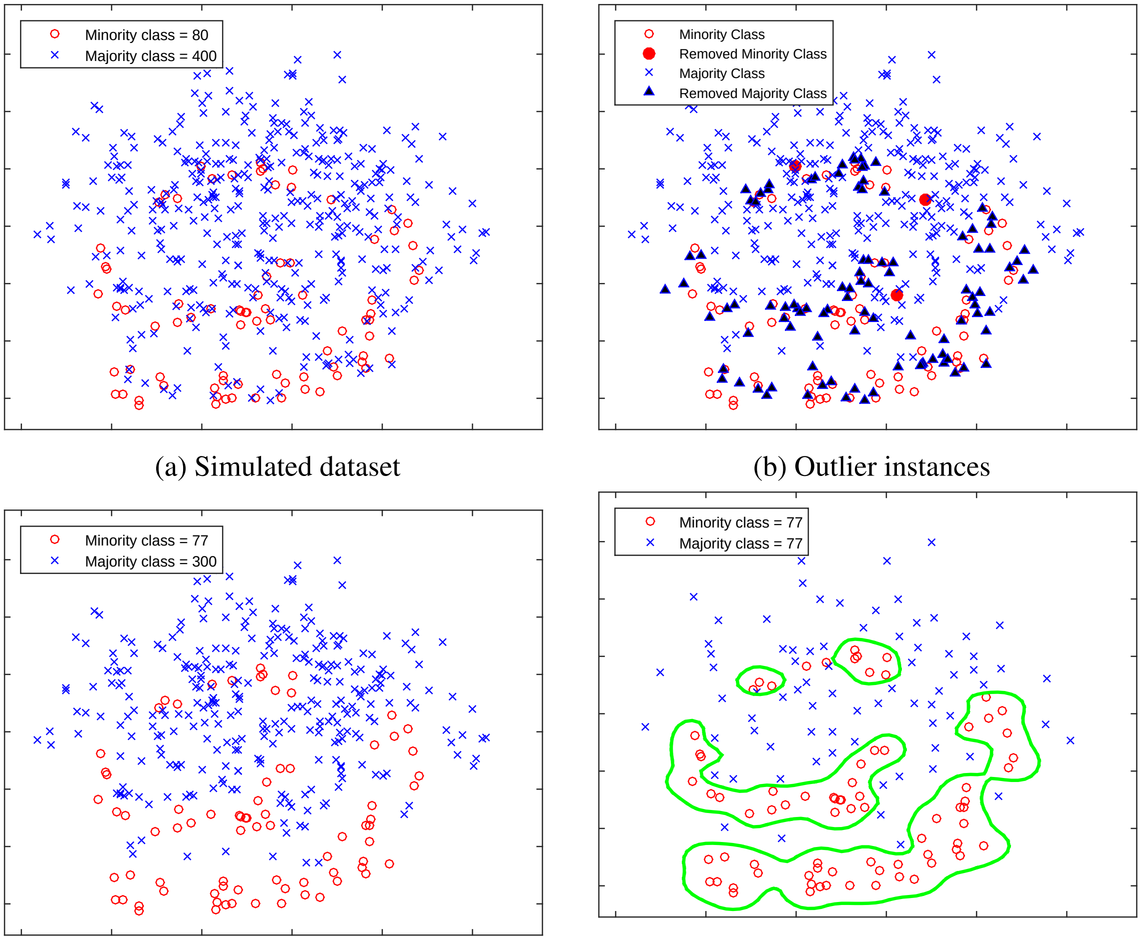
3.3.3 Process of the Proposed Classification Method

Sections 3.3.1 and 3.3.2 give the details of the two instance-selecting techniques — ensemble outlier filtering and normalized-cut sampling — that compose our proposed guided undersampling method (GUM). The following steps describe how GU–SVM uses GUM to construct the classification model for imbalanced data.

1. Given a training data $D_{train} = D_{majority} \cup D_{minority}$ with an imbalance ratio $r = \frac{|D_{majority}|}{|D_{minority}|}$, use our ensemble outlier-filtering technique described in Section 3.3.1 to obtain the outlier-filtered training data, $D'_{train} = D'_{majority} \cup D'_{minority}$.
2. Given the outlier-filtered data instances D'_{train}, apply to it the normalized-cut sampling method described in Section 3.3.2. The normalized-cut sampling method returns spread-out samples of $D'_{majority}$, $D''_{majority}$, such that $\mid D''_{majority} \mid = \mid D'_{minority} \mid$.
3. Apply the SVM algorithm on the data obtained from Step 2, $D''_{train} = D''_{majority} \cup D'_{minority}$ in order to construct the classification model.

The general idea behind GU–SVM is as follows. GU–SVM uses GUM to obtain a clean and well-represented subset of the training data and applies SVM on the sampled data in order to construct the classification model (i.e., the class boundary). More specifically, GUM first uses our ensemble outlier-filtering technique to remove the outlier instances from the training data (if not removed, those instances introduce bias in the classification model). Note that our ensemble outlier filtering is specially designed so that it has a good minority-outlier detection performance for imbalanced datasets. Next, GUM uses our normalized-cut sampling on the (outlier-filtered) majority instances to select a subset of the majority instances such that the selected instances are spread out over the majority class region. Last, GU–SVM applies SVM on the guided sampled data in order to construct the classification model.

Figure 7 gives a flowchart of the main procedures within GU–SVM. Figure 8 illustrates GU–SVM on a two-dimensional simulated imbalanced dataset. Figure 8(a) illustrates the simulated dataset. Both majority and minority classes have a spiral-like distribution; these two spiral distributions interlace in a jing-jang fashion. Figure 8(b) shows the outlier instances identified by our ensemble outlier-filtering technique. Figure 8(c) shows the outlier-filtered data using our ensemble outlier filtering. One hundred majority instances and three minority instances are identified as outlier instances and removed from the training data. Figure 8(d) shows the 77 majority instances that were selected by normalized-cut sampling and the class boundary constructed by GU-SVM. Note that GU-SVM selected a well-represented subset of the outlier-filtered training instances and constructed an accurate classification model.



(a) Simulated dataset

(b) Outlier instances

(c) Outlier-filtered dataset

(d) Normalized-cut sampled dataset and final classification boundary

Figure 8: Illustration of GU–SVM on a simulated imbalanced dataset.

3.4 Experiments

This section analyzes the performance of GU–SVM on several datasets and compares it with five state-of-the-art imbalanced-data classification algorithms. This section is organized as follows. Section 3.4.1 briefly reviews the five state-of-the-art classification methods against which GU–SVM is compared. Section 3.4.2 describes the datasets used in the performance evaluation. Section 3.4.3 explains the performance measures that are employed in this study. Finally, Section 3.4.4 gives the results of our computational experiments.

3.4.1 State-of-the-art Methods for Imbalanced-data Classification

This section briefly describes the five classification methods (raw-data SVM, cost-sensitive SVM, random-undersampling SVM, SMOTE SVM, and BSMOTE SVM) against which GU–SVM is compared. As mentioned in Section 3.2.1, with the exception of raw-data SVM, these state-of-the-art classification methods are widely used in the imbalanced-data classification literature. We include raw-data SVM in the comparison as a baseline classification method upon which every other method should improve. The other four classification methods were chosen so that each of the imbalanced-data classification literature was adequately represented. The selected four classification methods are the ones that have been commonly used in past studies (the cost-sensitive SVM – cost-sensitive approach, the random-undersampling SVM – sampling approach, and the SMOTE SVM and BSMOTE SVM – synthetic data generation approach).

Throughout this section, the imbalance ratio (defined as the number of majority instances divided by the number of minority instances) of the given original dataset is denoted as r. The raw-data SVM is the traditional SVM algorithm (Cortes and Vapnik, 1995) applied directly on the raw data (the original imbalanced data). The cost-sensitive SVM uses a traditional SVM algorithm with different penalty coefficients C_{major} and C_{minor} in the SVM objective function, where C_{major} and C_{minor} are the cost parameters of the majority and minority slack variables, respectively. Specifically, we choose the cost parameters so that the cost ratio between the majority and minority classes, C_{minor}/C_{major}, is equal to r. The random-undersampling SVM first undersamples the majority instances and then applies a traditional SVM to the sampled data. The undersampling is applied by randomly sampling without replacement the majority instances. Moreover, the final sample size is chosen so that the sampled dataset has an imbalance ratio of 1 (i.e., dataset is perfectly balanced). The SMOTE SVM (Chawla et al., 2002) generates new synthetic data instances by interpolating between each of the minority instances and one of its k nearest minority instances (k was set to five as in Chawla et al. 2002); then, the original training data and the synthesized data are used for training a traditional SVM classifier. In the BSMOTE method (Han et al., 2005), a borderline instance is defined as a minority instance that more than half of its k nearest neighbors belong to the majority class (k was set to five as in Han et al. 2005). The BSMOTE SVM method generates synthetic minority instances by interpolating between each borderline minority instance and one of its nearest minority instances; then, the original training data and the synthesized data are used for training a traditional SVM classifier.

Notice that one of the main components of each of the aforementioned methods is a traditional SVM. Throughout this experiment, for all of these methods, we used exactly the same traditional SVM. In particular, we used the radial basis function $K(x_i,x_j) = \exp(-\gamma \parallel x_i - x_j \parallel^2)$ as kernel function. We determined γ and the cost ratio between the margin penalty and the total sum of the slack variables in the SVM objective function through cross-validation of raw-data SVM (and that same γ and cost ratio were used for every method on the respective dataset). All of the aforementioned classification methods (as well as our proposed method) were implemented in MATLAB with the LIBSVM tool package (Chang and Lin, 2011).

3.4.2 Datasets

The base datasets used in this study have the following three characteristics: (1) they are open to the public; (2) they are widely used in previous imbalanced-data classification studies; and (3) the area-under-the-curve (AUC, defined in Section 3.4.3), obtained from the raw-data SVM on the dataset with an imbalance ratio of 5 (described in the eusuing paragraph), was less than 0.95. This third characteristic is important because it is very difficult to accurately discriminate/rank the performances of the methods on datasets that are relatively easy even for the naive raw-data SVM. Table 7 lists the 11 base datasets used in this study along with a brief description of each. Among the base datasets described in Table 7, multi-class datasets were modified by labeling the class with the fewest number of instances as the minority class and the rest of the instances were combined in the majority class. Finally, note that even though we considered a wide variety of sources, all

Table 7: Description of 11 open datasets

Dataset	Dim	# of Inst.	# of Maj.	# of Min.	Brief Description
Australian	14	690	383	307	Australian credit card approval
CMC	24	1473	1140	333	Contraceptive method choice
Ecoli	9	336	301	35	Localization site of protein
German	24	1000	700	300	German credit data
Glass	9	214	197	17	Glass type identification
Haberman	3	306	225	81	Breast cancer survival data
Heart	13	270	150	120	Heart disease detection
Liver	6	345	200	145	Liver disorder detection
Pima	8	768	500	268	Pima indians diabetes data
Vehicle	18	846	634	212	Vehicle type identification
Yeast	9	1484	1055	429	Localization site of protein

Note: Dim means Number of Features. # of Inst. means Number of Instances. # of Maj. means Number of Majority-class Instances. # of Min. means Number of Minority-class Instances.

of the selected datasets happen to be available in the UCI Machine Learning Repository (Lichman, 2013).

From each of the 11 base datasets in Table 7, we built four imbalanced datasets, each with a different imbalance ratio (namely, 5, 10, 20, and 30). Specifically, given an imbalance ratio r, $r \in \{5, 10, 20, 30\}$, if the imbalance ratio of the base dataset is greater than r, we selected all the minority instances in the base dataset and r times as many majority instances using sampling without replacement. Otherwise, if the imbalance ratio of base dataset is less than r, we selected all majority instances in the base datasets and undersampled without replacement the minority instances so as to obtain the desired imbalance ratio of r. Moreover, for each of 44 datasets (11 base datasets with four imbalance ratios), we created 10 random datasets for repetition in order to minimize any effect from the random sampling used to generate the datasets (throughout this chapter, these random datasets for repetition are denoted as repetitions). Therefore, we generated 440 total datasets (10 repetitions for each of 44 datasets). Consequently, in Section 3.4.4, whenever we report the AUC of a given method on a particular dataset such AUC is the AUC average of 10 repetitions, each obtained via a five cross-validation. Finally, to increase the accuracy of the AUC calculation, following the lead of Pourhabib et al. (2015), when performing the cross-validations, the data instances that were not selected from the original base dataset during the random sampling process were appended to the test set partition during the five-cross-validations. This appending procedure is particularly important when measuring the AUC of the datasets with large imbalance ratios as the AUC otherwise has huge variances due to the extreme scarcity of the minority instances.

3.4.3 Performance Measure

To measure and compare the classification performance, we used the area-under-curve (AUC) measure of the receiver operating characteristic (ROC) plot (Bradley, 1997). An ROC curve is a graphical plot that illustrates the performance of two-class classification when the classification algorithm (here, the SVM) provides a continuous output, and thus the class prediction can be varied by changing the classification threshold. The curve is created by plotting each $(False\, alarm, Detection\, power)$ point for the test data at various threshold settings, where the false alarm refers to the false positive rate (the ratio of the misclassified majority instances compared to all majority instances) and the detection power refers to the true positive rate (the ratio of the correctly classified minority instances compared to all minority instances). After plotting an ROC curve, an AUC is computed by measuring the area under the ROC curve. A higher AUC means a better classification performance in general since a higher AUC indicates that the corresponding classification model has the higher detection power points in average against the given false alarm points. The AUC of each classification method for each dataset is computed by the average value of AUCs over five cross-validations and ten repetitions (each over a randomly sampled dataset as described in Section 3.4.2).

After obtaining the AUC of each classification algorithm for each dataset, one may be tempted to consider taking the average value of the AUCs over all datasets to compare multiple classification methods. However, this classification methods comparison with average AUCs would be incorrect as the AUC value differences between the classification methods vary largely among the different datasets. For instance, the largest AUC difference between the classification methods of Ecoli data [imbalance ratio = 30] is 0.305 while the largest AUC difference between the classification methods of Australian data [imbalance ratio = 5] is 0.012; see, for details, Table 8. Thus, ranking the classification methods by using the average value of the AUCs over all datasets could be dominated by the datasets that have the high AUC value differences between the classification methods. Therefore, in order to compare the performance of classification algorithms without bias from the AUC value differences between the datasets, this study uses the Friedman test, as revised by Demšar (2006) for comparing of multiple classification methods.

The Freidman test is a non-parametric statistical test for detecting the differences between the rank data. The null-hypothesis is that all ranks of classification methods are equivalent, and if the null-hypothesis is rejected by the Freidman test, the post-hoc test is conducted to determine which classification methods have consistently higher or lower ranks compared to others.

Mathematically, given N algorithms and M datasets, the Freidman statistic is calculated as follows (Friedman, 1937):

$$\chi_F^2 = \frac{12M}{N(N+1)}[\sum_j R_j^2 - \frac{N(N+1)^2}{4}], \tag{3.3}$$

where R_j is the average rank value of algorithm j over all datasets. If M and N are large (as a rule of thumb, $M > 10$ and $N > 5$, by Demšar 2006), the probability of χ_F^2 can be approximated by the probability of a chi-squared distribution with $N-1$ degrees of freedom.

If the null-hypothesis is rejected at significance level α, the post-hoc test is conducted in order to determine which algorithms are significantly different from each other based on the average rank differences of the algorithms. In the post-hoc test, the performance of two algorithms is significantly different if the average rank difference gap between the two algorithms is greater than or equal to the critical difference:

$$CD = q_\alpha \sqrt{\frac{N(N+1)}{6M}}, \tag{3.4}$$

where we set the critical value q_α according to the post-hoc test designed by Nemenyi (Nemenyi, 1963).

3.4.4 Results

This section provides the computational results obtained by applying GU–SVM on datasets (described in Section 3.4.2), and compares its performance with that of the state-of-the-art classification methods (described in Section 3.4.1). This section also examines the classification performance of each of the two instance-selecting techniques — ensemble outlier-filtering technique and normalized-cut sampling — separately in order to investigate the effect that each instance-selecting technique has on the performance of GU–SVM.

Performance Analysis of GU–SVM

Table 8 gives the AUC obtained by each classification method on each dataset. As described in Section 3.4.2, each AUC reported in Table 8 is the AUC average of 10 repetitions, each obtained via a five cross-validation. The classification methods shown in Table 8 include the five state-of-the-art classification methods introduced in Section 3.4.1 and GU–SVM. In Table 8, RawSVM, CSSVM, and RandSVM represent raw-data SVM, cost-sensitive SVM, and random-undersampling SVM, respectively. The bold-faced values are the highest AUC value for each dataset. Note that, GU–SVM provides the highest AUC values on 30 datasets out of total 44 datasets. Moreover, on the datasets not built from the liver and glass base datasets, GU–SVM provides either the highest AUC value or an AUC value (extremely) close to the highest AUC value. Thus, it is reasonable to conclude that GU–SVM is the method of choice for 9 datasets out of the total 11 datasets. In Section 3.5, we will analyze with more depth why GU-SVM performed poorly on the liver and glass datasets. 3.5.

As argued in Section 3.4.3, it is not insightful to compare the techniques using the average value of AUCs over all datasets. Instead, we compared them in terms of their

Table 8: Average Area Under Curve (AUC). GU–SVM outperforms all other methods in terms of AUC in most datasets.

Dataset							
Base DataSet	Imbalance Ratio	RawSVM	CSSVM	RandSVM	SMOTESVM	BSMOTESVM	GU–SVM
Australian	5	0.882	0.887	0.892	0.889	0.888	**0.894**
	10	0.843	0.858	0.879	0.866	0.867	**0.890**
	20	0.780	0.820	0.851	0.819	0.825	**0.871**
	30	0.752	0.793	0.832	0.799	0.799	**0.879**
CMC	5	0.601	**0.709**	0.699	0.675	0.679	0.705
	10	0.552	0.686	0.677	0.614	0.620	**0.687**
	20	0.529	0.668	0.663	0.568	0.563	**0.677**
	30	0.499	0.639	0.644	0.545	0.529	**0.657**
Ecoli	5	0.932	0.948	0.946	**0.949**	0.941	0.945
	10	0.898	**0.944**	0.941	0.920	0.924	0.943
	20	0.716	0.940	0.940	0.868	0.870	**0.943**
	30	0.638	0.937	0.940	0.770	0.795	**0.943**
German	5	0.680	0.679	0.675	0.687	0.689	**0.696**
	10	0.659	0.657	0.642	0.664	**0.669**	0.669
	20	0.632	0.632	0.610	0.640	**0.642**	0.633
	30	0.631	0.630	0.604	0.636	0.634	**0.643**
Glass	5	0.834	**0.843**	0.742	0.827	0.822	0.735
	10	0.793	**0.830**	0.726	0.818	0.800	0.718
	20	0.671	**0.768**	0.630	0.713	0.697	0.641
	30	0.685	**0.759**	0.613	0.718	0.689	0.634
Haberman	5	0.566	0.638	0.656	0.648	0.649	**0.666**
	10	0.527	0.578	0.613	0.568	0.581	**0.639**
	20	0.504	0.562	0.604	0.534	0.534	**0.615**
	30	0.494	0.571	0.565	0.521	0.497	**0.591**
Heart	5	0.825	0.824	0.806	0.828	0.828	**0.829**
	10	0.819	0.818	0.787	0.826	0.827	**0.829**
	20	0.807	0.807	0.768	0.809	**0.821**	0.818
	30	0.819	0.819	0.777	0.822	0.823	**0.844**
Liver	5	0.701	0.713	0.689	0.723	**0.724**	0.679
	10	0.689	0.693	0.648	**0.712**	0.710	0.644
	20	0.663	0.670	0.598	**0.678**	0.674	0.595
	30	0.599	**0.619**	0.560	0.611	0.600	0.541
Pima	5	0.744	0.746	0.766	0.758	0.757	**0.783**
	10	0.717	0.712	0.745	0.725	0.725	**0.765**
	20	0.695	0.689	0.714	0.705	0.695	**0.745**
	30	0.678	0.677	0.714	0.681	0.679	**0.741**
Vehicle	5	0.775	0.782	0.786	0.781	0.781	**0.800**
	10	0.747	0.749	0.758	0.754	0.751	**0.792**
	20	0.708	0.708	0.717	0.717	0.713	**0.760**
	30	0.701	0.700	0.696	0.709	0.708	**0.740**
Yeast	5	0.703	0.731	0.743	0.738	0.735	**0.749**
	10	0.652	0.693	0.724	0.680	0.672	**0.733**
	20	0.615	0.651	0.687	0.624	0.616	**0.718**
	30	0.593	0.628	0.669	0.599	0.587	**0.703**

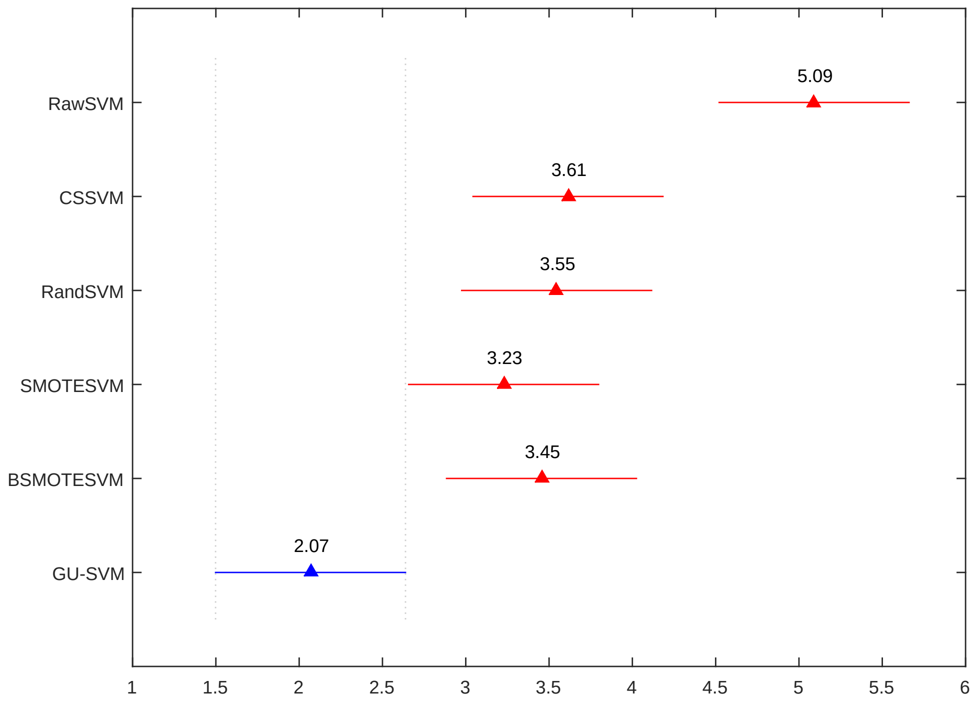


Figure 9: Post-hoc analysis on the ranks of state-of-the-art classification methods and GU–SVM. Average AUC rank of GU–SVM (2.07) is significantly lower than that of all other methods.

ranks. Specifically, for each dataset (i.e., each row in Table 8), we ranked the classification methods and then the methods are compared in terms of the average of their ranks, hereafter called *average AUC rank*, they obtained. In this sense, GU–SVM has the lowest average AUC rank (2.07) among all of the classification methods (the average AUC rank of each method is found in Figure 9). In order to test whether the lowest rank of GU–SVM is statistically significant, we performed the Friedman test, followed by a post-hoc analysis. Specifically, under the null hypothesis that all classification methods have the same average AUC rank, the Friedman statistic χ_F^2 was 58.7, and thus the null hypothesis is rejected with a p-value 1.07×10^{-11}. Moreover, the post-hoc Nemenyi analysis, as illustrated graphically in Figure 9, asserts that the average AUC rank of GU–SVM is significantly lower (at confidence level 0.05) than that of all other classification methods. Specifically, in Figure 9, the average AUC rank of each classification method is denoted as a triangle and the critical difference range obtained by the Nemenyi post-hoc analysis is denoted as lines laid on the triangles. In this Nemenyi post-hoc analysis, two methods are determined to be significantly different when their critical difference ranges do not overlap.

The results from this section show that GU–SVM has the improved classification performance compared to the state-of-the-art classification methods. The following subsection investigates whether the outstanding performance of GU–SVM can be attributed only (or mostly) to one of the two instance-selecting techniques — ensemble outlier-filtering technique and normalized-cut sampling — that compose GU–SVM.

Performance Analysis of GU–SVM's components

To independently examine the classification performances of the ensemble outlier-filtering and the normalized-cut sampling, we built two classification methods that employ only one of the two instance-selecting techniques respectively and compare them with the state-of-the-art classification methods. We refer to these methods as ensemble outlier-filtering SVM (EOF–SVM) and normalized-cut sampling SVM (Ncut–SVM).

Ensemble Outlier-filtering Technique (EOF–SVM) The EOF–SVM consists on applying the traditional SVM algorithm to the ensemble outlier-filtered data. In other words, EOF–SVM is the GU–SVM method, but without applying the normalized-cut sampling to

Table 9: AUC performance of ensemble outlier-filtering SVM (EOF–SVM)

Imbalance Ratio	Australian	CMC	Ecoli	German	Glass	Haberman	Heart	Liver	Pima	Vehicle	Yeast
5	0.896	0.686	0.947	0.697	0.753	0.680	0.833	0.696	0.787	0.800	0.747
10	0.891	0.670	0.940	0.668	0.716	0.659	0.831	0.664	0.772	0.793	0.730
20	0.875	0.657	0.934	0.630	0.658	0.630	0.822	0.634	0.751	0.759	0.717
30	0.873	0.642	0.931	0.642	0.640	0.608	0.837	0.568	0.741	0.748	0.699

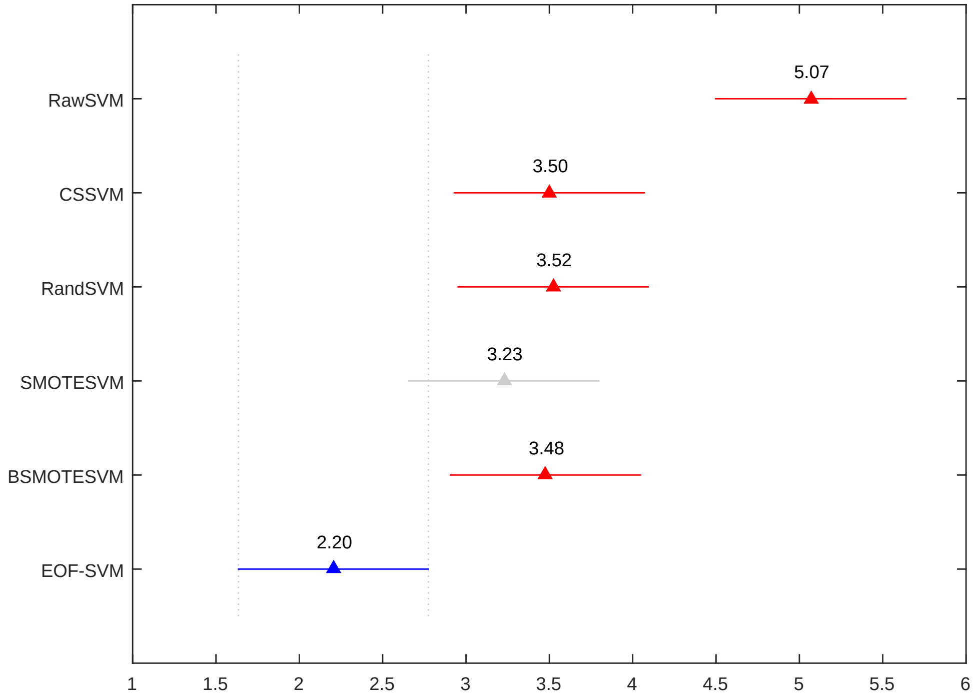


Figure 10: Post-hoc analysis on the ranks of state-of-the-art classification methods and EOF–SVM. Average AUC rank of EOF–SVM (2.20) is lower than that of all other methods; however, its rank is not significantly different from the rank of the SMOTE SVM (3.23) by the Nemenyi post-hoc analysis.

the ensemble outlier-filtered training data.

Table 9 shows the AUC obtained by EOF–SVM on each dataset. As described in Section 3.4.2, each AUC is the AUC average of 10 repetitions, each obtained via a five cross-validation. To compare the performance of EOF–SVM with the state-of-the-art classification methods, we ranked the AUCs of EOF–SVM in Table 9 and state-of-the-art classification methods in Table 8. In this sense, EOF–SVM has the lowest average AUC rank (2.20) among all of the state-of-the-art classification methods (the average AUC rank of each method is found in Figure 10). Then, we tested the significance of this result. Specifically, under the null hypothesis that all classification methods have the same average AUC rank, the Friedman statistic χ_F^2 was 53.0, and thus the null hypothesis is rejected with a p-value 1.59×10^{-10}. Moreover, the post-hoc Nemenyi analysis, as illustrated graphically in Figure 10, asserts that the average AUC rank of the EOF–SVM (2.20) is significantly lower (at confidence level 0.05) than the ranks of four classification methods — the raw-data SVM, the cost-sensitive SVM, the random-undersampling SVM, and the BSMOTE SVM. However, the post-hoc Nemenyi analysis asserts that the rank of the EOF–SVM is not significantly different from the rank of SMOTE SVM (3.23). This result shows that EOF–SVM has a competitive classification performance comparable to several state-of-the-art imbalanced data classification methods, though the rank of the EOF–SVM is not significantly lower than that of the SMOTE SVM.

Normalized-cut Sampling (Ncut–SVM) The Ncut–SVM consists on applying the traditional SVM algorithm to the normalized-cut sampled data. In other words, Ncut–SVM is the GU–SVM method, but without applying the ensemble outlier filtering to the training data.

Table 10 shows the AUC obtained by Ncut–SVM on each dataset. As described in Section 3.4.2, each AUC is the AUC average of 10 repetitions, each obtained via a five cross-validation. To compare the performance of Ncut–SVM with the state-of-the-art classification methods, we ranked the AUCs of Ncut–SVM in Table 10 and state-of-the-art

Table 10: AUC performance of normalized-cut sampling SVM (Ncut–SVM)

Imbalance Ratio	Australian	CMC	Ecoli	German	Glass	Haberman	Heart	Liver	Pima	Vehicle	Yeast
5	0.900	0.701	0.948	0.685	0.786	0.669	0.824	0.701	0.777	0.795	0.756
10	0.892	0.682	0.946	0.659	0.758	0.649	0.817	0.660	0.751	0.778	0.735
20	0.876	0.664	0.946	0.631	0.660	0.630	0.807	0.623	0.747	0.745	0.718
30	0.882	0.633	0.947	0.630	0.640	0.589	0.827	0.556	0.740	0.717	0.703

classification methods in Table 8. In this sense, Ncut–SVM has the lowest average AUC rank (2.41) among all of the state-of-the-art classification methods (the average AUC rank of each method is found in Figure 11). Then, we tested the significance of this result. Specifically, under the null hypothesis that all classification methods have the same average AUC rank, the Friedman statistic χ_F^2 was 45.1, and thus the null hypothesis is rejected with a p-value 6.48×10^{-9}. Moreover, the post-hoc Nemenyi analysis, as illustrated graphically in Figure 11, asserts that the rank of the Ncut–SVM (2.41) is significantly lower (at confidence level 0.05) than the ranks of two classification methods — the raw-data SVM and the cost-sensitive SVM. However, the post-hoc Nemenyi analysis asserts that the rank of the Ncut–SVM is not significantly different from the rank of cost-sensitive SVM (3.50), SMOTE SVM (3.09), and BSMOTE SVM (3.36). This result shows that Ncut–SVM has a competitive classification performance comparable to several state-of-the-art imbalanced data classification methods, though the rank of the Ncut–SVM is not significantly lower than that of the cost-sensitive SVM, the SMOTE SVM and the BSMOTE SVM.

Results From Performance Analysis of the Two Instance-selecting Techniques The computational results obtained from comparing each of the two instance-selecting techniques against the state-of-the-art classification methods respectively demonstrate

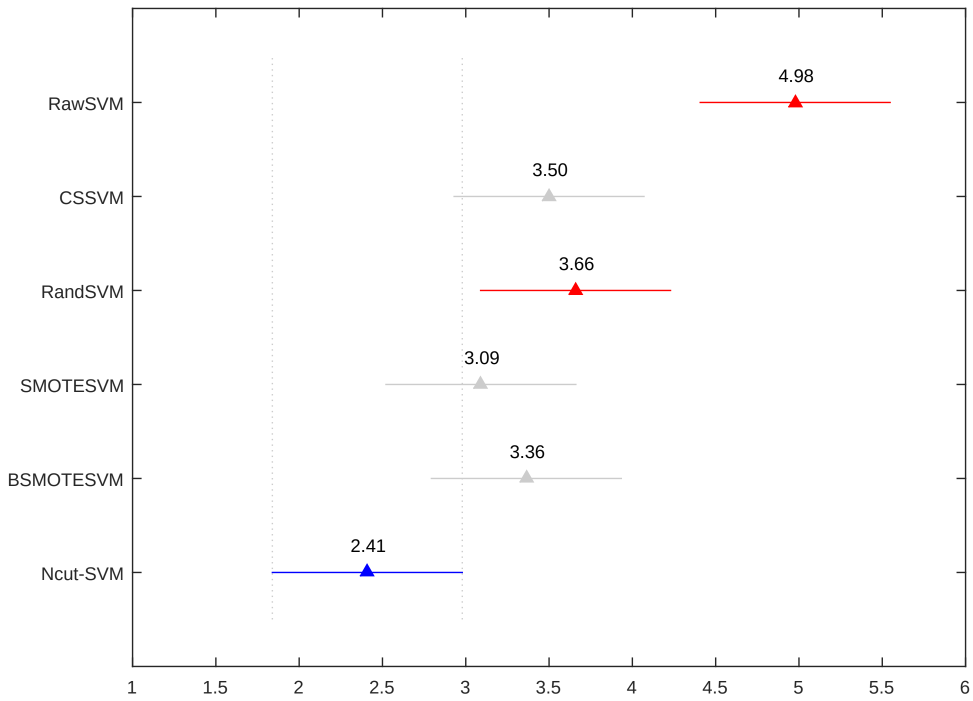


Figure 11: Post-hoc analysis on the ranks of state-of-the-art classification methods and Ncut–SVM. Average AUC rank of Ncut–SVM (2.41) is lower than that of all other methods; however, its rank is not significantly different from the ranks of the cost-sensitive SVM (3.50), the SMOTE SVM (3.09) and the BSMOTE SVM (3.36) by the Nemenyi post-hoc analysis.

that each of the two instance-selecting techniques — ensemble outlier-filtering and normalized-cut sampling — has a competitive classification performance comparable to several state-of-the-art imbalanced-data classification methods when combined to SVM. However, their average AUC ranks are not significantly lower than some of the state-of-the-art classification methods. Considering the significantly better classification performance of GU–SVM (which combined these two techniques) against the state-of-the-art imbalanced-data classification methods, the analysis of this section shows that both instance-selecting techniques comprising GU–SVM method are essential for its outstanding classification performance.

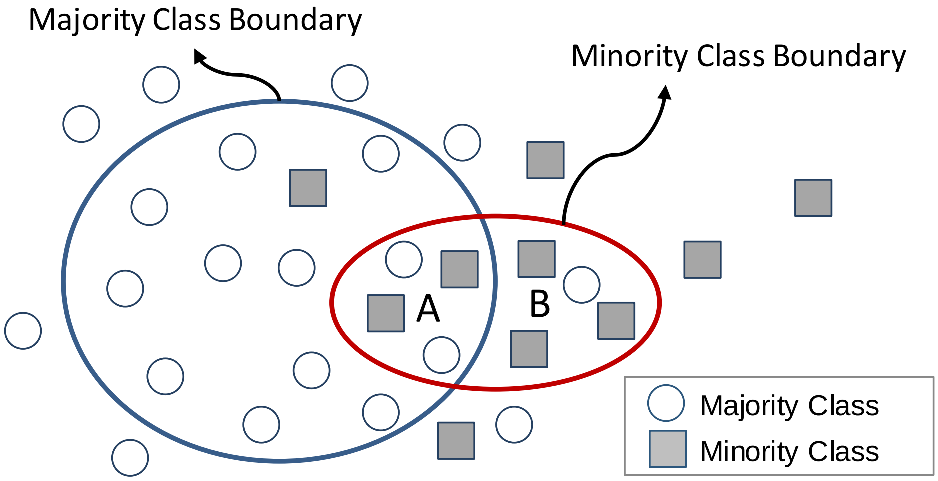


Figure 12: Illustration of the concept of minority overlap index. The minority overlap index is the proportion of the minority class region that is inside of the majority class region, $A/(A+B)$. The empirical minority overlap index is calculated by counting the number of minority instances in $(A+B)$ and dividing this number by the number of minority instances in A, which is $2/(2+3) = 0.4$ in this figure.

3.5 Analyzing GU–SVM in the Two Datasets Where It Performed Poorly

The computational results on the previous section demonstrated that GU–SVM performs significantly better than other state-of-the-art classification methods. However, from Table 8, it is evident that GU–SVM failed to outperform most methods on two datasets: the liver and glass base datasets. This section aims to explain the relatively poor performance of GU–SVM on the liver and glass datasets. Specifically, we argue that GU–SVM performance lags in datasets where a large portion of the region containing the minority class is inside of the region containing the majority class. The next paragraph formalizes this concept.

The *minority overlap index* is the proportion of the minority class region that is inside of the majority class region (Figure 12 illustrates this concept). To measure this index empirically, we use the following procedure: 1) Identify the class boundary of each class using a one-class SVM classifier. Specifically, in this study, we used Schölkopf et al. (2001)'s classifier with parameter $\nu = 0.5$ (Schölkopf et al. (2001) proved that given a $\nu \in (0, 1]$, their classifier is guaranteed to build a class boundary that includes at most, and approximately for $\nu \geq 0.5$, ν fraction of the class instances). 2) Calculate the empirical minority overlap index by counting the number of minority instances inside of both, the majority and minority, class boundaries and dividing this number by the number of minority instances inside the minority class boundary. Note that this calculation is empirically evaluating the ratio of the volumes of the intersection and the minority class; this empirical calculation is needed because calculating the volumes of high dimensional bodies is extremely difficult and time consuming.

Now, using the minority overlap index, we show that GU–SVM is indeed dominated by almost all of the other five state-of-the-art methods only when the empirical minority overlap index is very high (≥ 0.9). Figure 13 graphs, for each base dataset, the rank of GU–SVM (average of average AUC ranks of four different imbalance-ratio datasets [5, 10, 20, 30] against the other five state-of-the-art methods) versus the empirical minority overlap index. Here, for each base dataset, its empirical minority overlap index is calculated by using original datasets described in Table 7. Note that, among the 11 base datasets, the ranks of GU–SVM are less than or equal to two, with the exception of the liver and glass base datasets, which are the datasets with extremely high minority overlap indices (0.93

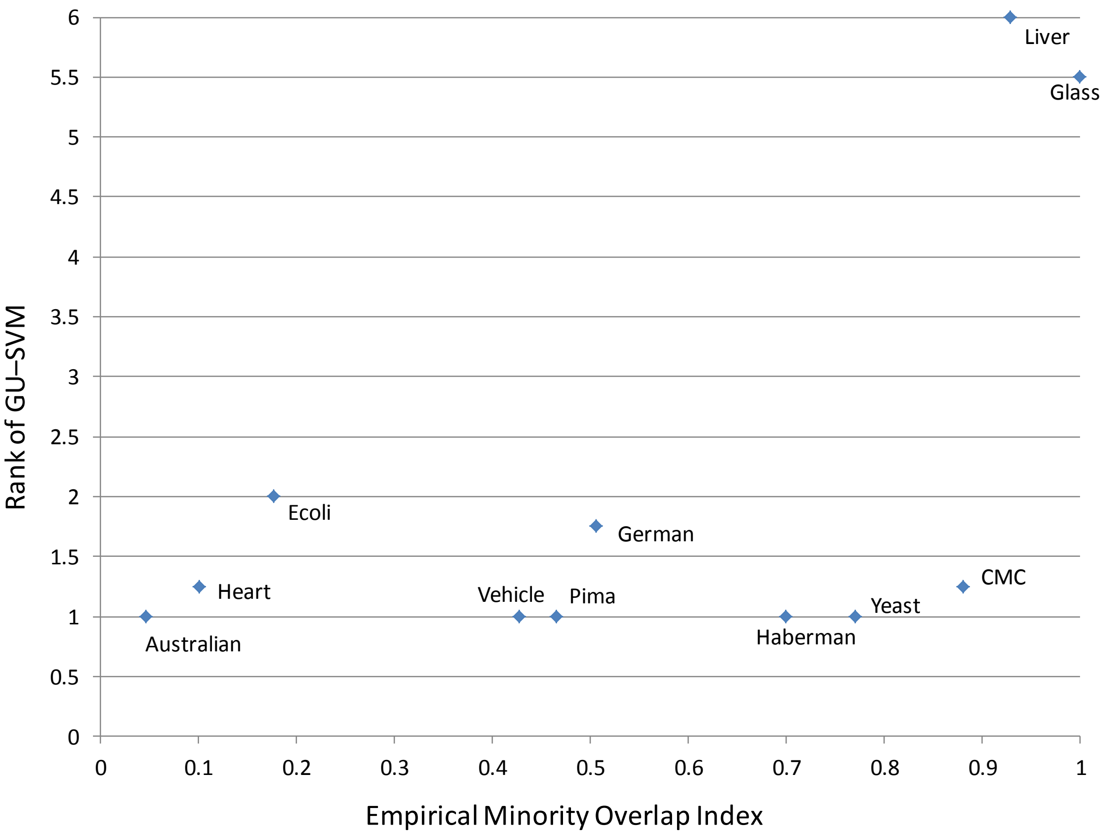


Figure 13: Average AUC ranks of GU–SVM in the 11 base datasets, which is plotted by the empirical minority overlap index. GU–SVM shows limited performance in the datasets where the minority overlap index is above a certain threshold.

and 1, respectively); all other datasets have minority overlap indices of less than 0.9. This result suggests that GU–SVM should be used for datasets with minority overlap index < 0.9 while other methods (specifically, SMOTE SVM, BSMOTE SVM, or cost-sensitive SVM) should be used for datasets with minority overlap index ≥ 0.9.

3.6 Conclusions

To improve classification performance in handling two-class imbalanced data, this chapter presents GU–SVM, a new imbalanced-data classification method. We believe that the take-away message from our investigation can be summarized as follows:

- Outlier-detection and removal from both classes is crucial for handling imbalanced data. In fact it makes a greater impact if one can identify and remove outliers in the minority class.
- Researchers understand the importance of selecting representative subsets of data while undersampling the majority class but how to best attain that goal is still under debate. We believe that the proposed normalized-cut base approach, aiming at spreading out the majority samples evenly, provides a new angle of looking at the problem and produces competitive results.
- Each component mentioned above in and by itself improves the performance for imbalanced data classification. Their combination makes a further, not negligible, enhancement.
- GU-SVM does not always outperform its competitors. But we discover a minority overlap index that can explain why and when GU-SVM may lose its edge. Using this minority overlap index, practitioners can apply GU-SVM when its strength can be taken advantage of or can apply the alternative methods otherwise.

4 SUMMARY

The principal objective of this dissertation was to develop data mining algorithms that outperform conventional data mining techniques on social and healthcare sciences. Toward this objective, this dissertation developed two data mining techniques, each of which addressed the limitations of a conventional data mining technique when applied in these contexts.

The first part of this dissertation addressed the problem of identifying important factors that promote or hinder population growth. When addressing this problem, previous studies included variables (input factors) without considering the statistical dependence among the included input factors; therefore, most previous studies exhibit multicollinearity between the input variables. Consequently, some of the results obtained via the regression analyses contradict each other. We proposed a novel methodology that, even in the presence of multicollinearity among input factors, is able to (1) identify significant factors affecting population growth and (2) rank these factors according to their level of influence on population growth. In order to measure the level of influence of each input factor on population growth, the proposed method combined decision tree clustering and Cohen's *d* index. We applied the proposed method to a real county-level United States dataset and determined the level of influence of an extensive list of input factors on population growth. Among other findings, we showed that poverty ratio is a highly important factor for population growth while no previous study found poverty ratio to be a significant factor due to its high linear relationship with other input factors.

The second part of this dissertation proposed a classification method for imbalanced data — data where the majority class has significantly more instances than the minority class. The specific problem addressed was that conventional classification methods have poor minority-class detection performance in imbalanced dataset since they tend to classify the vast majority of the test instances as majority instances. To address this problem, we developed a guided undersampling method (GUM) that combines two instance-selecting techniques — ensemble outlier filtering and normalized-cut sampling — in order to obtain a clean and well-represented subset of the original training instances. Our proposed imbalanced-data classification method uses GUM to select the training data and then applies support vector machines on the sampled data in order to construct the classification model (i.e., decide the final class boundary); the resulting method is referred to as GU–SVM. Our computational results showed that GU–SVM outperforms (with statistical significance) several state-of-the-art imbalanced-data classification methods, including cost-sensitive, sampling, and synthetic data generation approaches on eleven open datasets, most of them related to healthcare sciences.

GU–SVM showed superior classification performance in most of the eleven imbalanced datasets, though GU–SVM was outperformed by other state-of-the-art classification methods on datasets where a large portion of the region containing the minority class was inside of the region containing the majority class. We quantified this observation by defining the minority overlap index, devising a method to calculate said index empirically, and showing that indeed GU–SVM is only outperformed by other state-of-the-art methods in datasets with a very high empirically minority overlap index. Using this minority overlap index, practitioners can apply GU-SVM when its strength can be taken advantage of or can apply the alternative methods otherwise.

REFERENCES

Paul Attewell, David B. Monaghan, and Darren Kwong. *Data Mining for the Social Sciences: An Introduction*. University of California Press, 2015.

Francis R. Bach, David Heckerman, and Eric Horvitz. Considering cost asymmetry in learning classifiers. *The Journal of Machine Learning Research*, 7:1713–1741, 2006.

Patricia E. Beeson, David N. DeJong, and Werner Troesken. Population growth in US counties, 1840–1990. *Regional Science and Urban Economics*, 31(6):669–699, 2001.

Andrew P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

Leo Breiman, Jerome Friedman, Charles J. Stone, and Richard A. Olshen. *Classification and regression trees*. Chapman & Hall/CRC, 1984.

Carla E. Brodley and Mark A. Friedl. Identifying mislabeled training data. *Journal of Artificial Intelligence Research*, 11:131–167, 1999.

David L. Brown. Migration and community: Social networks in a multilevel world. *Rural Sociology*, 67(1):1–23, 2002.

David L. Brown, Glenn V. Fuguitt, Tun B. Heaton, and Saba Waseem. Continuities in size of place preferences in the united states, 1972–1992. *Rural Sociology*, 62(4):408–428, 1997.

Gavin Brown, Jeremy Wyatt, Rachel Harris, and Xin Yao. Diversity creation methods: A survey and categorisation. *Information Fusion*, 6(1):5–20, 2005.

Eunshin Byon, Abhishek K. Shrivastava, and Yu Ding. A classification procedure for highly imbalanced class sizes. *IIE Transactions*, 42(4):288–303, 2010.

Gerald A. Carlino and Edwin S. Mills. The determinants of county growth. *Journal of Regional Science*, 27(1):39–54, 1987.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

Samprit Chatterjee and Ali S. Hadi. *Regression analysis by example*. Wiley-Interscience, 2006.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

Nitesh V. Chawla, Nathalie Japkowicz, and Aleksander Kotcz. Editorial: Special issue on learning from imbalanced data sets. *ACM SIGKDD Explorations Newsletter*, 6(1):1–6, 2004.

Guangqing Chi and David W. Marcouiller. Isolating the effect of natural amenities on population change at the local level. *Regional Studies*, 45(4):491–505, 2011.

Guangqing Chi and Paul R. Voss. Small-area population forecasting: Borrowing strength across space and time. *Population, Space and Place*, 17(5):505–520, 2010.

David E. Clark and Christopher A. Murphy. County wide employment and population growth: An analysis of the 1980s. *Journal of Regional Science*, 36(2):235–256, 1996.

Jacob Cohen. The statistical power of abnormal-social psychological research: a review. *Journal of abnormal and social psychology*, 65(3):145–153, 1962.

Jacob Cohen. *Statistical power analysis for the behavioral sciences*. Lawrence Erlbaum, 1988.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3): 273–297, 1995.

Steven C. Deller, Tsung-Hsiu Sue Tsai, David W. Marcouiller, and Donald B.K. English. The role of amenities and quality of life in rural economic growth. *American Journal of Agricultural Economics*, 83(2):352–365, 2001.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

Pedro Domingos. Metacost: A general method for making classifiers cost-sensitive. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 155–164. ACM, 1999.

Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley & Sons, New York, NY, 2nd edition, 2001.

Andrew Estabrooks, Taeho Jo, and Nathalie Japkowicz. A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence*, 20(1):18–36, 2004.

Milton Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32(200):675–701, 1937.

Glenn V. Fuguitt and James J. Zuiches. Residential preferences and population distribution. *Demography*, 12(3):491–504, 1975.

Linda M. Ghelfi and Timothy S. Parker. A county-level measure of urban influence. *Rural Development Perspectives*, 12(2):32–41, 1997.

John S. Gilmore. Boom towns may hinder energy resource development. *Science*, 191 (4227):535–540, 1976.

Stephen M. Golant. Adjustment process in a system: A behavioral model of human movement. *Geographical Analysis*, 3(3):203–220, 1971.

Edith E. Graber. Newcomers and oldtimers: Growth and change in a mountain town. *Rural Sociology*, 39(4):503–513, 1974.

William H. Greene. *Econometric Analysis*. Pearson series in economics. Pearson Prentice Hall, 7th edition, 2012.

Michael J. Greenwood. Research on internal migration in the United States: A survey. *Journal of Economic Literature*, 13(2):397–433, 1975.

Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *Advances in Intelligent Computing*, pages 878–887. Springer, 2005.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: Data mining, inference and prediction*. Springer, 2nd edition, 2009.

Boyd Hamilton Hunter and Diane Evelyn Smith. Surveying mobile populations: Lessons from recent longitudinal surveys of indigenous australians. *Australian Economic Review*, 35(3):261–275, 2002.

John Iceland, Gregory Sharp, and Jeffrey M. Timberlake. Sun belt rising: Regional population change and the decline in black residential segregation, 1970-2009. *Demography*, 50(1):97–123, 2013.

Nathalie Japkowicz. The class imbalance problem: Significance and strategies. In *Proceedings of the 2000 International Conference on Artificial Intelligence (ICAI)*, pages 111–117, 2000.

George H. John. Robust decision trees: Removing outliers from databases. In *Knowledge Discovery and Data Mining*, pages 174–179. AAAI Press, 1995.

Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. *Advances in Neural Information Processing Systems*, 7:231–238, 1995.

Miroslav Kubat and Stan Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *Proceedings of the Fourteenth International Conference on Machine Learning*, volume 97, pages 179–186, 1997.

Gerald R. Leslie and Arthur H. Richardson. Life-cycle, career pattern, and the decision to move. *American Sociological Review*, 26(6):894–902, 1961.

Moshe Lichman. UCI Machine Learning Repository, 2013. URL `http://archive.ics.uci.edu/ml`.

Yi Lin, Yoonkyung Lee, and Grace Wahba. Support vector machines for classification in nonstandard situations. *Machine Learning*, 46(1-3):191–202, 2002.

Charles X. Ling, Qiang Yang, Jianning Wang, and Shichao Zhang. Decision trees with minimal costs. In *Proceedings of the Twenty-first International Conference on Machine learning*, page 69. ACM, 2004.

Wei Yin Loh. Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):14–23, 2011.

Marcus A. Maloof. Learning when data sets are imbalanced and when costs are unequal and unknown. In *International Conference on Machine Learning (ICML)-2003 Workshop on Learning from Imbalanced Data Sets II*, volume 2, 2003.

Hamed Masnadi-Shirazi and Nuno Vasconcelos. Risk minimization, probability elicitation, and cost-sensitive SVMs. In *International Conference on Machine Learning (ICML)*, pages 759–766, 2010.

David A. McGranahan. Natural amenities drive rural population change. *Agricultural Economic Report 781*, 1999.

David A. McGranahan and Calvin L. Beale. Understanding rural population loss. *Rural America*, 17(4):2–11, 2002.

David Mease, Abraham J. Wyner, and Andreas Buja. Boosted classification trees and class probability/quantile estimation. *The Journal of Machine Learning Research*, 8: 409–439, 2007.

Douglas C. Montgomery, Elizabeth A. Peck, and G. Geoffrey Vining. *Introduction to linear regression analysis*. Wiley, 5th edition, 2012.

Paul F. Nemenyi. Distribution-free multiple comparisons. *PhD thesis, Princeton University*, 1963.

Y. Pan and R.T. Jackson. Ethnic difference in the relationship between acute inflammation and serum ferritin in us adult males. *Epidemiology and infection*, 136(3):421, 2008.

Raymond Pearl and Lowell J. Reed. On the rate of growth of the population of the united states since 1790 and its mathematical representation. *Proceedings of the National Academy of Sciences of the United States of America*, 6(6):275–288, 1920.

Arash Pourhabib, Bani K. Mallick, and Yu Ding. Absent data generating classifier for imbalanced class sizes. *The Journal of Machine Learning Research*, 16:2695–2724, 2015.

Henry S. Pritchett. A formula for predicting the population of the united states. *Publications of the American Statistical Association*, 2(14):278–286, 1891.

John M. Quigley. Urban diversity and economic growth. *The Journal of Economic Perspectives*, 12(2):127–138, 1998.

Anil Rupasingha and Stephan J. Goetz. County amenities and net migration. *Agricultural and Resource Economics Review*, 33(2), 2004.

Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Larry A. Sjaastad. The costs and returns of human migration. *The Journal of Political Economy*, 70(5):80–93, 1962.

Stanley K. Smith, Jeff Tayman, and David Arthur Swanson. *State and local population projections: Methodology and analysis*. Kluwer Academic, Plenum Publ., 2001.

Alden Speare. Residential satisfaction as an intervening variable in residential mobility. *Demography*, 11(2):173–188, 1974.

Steven K. Thompson. *Sampling*. Wiley, 3rd edition, 2012.

U.S. Census Bureau. USA Counties. Retrieved July 26, 2011 http://censtats.census.gov/usa/usa.shtml, 2011.

U.S. Department of Agriculture. Natural amenity data. Retrieved July 26, 2011 http://www.ers.usda.gov/topics/rural-economy-population/natural-amenities.aspx, 2011.

Sofie Verbaeten and Anneleen Van Assche. Ensemble methods for noise elimination in classification problems. In *Multiple Classifier Systems*, volume 2709, pages 317–325. Springer Berlin Heidelberg, 2003.

Dongling Wang and Meng Shi. Density weighted region growing method for imbalanced data SVM classification in under-sampling approaches. *Journal of Information and Computational Science*, 11:6673–6680, 2014.

Gary Weiss and Foster Provost. The effect of class distribution on classifier learning: An empirical study. Technical report, ML-TR-44, Department of Computer Science, Rutgers University, New Jersey, 2001.

Dennis L. Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-2(3):408–421, 1972.

Show-Jane Yen and Yue-Shi Lee. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, 36(3):5718–5727, 2009.

Wilbur Zelinsky. The hypothesis of the mobility transition. *Geographical review*, 61(2): 219–249, 1971.

APPENDIX: DETAILED DESCRIPTION OF THE INPUT VARIABLES IN PART I

This appendix is dedicated to give a detailed description of the input variable appeared in Part I. For input variables, many previous studies tried to include as many input variables as possible (see, for example, Carlino and Mills 1987; Clark and Murphy 1996; Beeson et al. 2001; Deller et al. 2001). To avoid including an unnecessarily large number of input variables, we classified input variables into five categories (*Income*, *Policy*, *Race*, *Natural Amenity*, and *Others*) and attempted to minimize the number of input variables while ensuring each category was adequately represented; this was achieved by selecting the variables that are commonly used ones in past studies.

Next we give a description of each input–variable category followed by the explicit list of variables within the category.

Income

In this category, we include income–related indicators to capture factors that influence the economic circumstances of individuals.

1. *Median Income*: Median household income of the county as measured in \$US (data from 2000, average: $\$36,274$).
2. *Poverty Ratio*: People in poverty in the county given as a proportion (data from 2000, average: 13.31%).

Policy

This category is intended to capture the government's financial policy that may attract households and firms to establish in the region. The variable *Local Net* provides the local government's fiscal spending tendency (i.e., deficit or surplus) by giving the local government's budget balance as a proportion of the difference between the local government's revenue and expenditure to the local government's revenue. Specifically, the *Local Net* of a given county is the difference between the county's yearly revenue and expenditures divided by the yearly revenue of the county. Since the data from 2000 for *Local Net* and *Highway* are not available, we used the weighted averages of 1997 and 2002.

1. *Federal Expenditure*: Amount of money that the federal government expended in the county as measured in \$ (data from 2000, average: \$484,422,000).
2. *Local Net*: Local government budget balance given as a proportion of the difference between the local government's revenue and expenditure to the local government's revenue (weighted average of 1997 and 2002, average: 1.10%).
3. *Highway*: Local government highway expenditure as measured in \$ (weighted average of 1997 and 2002, average: \$12,557,000).

Race

The race category includes county measures of ratio for specific races.

1. *Black Ratio*: Percentage of Black persons in the county (data from 2000, average: 8.84%).
2. *Asian Ratio*: Percentage of Asian persons in the county (data from 2000, average:

0.77%).

3. *Hispanic Ratio*: Percentage of Hispanic or Latino origin persons in the county (data from 2000, average: 6.19%).

Natural Amenity

The natural amenity category includes local amenities that can affect the productivity of the corresponding county as well as migration patterns in and out of the county (McGranahan and Beale, 2002). To include information about summer temperature while removing its correlation with January temperature, we adopt *Temperature Gap*, a residual of regression of July on January temperature, introduced by McGranahan (1999). Low *Temperature Gap* means low temperature residual gap between January and July, which leads to consistent yearly temperature (low winter–summer temperature gap). Since residuals are not correlated with independent variables, *Temperature Gap* is not redundant with January temperature. We also included *Urban Influence Code*, an urban–rural classification code, designed by Ghelfi and Parker (1997) and *Topography Code*, a land formation classification code, designed by McGranahan (1999). To be associated with the decision tree model, these two codes are implemented in the input variable dataset as a set of dummy variables which take the value 0 or 1 to indicate the absence or presence of some categorical effect that may be expected to shift the outcome.

1. *WaterArea Ratio*: Proportion of water area in the county (data from 2000, average: 4.63%).

2. *January Temperature*: Average county January temperature (data from 1941 to

1970, average: $32.90^{\circ}F$).

3. *January Sun*: Average county January sunny hours (data from 1941 to 1970, average: 151.57 *hours*).
4. *Temperature Gap*: Temperature regression residual gap between January and July of the county (data from 1941 to 1970, average: $0.00^{\circ}F$).
5. *July Humidity*: Average July humidity of the county (data from 1941 to 1970, average: 56.13%).
6. *Urban Influence Code*: Urban–rural classification for counties. Responses range from 1 to 9 with 1 being a metro county of 1 million population or more and 9 being a non-metro with a population less than 2,500 (data from 1941 to 1970).
7. *Topographic Code*: Topographic classification of land formation. Responses range from 1 to 21 with 1 being a flat plain and 21 being a high mountain (data from 1941 to 1970).

Others

This category includes input variables that do not fit in the previous categories but that we think are relevant and other researchers have also included them.

1. *Crime Rate*: Number of violent crimes per 1000 persons in the county (data from 2000, average: 0.24).
2. *College Ratio*: Proportion of bachelor's degree of the county (data from 2000, average: 16.50%).
3. *Employment Rate*: Proportion of employed people to population (15 and over) of

the county (data from 2000, average: 65.94%).

4. *Population Density*: Population per square mile of the county (data from 2000, average: 245.10 persons per square mile).