August 2018

Application of Machine Learning in Cancer Research

Mandana Bozorgi
mandanbozorgi@gmail.com

APPLICATIONS OF MACHINE LEARNING IN CANCER RESEARCH

by

Mandana Bozorgi

Master of Science (M.Sc.)
University of Nevada, Las Vegas
2015

A thesis submitted in partial fulfillment of
the requirements for the

Doctor of Philosophy – Computer Science

Department of Computer Science
Howard R. Hughes College of Engineering
The Graduate College

University of Nevada, Las Vegas
August 2018

UNLV | GRADUATE COLLEGE

Dissertation Approval

The Graduate College
The University of Nevada, Las Vegas

February 27, 2018

This dissertation prepared by

Mandana Bozorgi

entitled

Applications of Machine Learning in Cancer Research

is approved in partial fulfillment of the requirements for the degree of

Doctor of Philosophy - Computer Science
Department of Computer Science

Kazem Taghva, Ph.D.
Examination Committee Chair

Kathryn Hausbeck Korgan, Ph.D.
Graduate College Interim Dean

Laxmi Gewali, Ph.D.
Examination Committee Member

Fatma Nasoz, Ph.D.
Examination Committee Member

Justin Zhan, Ph.D.
Examination Committee Member

Ashok Singh, Ph.D.
Graduate College Faculty Representative

Abstract

This thesis revisits the problem of five year survivability predictions for breast cancer using machine learning tools. This work is distinguishable from the past experiments based on the size of the training data, the unbalanced distribution of data in minority and majority classes, and modified data cleaning procedures. These experiments are also based on the principles of TIDY data and reproducible research. In order to fine-tune the predictions, a set of experiments were run using naive Bayes, decision trees, and logistic regression. Of particular interest were strategies to improve the recall level for the minority class, as the cost of misclassification is prohibitive. One of The main contributions of this work is that logistic regression with the proper predictors and class weight gives the highest precision/recall level for the minority class.

In regression modeling with large number of predictors, correlation among predictors is quite common, and the estimated model coefficients might not be very reliable. In these situations, the Variance Inflation Factor (VIF) and the Generalized Variance Inflation Factor (GVIF) are used to overcome the correlation problem. Our experiments are based on the Surveillance, Epidemiology, and End Results (SEER) database for the problem of survivability prediction. Some of the specific contributions of this thesis are:

1. detailed process for data cleaning and binary classification of 338,596 breast cancer patients.
2. computational approach for omitting predictors and categorical predictors based on VIF and GVIF.
3. various applications of Synthetic Minority Over-sampling Techniques (SMOTE) to increase precision and recall.
4. An application of `Edited Nearest Neighbor` to obtain the highest F1-measure.

In addition, this work provides precise algorithms and codes for determining class membership and execution of competing methods. These codes can facilitate the reproduction and extension of

our work by other researchers.

Acknowledgements

I would like to take this opportunity to express my gratitude to my advisor, Prof. Kazem Taghva, whose guidance, mentorship, patience, and generosity have always made him the best mentor who guided me through this process. His continuous teaching, support, and understanding throughout this process have allowed me to learn, innovate, enjoy, and made it a rewarding experience. I would like to thank Dr. Laxmi Gewali, Dr. Ashok Singh, Dr.Fatma Nasoz and Dr. Justin Zhan for their time, valuable comments input and for agreeing to serve on my defense committee. My most profound gratitude to Prof. Datta for bestowing on me the opportunity to benefit from his advice, wisdom, and guidance throughout these past years. My dearest family- my father Dr. Fattah Bozorgi, my mother Mrs. Azadeh Mehran, and my brother Mr. Mike Bozorgi. Thank you for your unconditional love, support, and encouragement throughout the years, without which I would not be where I am today. Special thanks to my dear friend, Dr. Jaleh Pourhamidi for her valuable friendship, collaboration, and support over the years. To all my teachers, family, friends, and colleagues past and present: I humbly thank you for the positive impact you have had on my life, both personally and professionally.

Mandana Bozorgi

University of Nevada, Las Vegas

August 2018

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

According to the National Breast Cancer Organization [Cancer, 2016], "Breast cancer is a disease in which malignant (cancer) cells form in the tissues of the breast." Over 230,000 women are diagnosed with breast cancer in the United States annually[Cancer, 2016][breastcancer.org]. In addition about one in eight women will develop breast cancer. These alarming statistics have led to tremendous research efforts and studies associated with breast cancer in recent years. In addition, many organizations have compiled statistical data pertaining to individual patients. One such database is Surveillance, Epidemiology, and End Results (SEER) database which is maintained by National Cancer Institute (NCI) [NCI, 2016]. The SEER database is a rich source of information for statistical learning analysis. For example, Bellaachia and Guven in 2006 carried out a comparative study of three data mining techniques in order to predict five year survivability based on SEER data [Bellaachia and Guven, 2006].

Since SEER database is updated on a regular basis with new patients, it is logical to repeat some of the past experiments. As the first step, we wanted to repeat the same experiments to establish a basis for comparison with a new updated SEER data. It turns out that we could not repeat experiments reported by Bellaachia and Guven [Bellaachia and Guven, 2006] and by Delen, Walker, and Kadam [Delen et al., 2005]. This is because the reported data preparation, clean up, and data processing were incomplete and ambiguous. As a result, these cited works were not reproducible research [Peng, 2011].

This thesis revisits the topic of prediction of five year survivability for breast cancer with machine learning tools, following the principles of TIDY data and reproducible research as discussed by Peng [Peng, 2011] and Wickham [Wickham, 2014]. Of particular interest in how to set up an environment that other researchers could use to apply the same techniques on other types of cancer.

This thesis is organized into six chapters, including this introduction.

Chapter 2 gives a detailed background on SEER database, clinical definition of five year survivability, TIDY data and reproducible research.

In chapter 3, we describe our approach to data cleaning and category identification for patients in the SEER database. We also provide the list of the chosen attributes from the SEER data that characterizes each patient.

Chapter 4 gives a brief introductions to specific machine learning techniques used in this work. More specifically, we provide a short introduction to Naive Bayes, decision trees, linear regression, and logistic regression. Also we summarize some of the notable works associated with data science, SEER database, and machine learning techniques.

Chapter 5 describe the comparison results for Naive Bayes, decision trees, and logistic regression. The metrics of comparison are precision, recall, F1-measure, and ROC that are also defined in this chapter.

In chapter 6, we explain our ideas on correlation, prediction, and fine tuning of our regression model. In particular, we describe how VIF and GVIF are used to overcome the correlation problem.

Chapter 7 concludes this thesis with a short summary of our results and explores directions for future work.

Chapter 2

Cancer Data, Survivability, and Reproducible Research

Typically, many projects use data sets that were not necessarily collected for those projects. For example, SEER database is built for summarizing cancer data and not survivability prediction. The survivability prediction problem is a binary classification with uneven distribution of data points [Vapnik, 1995][Xiao et al., 2009]. In order to prepare SEER data for binary classification, we must first decide how to assign data points to each class. According to Parkin and Hakulinen, a well-accepted methodology in predicting patient survival involves summarizing and analysis [Parkin and Hakulinen, 1991]. The most widely used metric involves calculating the percentage of patients alive after five years, using a direct method as outlined by Parkin and Hakulinen [Parkin and Hakulinen, 1991].

The following three sections provide detailed description for SEER data, five year survivability, and reproducible research:

2.1 SEER data

Every year the National Cancer Institute (NCI) releases the latest cancer statistics. The NCI recognizes the need for greater research of a more diverse population in order to better understand and to support the researchers in the field. Since SEER shares this same sentiment, the NCI has funded SEERs registries. SEER stands for `surveillance`, `epidemiology` and `end results`. The data used is based on the SEER registry program.

On January 1, 1973, SEER began to collect cancer data from Connecticut, Iowa, New Mexico,

Utah, Hawaii and the metropolitan areas of Detroit and San Francisco/ Oakland. In 1974, it expanded to the metropolitan area of Atlanta and the 13 county Seattle/Puget Sound Area. By 1978, 10 predominately African American rural counties in Georgia were added. In 1980, Native Americans residing in Arizona were included. By the end of 1990, New Orleans, Louisiana (1974-1977, rejoined 2001); the state of New Jersey (1979-1989, rejoined 2001); and Puerto Rico (1973-1989) were added to the SEER registries.

But it didnt stop there. By 1992, SEER had increased its coverage of minority populations, especially the Hispanic population. In California, Los Angeles County and 4 counties in the San Jose/Monterey area south of San Francisco were added. In 2001, the state of Kentucky and the remaining counties of California were added[NCI, 2016].

SEER is also part of a larger national cancer registration program, which includes registries managed by the CDC (Cancer for Disease and Prevention). SEER, in conjunction with the NCI and the CDC, covers the vast majority of the United States. The SEER registry is a fundamental component of the data system for cancer research.

In 2010, the state of Georgia was added to the SEER registry. In some areas like New Jersey, greater California and Louisiana, funds from the NCI and the CDC (Center for Disease Control and Prevention), SEER received combined funds from NCI and CDC.

Each November SEER registrars report the latest cancer cases to the NCI. Every year, NCI, CDC, American Cancer Society (ACS) and the North American Association of Central Cancer Registries (NAACCR) collaborate for providing updates on cancer incidence and death rates. The first report of them published in 1998[Edwards et al., 2014], and the most recent report provides update cancer rates and trends for all cancer types or combination[Edwards et al., 2014].

The current NCI statistics are from 1973 to 2013. The NCI always releases data with a two/three-year gap, due to the complicated collecting process. Also, they wish to ensure the quality of the data.

SEER collects information on up to 94 different types of cancer including: Liver, Lung, Prostrate, Breast , Colon, Skin, Thyroid, Melanoma, Middle Ear, Ovary, Testis, Kidney, Orbit, KS, Brain, OthEye, Lymphoma, HeartMediastinum, KidneyParenchyma , NETColon, etc.

SEER reports the cancer data in 143 attributes. For instance, information such as Patient ID, Race, Marital Status, Primary Site Code, Histologic Type, Behavior Code, Grade, Extension Of Tumor, Lymph Node Involvement, RXSUMM surgery primary site, Radiation, Stage Of Cancer, Age, Tumor Size, Number Of Positive Nodes, Vital Status, Survival Month, Year Of Diagnosis,

Table 2.1: Tumor Size categories in SEER Data

Code	Description
000	Indicates no mass or no tumor found
001-988	Exact size in millimeters
989	989 millimeters or larger
990	Microscopic focus or foci only; no size of focus is given
991	Described as less than 1 cm
992	Described as less than 2 cm
993	Described as less than 3 cm
994	Described as less than 4 cm
995	Described as less than 5 cm
996-998	Site-specific codes where needed
888	Not applicable
999	Unknown; size not stated; not stated in patient record

Table 2.2: Cancer Stage categories in SEER Data

Code	Description
0	In situ
1	Localized
2	Regional
3	Microscopic focus or foci only; no size of focus is given
4	Distant
8	Localized/Regional Only used for Prostate cases.
9	Unstaged

Month of Diagnosis and Cause Of Death, etc. We start our research by first looking at Breast cancer information, and not all the attributes are related to Breast cancer. The attributes such as Brain, Lung, Bone , Liver, etc doesn't have any information related to Brest cancer.

In SEER data, we have different distinctions for each attribute. For example,there are information such as what kind of radiation patient received or if the patient refused radiation even though it was recommended and so on.

Another example is Tumor sizes that represented in 12 different categories shown in the table 2.1.

In SEER data, cancer's stages are represented in 5 categories: in situ, localized, regional, distant and unknown that shown in the table 2.2.

In SEER data, we have detailed information related to grading. Besides grade I, II, III and IV, we have the T-Cell, B-Cell, Null Cell and N K Cells information. These are shown in the table 2.3.

SEER data used in the vast area of cancer research. For instance, [Al-Bahrani et al., 2013] Used SEER data to find actual survival rate for Colon Cancers patients. In this study, the multiple

Table 2.3: Cancer Grade categories in SEER Data

Code	Description
1	Grade I
2	Grade II
3	Grade III
4	Grade IV
5	T-cell
6	B-cell
7	Null cell
8	N K cell
9	cell type not determined

classification schemes used to estimate the risk of mortality after one, two and five years of diagnosis.

In that study [Al-Bahrani et al., 2013] compared basic classifiers, J48 decision tree, reduced error pruning tree, random forest , alternating secision tree and logistic regression, with Meta classifiers, Bagging, AdaBoost, Random SubSpace and Voting With selected 13 attributes. The result shows that the voting method has the best and more accurate survivability rate. Another study [Davies and Welch, 2014] used SEER data for analyzing the increasing thrend in Thyroid cancers patients. Since 1975, the patients who diagnosed with Thyroid cancer are nearly tripled while its mortality has remained stable. That at the end of this research it appears that its just overdiagnosis of papillary thyroid.

Another example is research by [Abdel-Rahman, 2017], the SEER data used on the Mediastinal tumors' research. Mediastinal tumors can be benign or malignant, and it's just growing in the area of the chest like heart. The results of this study are shown that surgical resection plays a particularly important role in the management of this disease.

In our research, we use Python 3.5.1, Anaconda 2.4.0 and Pandas version 0.17.0. At the beginning, we divide our data into two parts, information gathered from 1973-2003 and information gathered from 2004-2013 (2004+). We then merge them together to find proper data. The reason for this is due to some information in the description being stored in a different position; for example, information about tumor size collected in position 61-63 for the years 1988 to 2003 and after 2004 stored in position 96-98.

2.2 Survivability

One way to determine a patients survivability is to use biostatistics as a survival analysis methodology. This methodology can help to quantify and describe survival time. In addition, it examines the greatness of differences in survival time [Fink and Brown, 2006].

In an ideal study, all patients would be diagnosed at the same time, stay in the study until an outcome was achieved (possibly death), and participate in follow-ups. However, it is an almost impossible task to find a large group of patients with the ideal conditions. Some patients were diagnosed prior to entering our data set, and they had already begun treatment. Others decided to leave the study, so they never followed up. Since we do not always have an ideal dataset, we need to develop statistical strategies to obtain good information from the incomplete dataset. This is the reason we are led to use the survival analysis techniques as defined in [Fink and Brown, 2006].

The life of the patient (survivability) is an important variable in our research. Generally, a patient diagnosed with any kind of cancer who lives 5 years or more is considered to be in remission. Survivability depends on many factors, such as cancer stage, age group, tumor size, amount of positive nodes etc. In SEER data, VitalStatus is used to represent if the patient is still alive or not. Also, we have additional information regarding survival months and cause of death. Used together, all this information helps us to evaluate the different survival techniques and to select one. There are several different techniques used in calculating survivability. A few of the techniques used are the Direct Method, Actuarial Method and Kaplan-Meier Method which will be described in the followings [Parkin and Hakulinen, 1991]:

2.2.1 Direct Method

The Direct Method [Parkin and Hakulinen, 1991] is the most cited method for calculating lifetime probability. In this method, the patients survival rate is evaluated at the end of a specific time interval. If a patient had survived for a minimum of 60 months (5 years) after being diagnosed, the patient would be notated as `Survived`, even if the patient were no longer alive. The key factor being that the patient lived 5 years after the initial diagnosis. On the other hand, if a patient dies prior to 60 months (5 years) and the cause of death is cancer, then the patient is considered `Not-Survived`. Patients who live less than 5 years and die from any other cause than cancer are not considered.

Table 2.4: Actuarial Life Type

`Value`
Number of patients at the beginning of the interval
Number of patients who died during the interval time
Number of patients stop/lost to follow up during the interval time
Number of patients exposed to risk of death
Number of patients withdrawn alive during the interval
Conditional probability of death
Conditional probability of survival

2.2.2 Actuarial Method

Cutler and Ederer used the Actuarial Method in 1958 to develop the life-table analysis [Dawson and Trapp, 2004] used in todays survivability analysis. In this method, a dataset table contains information, such as the number of patients at the beginning of an interval, the number of deceased patients etc. [Lucijanic and Petrovecki, 2012][Parkin and Hakulinen, 1991]. Please see the table 2.4.

The survival rate will have been calculated based on these variables.

In Actuarial method `Number of patients exposed to risk of death` is the average of the `Number of patients withdrawn alive during the interval` and the `Number of patients stop Or lost to follow up during the interval`.

The Conditional probability of death calculated as:

$$\frac{Number\ Of\ patients\ who\ died\ during\ the\ interval\ time}{Number\ Of\ Patients\ exposed\ to\ risk\ of\ death} \tag{2.1}$$

The conditional probability of survival is defined similarly as:

$$\frac{Number\ of\ patients\ who\ died\ during\ the\ interval\ time}{Number\ of\ patients\ exposed\ to\ risk\ of\ death} \tag{2.2}$$

The `Survival rate` is calculated by multiplying the Conditional probability of survival for each interval time [Fink and Brown, 2006].

This information is then added to the Life-Table or Actuarial Table. And is represented by the survival curve. This method is good only if the interval is for a short period of time; it is not designed for long intervals. Also, we are interested in working with patients who wish to enter our data set at any given time. Thats why the Actuarial Method is one that we are less likely to use.

2.2.3 Kaplan-Meir Method

The Kaplan-Meir method is similar to the Actuarial method [Parkin and Hakulinen, 1991], However instead of a cumulative survival rate at the end of each year of follow up, the proportion of patients still surviving can be calculated at intervals as short as the accuracy of recording date of death permits[Austin, 2014]. The Kaplan-Meir method also evaluates in tabular form [Kaplan and Meier, 1958]. In this method time consider as reference point, different points of calculation divided by time. It evaluate estimation of survivability over time, even though some patients dont have any follow up records, and repeat the study for different length of time. One disadvantage of Kaplan-Meir method is that its difficult to find best proportion due to the censoring the data[Austin, 2014]. In addition, the choice of time is arbitrary and it is misleading the survival curve comparison [Lucijanic and Petrovecki, 2012].

2.3 Reproducible Research

Machine learning is used in a variety of statistical, probabilistic and optimization techniques in many different complex data sets. In machine learning techniques, we learn historical information and can then detect a pattern from the data sets. These learning techniques are used to discover new facts from the data and to interpret the data patterns. This helps researchers to better prepare and issue useful information. Machine learning techniques are frequently used in cancer diagnosis and detection.

More recently, they have been used for cancer prediction. Unfortunately, very few research papers have fully addressed their process. In all machine learning experimental studies, preparing and cleaning of the data has been a major factor. However, data computation steps have been ignored in everyday research publications [Millman and Pérez, 2014]. They are mainly considered to be a task for fellow researchers to figure out. Most of the previous research data and software have been poorly saved and organized, making it almost impossible to reach the identical result as the publisher. In other words, many of the codes and closed-sources make it hard to completely understand the research [Sonnenburg et al., 2007]. In much of this research, it is difficult and almost impossible to reproduce the same results due to lack of information, and insufficient data descriptions, data processing, source codes, and so on. There is no doubt that reproducible research can be a foundation for further studies by providing the software, source codes and data sources. Then researchers will be able to easily and quickly adopt the methods [Sonnenburg et al., 2007].

Because of the Internet and social media, today everybody has a chance to voluntarily share their ideas. Web 2.0 is one of the defining characteristics of these systems, for instance, YouTube, Wikipedia, Open Source Software (OSS) etc [Oreilly, 2005]. OSS usually refers to computer software products, which users are allowed to freely use, modify and redistribute. GitHub and BitBucket are examples. The largest OSS community on the web is called SourceForge,in March 2014,contains more than 430,000 projects and over 3.7 million registered members.[Wikipedia, 2017] and it's competing with other "providers such as GitHub, Bitbucket, RubyForge, Tigris.org, BountySource, Launchpad, BerliOS, JavaForge, GNU Savannah, and GitLab" [Wikipedia, 2017].

SourceForge allows developers to manage their own source code along with the people who have access to the code, to keep track of different updates on their work, and to give others permission to download their code. SourceForge has a variety of sub-communities. With the Internet and social media technologies help everybody has a chance to share their ideas voluntarily. The Web 2.0 is one of the defining characteristics of this system, for instance, YouTube, Wikipedia, Open Source Software(OSS), etc [Oreilly, 2005]. OSS usually refers to computer software products which users are allowed to freely use, modify and redistributed.also let the others download their materials.

The idea of using reproducible research is straightforward and convenient because programmers or users can read, modify and republish the study's result [Millman and Pérez, 2014]. When using reproducible research, we need to use open source software, which allows both the free use and the exchange of information. The Open Source Software required for reproducible research must:

- Be free and easy to access
- Allow researchers to build, to modify, and to redistribute the information, such as source codes, citations, and graphs, etc., as many times as necessary
- Permit others access to the code of origin. This helps researchers to understand the model better and to develop new methods quickly [Sonnenburg et al., 2007].

In August 2004, an open letter signed by 25 Nobel laureates was sent to the United States Congress stating, *'Open access indeed expands shared knowledge across scientific fields, it is the best path for accelerating multi-disciplinary breakthroughs in research.'* [Sonnenburg et al., 2007]. It is necessary to follow the same experimental and data to obtain the same result. Same experimental means that by downloading and running the code on the same data on different machines, the researcher will end up with the same result.' the Same result means identical result or out-

put[Schaffner, 1994]. We firmly believe reproducible research has many benefits with few if any, disadvantages. There are many different fields using machine learning techniques. Using open source software and making machine learning techniques reproducible are preferred[Feller and Fitzgerald, 2000]. By sharing the code of origin, paper, and data, we can achieve the reproducibility of machine learning research. The advantages of having data availability, reproducibility, and testability allow faster progress in all areas of research.

Reproducible research and organized steps of computing not only help future researchers but also help publishers to go back and make any necessary changes before publishing a paper. For example, a researcher uses Python for analyzing and developing performance code in Java (related to research), and uses Tableau for making good-looking plots. Unfortunately, months later the researcher realizes there is a problem either with the work or the result. Without having a comprehensive workflow, are they able to validate the issue without making any errors or changing the complete process? Will other researchers be able to easily understand their new idea without having the good and complete source?

In our research, we used Python. Python is a simple language, installable from almost all different platforms, and powerful enough to deal with complex, experimental, significant data. Python supports functional programming, object-oriented programming, and meta-programming [Millman and Pérez, 2014][Demšar et al., 2013]. Due to excellent support for scripting tools written in other languages (like C, and R), Python is often used as an integration language for calling routines from a broad range of high-quality scientific libraries.

Python is used in a substantial amount of libraries. Python has been built in libraries for different purposes, such as database access, data compression and so on. In our research, we used Pandasa, NumPy, Matplotlib and scikit-learn. These libraries are designed to simplify the data analysis workflow. In this research we are going to use the Pandas library, which has a more attractive and practical statistical computing environment. We call Pandas as follows:

```
Import Pandas as pd
```

By adding pd. in front of our command, we can then use the Pandas library. With Pandas help, our data set arrives in tabular format, making it easier to explore. Because we initially didnt have a specific table with two dimensions, or observation and column names, we had to create this table first. The SEER data sample is shown in figure **??**.

With Pandas and the SEER registry guidebooks help, we were able to make a table. The first

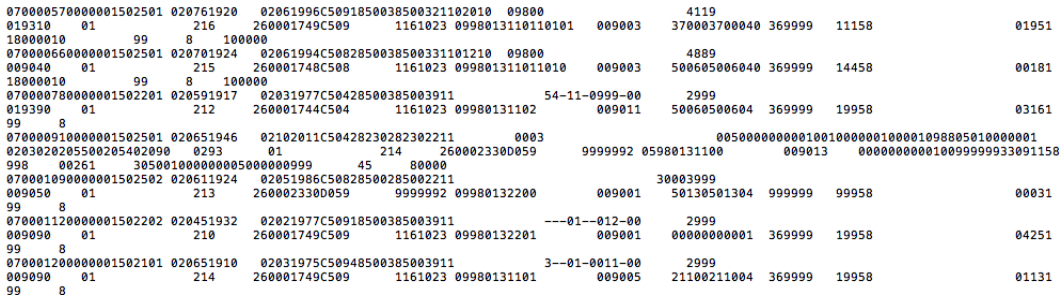


Figure 2.1: Sample of SEER raw data

Table 2.5: SEER Data

0	07000003	01	2
1	07000057	01	5
2	07000066	01	5
3	07000078	01	2

step was to find about the position of each observation and then to assign them to a relative column. For instance, having information related to Race and a patients marital status as seen below:

```
Import Pandas as pd

ColumnName=["PatientID","Race","MaritalStatus"]

data=pd.DataFrame([(line[0:8],line[19:21], line[18:19])
for line in open("BREAST_new.TXT","r"), columns= ColumnName)
```

and the result in a two-dimensional tabular format is shown in the table 2.5.

And as mentioned before, each of these codes has a description guide in the SEER Registry book. For instance, in our first and last examples, the patients are married. However, in the second and third examples, they are widowed. All have the same ethnicity, which is white. With Pandas help, we read our data set once and use it as many time as we wish. So far, we have read the data and have made it easily callable. Scikit-Learn is another library that is widely used in Machine Learning research. Scikit-Learn is excellent for the implementation of many supervised and unsupervised learning algorithms. Its easy to use and understand and can easily interface with other programs. Scikit-Learn is distributed under the BSD license, non-copy left license. (Also, it is Bare-bone design, for lowering the barrier the entity), moreover, it incorporates complied code for efficiency.

It provides reference implementation of different machine learning algorithms.[Pedregosa et al., 2011b] Numpy, another library, is used for data and model parameters. The view base modeling in Numpy minimizes copies and provides advance arithmetic operations[Walt et al., 2011].

Reproducibility is not just limited to computing complex code, but more importantly to cleaning the data. The cleaning and preparing of the data are the most time-consuming parts of data analysis. Unless we write and exact the same data, we can never achieve the same results by running the machine learning computations code. The first step is to clean the data. Based on the needs of the topic, cleaning may have to be repeated many times. In the next section, we will discuss Tidy data in detail. Its important to have an accurate, reliable way to provide information related to the necessary steps to clean the data.

One of the tool that help researcher is BitBucket; The researchers aware of the difficulty of Source Code control during the research duration, like store the project safely, modifying with the ability of keeping track of each step, be able to go back steps based on the project needs, and giving the chance to have experiment with new features without damaging the whole work. One question is that where so we have a plan to save our source code? Git Hub is one well known (write about Git Hub) But Git Hub is not the only option, BitBucket is another "BitBucket has been around for a long time, having been founded in 2008 and bought out in 2010 by Aussie tech giant Atlassian after having developed its own committed contingent of die-hard fans."

The Virtual machines designed to let the softwares running on top of the servers in order to use the specific needed hardware. Virtual Machines sits between Operating system and hardware, and its virtualize the server. Each Virtual machines runs a unique operating system, we can use different virtual machines, with different operating systems that all can be run on the same physical server. Each virtual machines has their own libraries.

Containers sit on top of server as well, and it host an Operating System. And contain libraries. In container, the libraries are read only and so do all the shared component. In contrast of VM that they can be as large as gigabytes the containers have megabyte sizes. Containers are fast and variety of containers can be put on top of a server. We can share containers and they are shareable in public and private cloud deployments. We do have less bug fixes, patches and etc. when we are working with containers, due to sharing a common operating system. In containers, the Operating system is virtualized, and shared Operating system is used, however in virtual machine hardware is virtualized then they are complicated in terms of system requirements.

Chapter 3

Data Processing

The performance of statistical learning algorithms such as logistic regression depends on training data. The data preparation is one of the crucial steps in training of the classifiers. In the next two sections, we describe our steps in data preparation based on the concept of TIDY data as described in [Peng, 2011, Baggerly and Coombes, 2009, Wickham, 2014, Taghva and Bozorgi, 2016]:

3.1 Patient Attributes

The raw data we used is the data repository as reported in "SEER RESEARCH DATA RECORD DESCRIPTION CASES DIAGNOSED IN 1973-2013" [NCI, 2016]. This repository contains 769,261 records with 134 attributes. Since the records cover various kinds of cancer, not all attributes apply to our work on breast cancer. Furthermore, there is a set of attributes that only applies to data collected after 1988. One such set used for this study was `EOD Tumor Size`, `EOD Extension`, `EOD Lymph Nodes`; the data for this set were collected from 1988 to 2003. The same data was collected after 2003 with different labels and positions (columns), namely, `CS Tumor Size`, `CS Extension`, and CS Lymph Node Involv, respectively. We used 18 attributes, as `described` in Table 3.1. The attributes patientId, COD, yearOfDiagnosis, and survivalMonths were not used as features for classification. However, survivalMonths, yearOfDiagnosis, and COD were used to label the two classes for binary classification.

3.2 Class Definition for Patients

The next step in data preparation and cleaning was to label records based on five year survivability according to direct method as outlined in [Parkin and Hakulinen, 1991]. It worth mentioning that

Table 3.1: The Eighteen Attributes Used in Our Experiments

Variable	Variable Definition	Values
patientIdNumber	uniquely identifies a patient	up to 8 digiits
race	two digit code race identifier	01-99, 01 for white,02 for black
maritalStatus	one digit code for marital status	1-9, 1 for single, 2 for married
behaviorCode	code for benign etc.	o-4,0 for benign,1 for malignant, etc.
grade	cancer grade	1-9, 1 for Grade I, etc.
vitalStatusRecord	alive or not	1-4, 1 for alive, 4 for dead
histologicType	microscopic composition of cells	4-digit code
csExtension	extension of tumor	2-digit code
csLymphNode	involvement of lymph nodes	2-digits code
radiation	radiation type code	0-9, for none, 1 for Beam, etc.
SEERHistoricStageA	codes for stages	0-9, 0 for in situ, 1 for localized
ageAtDiagnosis	First diagnosis age	00-130, actual age, 999 for unknown
csTumorSize	size in millimeters	000-888, 000 for no tumor
regionalNodesPositive	negative vs positive	00-99, number of positive nodes
regionalNodesExamined	positive, negative nodes examined	00-99, exact number
survivalMonths	number of months alive	000-998, 9999 for unknown
COD	Cause of Death	5-digit, 2600 for breast, 00000 alive
yearOfDiagnosis	This visit year	4-digit code

many of the studies on SEER data ignored this step [Bellaachia and Guven, 2006, Delen et al., 2005]. Consider the three patient records as shown in Table 3.2. There are four records for patient 1. The first record shows that the patient has survived 110 months from the visit on October of 2004. Based on this record, patient 1 will be labeled as `survived`. Patients 2 has survived 47 months from the date of first visit on January 2010. This patient will be marked as `ignore` and will not be used for training. Patients 3 and 4 are both deceased and the cause of death for both patients is breast cancer. Patient 3 has survived beyond five years, so she will be labeled as `survived`. Patient 4 is labeled as `not-survived`. We only keep the record of the first visit for each patient for training purposes. Finally we remove any record which has empty or unknown values in regionalNodesPositive, regionalNodesExamined, CSTumorSize, and EODTumersize. We net total of 338,596 patients of which 300,215 are labeled `survived` and 38,381 are labeled `not-survived`.

We want to point out that the number of `survived` data points are almost eight times the number of `not-survived` data points.

As mentioned in chapter 1, we first were interested in reproducing the experiments reported by Bellaachia and Guven [Bellaachia and Guven, 2006] in order to extend the work on the more recent SEER data. Unfortunately, neither the data sets nor the results could be reproduced, mainly due

Table 3.2: Four Patients Records

patientId	VSR	STR	monthOfDiagnosis	yearOfDiagnosis	COD
1	1	110	10	2004	00000
1	1	85	11	2006	000000
1	1	15	9	2012	00000
1	1	14	10	2012	00000
2	1	47	1	2010	00000
2	1	9	3	2013	00000
2	1	8	5	2013	00000
3	4	96	3	2005	2600
3	4	46	5	2009	2600
4	4	23	7	2006	2600
4	4	22	8	2006	2600

to the lack of exact and explicit instructions for data preparation. This is very common in scientific literature and major obstacle in reproducible research [Peng, 2011, Baggerly and Coombes, 2009]. Following [Wickham, 2014], the data preparation must include four components:

1. The raw data

2. A TIDY data set

3. A code book describing each variable and its value

4. An explicit and exact recipe from which one needs to produce components one and two from component one.

Chapter 4

Machine Learning Tools

The primary goal of many artificial intelligence (AI), machine learning, and data science is the discovery of new facts from data based on statistical and logical methods. The secondary goal of these disciplines is to communicate the new facts [Aumann et al., 2003][Dhar, 2013]. Of course, the discovery should be valid and reproducible. Unfortunately,many reported discoveries are not reproducible due to sloppy data preparation and clean up [Editors, 2012][Economist, 2013].

Typically, many projects use data sets that were not necessarily collected for those projects. For example, SEER database is built for summarizing cancer data and not survivability prediction. The survivability prediction problem is a binary classification with uneven distribution of data points [Vapnik, 1995][Xiao et al., 2009]. In order to prepare SEER data for binary classification, we must first decide how to assign data points to each class. According to Parkin and Hakulinen, a well-accepted methodology in predicting patient survival involves summarizing and analysis [Parkin and Hakulinen, 1991]. The most widely used metric involves calculating the percentage of patients alive after five years, using a direct method as outlined by Parkin and Hakulinen [Parkin and Hakulinen, 1991]. Chapter 3 gives our detailed explanation of our approach to data assignments based on direct method.

One of the earliest and most cited work on survival predictability with machine learning tools are the experiments reported by Delen et al. [Delen et al., 2005]. These experiments identified decision tree as the best predictor, compared with artificial neural networks (ANN) and logistic regression. A follow-up set of experiments by Bellaachia and Guven [Bellaachia and Guven, 2006] reported similar results that decision tree was superior to naive Bayes and ANN. Neither work was reproducible research, as there are no code book description of recipes on data preparation and algorithms. Furthermore, it is not clear which methods (direct vs actuarial) that both studies used

to identify patient five year survival status of patients.

Both of the above-mentioned studies were conducted using SEER data. Closely related studies on lung cancer, also using SEER data, found that decision tree was the best predictor [Agrawal et al., 2012]. This study further identified the importance of two out of 11 features when predicting survivability. In another interesting and related study using SEER data, Zolbanin et al.[Zolbanin et al., 2015] based the prediction of survivability on comorbidity of cancers, for example, breast and prostate cancer.

Salma et al. [Salama et al., 2012] performed comparison studies on Wisconsin Breast Cancer (WBC) database [Lichman, 2013], and reported that Multi-Layer Perception (MLP) was superior to decision tree for that database. It is important to point out that WBC collects a different set of features for breast cancer than does SEER. It is also worth mentioning that another study by Christobel and Sivaprakasam [Angeline Christobel. Y, 2011] identified the Support Vector Machine (SVM) as the best predictor for the WBC database. Finally, we want to draw attention to binary classification based on missense mutation in genome [Wei and Dunbrack Jr, 2013].

The patient survival summarizing and analysis is a well accepted methodology [Parkin and Hakulinen, 1991]. The most widely used metric is the calculation of percentage of patients alive after five years by direct method as outlined in [Parkin and Hakulinen, 1991]. One of the earliest and most cited work on survival predictability with machine learning tools is the experiments reported by Delen et al. [Delen et al., 2005]. These experiments identified decision tree as the best predictor compared with artificial neural networks (ANN) and logistic regression. A follow up set of experiments by Bellaachia et al. [Bellaachia and Guven, 2006] reported similar results that decision tree was superior to Naive Bayes and ANN. It is not clear which method (direct vs actuarial) both studies use to identify patient five year survival status.

Both of the above-mentioned studies were conducted using SEER data. Closely related studies on lung cancer, also using SEER data, found that decision tree was the best predictor [Agrawal et al., 2012]. This study further identified the importance of two out of 11 features when predicting survivability. In another interesting and related study using SEER data, Zolbanin et al.[Zolbanin et al., 2015] based the prediction of survivability on comorbidity of cancers, for example, breast and prostate cancer.

Salma et al. [Salama et al., 2012] performed comparison studies on Wisconsin Breast Cancer (WBC) database [Lichman, 2013], and reported that Multi-Layer Perception (MLP) was superior to decision tree for that database. It is important to point out that WBC collects a different set

of features for breast cancer than does SEER. It is also worth mentioning that another study by Christobel and Sivaprakasam [Angeline Christobel. Y, 2011] identified the Support Vector Machine (SVM) as the best predictor for the WBC database. Finally, we want to draw attention to binary classification based on missense mutation in genome [Wei and Dunbrack Jr, 2013].

The specific machine learning tools used in these experiments are binary classification techniques. In general, we use features such as stages of cancer to help with this classification. The NCI collects a large number of attributes for each cancer patient. Most researchers use a subset of these attributes as features for binary classification. A fundamental question associated with these experiments is the test of significance. In other words, how many of the selected features can be eliminated without degrading the classifiers.

In our initial studies [Taghva and Bozorgi, 2016], we were interested in a predictive model which estimates the odds of a female subject surviving breast cancer based upon the subject attributes. The performance of logistic regression was compared to other machine learning tools such as Naive Bayes and decision trees. We identified logistic regression as a strong candidate for classification task based on F1 measure.

4.1 Naive Bayes

This section provides a brief introduction to binary classification with naive Bayes, logistic regression, and decision tree. In general, classification starts with a vector of features $\overrightarrow{X} = (x_1, x_2, \ldots, x_n)$ which can serve as a template for each data point in the data set. We wanted to build a binary classifier Y that predicts survivability. Essentially this construction was based on the characteristics of the initial data set, in this case, the SEER database.

The simplest learning algorithm is the naive Bayes [Friedman et al., 1997]. This classification technique relies on Bayes' rule that the the outcome of an event A can be predicted from evidence B:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{4.1}$$

In practice, there are more events (or features) that contribute to this equation. The word `naive` stems from the fact that features x_i's are assumed to be independent of each other. Notice that the numerator is the joint probability $P(A, B)$. For a more general vector of features $\overrightarrow{X}$, this joint probability for a new data point to be classified is simply the product of the individual

probabilities:

$$P(X_1, X_2, \ldots X_n) = P(X_1) \cdot P(X_2), \ldots, \cdots P(X_n) \tag{4.2}$$

4.2 Decision Trees

The decision tree [Quinlan, 1986a] uses a tree structure to classify the data points. The leaves represent classes (survived or not), and branches represent conjunction of features from the feature vector. This is a popular method as it represents a conceptual thought process that one can start at the root and make conclusions at the leaves.

Decision Tree is a hierarchical acyclic graph that start with one node called rootGehrke et al. [1999]. Each node describes a variable and edge represent the decision, and depending on the assignment, each leaf has a distinct meaning. Each node can have as many edge as possible. The most common tree is used for binary classification that each node just has two branches. There are different methods for creating a Decision tree, one approach is to create a big tree and reach out on the best tree by pruning (eliminating the useless nodes) the nodes. On the other hand, generate a introduction algorithm to guide us to split the data into the finite subsets, Instead of creating many different trees and pick the best one.

There are different decision tree algorithms available like Iterative Dichotomizer3(ID3)-1986 [Quinlan, 1986b], C4.5-1993 [Quinlan, 2014], Chi-square Automatic InteractionDetection (CHAID)-1980 [Kass, 1980], Classification and Regression Tree (CART)-1984 [De'ath and Fabricius, 2000], Quick-Unbiased-Efficient Statistic Tree(QUEST)-1997 [Loh and Shih, 1997], GUIDE-2002, Classification Rule with Unbiased Interaction Selection and Estimation(CRUISE)-2001 [Kim and Loh, 2001] and Conditional inference tree(CTREE)-2006 [Hothorn et al., 2006]. Probably the most popular one in machine learning are ID3 (and its successor), C4.5 and CART. Quinlan develop the ID3 Decision Tree in 1986 at University of Sydney, and improved the tree in 1993 [Quinlan, 2014], and named it C4.5. Ross Quinlan has various publications, he was actively works on the Decision Tree algorithms, in the late 80s he developed ID3.

4.2.1 Iterative Dichotomizer 3(ID3)

Quinlan considered the theory of Shannon as the base of the ID3 and C4.5 algorithms. Shannon Theory is based on Information Theory; In general Information Theory is based on statistic and

probability, the useful information created by measuring the distribution associated to the random variable that it called entropy. A entropy can be associated of the measure information between single random variable or between two random variables. The Shannon entropy shows in equation 4.3.

$$H(s) = -\sum_{j=1}^{c}(p(j)\log_2 p(j) \tag{4.3}$$

That p(j) is represent the probability of the j-th class, and C represent the number of classes of the output variable. The simple example below is for better understanding the Shannon entropy .

Table 4.1: Patient Information

Patient Id	Type of Cancer	Doctor Visit	Surgery	Chemotherapy	Survivability
1	Breast Cancer	Regularly	Yes	No	No
2	Breast Cancer	Regularly	Yes	Yes	No
3	Colon Cancer	Regularly	Yes	No	Yes
4	Prostate Cancer	Often	Yes	No	Yes
5	Prostate Cancer	Rarely	No	No	Yes
6	Prostate Cancer	Rarely	No	Yes	No
7	Colon Cancer	Rarely	No	Yes	Yes
8	Breast Cancer	Often	Yes	No	No
9	Breast Cancer	Rarely	No	No	Yes
10	Prostate Cancer	Often	No	No	Yes
11	Breast Cancer	Often	No	Yes	Yes
12	Colon Cancer	Often	Yes	Yes	Yes
13	Colon Cancer	Regularly	No	No	Yes
14	Prostate Cancer	Often	Yes	Yes	No

In this example the attribute's values is as follow:

Type of Cancer= { Breast, Colon , Prostate }

Doctor Visit={Regularly,Often,Rarely}

Surgery={Yes,No}

Chemotherapy ={Yes,No}

Survivability={Yes,No}

The Shannon entropy calculated as shown in equation 4.4.

$$H(S) = \frac{-9}{14} \times \log_2(\frac{9}{14}) - \frac{5}{14} \times \log_2 \frac{5}{14} = 0.94 \tag{4.4}$$

In addition, Quinlan uses the Concept Learning System (CLS) algorithm as anther base for ID3 algorithm. ID3 is a supervised learning algorithm and it use the training data to create a tree. The produced tree is used to classify the testing data sets.

In ID3 model a tree generates based on the categorical input and output. It goes through all the categorical attributes, hence generate a wide and shallow tree. At the beginning ID3 algorithm assign one split for every node (attribute) where these splits create branches of the categorical attributes. Then with Information Gain method the best split for ID3 Decision tree measured and evaluated. Previous two steps process recursively applied to the new branches. [Suknovic et al., 2012] Information Gain method calculated based on equation 4.5.

$$H(U,S) = \sum_{i=1}^{K}(\frac{\mid S_i \mid}{\mid S \mid}E(S_i)) \tag{4.5}$$

Where, the H(U,S) is represent the expected entropy of the input U that has K categories. $E(S_i)$ is the entropy as well that represent the output attribute.

Information Gain represent as I(U,S) in equation 4.6 as defined in [Suknovic et al., 2012].

$$I(U,S) = H(S) - \sum_{j=1}^{K}(s_j * H(s_j)) \tag{4.6}$$

The results of the Information gain for previous example calculated as shown in equations 4.7, 4.8, and 4.9 and.

$$H(S) = \frac{9}{14}\log(\frac{9}{14}) + \frac{5}{14}\log(\frac{5}{14}) = 0.94 \tag{4.7}$$

$$\begin{aligned} I(TypeOfCancer,S) = H(S) - \frac{5}{14} \times H(S_{BreastCancer}) - \frac{4}{14}H(S_{ProstateCancer}) \\ - \frac{5}{14} \times H(S_{ColonCancer}) = \\ 0.94 - \frac{5}{14} \times 0.9710 - \frac{4}{14} \times 0 - \frac{5}{14} \times 0.9710 = 0.246 \end{aligned} \tag{4.8}$$

$$\begin{aligned} H(S_{Breastcancer}) = \frac{2}{5}\log(\frac{2}{5} + \frac{3}{5}\log(\frac{3}{5} = 0.971 \\ H(S_{ProstateCancer}) = \frac{4}{4}\log(\frac{4}{4} = 0 \\ H(S_{colon}) = \frac{3}{5}\log(\frac{3}{5} + 5 \times \log(\frac{2}{5} = 0.971 \end{aligned} \tag{4.9}$$

The results of the Information Gain for the other attributes of the example are shown in Table 4.2.

Table 4.2: Patient Information

I(Chemotherapy,S)	0.048
I(Doctor Visit,S)	0.0289
I(Surgery,S)	0.1515

ID3 is not the perfect decision tree due to the different limitations, it can just work with categorical data , and it's not designed to work with numerical data. In addition, ID3 is sensitive to features with large individual number of values. For instance unique Patient Id(Or Social Security Number), these unique values can cause the low conditional entropy value.

4.2.2 C4.5

Quinlan worked on the ID3 problems and discover a new method in 90s. He used the gain ratio method to improve the ID3. The improved ID3 by Quinlan named C4.5 Decision Tree [Hssina et al., 2014]. In C4.5 the Gain ratio method used to calculate the splitting attributes. It can work with numerical and categorical input attributes. C4.5 Decision tree also has the feature to work with unknown values. If an attribute has an unknown value/values the C4.5 manage those values by evaluating the gain ratio. In the first step,in C4.5 Decision Tree , all possible binary splits for all numerical attributes are considered, the splits are always binary in this model. Then the best split selected by evaluating the gain ratio measurement. These two steps recursively applied to all attributes[Suknovic et al., 2012]. Until reached the stop point of the tree. The "gain ratio" calculation is show in equations 4.10, and 4.11.

$$G(U,S) = \frac{I(U,S)}{SI(U,S)} \tag{4.10}$$

$$SI(U,S) = -\sum_{i=1}^{K}(\frac{\mid S_i \mid}{\mid S \mid} \times \log(\frac{\mid S_i \mid}{\mid S \mid}) \tag{4.11}$$

The c4.5 designed to work with categorical and numerical attributes, in C4.5 algorithm process the categorical attributes generates the multiway splits ,however the numerical attributes always generate binary splits.

4.2.3 Classification and Regression Tree (CART)

CART is a decision tree that can evaluate classification and regression. It can only work with binary splits and produce narrow and deep tree. In this tree, all possible splitting will be generating (can be numerical attributes or categorical attributes). The best splits be selected based on the evaluation Measure method, that this Evaluation Measure can be based on different method like Gini, Twoing, and order Twoing. these two steps recursively repeated until stopping criteria has been reached.

The Gini measure evaluation [Suknovic et al., 2012] is calculated as shown in equations 4.12 and 4.13.

$$G(U,S) = j(S) - P_L \times j(S_L) - P_R \times j(S_R) \tag{4.12}$$

$$j(S) = \sum_{j,i} P(\frac{j}{S})P(\frac{i}{S}), i \neq j \tag{4.13}$$

The smaller the value of the Gini Index shows the better split. Gini method doesn't designed to work with the data with the wildly spread domain of the target, in those cases Towing criteria can be used as shown in equation 4.14.

$$TwoingCriteria(t) = \frac{P_L P_R}{4}((\sum(\mid P(\frac{i}{t_l})P(\frac{i}{t_R} \mid))^2) \tag{4.14}$$

$P(\frac{i}{t}$ is the probability of the fraction of class i at node [Ture et al., 2009]. The P_L is shown the probability of a case to be at the left branch and P_R is the probability that a case shown in the right side of the tree. In addition the Mean Square Error is used in this tree to achieve the best splits.It also designed to work with missing values.

4.2.4 Chi-square Automatic InteractionDetection (CHAID)

CHAID is a decision tree algorithm that like ID3 is just work with categorical data. CHAID designed to find the most significant attributes based on the Chi-Square statistic. In the first step, the most significant split for each attributes produced by generating two tables for every pair of categorical attributes, and evaluate the Chi-Square for each table. For each pair the results compared with the threshold, the two least significant different categories will be merged, and this step repeat again for the new pairs.Then each remaining category generated from two or more original categories. In this step the most significant categories will be found by dividing

the component category into all possible two categories division [Mohankumar et al.]. The most significant split implemented by using Chi-Square statistic, also the Benferroni multiplier used for finding compound categories.These steps repeated recursively until the stopping criteria has been reached. Chi-Square is used for classification and prediction [Pereira et al., 2017]. calculated as shown in equation 4.15.

$$Chi - Square = \frac{(actual - excpected)^2}{excpected} \tag{4.15}$$

4.2.5 Quick Unbiased Efficient Statistic Tree

quest Quest is a classification tree that works with numerical and categorical input attributes . In this method, For numerical input attributes ANOVA f-test evaluated and for categorical variables Chi-Square.

In addition, Benferroni adjustment is used in this algorithm to make sure the bias is insignificant.All categorical variables transform to numerical variable with "discriminant coordinate or canonical variate " CrimCoord transformer. The CrimCoord is used in the Quest and CRUISE decision tree, to convert numerical data to categorical data. Then the split point will be found in the selected numerical attributes. In addition for finding the best split 2-means(group classes to two super classes) is applied.The previous steps recursively repeats until the stop point reached.

4.2.6 Classification Rule with Unbiased Interaction Selection and Estimation

CRUISE is another supervise learning algorithm. The CRUISE algorithm is the FACT and QUEST improved algorithm. Hyunjoong Kim and Wei-Yin Loh from Yonsei University, Korea and University of Wisconsin-Madison, USA designed this algorithm in 2001 [Kim and Loh, 2001]. It designed to use pruning and also compatible to work with missing values. In CRUISE Decision tree the attribute selection is done by a Chi-Square testing and normalizing with Peizer-Pratt transformation. The best attributes have maximum normalized value from Peizer-Pratt transformation.The different tables generated for each numerical or categorical attributes. It generate K tables for pair of categorical attributes, and 4 tables for each pair of numerical attributes. All categorical variables transform to numerical variable with CRIMCOORD transformer. CRIMCOORD transformers all numerical attributes to categorical attributes.Then for finding the best splits the Box-Cox transformation used before applying LDA, LDA is designed to work on the normal distribution data.The previous steps recursively repeat until the stop point reached.

4.2.7 Conditional inference tree(CTREE)

Another tree that is designed to work with categorical and numerical data is CTREE. It used for analyzing the classification and regression. The most significant attributes selected based on the H_0 hypothesis. If the the minimum adjusted $p - value$ is smaller than the threshold or if the H_0 is not rejected then the tree splitting will stop. In this model, the best split is selected with two samples test linear statistics. The most significant attributes chosen with Permutation(randomization) test calculation. The significant split evaluated by permuting the response under null H_0 of "independence between covariates and response variable" [Hothorn et al., 2006]. CTREE decision tree doesn't work with missing values.

4.2.8 Comparison between different Decision tree algorithm

In general, in different Decision Tree algorithms finding significant attributes is the first challenge. Each algorithm used different methods to find the significant attributes, like ANOVA f-test, Chi-Square test, Permutation test. In creating splits process, the Binary, QDA and LDA methods were used. In addition , for evaluating the efficient splits, Information Gain, Gain Ratio, Gini Index, Twoing, Ordered Twoing, Chi-Square test, AUC, Mean Square Error (MSE) and Permutation two-sample test methods were used.

The other challenge is to find the stopping point of the tree, Rokach and Maimn in 2008 solved this issue by considering the Pure node, Maximum tree depth or Minimum evaluate Split threshold. CHAID uses $p - value$ to measure the desirable of a split, while CART uses the reduction of an impurity measure. CART is designed to generates only binary splits, while CHAID searches for multi-way splits. In ID3 and C4.5, for each categorical attributes there is just one possible split. If there is N possible categories, then 2k-1-1 binary splits can be generated. In CHAID algorithm, the similar categories can be grouped and produce neither multiway nor binary. The Binary splits used in CART, GUIDE and CTREE, the multiway splits used by ID3 and C4.5 and the CHAID used significant split methods. The C4.5 and CART generates splits based on the numeric attributes. QUEST use QDA and CRUISE generate splits with linear analysis. Every Decision tree has stopping criteria. Some method considers maximum predefined depth of the tree as the stopping point, some algorithms suggested to grow the tree and afterward prune the not significant nodes to guarantee that the most significant tree as shwn in Table 4.3

Table 4.3: Decision Tree comparison

Decision Tree	Missing Values	Splits per node	Unbiased splits
C4.5	Yes	$\geq$2	No
CTREE	No	2	Yes
CRUISE	Yes	$\geq$2	Yes
CART	Yes	2	No

Figure 4.1: Galton's Data

`Father's Height`	`Son's Height`
68.0	66.5
69.0	72.1
78.5	75.3
75.5	79.2

4.3 Linear Regression

One of the oldest statistical methods for inferencing is `Linear Regression` discovered by Francis Galton in 1886 [Galton, 1886]. Galton was interested in estimating the son's height based on father's height. The assumption being that there is a linear relation between the son and father heights. If we let x and y represent the father and son heights respectively, then the following equation could represent the desired relationship:

$$y = \beta_1 x + \beta_0 \tag{4.16}$$

In this equation, β_1 and β_0 represent slope and intercept of the line, respectively. In general, we are interested in identifying slope and intercept values that minimizes the error. These values are obtained based on observed data. Consider a snippet of data from Galeton's data as represented in 4.1.

The Python code in 4.1, represents the graph in 4.2 as four red points. In addition, one can guess that the regression lines $y = 0.9x + 5.8$ (in blue) or $y = 0.9x + 5.0$ (in green) may be a good fit for this data as it is displayed in 4.2.

Listing 4.1: Python code for regression lines

```
\import numpy as np
\import matplotlib.pyplot as plt
X = [68, 69, 78.5, 75.5]
```

```
Y = [66.5, 72.1, 75.3, 79.2]
t = np.linspace(60,85,20)
plt.plot(X, Y, 'ro', t, 0.9*t + 5.8, 'b ', t,0.9*t + 5.0, 'g ')
plt.xlabel("Father's Height")
plt.ylabel("Son's Height")
plt.title("possible regression lines")
plt.show()
```

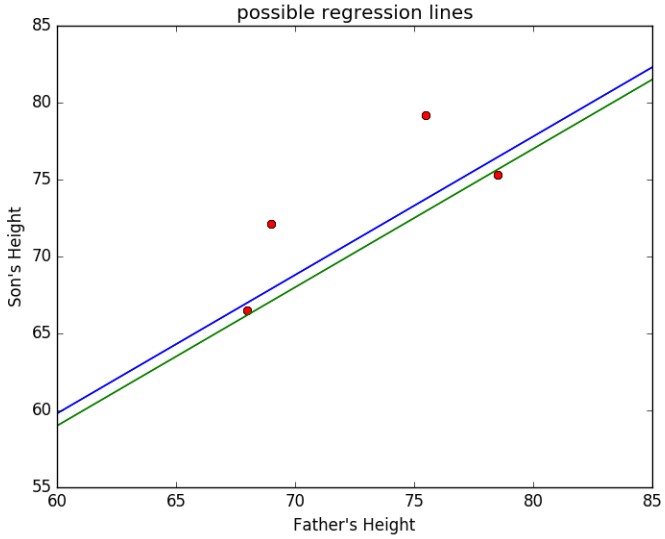


Figure 4.2: Display of the Possible Regression Lines

The regression line can be used to predict son's height given father's height. Conventionally, for a given value x (father's height), there are two values for son's height, the actual value known as `observed` and value obtained from the regression line known as `fitted`. we use y and $\hat{y}x$ to denote observed and fitted values, respectively. for example for the point $68, 66.5$ and regression line $y = 0.9x + 5.0$, the observed value is 66.5 and the fitted value is $y^{=}0.9 * 68.0 + 5.0$ which is 66.2. The difference between these two values $66.5 - 66.2 = 0.3$ is the error. Typically, we want to minimize this error based on the proper values of β_1 and β_0.

For a given list of data points $(x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)$, we would like to calculates the slope and intercept of the regression line by minimizing the `Square Error`, SE defined by:

$$SE = \sum_{i=1}^{n} (y_i - (\beta_1 x_i + \beta_0))^2 \tag{4.17}$$

Let $\bar{x}$ denote the mean of, namely, $\bar{x} = \frac{x_1+x_2 \ldots x_n}{n}$ or equivalently $(x_1 + x_2 \cdots + x_n) = n\bar{x}$, then we proceed with the following to find values 0f β_1 and β_0 to minimize the SE.

We start by expanding SE as follows:

$$SE = (y_1^2+y_2^2 \cdots+y_n^2)-2\beta_1(x_1y_1+x_2y_2 \cdots+x_ny_n)-2\beta_0(y_1+y_2 \cdots+y_n)+\beta_1^2(x_1^2+x_2 \cdots+x_n^2)+2\beta_1\beta_0(x_1+x_2 \cdots+x_n)$$

This is equivalent to:

$$SE = n\overline{y^2} - 2n\beta_1\overline{xy} - 2n\beta_0\overline{y} + \beta_1^2\overline{x^2} + 2n\beta_0\beta_1\overline{x} + n\beta_0^2 \tag{4.19}$$

To minimize equation 4.19, we take two partial derivatives with respect to β_0 and β_1:

$$\frac{\partial SE}{\partial \beta_1} = -2n\overline{xy} + 2n\beta_1\overline{x^2} + 2n\beta_0\overline{x} = 0 \tag{4.20}$$

$$\frac{\partial SE}{\partial \beta_0} = -2n\overline{y} + 2n\beta_1\overline{x} + 2n\beta_0 = 0 \tag{4.21}$$

by dividing both side of these two equations by $2n$, we get two equations in 4.22, and 4.23:

$$-\overline{xy} + \beta_1\overline{x^2} + \beta_0\overline{x} = 0 \tag{4.22}$$

$$-\overline{y} + \beta_1\overline{x} + \beta_0 = 0 \tag{4.23}$$

We can rewrite these two equations as equations 4.24, and 4.25.

$$\beta_1\overline{x^2} + \beta_0\overline{x} = \overline{xy} \tag{4.24}$$

$$\beta_1\overline{x} + \beta_0 = \overline{y} \tag{4.25}$$

From equation 4.25, we observe that the point $(\overline{x}, \overline{y})$ lies on the regression line. Also, if we divide the equation 4.24 by $\overline{x}$, we get the equation 4.26 which implies that the point $(\frac{\overline{x^2}}{\overline{x}}, \frac{\overline{xy}}{\overline{x}})$ lies on the regression line.

With these two points, we can calculate slope and intercept of the regression as shown in equations 4.26, and 4.27.

$$\beta_1 = \frac{\overline{y} - \frac{\overline{xy}}{\overline{x}}}{\overline{x} - \frac{\overline{x^2}}{\overline{x}}} \tag{4.26}$$

$$\beta_0 = \overline{y} - \beta_1 \overline{x} \tag{4.27}$$

Mathematically, the slope of this regression line is equivalent to equation 4.28.

$$cor(y, x)\frac{sd(y)}{sd(x)} \tag{4.28}$$

We can revisit the four points in 4.1 in order to compute the slope and intercept of the regression line. Based on equation 4.26 and 4.27, we obtain 0.835 and 12.53 for slope and intercept, respectively. The four point and the regression line is shown in 4.3.

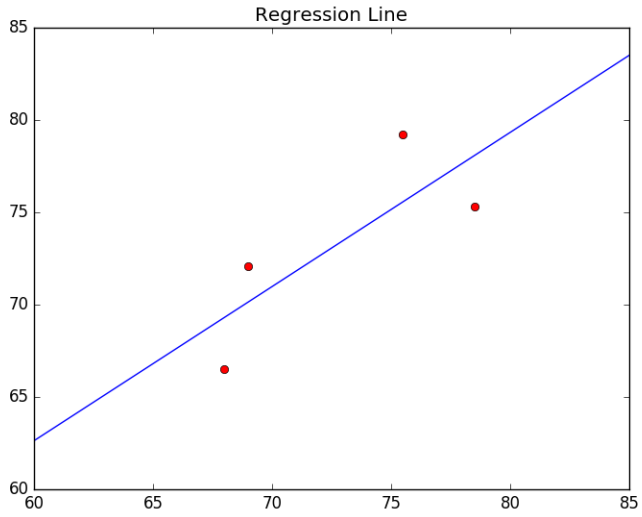


Figure 4.3: Display of the Regression Line

In practice, the analytical solution is hard when the number of observed data and features are high. In the next section, we will describe numerical method of `gradient descent` as is commonly used in machine learning applications.

4.3.1 Gradient Descent

In order to minimize the error in prediction as described in equation 4.17, one can employ the method of gradient descent to approximate the minimum point on the error curve. In what follows, we describe this method using derivatives, or more precisely partial derivatives.

Figure 4.4: Computing Example for Gradient Descent

$$
\begin{aligned}
y &= x^2 \\
y\prime &= 2x \\
at\ x = 1, y = 1, y\prime &= 2 \\
tangent\ line\ is : y &= m(x - x_1) + y_1 = 2x - 1 \\
at\ x = 1/2, y = 1/4, y\prime &= 1 \\
tangent\ line\ is : y &= x - 1/4 \\
at\ x = 0, y = 0, y\prime &= 0 \\
tangent\ line\ is : y &= 0
\end{aligned}
\tag{4.29}
$$

Listing 4.2: Python code for Gradient

```
import numpy as np
import matplotlib.pyplot as plt
X = [1, 0.5, 0]
Y = [1, 0.25, 0]
t = np.linspace(-3,3,200)
plt.plot(X, Y, 'ro', t, t**2, 'y ', t,2*t   1, 'b ', t, t  0.25, 'b ',
         t, 0*t, 'g ')
plt.xlabel("x values for parabola")
plt.ylabel("y values for parabola")
plt.title("Gradient Descent Example")
plt.slim(-3,3)
plt.ylim(-3, 3)
plt.show()
```

Assume a parabola function $y = x^2$, it is clear that the minimum point on this parabola is the point $(0, 0)$. Equivalently, the slope of the tangent line at this point is 0.0. The derivative of this equation is $y\prime = 2x$. In the equation 4.4, we compute the slope of the tangent line at three different points.

In gradient descent, we pick a point at random (more precisely, the slope), say $x = 1$. We calculate the slope of the tangent line. In the next step, we pick a point based on the calculated slope by subtracting a small value from current x, say $x = 1/2$. We continue the process and pick the next value say, $x = 0$. The idea is that we take steps toward the minimum point on the curve. The Python code and display of the example are shown in listing 4.2 and figure 4.5, respectively.

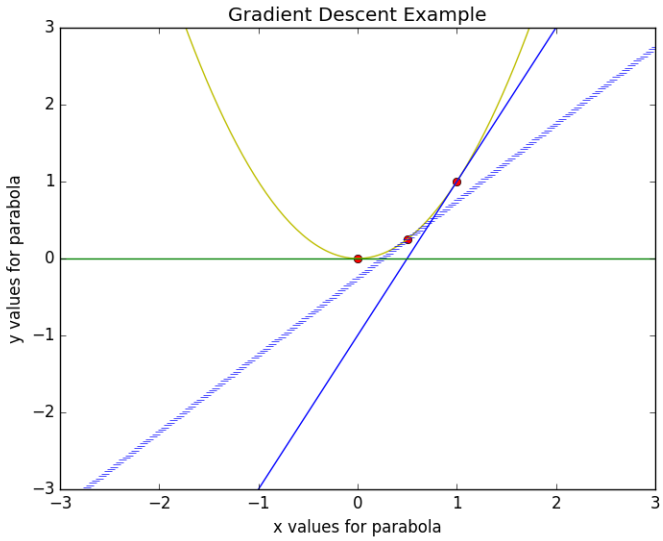


Figure 4.5: Display of the Possible Gradient Descent

Figure 4.6: Galton's Data

Father's Height	Son's Height
68.0	66.5
69.0	72.1
78.5	75.3
75.5	79.2
65.3	66.2
71.3	68.4
58.4	59.2
59.6	57.8
54.5	55.1

To make gradient descent more precise, we start with a hypothetical set of son's and father's height as shown in 4.6.

A common approach to normalizing data is `min-max` which normalizes according to equation 4.30 and coded in 4.3.

$$\frac{x - min}{max - min} \tag{4.30}$$

Listing 4.3: Python code for min-max

```
def min_max(list):
```

Figure 4.7: Normalized Galton's Data

Father's Height	Son's Height
0.56	0.47
0.6	0.71
1.0	0.84
0.88	1.0
0.45	0.46
0.7	0.55
0.16	0.17
0.21	0.11
0.0	0.0

```
min_value = np.min(list)
max_value = np.max(list)
new_list = [(x    min_value)/(max_value    min_value) for x in list]
print new_list
return new_list
```

The result of this normalization is shown in figure 4.7.

For a regression line $y = mx+b$ and an observed point (x_i, y_i), we denote the predicted value by $\hat{y}_i = mx_i + b$. The error $SE = \sum(y - \hat{y})^2 = (y - (mx + b))^2$ is defined as the sum of the differences between the observed and predicted values over the training data. The gradient is defined as the partial derivatives of SE with respect to m and b as defined in equations 4.32 and refeq:b-gradient.

$$b - gradient = \frac{\partial SE}{\partial b} = -2\sum_{i}^{n}(y_i - (mx_i + b)) \tag{4.31}$$

$$m - gradient = \frac{\partial SE}{\partial m} = -2\sum_{i=1}^{n} x_i(y_i - (mx_i + b)) \tag{4.32}$$

We can simplify these gradient as displayed in equations 4.33 and 4.34.

$$-2X(Y - \hat{Y}) \tag{4.33}$$

$$-2(Y - \hat{Y}) \tag{4.34}$$

The algorithm proceeds according to the following steps:

Figure 4.8: Gradient Descent Calculations

b	m	x	y	$\hat{y}$	SE	$-2(Y-\hat{Y})$	$-2X(Y-\hat{Y})np$
0.37	0.87	0.56	0.47	0.86	0.15	0.77	0.43
		0.6	0.71	0.89	0.033	0.36	0.22
		1.0	0.84	1.14	0.02	0.27	0.24
		0.88	1.0	1.14	0.02	0.27	0.24
		0.45	0.46	0.76	0.091	0.60	0.27
		0.7	0.55	0.98	0.18	0.86	0.60
		0.16	0.17	0.51	0.12	0.68	0.11
		0.21	0.11	0.55	0.20	0.89	0.19
		0.0	0.0	0.37	0.14	0.74	0.0
					0.95	5.44	2.3

1. initialize m and b randomly. (i.e. [m,b] = np.random.rand(2))

2. calculate the gradients based on the equations 4.33 and 4.34.

3. Update m and b according to a learning step α and gradients as in equations 4.35 and 4.36.

4. repeat the the two previous steps until the error reaches a stable state.

$$b = b - \alpha * \frac{\partial SE}{\partial b} \tag{4.35}$$

$$m = m - \alpha * \frac{\partial SE}{\partial m} \tag{4.36}$$

The learning step α is obtained experimental. Typical values are 0.1, 0.01, or 0.001. In general, It is hard to obtain an ideal error, so the repeating steps in the algorithm runs in increment of thousands. Table reffig:regression-calc shows the process for our example data.

At this step, the error is 0.95. If we set our learning step to 0.01, the gradient values of m and b will be updated to according to the equations 4.38 and eq:m-gradient3.

$$m = m - \alpha * \frac{\partial SE}{\partial m} = 0.87 - 0.01 * 2.3 = 0.847 \tag{4.37}$$

$$b = b - \alpha * \frac{\partial SE}{\partial b} = 0.37 - 0.01 * 5.44 = 0.316 \tag{4.38}$$

The algorithm repeats the calculation as it is done in 4.8 with the updated values of m and b. If we run this process for one thousand times we arrive at $SE = 0.0167$, $m = 0.847$, and $b = 0.127$, respectively. The final regression line is displayed in Figure 4.9.

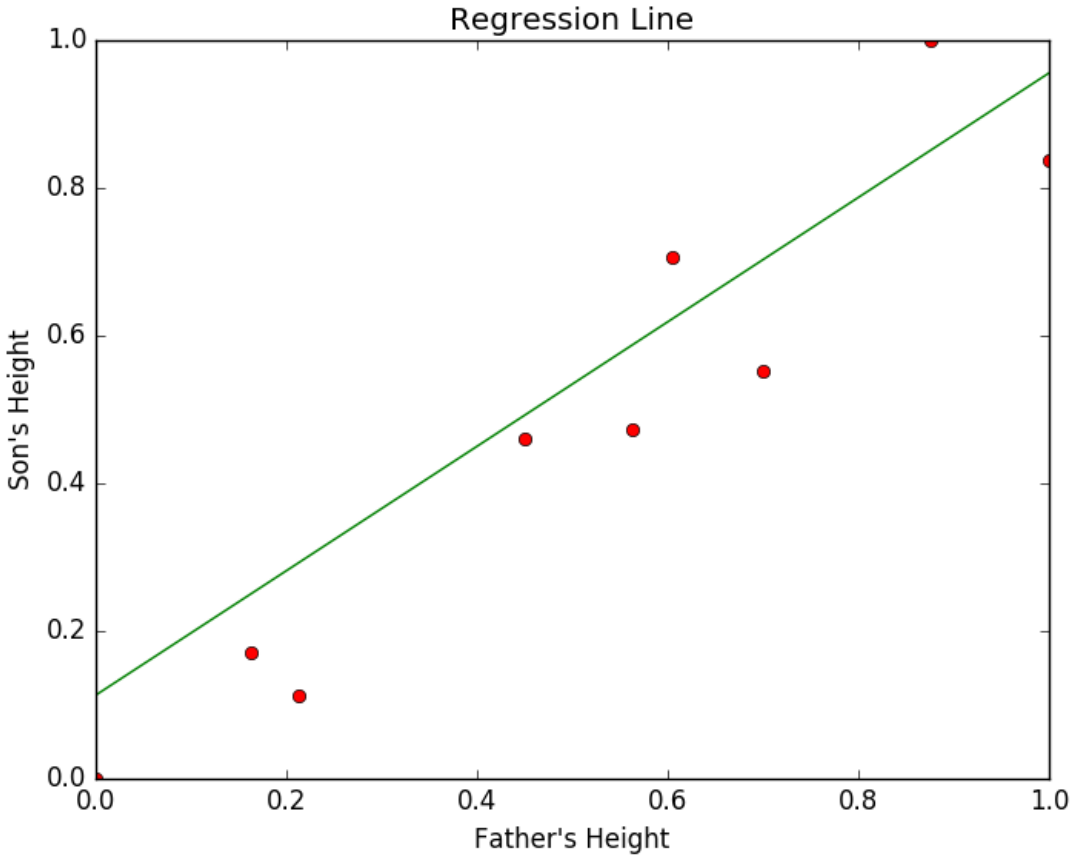


Figure 4.9: Display of the Final Regression Line

4.4 Logistic Regression

The Least Square Method was published by Adrian-Marie Legendre in 1805; however, Carl Fredrick Gauss had used it previously. In fact, he developed the Least Square Method in 1795 at the age of 18. Gauss's Least Square Method was first used in astronomy by an Hungarian astronomer. The astronomer had used the method to monitor Ceres before it became lost for 40 days in January

1802. He used the 24-year-old Gauss Method instead of the more complicated Kepler nonlinear equation method. In 1809, Gauss published the Least Square Method; The [Gauss, 1877]credit for its discovery went to Legendre(1805) [Legendre, 1805] and Gauss(1809) [Gauss, 1809].

The Logistic Regression Model was developed in 1958 by David Cox [Cox, 1958]. In this model, the probability of a binary response is based on one or more independent variables. Regression analysis is used for estimating the relationship between independent variables, which are also known as predictors. Regression analysis is used to understand the relationship between predictors and dependent (target) variable. The performance of the results is closely related to the data usage. In 1821, Gauss developed the Gauss-Markov Theorem, which was based on his previous method, the Least Square Method [Gauss, 1823].

The Logistic Regression Model is one of the most commonly used statistical procedure methods. It is used in many different areas, especially in medical research. Logistic Regression is designed to respond to the zero or one's shape of outcomes, for instance, "success" or "fail", "yes" or "no", etc. On the other hand, Ordinary Least Square(OLS) is designed for beyond the range variables, not binomial variables. Due to the [Loh and Shih, 1997] error variances and normal distribution results, the OLS Method is not recommended for binary variables. However, if the target variables are binomial, the Logistic Regression Method is recommended. The OLS Method is recommended when there are a range of target variables. They both are sufficient when there is a variety of independent variables, which may be categorical, continuous or ordinal.

There are many techniques for calculating the coefficient, such as Fisher's correlation coefficient, Spearman's coefficient and Pearson's coefficient. The Pearson coefficient was developed by Bravais in 1846 [Denis, 2001] and described by Karl Pearson in 1895. [Pearson, 1895]. In 1904, C. Spearman[Spearman, 1904] used the Pearson Method as another way in which to calculate the relationship strength between two variables. The historic milestone of this correlation and regression is presented in the table **??**:

Both the Logistic Regression Method and the OLS Method can be used for the Pearson and Spearman correlation coefficient methods. When we have more than one variable, the correlation between variables can be measured by using a different index(Coefficients).

One difference, however, is that Spearman's coefficient was developed to measure the rank correlation, while Pearson's coefficient was developed to measure the "linear association between the OLS and the Logistic predicted values.

With logistic regression [Lin et al., 2008], the feature vector $\overrightarrow{X}$ is used to fit the data point in

Table 4.4: History Of Correlation

Date	Person	Title
1823	Carl Fredrich GaussGauss [1823]	German mathematician
1843	John Stuart MillMill [1843]	British philosopher
1846	Argusts Bravias	French physicist
1868	Charles Darwin	British natural philosopher
1877- 1885-1888	Sir Francis Gallon	British mathematician
1895-1896	Karl Pearson	British statistician
1904	Spearman	
1920	Karl PearsonPearson [1920]	French

the equation:

$$P(\overrightarrow{X}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \cdots + \beta_n x_n \tag{4.39}$$

Since this value is not necessarily between 0 and 1, a link function, `logit` is used:

$$P(\overrightarrow{X}) = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2 \cdots+\beta_n x_n}}{1 + e^{\beta_0+\beta_1 x_1+\beta_2 x_2 \cdots+\beta_n x_n}} \tag{4.40}$$

The Maximum Likelihood Estimate (MLE) is used to find the values of the coefficients β_i's from the data.

In this section, we give a brief introduction to binary classification with logistic regression. In general, we start with a vector of features $\overrightarrow{X} = (x_1, x_2, \ldots, x_n)$ that can serve as a template for each data point in our data set. We want to build a binary classifier Y that predicts survivability. This construction is essentially based on the characteristics of the training data set. In our case, this is SEER database.

In the classical regression Lin et al. [2008], the feature vector $\overrightarrow{X}$ is used to fit the data point in the equation:

$$P(\overrightarrow{X}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \cdots + \beta_n x_n \tag{4.41}$$

Since this value is not necessarily between 0 and 1, we can not use it as probability to assign a class to the data point. In general, a link function, `logit` is used to convert $p(\overrightarrow{X})$ to a value between 0 and 1.

$$P = \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2 \cdots+\beta_n x_n}}{1 + e^{\beta_0+\beta_1 x_1+\beta_2 x_2 \cdots+\beta_n x_n}} \tag{4.42}$$

The training data is used to estimate the coefficients of the equation 4.42. We use log likelihood to decide on class assignment. For binary classification, each training data point $\bar{x_i}$ has a class assignment y_i (e.g. 0 for `not-survived`, 1 `survived`). We then substitute a data point in equation 4.42 in order to calculate the probability p_i. The log likelihood is:

$$\sum_{i=1}^{n} y_i log p_i + (1 - y_i) log(1 - p_i) \tag{4.43}$$

The equation 4.43 is solved numerically to obtain Maximum Likelihood Estimate (MLE) for coefficients β_i's from the data.

Chapter 5

Comparison

Many natural problems can be solved using binary classification techniques. Known examples of binary classifications are the detection of fraudulent credit card fraudulent transactions [Phua et al., 2004], spam identification [Benevenuto et al., 2009], classified documents [Taghva, 2009], and privacy detection [Taghva et al., 2006]. Naive Bayes, decision trees, logistic regression, artificial neural network(ANN), and support vector machine (SVM) are among the most popular techniques for binary classification.

One of the earliest and most cited work on survival predictability with machine learning tools are the experiments reported by Delen et al. [Delen et al., 2005]. These experiments identified decision tree as the best predictor, compared with artificial neural networks (ANN) and logistic regression. A follow-up set of experiments by Bellaachia and Guven [Bellaachia and Guven, 2006] reported similar results that decision tree was superior to naive Bayes and ANN. Neither work was reproducible research, as there are no code book description of recipes on data preparation and algorithms. Furthermore, it is not clear which methods (direct vs actuarial) that both studies used to identify patient five year survival status of patients.

Both of the above-mentioned studies were conducted using SEER data. Closely related studies on lung cancer, also using SEER data, found that decision tree was the best predictor [Agrawal et al., 2012]. This study further identified the importance of two out of 11 features when predicting survivability. In another interesting and related study using SEER data, Zolbanin et al.[Zolbanin et al., 2015] based the prediction of survivability on comorbidity of cancers, for example, breast and prostate cancer.

Salma et al. [Salama et al., 2012] performed comparison studies on Wisconsin Breast Cancer (WBC) database [Lichman, 2013], and reported that Multi-Layer Perception (MLP) was superior

Figure 5.1: Confusion Matrix

	Predict No	Predict Yes	
Actual No	True Negative (TN)	False Positive (FP)	Neg
Actual Yes	False Negative (FN)	True Positive (TP)	Pos
	PNeg	PPos	n

to decision tree for that database. It is important to point out that WBC collects a different set of features for breast cancer than does SEER. It is also worth mentioning that another study by Christobel and Sivaprakasam [Angeline Christobel. Y, 2011] identified the Support Vector Machine (SVM) as the best predictor for the WBC database. Finally, we want to draw attention to binary classification based on missense mutation in genome [Wei and Dunbrack Jr, 2013].

5.1 Comparison Metrics

Regarding the prediction accuracy when using precision/recall metrics and ROC curve, in the 10-fold cross validation method, the entire data set was split into 10 random sub-samples. Each classifier uses nine folds for training and one fold for testing. The final confusion matrix is the average of the 10 runs.

Suppose we start with n data points divided into `positive` (Pos) and `negative` (Neg) examples. Let TP be the number of true positives, that is, the number of patients which the classifier predicts `survived` and the patients actually have `survived`. Let FN be the number of false negatives, i.e., the number of patients that actually `survived` but the classifier predicts `not-survived`. The TN is defined as the number of patients that have `not-survived` and the classifier also predicts `not-survived`. The FP is the number of patients that have `not-survived` but the classifier falsely predicts `survived`. These four metrics are typically summarized in a confusion matrix as shown in Figure 5.1. The total of TN and FN is denoted by $PNeg$. Simialarly, the total of FP and TP is denoted by $PPos$.

Recall or True Positive Rate *tpr* then is defined as:

$$recall = tpr = \frac{TP}{TP + FN} \tag{5.1}$$

And the *precision* is defined as:

$$precision = \frac{TP}{TP + FP} \tag{5.2}$$

The `harmonic mean` of precision and recall is called the $F1$ measure, defined as:

$$F1 = \frac{2}{1/precision + 1/recall} \tag{5.3}$$

The False Positive Rate, *fpr* is defined as:

$$fpr = \frac{FP}{FP + TN} \tag{5.4}$$

The accuracy of the classifier, *acc* is defined as the weighted average of true positive and true negative rates.

$$acc = Pos * tpr + Neg * (1 - fpr) \tag{5.5}$$

Another popular metric for comparison of binary classifiers is the Receiver Operating Characteristic (ROC) curve. The ROC is extensively used in ther literature. The ROC curve exhibits the tradeoff between true positive and false positive error rates [Duda et al., 2012]. The X-axis ad Y-axis in ROC curve are *fpr* and *tpr*, respectively.

The Area Under the ROC Curve (AUC) is also an accepted measure of the binary classification performance and is widely used.

5.2 Base Experiment

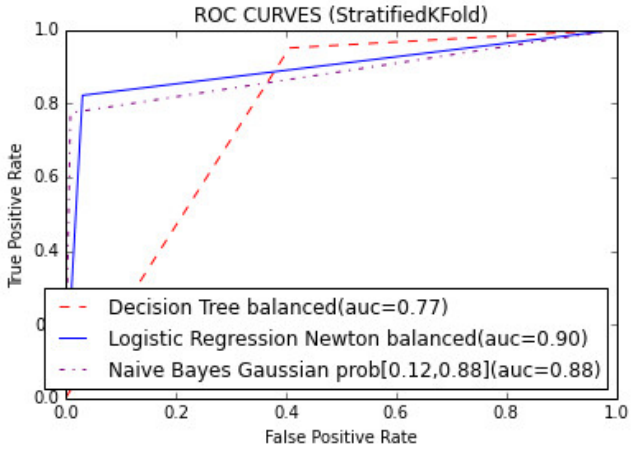
In this study, the performance of naive Bayes, decision trees, and logistic regression were evaluated for their performance in predicting five-year survivability of breast cancer patients. These three approaches were chosen because they were techniques used in past studies on survivability prediction. The implementations for these three approaches developed by Pedregosa et al. [Pedregosa et al., 2011a] were used in these experiments.

As mentioned previously, the number of data points in the `survived` class is eight times the number of `not-survived` data points. Typically, this imbalance affects the classification accuracy [Wei and Dunbrack Jr, 2013]. Many approaches have been developed to overcome the problems associated with the unbalanced training data. The simplest one is to provide the prior weights of the training class to the classifier. The `balanced` value for class-weight parameter for both

Figure 5.2: Performance of the Classifiers

Classifier	class	Precision	Recall	F1
Naive Bayes	`survived`	0.36	0.99	0.53
	`not-survived`	1.00	0.77	0.87
Logistic Regression	`survived`	0.41	0.97	0.58
	`not-survived`	1.0	0.82	0.90
Decision Tree	`survived`	0.60	0.59	0.60
	`not-survived`	0.95	0.95	0.95

Figure 5.3: ROC Curve



decision tree and logistic regression experiments. In addition, the class prior $[0.12, 0.88]$ was used for naive Bayes experiments. Stratified 10-fold cross validation was used for training and testing to make sure that each fold preserved a similar distribution as the original classes. Aside from the default setting, the only other parameter used was `newton method` for the solver method of the logistic regression.

The performance of the tree classifiers with 10-fold cross validation is summarized in Figure 6.7 and Figure 5.3.

The *precision* reports the percentage of data points that are classified as positive that are actually positive. The *recall* reports the percentage of correctly labeled data points. Precision is sensitive to the class distribution. In general, the precision is affected by the class distribution while recall is not. All three methods have low precision for the `not-survived` class, but both logistic regression and Naive Bayes have very high recall values for this class. This is a crucial point as the cost of misclassification is prohibitive for this class. The idea being that when a patient is put in the `not-survived` class, then we may require further test to be assured of the patient condition. The ROC curve suggests that logistic regression is also superior based on the AUC value. The difference

between AUCs for Naive Bayes and logistic regression may not be statistically significant.

A closer look at the coefficients reveals that `race` and `vitalStatusRecord` are not significant and can be eliminated.

There are many methods to improve our base experiments. One of the most widely used techniques is to use the correlation among predictors to computationally categorize certain attribute values to improve the recall and precision level. In the next section, we will give a detailed introduction to VIF and GVIF to overcome the correlation problems.

5.3 VIF and GVIF

Our previous work on cancer data [Taghva and Bozorgi, 2016] has identified logistic regression as a superior choice over Naive Bayes and decision trees. Our classification is based on 14 attributes. Many of these features are categorical.In regression modeling with large number of predictors, correlations among predictors is quite common, and the estimated model coefficients might not be very reliable. In these situations, the variance inflation factor (VIF) is computed for each predictor; a rule of thumb is to omit any predictor which has VIF larger than 5 [Fox, 2002]. When there are categorical predictors present, VIF does not apply; [Fox and Monette, 1992] developed a generalized VIF. Since the present data set has several categorical predictors, we have computed GVIF for all predictors. For continuous predictors, GVIF and VIF are the same.

Figures 5.4,5.5, and 5.6 are bar charts of ten categorical predictors in the present data set. It can be seen that the predictors race, histologicType , grade, csLymphNod and COD have large number of levels or values, which makes the fitting of logistic regression models numerically inaccurate, as indicated by extremely large values of GVIF. For this reason, we recoded these predictors by combining levels with low sample sizes into one category which we called `other`; this was done for each of the predictors mentioned above.

Another prominent technique is Synthetic Minority Over-sampling Techniques known as SMOTE. In the next chapter, we will a give detailed introduction to SMOTE. We then combine GVIF and SMOTE to improve our results.

Figure 5.4: Barplots of (a) race, (b) marital status, (c) histologic type, (d) beha vior code

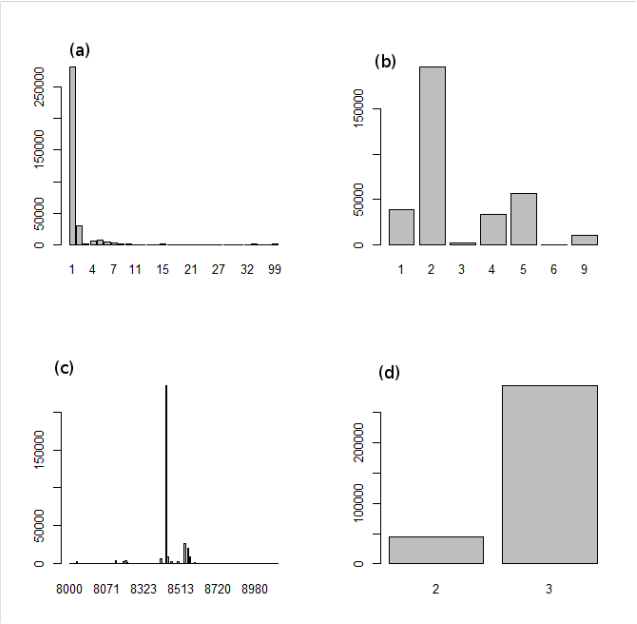


Figure 5.5: Barplots of (a) grade, (b) csEODLymphNode, (c) radiation, (d) seerHistoricStageA

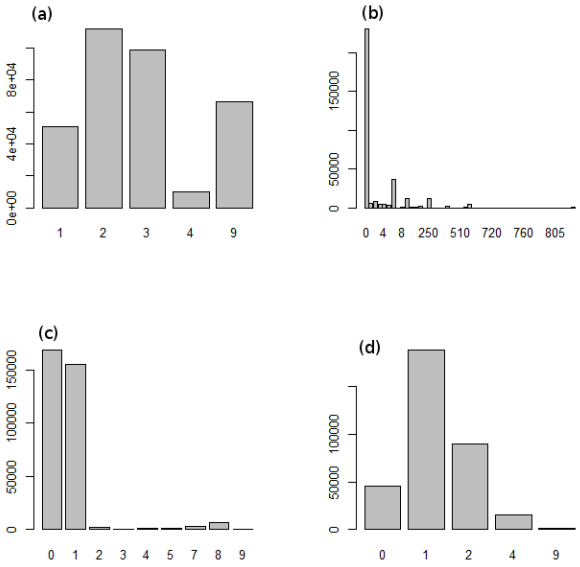
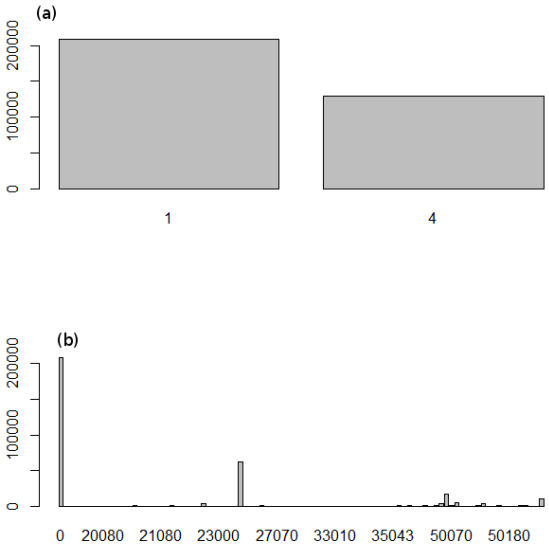


Figure 5.6: Barplots of (a) VitalStatusRecord, (b)causeOfDeathToSEERSiteRecord



Chapter 6

Correlation, Sampling, and Estimation of Models

This chapter describes our optimization approaches to improve the recall and precision level of our classifier based on logistic regression. We start with technical development of different SMOTE techniques in section 6.1. We then combine the GVIF and SMOTE to finalize our experiments.

6.1 Synthetic Minority Over-sampling Techniques

As mentioned in section 1, the cost of misclassification of a minority record is higher than the cost of miscalssification of a majority record. The re-sampling techniques' main objective is to correct this misclassification cost. These re-sampling tools usually under-sample the majority class or over-sample the minority class. Some side effects of these techniques are that under-sampling may throw away good data and over-sampling may cause over-fitting.

When working with imbalanced data sets, there are two things to consider, between-class imbalances and within-class imbalances[Chawla et al., 2004]. In imbalanced data sets, the majority class has more samples, while the minority class has fewer. Imbalanced data sets are found in many different areas, such as in the detection of fraud phone calls[Fawcett and Provost, 1996], detect the possibly cancerous cells in Mammography image [Chawla, 2003] or discover oil spills in satellite radar images[Kubat et al., 1998].

In these examples, we are mostly interested in the results of the minority class rather than the results of the majority class. Unfortunately, traditional data mining algorithms are not designed for imbalanced data sets. When dealing with imbalanced data sets, different methods must be

Table 6.1: Sampling Technique

Models	Techniques
Over-Sampling	SMOTE (Synthetic Minority Over-Sampling Technique) B-SMOTE (Borderline SMOTE) B-SMOTE (Borderline SMOTE) B-SMOTE (Borderline SMOTE) Random majority Over-Sampling with replacement ADASYN(Adaptive Synthetic)
Under-Sampling	Random Majority Under-Sampling with replacement Tomek Links Near-Miss Under-Sampling with Cluster Centroid One Side Selection Neighborhood Cleaning Rule (ENN) Edited Nearest Neighbor Repeated Edited Nearest Neighbor Condensed Nearest Neighbor Instance Hardness Threshold AIKNN

used. Solutions can be determined either by `data levels` or `algorithm levels`. For data levels, changing the distribution and using sampling techniques results in more balanced data sets. On the other hand, for algorithm levels, improving and modifying the existing data mining to find a new algorithm [Han et al., 2005] .

In this study, we are considering the between-class imbalance, where some classes have a lot more samples than other classes. One solution for solving imbalanced data sets issues is to use Sampling techniques. We have two major Sampling techniques, Over-Sampling and Under-Sampling, plus a combination of the two. In Over-Sampling, minority class examples must be replicated to achieve a more balanced distribution; however, in Under-Sampling, some examples are eliminated from the majority class to find more balanced sets. The list of the some of the sampling techniques presented in table 6.1.

We have two different categories for sampling techniques; one is combining the Over-and Under-Sampling technique and the other is creating an Ensemble balanced set as shown in table 6.2.

In Random Over-Sampling, the random sample of the minority class is duplicated to achieve a more balanced dataset.

A well-known Over-Sampling technique is Synthetic Minority Over-Sampling Technique (SMOTE). SMOTE was inspired by the Ha & Bunke 1997 handwriting recognition technique [Chawla et al.,

Table 6.2: Sampling Technique

Models heightOver-Sampling Followed by Under-Sampling	Techniques SMOTE and Tomek Link SMOTE and ENN SVM (Support Vector Machine)
Ensemble Sampling	Easy Ensemble Balance Cascades

2002]. In this model, the minority class is not over sampled by replacement. Instead, SMOTE creates synthetic examples and uses the nearest K neighbor technique of the minority class, which usually considers the 5 nearest neighbors. However, it depends on the amount of over sampling. If the needed over sampling is 300%, then 3 of the 5 randomly chosen nearest neighbors have one sample generated for each. Amongest all the minority class neighbors, the samples with the smallest Euclidean Distance are selected and identified as the select minority class neighbors. (We represent the minority class with M) One of these five neighbors is then randomly selected and a new synthetic sample is created as shown in 6.1.

$$\begin{aligned} S &= M_{n_{Neigbor}} \\ M &= m_1, m_2, m_3, ..., m_{numMinority} \\ N &= n_1, n_2, n_3, ..., n_{numMajority} \end{aligned} \tag{6.1}$$

y is the number between zero and 1. $M_{n_{Neigbor}}$ is randomly chosen among 5 neighbors in sample P. It can be different on the other SMOTE sample minority node. There is an assumed line between the minority node, the first selected neighbor, and the new generated sample, which lies on this joining line. The minority class over sampled by each minority class sample and introducing synthetic examples along the line segments join any/all of the k nearest neighbors. It calculates the difference between sample and its selected neighbor, multiply the number by a random number y and add it to feature vector.

SMOTE performs better than random over sampling and is being used in many different areas. It is being used in bioinformatics for gene prediction [Lusa et al., 2013]. Also, Nitesh et al. , integrated SMOTE into a standard boosting procedure [Han et al., 2005], this improved the prediction of the minority class.

In the SMOTE process, the first step is to consider the minority class and ignore the majority class. For every minority, find the K nearest neighbor. In this example, consider k=5.

Then choose the 5 neighbors with the smallest distance. Create a new sample along the joining (you may want to use adjoining line) line for each neighbor as shown in figure 6.1.

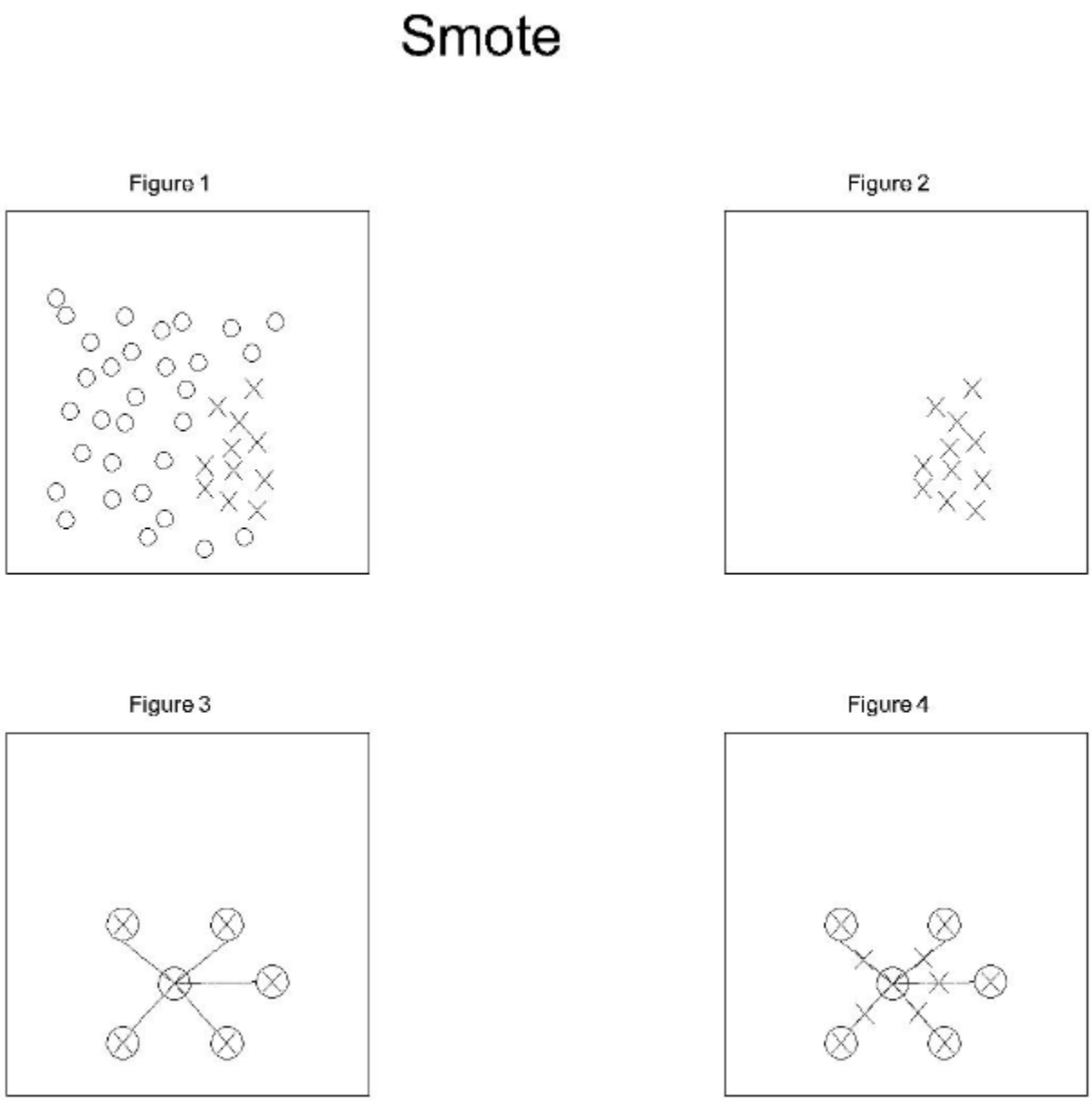


Figure 6.1: Synthetic Minority Over Sampling Technique -SMOTE

To better understand SMOTE, this simple example can help. Assume you have a sample of (7,4) and the first nearest neighbor is (5,3). First we find the differences of these two points (5-7, 3-4) = (-2,-1).

and we need to assume y between 0 and 1, here we pick 0.5. The new Synthetic example would be: $(7,4) + 0.5 * (-2,-1)$ which is $(6, 3.5)$.

Borderline-SMOTE-1 is very much the same as SMOTE.In both techniques, Borderline-SMOTE-1(B-SMOTE1) and SMOTE, the synthetic examples generated along on the line joining the two nearest neighbors are the same but there is one difference. In SMOTE, we only consider the K minority nearest neighbors; however, in Borderline-SMOTE-1, the k nearest of the whole data set is considered. In whole data set, we have the minority class (M) and the majority class (N) as shown in equations 6.2 and 6.3.

$$M = m_1, m_2, m_3, \cdots, m_{numMinority} \tag{6.2}$$

$$N = n_1, n_2, n_3, \cdots, n_{numMajority} \tag{6.3}$$

In the B-SMOTE method, first find the k nearest neighbor of one minority randomly selected point. Then count to see how many of these selected neighbors are in the minority group l and how many are in the majority group $l\prime$.

If $0 \leq l\prime \leq l/2$ this point is ignored. If $l = l\prime$ we consider it as noise and ignore this node.

If $l2 \leq l\prime \leq l$ the number of selected nearest neighbors of the majority class is greater than the selected nearest neighbors of the minority class. This point will be placed in a set called DANGER, which is a list of selected minority class points.

For each point in the DANGER set $DANGER = \{m_1\prime, m_2\prime, \ldots, m_{num}\prime$, we only calculate the k nearest neighbors of the minority class.

Each point on the DANGER list is based solely on the k nearest minority class neighbors. Synthetic samples are generated for each point. From the k nearest neighbor, the t nearest one gets selected $0 \leq t \leq k$.

The same goes for SMOTE. We find the differences and multiply them by the random number between zero and 1(y). This process is repeated for all the points in the DANGER set.

The other over sampling method is the Adaptive Synthetic Sampling Approach (ADASYN). This method was inspired by SMOTE, SMOTEBoost and DataBoost-IM [He et al., 2008]. The ADASYN idea is to reduce the bias by adjustment weight and adaptive learning. The main thought in this model is to use a weighted/density distribution to decide the number of synthetic examples needed for each selected minority data point. In ADASYN, more synthetic data samples are

generated from the points that are hard to learn and less samples are generated from the points that are easier to learn[He et al., 2008].

We represent the training data with T,with the number ofM_{num} instances in the minority group andN_{num} examples in the majority group.

In ADASYN model, the first step is calculating the degree of freedom D by dividing the number of minority classes (M_{num}) by the number of majority classes(N_{num}). This model then considers how much the degree of difficulty of learning the minority class examples is.

If the degree of freedom is smaller than the threshold for the maximum tolerated degree of class imbalance ratio $D_{threshold}$, then w synthetic examples are created.

One of the first earliest under-sampling techniques used was Condensed Nearest Neighbor(CNN). This technique was based on the k nearest neighbor rule. The idea of this method was to shrink the sample space. The Nearest neighbor implementation was nave and required a lot more space compared to other of all the previous classified data[Angiulli, 2005]. Several different solutions were offered at that time in order to avoid using so much space. These were methods known as lazy, instance-base, memory-based and case-based.[Shekarforoush et al., 2017] Later on they were grouped into the following three categories:

- Competence preservation
- Competence enhancement
- Hybrid approach

The Tomek link method is based on the Condensed Nearest Neighbor. Tomek Link is an Under-Sampling technique, which creates more balanced data sets by removing the examples from the minority group. After under-sampling, in this model the training data number will be less than the number of the total data set.

Tomek link looks at pairs of data points located in different groups which have no points between them and are very close to each other. Consider these two examples in equation 6.4:

$$M_n \text{ and } N_i, n = 1, \cdots, M_{num} \text{ and } i = 1, \cdots, N_{num} \tag{6.4}$$

The distance between M_n and N_i is represented by $d(M_n, N_i)$.

We consider these two points as Tomek link only if there are no other data points between them. Tomek Link creates more balanced data sets by removing the majority data point in each

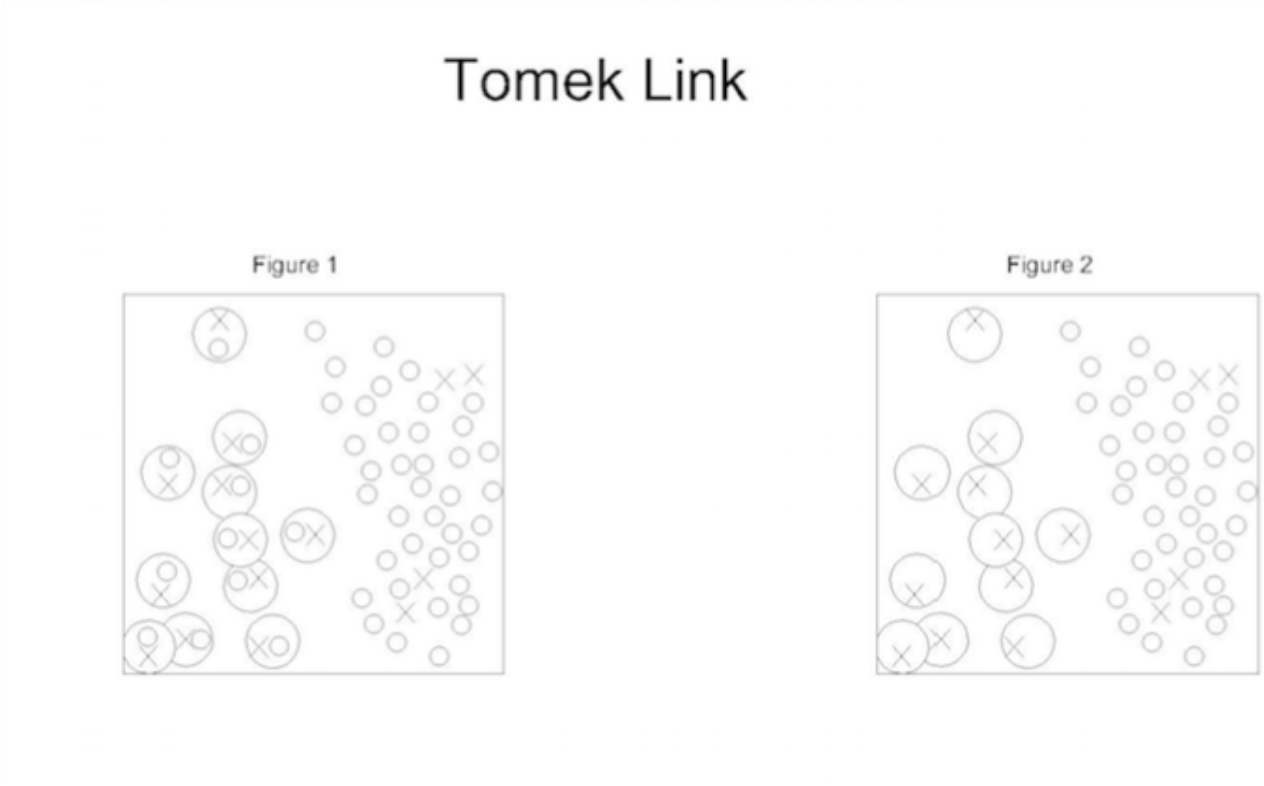


Figure 6.2: Tomek Link

pair as shown in figure 6.2.

Another method is called the Edited Nearest Neighbors. This method was developed by Wilson in 1972 [Wilson, 1972]. This particular method is based on the K nearest Neighbor but with a few differences. In this method, the editing procedure is used to balance the data set. The first step entails finding the KNN of one selected point from the training set. The next step is finding the KNN of that sample. If the majority of the selected neighbors are the same class as the selected point, then assign $flag_0$.However,if the majority of the selected neighbors are from a different group, then assign $flag_1$. Then continue on to the next node. Repeat this process until all the nodes in the training set are covered. At the end, remove the nodes that were assigned $flag_0$. Consider a node x_i from the minority group. After finding the kNN, count the nearest neighbors. If the majority goes to the Minority group, then assign $flag_0$, since they are in the same group. And if the majority vote goes to the majority class, then assign $flag_1$. Continue with the same process on the next node x_{i+1}. Stop after all the nodes get the proper flags. Then remove all the nodes with $flag_0$ [Tomek, 1976].

The next method that we are going to present is One Side Selection method. This is yet another under-sampling technique, which came from the same idea as Tomek Link. To evaluate the accuracy of the sampling and classifiers, as was mentioned before, measures are formulated in a confusion matrix. We presented the Accuracy, precision, and recall measurement formula. In this method, we first need to evaluate the a^+ and a^-. a^+ represents the accuracy of positive examples, and $a^-$is a measure for accuracy in negative examples. Before trying this method, we need to find these two numbers. If they are somewhat similar or close, this technique will not be a good option, but if a^+ and a^- have totally different results, then we can consider using the One-Sided Selection sampling technique as an option. The other measure is $G = -\sqrt{a^+ \times a^-}$, `mean of accuracy`. G is maximized when two a^+ and a^- are balanced.

The next method that we are going to present is One Side Selection method. This is yet another under-sampling technique, which came from the same idea as Tomek Link. To evaluate the accuracy of the sampling and classifiers, as was mentioned before, measures are formulated in a confusion matrix. We presented the Accuracy, precision, and recall measurement formula. In this method, we first need to evaluate the a^+ and a^-. a^+ represents the accuracy of positive examples, and $a^-$is a measure for accuracy in negative examples. Before trying this method, we need to find these two numbers. If they are somewhat similar or close, this technique will not be a good option, but if a^+ and a^- have totally different results, then we can consider using the One-Sided Selection sampling technique as an option. The other measure is G, mean of accuracy. G is maximized when two a^+ and a^- are balanced. By looking at the figure2.3, we can see that its hard to draw the decision surface line, because the circle points have square close neighbors.

By removing the redundant in the majority class, we have a lesser number in the majority class. Tomek link, when applied to remove borderline and noise, improves the value of the Geometric. The accuracies of a^+ and a^- are more balanced[Kubat et al., 1997]. This technique is used in different kinds of research, for example, the identification of carbonylated sites of human proteins[Zuo and Jia, 2017].

One Sided sampling is used to help reduce the number of majority class by adapting the Tomek Link technique. There are four examples:

- Borderline: examples that are close to the borderline surface
- Noise: Those further away from their own groups and closer to the other groups (like the square in the bottom right corner)

- Redundant: They can be represented by other instances.
- Safe: The type that matters most in our technique.

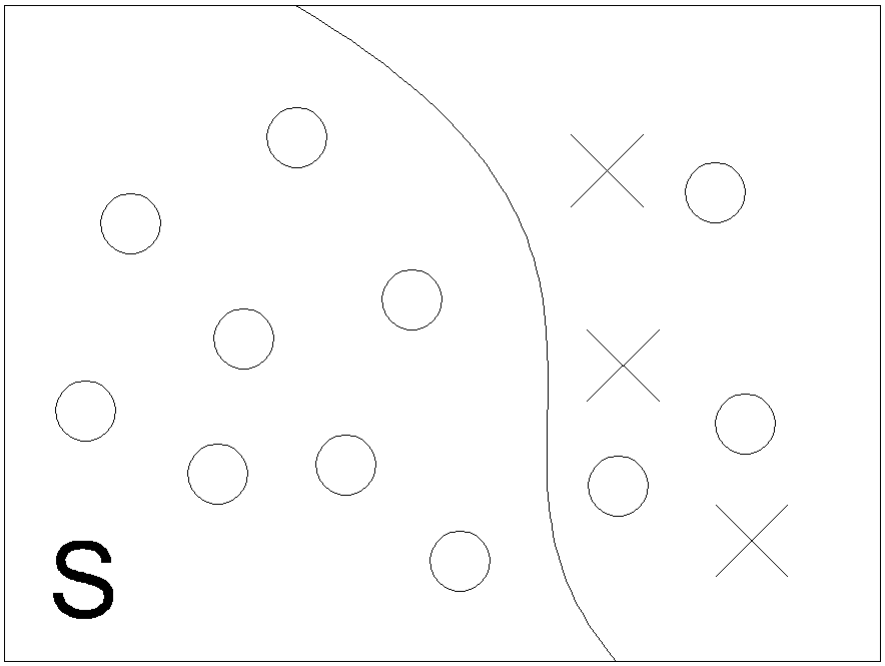


Figure 6.3: dividing minority and majority group is not easy in this spread

We then create a new set called C by keeping all the minority class (in this example the circles) and one randomly selected from the majority class. Using the samples in C, we Classify S with 1-NN rule; now we dont have any redundant examples in our new C data sets. At the end, we remove all the majority class which are borderline (close to the borderline surface) and the noise. Respectable balanced data sets are the result (T) as seen in figures 6.3 and 6.4.

We then create a new set called C by keeping all the minority class (in this example the circles) and one randomly selected from the majority class. Using the samples in C, we Classify S with 1-NN rule; now we dont have any redundant examples in our new C data sets. At the end, we remove all the majority class which are borderline (close to the borderline surface) and the noise. Respectable balanced data sets are the result (T).

The next section is the result of our experiments.

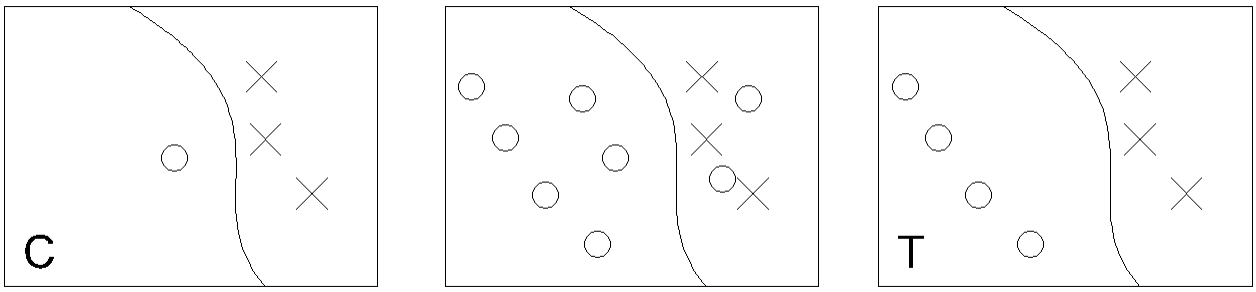


Figure 6.4: One Side Selection - Under Sampling

6.2 Logistic Regression, Correlation, SMOTE

We run a set of logistic regression experiments with the `balanced` value for class-weight parameter. Stratified 10-fold cross validation was used for training and testing to make sure that each fold preserved a similar distribution as the original classes. Figure 6.5 shows the model coefficients and corresponding p-values.

Since the P-value for at least one level of the factors in the model are less than 0.05, each categorical predictor is statistically significant; each of the five continuous predictors in the model are also statistically significant.

The GVIF values for the predictors in the above model are shown in Figure 6.6.

The GVIF values are all quite close to 1, indicated that the fitted logistic regression model does not suffer for multicollinearities among predictors.

The commonly used pseudo-rsquare values [Hu et al., 2006] are of moderate size suggesting that the fitted model is reasonable. The P-value of the Walds test for overall model fit is 0.00, indicating that fitted model is a significant improvement over the null model.

The performance of our base experiments with 10-fold cross validation is summarized in Figure 6.7. It can be seen that this model has a low precision and f1-score for the minority class.

The goal of the next set of experiments were to improve this deficiency. We performed four additional experiments with logistic regression in order to increase the recall and f1-measure for the minority class. The performances of these experiments are summarized in Figure 6.8.

We observe that `SMOTE`, `SMOTE Borderline`, and `Tomek Link` did not improve the recall or f1-score for the minority class. The performance of the `Editted Nearest Neighbors` is as good as anything reported in the literature including experiments on Support Vector Machine (`SVM`). The AUC for these five experiments are shown in 6.9 which support our findings on `Editted Nearest`

Figure 6.5: The ML estimates of the logistics regression model coefficients and cor responding P-vales, based upon the training set of 75% of all cancer data

	Estimate	SE	Z	P-value
Intercept	8.516e+00	8.400e-2	101.381	¡2e-16***
race 2	-5.479e-1	1.937e-2	-28.279	¡2e-16***
race (Other)	1.677e-01	2.637e-02	6.359	2.04e-10***
factor(maritalStatus)2	2.155e-01	2.016e-02	10.691	¡ 2e-16 ***
factor(maritalStatus)4	-3.040e-02	2.636e-02	-1.153	0.249
factor(maritalStatus)5	-1.239e-01	2.441e-02	-5.073 3.91e-07 ***	
factor(maritalStatus)Other	1.599e-01	3.659e-02	4.371	1.24e-05 ***
factor(behaviorCode)3	-4.050e+00	6.964e-02	-58.163	¡ 2e-16 ***
factor(grade)2	-1.132e+00	3.277e-02	-34.550	¡ 2e-16 ***
factor(grade)3	-2.247e+00	3.185e-02	-70.542	¡ 2e-16 ***
factor(grade)4	-2.281e+00	4.706e-02	-48.478	¡ 2e-16 ***
factor(grade)9	-1.548e+00	3.359e-02	-46.070	¡ 2e-16 ***
factor(radiation)1 3.680e-01	1.359e-02	27.088	¡ 2e-16 ***	
factor(radiation)Other	3.915e-02	2.927e-02	1.338	0.181
ageAtDiagnosis	-6.607e-03	5.097e-04	-12.962	¡ 2e-16 ***
csEODTumorSize	-1.554e-03	3.582e-05	-43.381	¡ 2e-16 ***
regionalNodesPositive	-1.631e-02	1.655e-04	-98.559	¡ 2e-16 ***
csEODExtension	-3.893e-03	6.456e-05	-60.298	¡ 2e-16 ***
regionalNodesExamined	-1.094e-02	4.232e-04	-25.855	¡ 2e-16 ***
		—		
Signif. codes:	0 *** 0.001	** 0.01	* 0.05	. 0.1 1

Figure 6.6: GVIF values for predictors of model

	GVIF	Df	GVIF(1/(2*Df))
race - categorical	1.06	2	1.02
maritalStatus - categorical	1.38	4	1.04
behaviorCode - categorical	1.03	1	1.01
grade - categorical	1.11	4	1.01
radiation - categorical	1.04	2	1.01
ageAtDiagnosis - numeric	1.42	1	1.19
csEODTumorSize - numeric	1.03	1	1.01
regionalNodesPositive - numeric	1.19	1	1.09
csEODExtension - numeric	1.04	1	1.02
regionalNodesExamined - numeric	1.09	1	1.04

Figure 6.7: Performance of the Base Experiment

Classifier	class	Precision	Recall	F1	AUC
Logistic Regression	`not-survived`	0.27	0.77	0.40	0.75
	`survived`	0.96	0.73	0.83	

Figure 6.8: Performance of the Base Experiment

Classifier	class	Precision	Recall	F1	AUC
SMOTE	`not-survived`	0.73	0.79	0.76	0.75
	`survived`	0.77	0.71	0.74	
Tomek Link	`not-survived`	0.28	0.79	0.41	0.76
	`survived`	0.96	0.73	0.83	
SMOTE Borderline	`not-survived`	0.72	0.82	0.77	0.75
	`survived`	0.79	0.68	0.73	
Edited Nearest	`not-survived`	0.95	0.90	0.92	0.92
	`survived`	0.90	0.95	0.93	

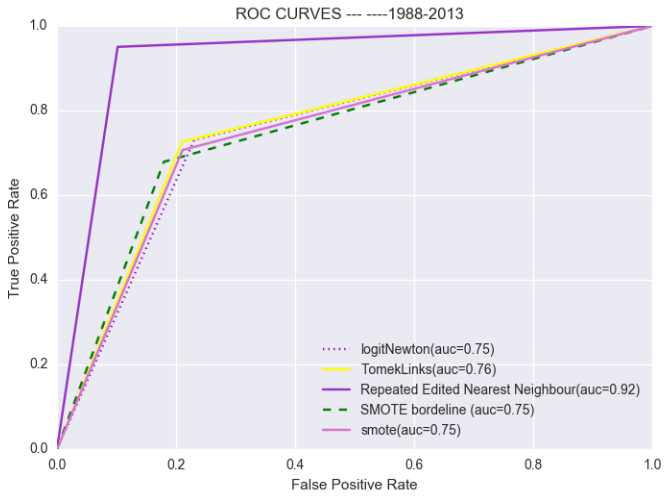


Figure 6.9: Area Under the Curves

`Neighbors`.

These four post processing sampling techniques were used for reducing classification bias in favor of the majority class, so even though the estimated coefficients of the logistic regression model might be biased, the classification results are improved for both classes. In addition, since these post processing techniques add or delete some records, the size of the training sets are slightly changed. The respected sizes of these training sets are reported in Figure 6.10.

Figure 6.10: Sizes of the Training Data Sets

	`survived`	`not-survived`
logistic regression	300215	38381
Tomek Links	289816	38381
SMOTE Borderline	300215	150107
Edited Nearest	90454	38381

Chapter 7

Conclusions and Future Works

This thesis reports on application of machine learning tools for predicting cancer survivability. This work was based on reproducible research principle, a larger data set, and unbalanced nature of cancer data set. Results indicate that logistic regression is an excellent choice for cancer prediction as compared to decision trees and naive Bayes.

Our experiments are also focused on identification of correlation between features and categorical predictors. We used VIF and GVIF to overcome the problems associated with categorical predictors.

The most significant contributions of this work are various applications of under-sampling and over-sampling techniques in order to increase the accuracy performance for the minority class.

Our work was motivated by recent discoveries in reproducible research. Many of the past work on the topic of cancer survival rate is based on SEER data. Unfortunately, most of these works are difficult to reproducible due to poor record keeping. In some cases, it is not clear what methods were used in data preparation or how the experiments carried out. We believe that this thesis provides a remedy for data preparation and cleaning in addition to record keeping.

We were also motivated by the idea that the recall level in cancer prediction must be almost perfect. It is a costly mistake to classify a not-survived member as a survived member. On the other hand, if the error is reversed, a doctor can rely on further testing to reverse the classifier prediction. Most of our experiments as reported in this thesis are based on optimization techniques to improve the recall level.

There are four possible extensions to this project that we are currently pursing. The first extension is to apply other the Synthetic minority over-sampling technique (SMOTE) to re-balance the the training set in order to improve the recall. Second extension is to apply these experiments to other types of cancers using SEER data. The third extension is to build a web-based application

that could be used as an advisory tool for surviability prediction. The fourth extension is to apply ANN in the mind set of logistic regression. The ANN can improve our result assuming more training data and features become available over time.

Bibliography

Omar Abdel-Rahman. An analysis of clinical characteristics and patient outcomes in primary mediastinal sarcomas. *Expert Review of Anticancer Therapy*, (just-accepted), 2017.

Ankit Agrawal, Sanchit Misra, Ramanathan Narayanan, Lalith Polepeddi, and Alok Choudhary. Lung cancer survival prediction using ensemble data mining on seer data. *Scientific Programming*, 20(1):29–42, 2012.

Reda Al-Bahrani, Ankit Agrawal, and Alok Choudhary. Colon cancer survival prediction using ensemble data mining on seer data. In *Big Data, 2013 IEEE International Conference on*, pages 9–16. IEEE, 2013.

Dr. Sivaprakasam Angeline Christobel. Y. An empiricalcomparison of data mining classification methods. *Journal of Computer Information Systems*, 3(2):10–110, 2011.

Fabrizio Angiulli. Fast condensed nearest neighbor rule. In *Proceedings of the 22nd international conference on Machine learning*, pages 25–32. ACM, 2005.

H. H. Aumann, M. T. Chahine, C. Gautier, M. D. Goldberg, E. Kalnay, L. M. McMillin, H. Revercomb, P. W. Rosenkranz, W. L. Smith, D. H. Staelin, L. L. Strow, and J. Susskind. Airs/amsu/hsb on the aqua mission: design, science objectives, data products, and processing systems. *IEEE Transactions on Geoscience and Remote Sensing*, 41(2):253–264, Feb 2003. ISSN 0196-2892. doi: 10.1109/TGRS.2002.808356.

Peter C Austin. The use of propensity score methods with survival or time-to-event outcomes: reporting measures of effect similar to those used in randomized experiments. *Statistics in medicine*, 33(7):1242–1258, 2014.

Keith A. Baggerly and Kevin R. Coombes. Deriving chemosensitivity from cell lines: Forensic bioinformatics and reproducible research in high-throughput biology. *Ann. Appl. Stat.*, 3(4):1309–1334, 12 2009. doi: 10.1214/09-AOAS291. URL `http://dx.doi.org/10.1214/09-AOAS291`.

Abdelghani Bellaachia and Erhan Guven. Predicting breast cancer survivability using data mining techniques. *Age*, 58(13):10–110, 2006.

Fabrício Benevenuto, Tiago Rodrigues, Virgílio Almeida, Jussara Almeida, and Marcos Gonçalves. Detecting spammers and content promoters in online video social networks. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 620–627. ACM, 2009.

breastcancer.org. U.s. breast cancer statistics. http://www.breastcancer.org/symptoms/understand_bc/statistics.

National Breast Cancer. What is Breast Cancer. http://www.nationalbreastcancer.org/what-is-breast-cancer, 2016.

Nitesh V Chawla. C4. 5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In *Proceedings of the ICML*, volume 3, page 66, 2003.

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. Special issue on learning from imbalanced data sets. *ACM Sigkdd Explorations Newsletter*, 6(1):1–6, 2004.

David R Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 215–242, 1958.

Louise Davies and H Gilbert Welch. Current thyroid cancer trends in the united states. *JAMA otolaryngology–head & neck surgery*, 140(4):317–322, 2014.

Beth Dawson and Robert G. Trapp. Analyzing research questions about survival. *New York*, 2004.

Glenn De'ath and Katharina E Fabricius. Classification and regression trees: a powerful yet simple technique for ecological data analysis. *Ecology*, 81(11):3178–3192, 2000.

Dursun Delen, Glenn Walker, and Amit Kadam. Predicting breast cancer survivability: a comparison of three data mining methods. *Artificial intelligence in medicine*, 34(2):113–127, 2005.

Janez Demšar, Tomaž Curk, Aleš Erjavec, Črt Gorup, Tomaž Hočevar, Mitar Milutinovič, Martin Možina, Matija Polajnar, Marko Toplak, Anže Starič, et al. Orange: data mining toolbox in python. *The Journal of Machine Learning Research*, 14(1):2349–2353, 2013.

D Denis. The origins of correlation and regression: Francis galton or auguste bravais and the error theorists. *History and Philosophy of Psychology Bulletin*, 13(2):36–44, 2001.

Vasant Dhar. Data science and prediction. *Communications of the ACM*, 56(12):64–73, 2013.

Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.

The Economist. How science goes wrong, 2013. URL http://www.economist.com/news/leaders/.

Nature Editors. Must try harder. *Nature*, 483(7391):509–509, 2012.

Brenda K Edwards, Anne-Michelle Noone, Angela B Mariotto, Edgar P Simard, Francis P Boscoe, S Jane Henley, Ahmedin Jemal, Hyunsoon Cho, Robert N Anderson, Betsy A Kohler, et al. Annual report to the nation on the status of cancer, 1975-2010, featuring prevalence of comorbidity and impact on survival among persons with lung, colorectal, breast, or prostate cancer. *Cancer*, 120 (9):1290–1314, 2014.

Tom Fawcett and Foster J Provost. Combining data mining and machine learning for effective user profiling. In *KDD*, pages 8–13, 1996.

Joseph Feller and Brian Fitzgerald. A framework analysis of the open source software development paradigm. In *Proceedings of the twenty first international conference on Information systems*, pages 58–69. Association for Information Systems, 2000.

Scott A Fink and Robert S. Brown. Survival analysis. *Gastroenterology Hepatology*, 2.5(2):380383, 2006.

John Fox. *An R and S-Plus companion to applied regression*. Sage, 2002.

John Fox and Georges Monette. Generalized collinearity diagnostics. *Journal of the American Statistical Association*, 87(417):178–183, 1992.

Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Mach. Learn.*, 29(2-3):131–163, November 1997. ISSN 0885-6125. doi: 10.1023/A:1007465528199. URL `http://dx.doi.org/10.1023/A:1007465528199`.

Francis Galton. Regression towards mediocrity in hereditary stature. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 15:246–263, 1886. ISSN 09595295. URL `http://www.jstor.org/stable/2841583`.

Carl Friedrich Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium auctore Carolo Friderico Gauss*. sumtibus Frid. Perthes et IH Besser, 1809.

Carl-Friedrich Gauss. *Theoria combinationis observationum erroribus minimis obnoxiae*, volume 1. Henricus Dieterich, 1823.

Carl Friedrich Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, volume 7. FA Perthes, 1877.

Johannes Gehrke, Venkatesh Ganti, Raghu Ramakrishnan, and Wei-Yin Loh. Boatoptimistic decision tree construction. In *ACM SIGMOD Record*, volume 28, pages 169–180. ACM, 1999.

Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. *Advances in intelligent computing*, pages 878–887, 2005.

Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 1322–1328. IEEE, 2008.

Torsten Hothorn, Kurt Hornik, and Achim Zeileis. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics*, 15(3):651–674, 2006.

Badr Hssina, Abdelkarim Merbouha, Hanane Ezzikouri, and Mohammed Erritali. A comparative study of decision tree id3 and c4. 5. *International Journal of Advanced Computer Science and Applications*, 4(2):13–19, 2014.

Bo Hu, Jun Shao, and Mari Palta. Pseudo-r 2 in logistic regression model. *Statistica Sinica*, pages 847–860, 2006.

Edward L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

Gordon V Kass. An exploratory technique for investigating large quantities of categorical data. *Applied statistics*, pages 119–127, 1980.

Hyunjoong Kim and Wei-Yin Loh. Classification trees with unbiased multiway splits. *Journal of the American Statistical Association*, 96(454):589–604, 2001.

Miroslav Kubat, Stan Matwin, et al. Addressing the curse of imbalanced training sets: one-sided selection. In *ICML*, volume 97, pages 179–186. Nashville, USA, 1997.

Miroslav Kubat, Robert C Holte, and Stan Matwin. Machine learning for the detection of oil spills in satellite radar images. *Machine learning*, 30(2-3):195–215, 1998.

Adrien Marie Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes*. F. Didot, 1805.

M. Lichman. UCI machine learning repository, 2013. URL `http://archive.ics.uci.edu/ml`.

Chih-Jen Lin, Ruby C. Weng, and S. Sathiya Keerthi. Trust region newton method for logistic regression. *J. Mach. Learn. Res.*, 9:627–650, June 2008. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=1390681.1390703`.

Wei-Yin Loh and Yu-Shan Shih. Split selection methods for classification trees. *Statistica sinica*, pages 815–840, 1997.

Marko Lucijanic and Mladen Petrovecki. Analysis of censored data. *Biochemia medica: Biochemia medica*, 22(2):151–155, 2012.

Lara Lusa et al. Smote for high-dimensional class-imbalanced data. *BMC bioinformatics*, 14(1): 106, 2013.

John Stuart Mill. Of the four methods of experimental inquiry. *A System of Logic, Raciocinative, and Inductive*, 1843.

K Jarrod Millman and Fernando Pérez. Developing open-source scientific practice. 2014.

M Mohankumar, S Amuthakkani, and G Jeyamala. Comparative analysis of decision tree algorithms for the prediction of eligibility of a man for availing bank loan. *Age*, 19:60.

NCI. Surveillance, epidemiology, and end results (seer). www.seer.cancer.gov, 2016.

Tim Oreilly. What is web 2.0, 2005.

DM Parkin and T Hakulinen. Analysis of survival. *Cancer registration. Principles and methods. IARC Scientific Publications*, (95):159–176, 1991.

Karl Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242, 1895.

Karl Pearson. Notes on the history of correlation. *Biometrika*, 13(1):25–45, 1920.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011a.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011b.

R.D. Peng. Reproducible research in computational science. *Science*, (334(6060)):1226–1227, 2011.

Bevinda Alisha Pereira, Anusha Pai, and Cassandra Fernandes. A comparative analysis of decision tree algorithms for predicting students performance. *International Journal of Engineering Science*, 10489, 2017.

Clifton Phua, Damminda Alahakoon, and Vincent Lee. Minority report in fraud detection: classification of skewed data. *Acm sigkdd explorations newsletter*, 6(1):50–59, 2004.

J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, March 1986a. ISSN 0885-6125. doi: 10.1023/A:1022643204877. URL http://dx.doi.org/10.1023/A:1022643204877.

J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986b.

J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

Gouda I. Salama, M. B. Abdelhalim, and Magdy Abd elghany Zeid. Breast cancer diagnosis on three different datasets using multi-classifiers, int. Technical report, J. of Comput. and Inform. Technology, 2012.

Ann C Schaffner. The future of scientific journals: Lessons from the past. *Information technology and libraries*, 13(4):239, 1994.

SeyedHamid Shekarforoush, Robert Green, and Robert Dyer. Classifying commit messages: A case study in resampling techniques. In *Neural Networks (IJCNN), 2017 International Joint Conference on*, pages 1273–1280. IEEE, 2017.

SÃ¸ren Sonnenburg, Mikio L Braun, Cheng Soon Ong, Samy Bengio, Leon Bottou, Geoffrey Holmes, Yann LeCun, Klaus-Robert MÃžller, Fernando Pereira, Carl Edward Rasmussen, et al. The need for open source software in machine learning. *Journal of Machine Learning Research*, 8 (Oct):2443–2466, 2007.

Charles Spearman. " general intelligence," objectively determined and measured. *The American Journal of Psychology*, 15(2):201–292, 1904.

Milija Suknovic, Boris Delibasic, Milos Jovanovic, Milan Vukicevic, Dragana Becejski-Vujaklija, and Zoran Obradovic. Reusable components in decision tree induction algorithms. *Computational Statistics*, 27(1):127–148, 2012.

Kazem Taghva. Identification of sensitive unclassified information. In *Computational Methods for Counterterrorism*, pages 89–108. Springer, 2009.

Kazem Taghva and Mandana Bozorgi. Revisiting survivability prediction of breast cancer with machine learning tools. *To appear*, 2016.

Kazem Taghva, Russell Beckley, and Jeffrey Coombs. The effects of ocr error on the extraction of private information. In *Document Analysis Systems VII*, pages 348–357. Springer, 2006.

Ivan Tomek. An experiment with the edited nearest-neighbor rule. *IEEE Transactions on systems, Man, and Cybernetics*, (6):448–452, 1976.

Mevlut Ture, Fusun Tokatli, and Imran Kurt. Using kaplan–meier analysis together with decision tree methods (c&rt, chaid, quest, c4. 5 and id3) in determining recurrence-free survival of breast cancer patients. *Expert Systems with Applications*, 36(2):2017–2026, 2009.

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995. ISBN 0-387-94559-8.

Stéfan van der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

Qiong Wei and Roland L Dunbrack Jr. The role of balanced training and testing data sets for binary classifiers in bioinformatics. *PloS one*, 8(7):e67863, 2013.

Hadley Wickham. Tidy data. *Journal of Statistical Software*, 59(10), 2014.

Wikipedia. Sourceforge. `https://en.wikipedia.org/wiki/SourceForge`, 2017.

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

Y. Xiao, B. Liu, L. Cao, X. Wu, C. Zhang, Z. Hao, F. Yang, and J. Cao. Multi-sphere support vector data description for outliers detection on multi-distribution data. In *2009 IEEE International Conference on Data Mining Workshops*, pages 82–87, Dec 2009. doi: 10.1109/ICDMW.2009.87.

Hamed Majidi Zolbanin, Dursun Delen, and Amir Hassan Zadeh. Predicting overall survivability in comorbidity of cancers: A data mining approach. *Decision Support Systems*, 74:150–161, 2015.

Yun Zuo and Cang-Zhi Jia. Carsite: identifying carbonylated sites of human proteins based on a one-sided selection resampling method. *Molecular BioSystems*, 13(11):2362–2369, 2017.

Curriculum Vitae

Graduate College
University of Nevada, Las Vegas

Mandana Bozorgi

email: mandanbozorgi@gmail.com

Degrees:

Master of Science in Computer Science 2015
University of Nevada Las Vegas

Thesis Title: Applications of Machine Learning in Cancer Research

Thesis Examination Committee:

Chairperson, Dr. Kazem Taghva, Ph.D.
Committee Member, Dr. Laxmi Gewali, Ph.D.
Committee Member, Dr. Justin Zhan, Ph.D.
Committee Member, Dr. Fatma Nasoz, Ph.D.
Graduate Faculty Representative, Dr. Ashok Singh, Ph.D.