YAŞAR UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MASTER THESIS

PREDICTION OF PRODUCTION WASTAGE VIA DATA MINING

GÖZDE KARADAĞ

THESIS ADVISOR: ASSIST.PROF. (PHD) İBRAHIM ZINCIR

DEPARTMENT OF COMPUTER ENGINEERING

PRESENTATION DATE: 28.08.2018

BORNOVA / İZMİR
AUGUST 2018

We certify that, as the jury, we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Jury Members:	**Signature:**
Asst. Prof. Dr. İbrahim Zincir Yasar University	....
Assoc. Prof. Dr. M. Süleyman Ünlütürk Yasar University	M.S.Ünlüt ....
Asst. Prof. Dr Samsun M. Başarıcı Aydın Adnan Menderes University	....

Prof. Dr. Cüneyt Güzeliş
Director of the Graduate School

ABSTRACT

PREDICTION OF PRODUCTION WASTAGE VIA DATA MINING

KARADAĞ, GÖZDE

Msc, DEPARTMENT OF COMPUTER ENGINEERING

Advisor: Assist.Prof. (PhD) İbrahim Zincir

August 2018

Efficiency is the most important criteria in production systems. The more wastage rate means the less efficiency in the production. However, companies accept wastage to some extent due to dynamism of production. There are two wastage type in production such as setup wastage and production wastage. Setup wastage is not changeable without changing resources. It occurs while preparing materials and machines to the production. This part forms fifteen percent of the total wastage in this study. Therefore, the acceptable wastage is fifteen percent. This research aims to estimate the wastage rate of a packaging company by using data mining methods. In addition, the wastage rate range is decreased gradually in order to make more accurate estimations.

The predicted result is specified as yes, no. If the wastage range is less than 15%, the prediction is no, otherwise yes. In the second analysis, the range is divided to three categories to make prediction more specific which are S, M, L. If the wastage rate is less than 14% it is defined as S. If the wastage is between 14% and 16%, the prediction is M and if it is more than 16%, then it is L. In order to investigate the effect of the parameters in production, 10 versions are generated for each group. Each version has different production parameters. Twenty algorithms are applied these twenty data sets.

As a result of the study, either the range limit is taken as 15% or in the range of 14% and 16% same results are obtained. Naïve Bayes algorithm with version 4 has the best result for Group1 and Group2. It shows that the wastage is not related with month and customer information as much as other parameters.

Key Words: production systems, data mining, wastage, production attributes

ÖZ

VERİ MADENCİLİĞİ İLE ÜRETİM FİRESİNİ TAHMİNLEME

KARADAĞ, GÖZDE

Yüksek Lisans Tezi, Bilgisayar Mühendisliği Bölümü

Danışman: Dr.Öğr.Üyesi İbrahim Zincir

Ağustos 2018

Üretim sistemlerinde verimlilik en önem verilen kriterlerden biridir. Üretimde yaşanan kayıp, fire miktarı ile doğru orantılıdır ve fire verimliliği azaltır. Üretim sürecinin dinamizmine bağlı olarak şirket tarafından belirlenen kabul edilebilir bir fire oranı olabilir. Bu çalışma bir ambalaj üretim firmasının belirlediği kabul edilebilir fire miktarı üzerindeki fireyi veri analizi yöntemlerini kullanarak tahmin edebilmeyi araştırmaktadır. Bu şirkette dikkate alınan 2 çeşit fire vardır. Bunlar setup firesi ve üretim firesidir. Setup firesi, makineyi ve ürünleri üretime hazırlarken ortaya çıkar. Aynı kaynaklar kullanılarak değiştirilmesi kolay değildir. Setup miktarı bu çalışma için %15 olarak belirlenmiştir. Ek olarak, çalışmanın detaylarında, belirlenen miktar aralığı daraltılarak daha detaylı tahminleme yapılmıştır. İlk analiz için tahminleme yapılan değer yes, no olarak belirlenmiştir. Fire yüzdesi %15 ten küçük ise no, büyük ise yes denilmiştir. Tahmin edilecek fire aralığı genişletilirken gruplar üçe ayrılarak S, M, L olarak belirlenmiştir. Fire miktarı %14 ten küçükse S, %14-%16 arasında ise M, %16 dan büyükse L olarak belirtilmiştir. Üretim parametrelerinin fire üzerindeki etkisini gözlemlemek amacıyla her grup için 10 tane version yaratılmıştır. Versiyonlar farklı üretim parametrelerini içerektedir. 20 algoritma bu 20 veri setine uygulanmıştır.

Çalışmanın sonucunda, fire oranı yüzde 15 kabul edildiğinde ve yüzde 14 – 16 aralığında kırılım dikkate alınarak analiz yapıldığında NaiveBayes algoritmasının Grup1 ve Grup2 nin 4. versiyonlarında en iyi sonucu ürettiği gözlemlenmiştir. Bu ay ve müşteri bilgisinin fireyi diğer üretim parametreleri kadar çok etkilemediğini gösterir.

Anahtar Kelimeler: üretim sistemleri, veri analizi, fire, üretim parametreleri

ACKNOWLEDGEMENTS

First of all, I would like to thank my supervisor İbrahim Zincir for his guidance and patience during this study.

I would like to express my enduring love to my parents, who are always supportive, loving and caring to me in every possible way in my life.

Gözde Karadağ
İzmir, 2018

TEXT OF OATH

I declare and honestly confirm that my study, titled "PREDICTION OF PRODUCTION WASTAGE VIA DATA MINING" and presented as a Master's Thesis, has been written without applying to any assistance inconsistent with scientific ethics and traditions. I declare, to the best of my knowledge and belief, that all content and ideas drawn directly or indirectly from external sources are indicated in the text and listed in the list of references.

Gözde Karadağ

Signature

....

September 18, 2018

TABLE OF CONTENTS

ABSTRACT v

ÖZ vii

ACKNOWLEDGEMENTS ix

TEXT OF OATH xi

TABLE OF CONTENTS xiii

LIST OF FIGURES xv

LIST OF TABLES xvii

SYMBOLS AND ABBREVIATIONS xix

CHAPTER 1 INTRODUCTION 1

CHAPTER 2 LITERATURE REVIEW 4

CHAPTER 3 BACKGROUND 10

3.1. Taxonomy Of Data Mining 10

3.2. Definition Of Data Mining 12

3.3. Fundamentals Of Data Mining 14

3.4. Data Mining Approaches 15

3.4.1. Statistical Approaches 16
3.4.2. Machine Learning Approaches 16
3.4.3. Database-Oriented Approaches 17
3.4.4. Neural Network Approaches 17

3.5. Implemented Algorithms 17

3.6. Weka Program 23

CHAPTER 4 RESEARCH 24

4.1. Factory Information 24

4.2. Manufacturing Process 25

4.3. Explanation Of Data 27

4.4. Implementation 31

4.5. Evaluation 31

CHAPTER 5 CONCLUSION AND FUTURE WORK 39

REFERENCES 41

APPENDIX A – Group1 Algorithm Results..45

APPENDIX B – Group2 Algorithm Results..50

APPENDIX C – Java Code 55

LIST OF FIGURES

Figure 1 Data Set Explanation 2

Figure 2 Literature Review Structure..4

Figure 3 Evolution of Data Science 11

Figure 4 KDD Process 13

Figure 5 Sample Products 25

Figure 6 Production Line Sample 26

Figure 7 Data Set Explanation 29

Figure 8 Java Imported Packages..32

Figure 9 Group 1 Version 4 NaiveBayes - ROC Curve..................................36

Figure 10 Group 1 Version 4 NaiveBayes - ROC Curve..................................36

Figure 11 Group 1 and Group 2 Version 4 NaiveBayes Results 37

Figure 12 Java Code..63

LIST OF TABLES

Table 1 Production Groups 28

Table 2 Data Set Example Part 1 29

Table 3 Data Set Example Part 2 30

Table 4 Data Set Example Part 3 30

Table 5 Data Distribution of Group 1 30

Table 6 Data Distribution of Group 2 30

Table 7 Version Creation 31

Table 8 Confusion Matrix 33

Table 9 Confusion Matrix 33

Table 10 Group1 Best Algorithms Results 34

Table 11 Group2 Best Algorithms Results 35

Table 12 Results for Group1 and Group2 36

Table 13 Group 1 Algorithm Results 45

Table 14 Group 2 Algorithm Result 50

SYMBOLS AND ABBREVIATIONS

ABBREVIATIONS:

WR	Wastage Rate
KDD	Knowledge Discovery Database
LWL	Locally Weighted Learning
MLP	Multilayer Perceptron
IBK	Instance Based Learning
AM	Agent Miner
MRD	Mobile Agent Resource Discoverer
MADM	Mobile Agent Decision Maker
SMO	Support Vector Machine

CHAPTER 1
INTRODUCTION

Wastage is one of the most important factors affecting production efficiency. It is very essential to know the variables affecting the rate of waste and to estimate effects of these variables to production. In addition, defining variables could increase the production efficiency and process flows. Actions should be taken to reduce the rate of wastage. Reducing the rate of wastage means increasing efficiency in production. Companies accept wastage to some extent due to dynamism of production. There are two wastage type in production such as setup wastage and production wastage. Setup wastage is not changeable without changing resources. It occurs while preparing materials and machines to the production. This part forms fifteen percent of the total wastage in this study. Therefore, the acceptable wastage is fifteen percent. This thesis aims to predict production wastage amount for a packaging company. Wastage rate is defined as fifteen percent for the first analyze. The study investigates the wastage exceeds fifteen percentage. According to study results, the company can focus its production parameters to decrease wastage.

The predicted result is specified as yes, no. In the second analyze, the range is divided to three categories to make prediction more specific which are S, M, L.

There are two main data sets. First data set is used for the first analyze and second data set is used for the second analyze. These data sets have same attributes except the predictable attributes which are Group1 and Group2.

Group1 has yes and no values. If wastage rate (WR) is smaller than 0.15, value is “NO” otherwise “YES”. Group2 has S, M, L values. If wastage rate is smaller than 0.14, value is “S”. If WR is between 0.14 and 0.16, value is “M”. Finally, if WR is bigger than 0.16, value is “L”.

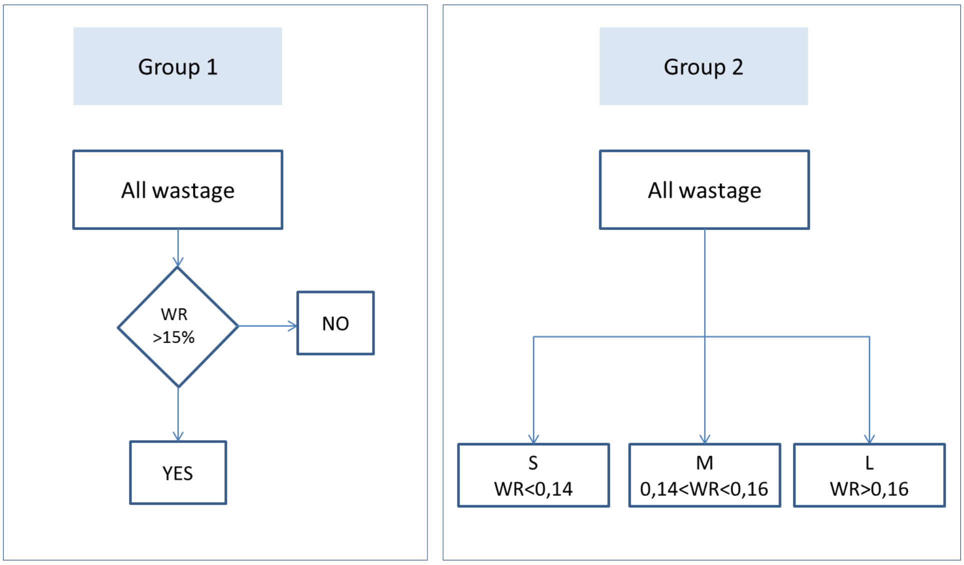


Figure 1. Data Set Explanation

Moreover, there are ten different versions of these groups. Each version is created by selecting different columns. All columns are covered in first version. Different columns are removed to generate other versions.

Twenty algorithms are applied to twenty data sets by using Weka. Total accuracy, true positive, true negative, ROC curve, precision and recall are calculated for the analyze.

Firstly, data collection and information process of manufacturing and the structure explained in detail in chapter one. Literature review takes part in chapter two. Background information is explained in chapter three. Historical background of data mining is explained in first part of the background. After that, fundamentals of data mining are specified. Following, data mining application approaches are mentioned such as; statistical, machine learning, database-oriented and neural network. Algorithms that are applied in this thesis are explained briefly which are AdaBoostM1, Bagging, BayesNet, ClassificationViaRegression, DecisionTable, HoeffdingTree, IBK (Instance Based Learning), J48(Decision Tree), JRip (Ripper), LogitBoost, LWL (Locally Weighted Learning), MultiClassClassifier, MultiScheme, NaiveBayes, NaiveBayesMultinomialText, NaiveBayesUpdateable, RandomForest, RandomTree, SMO (Support Vector Machine), Vote.

The WEKA program which is used for applying and executing algorithms is explained in last part of the background. The research is explained briefly in chapter 4. In addition, necessary information about factory, manufacturing process, collected data and evaluation are explained in chapter 4. Necessary information about analyzed data can be found in appendixes section. Results of the execution of algorithms and future works about the research is shown in chapter five.

CHAPTER 2

LITERATURE REVIEW

The literature review generally covers two topics, faults in production systems and data mining strategy which are illustrated in figure 2. The first topic involves an overview of the faults in production systems whilst the second topic involves data mining strategies. The intersection between the two topics which is decreasing faults in production systems by using data mining strategies will be the main criteria to meet the researches objectives.

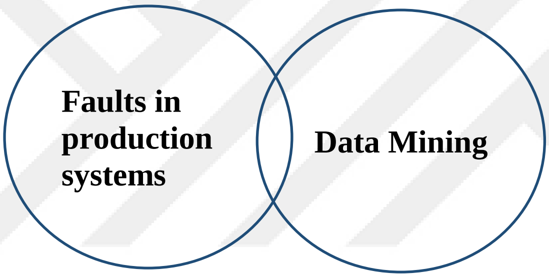


Figure 2. Literature Review Structure

2.1. Faults in Production Systems

To begin with, there are three specialties in manufacturing enterprises to survive in the global market which are quality, cost and cycle time in which the quality is the most critical factor among all. Using data mining for quality improvement of the production process is focused (He, Zhen, Wang, & Li, 2009). This paper proposes a knowledge-based quality improvement model. The model applies data mining process to the mass data collected from the manufacturing process. One of the goals of the proposed model is to improve the quality performance of manufacturing process.

In addition, new competitive enterprise domain reveals new trends in recent years such as increasing product complexity, changing customer requirements and decreasing production time. These challenges becloud the function of quality, improvement and prediction. As a result, the number of unexpected events is exploding. Because of these reasons, new tools are needed. The researchers (Menon, Tong, Sathiyakeerthi, Brombacher, & Leong, 2004) investigate the use of textual data mining to manufacture quality and production accuracy within the product development process. However, with efficient data collecting and data analyzing methodologies making improvements and accurate predictions can be possible.

Furthermore, increasing complexity of engineering systems causes reliability to decrease. The reliability of a manufacturing systems can be defined by fault tree analysis (FTA). Flexible Manufacturing system (FMS), which is using FTA, proposed in Analysis of a Flexible Manufacturing System article (Mahmood, Karaulova, Otto, & Shevtshenko, 2017). Basically, FMS can adapt itself to different kind of situations and can identify and differentiate the different incoming stages or products. Researchers focus on flexibility on proposed method for improving performance and competitiveness.

Moreover, not only are the system’s input and output statuses are fallowed in the control of production fault rates are considered, but also the system’s real time statuses are fallowed. There are some critical points proposed as a result of this research, when the fault source has a weak correlation the control chart performance is very good; when the fault source has a strong correlation, although the control chart performance decreases slightly, the alarm occurs quickly; when the correlation coefficient of the fault source is the same, the greater the offset of the fault source is, the faster the control chart creates alarm (Deng, Zhou, Liu, & Lu, 2018).

2.2. Data Mining Strategy

First of all, data mining strategy “the assemble-to-order” aims to reduce customer-waiting time for the product delivery. Apart from basic quality testing implications, “the assemble-to-order” strategy uses production data to determine the assembly sequences to minimize the risk of production faulty. The objectivity of “Zero fault” is aimed in Cunha, Agard and Kusiak’s paper (Cunha, Agard, & Kusiak, 2010). For this study, it is essential to find admissible past performance of production system. Data mining strategy is used to extract valuable information from large data sets to decrease the production faulty. The data mining strategy designed in this paper is well suited when the customer tolerance for product delivery time is low.

According to Witten, Frank & Hall (2011) the bigger data means the more difficult to understand the data. In the old times, hunter tried to figure out animals' patterns or farmers seek patterns in crop growth. Today we have digitalized huge amount of data and we are focusing on defining valuable patterns of that data. Data mining is an essential way to understand and define patterns of data. It is about improving situations by testing the data already stored in databases. The process considered by data mining must be automatic. The patterns must be meaningful. By this way, scientists process the data and turn it into valuable information. The outcome can be useful to predict or solve predefined problems (Witten, Frank, & Hall, 2011). Furthermore, because of the gigantic number of rules produced by data mining algorithms, there are needs to develop visualization framework for clear understanding. The visual data-mining framework known as Opportunity Map is presented (Zhao, Liu, Street, Tirpak, & Xiao, 2005). This framework provides an opportunity to identify useful knowledge in Quality Engineering

Because of the complexity of knowledge extraction from large databases, some of the data decomposition methods are discussed (Kusiak, 2000). Decomposition process defines the character of data mining task. Defined knowledge extraction methods are provided by data mining approach and used for prediction and prevention of manufacturing faults in wafers. There are key facts for implementing data mining process which are;

- understanding the application domain,
- eliminating target data set that is stored in computer systems,
- data preprocessing (handling unknown values),
- extracting patterns from data,
- process extracting data and convert it to human readable version,
- discovery of knowledge from extracted data.

2.3. Decreasing Faults in Production Systems by Data Mining

Firstly, data mining is used for diagnosing faults and managing manufacturing process in Chien, Wang & Chen (2007) study. Required data is collected and stored in databases while monitoring the manufacturing process. Researchers aim to solve the problems which the engineers cannot find the root causes of defects. The study proposed to develop a framework for data mining and knowledge discovery from the pre-collected database. In addition, Gebus and Leiviski (2007) show that how defect-related knowledge on the production line can be combined with decision-making process on real-time.

Niaki & Abbasi (2005) combines statistical methods and data mining followed by applying them to the quality improvement of production lines (Niaki & Abbasi, 2005). The proposed model is based on an artificial neural network to diagnose faults in out-of-control conditions and to help clarify abnormal variables when multivariate control charts based on Hotelling's T-square is used. The research focuses on how defect-related knowledge on the production line can be combined with decision-making process on real-time operations. It is needed to deal with irrelevant data while optimizing and diagnosing the production line in electronics assembly line.

Moreover, every collected data for applying data mining for improving manufacturing quality has unique attributes. Example, unbalanced distribution of the target attribute or a small training set relative to the number of input features. Regular data mining techniques cannot adapt to this situation although it is being used to control production quality. A new algorithm called Breadth-Oblivious-Wrapper (BOW) for that situations has been represented. The algorithm achieves breadth-first search while using a new F-measure splitting criterion for multiple oblivious trees. This paper focuses on mining quality-related data in manufacturing (Rokach & Maimon, 2006).

Mere & Ascacibar (2005) highlight how data mining can be applied to some typical industrial environments. It emphasis difficulties of research and how the result should be improvement. Data mining is an application strategy that comes from the knowledge discovery. Data mining focuses on analyzing data and finding regularities and patterns in it. It is mentioned that data mining is a discipline and process that must be mastered, it is not a product that can be bought. Data mining is a completely problem-solving cycle. In addition, U. Fayyad, Piatetsky-Shapiro, & Smyth (1996) focus on to identify how data mining and knowledge discovery in databases are related to each other. Because of the need of extracting knowledge from growing digital data, knowledge discovery is an emerging subject. Main understanding of knowledge extraction is gathering low-level data like production statistics and different variables and convert them to more compact, abstract and useful ones.

There are some basic applications of data mining such as fault detection, maintenance, and quality improvement (Harding, Shahbaz, Srinivas, & Kusiak, 2006). Data mining and manufacturing industry are connected to each other. The connection comes from the characteristics of complex databases. Characterization process is known as knowledge extraction from data. Knowledge can be difficult to gather and operate. Data mining methodologies provide researchers to manage that operation. Some of data mining models for manufacturing process is mentioned on several papers like CRISP-DMTM, (Cross Industry Standard Process for Data Mining), SEMMA, SolEuNet (Data Mining and Decision Support for Business Competitiveness: A European Virtual Enterprise), Kensington Enterprise Data Mining (Imperial College, Department of Computing, London, UK), and Data Mining Group. (Neaga, E. I., and Harding, J. A., 2002) (Neaga, E. I., and Harding, J. A., 2001)

Data mining method is developed for understanding, analyzing and conditioning the big amount of digital data. Data mining is the process of characterizing valuable patterns in large databases. It is the core part of knowledge discovery in database (KDD) process. There are steps in KDD such as data selection, data cleaning, data transformation, pattern searching, finding the presentation, finding interpretation and finding evaluation. Most of the time data mining and KDD are being used together to resolve improvement issues. Data mining has also some distinct tasks; summarization, classification, clustering, association and trend analysis. Also, data mining can adapt other research fields techniques like; statistical approaches, machine learning, database systems, neural networks, rough sets and visualization (Fu, 1997).

Knowledge discovery and data mining methodologies reviewed (Choudhary, Harding, & Tiwari, 2009). Characterization and description, association, classification, prediction, clustering and evolution analysis are labeled as the major data mining functions. There are very narrow financial margins that specify between success and failure. Data mining is a must technique for achieving low-cost production and it is also essential to clarify which industries need to produce and sell at global market. However, for reaching these levels of manufacturing, companies need to understand the identification of knowledge in big databases generated during production.

Outlier detections are one of the first steps in data mining applications. Outliers are considered as the noise of the data (unwanted or error). An exact application of outlier depends on assumptions of the derived data's detection method. There are outlier methods reviewed, these are; univariate, multivariate and parametric vs. nonparametric procedures. Because of the disjoint sets of assumption, a comparison between the methodologies is not always suitable (Ben-gal, 2005).

Last but not least, states the Waikato Environment for Knowledge Analysis (WEKA) workbench and also provides a beginning point. The tool is not only providing learning algorithms, it is also a framework inside which researchers could implement new environments without having data manipulation. Nowadays WEKA is recognized as a landmark system in data mining and machine learning. It allows users to implement their process and compare different algorithms in flexible data sets. Regression, classification, clustering, association rule mining and attribute selection algorithms can be found in the workbench. In addition, acceptance of a variety of data gives the real power to the WEKA.

WEKA is a collection of machine learning algorithms for solving real-world data mining problems. It is written in Java and runs on almost any platform. The algorithms can either be applied directly to a dataset or called from Java code as a library. (Hall et al., 2009)

CHAPTER 3
BACKGROUND

3.1. Taxonomy of Data Mining

Archaically, the method of analyzing valuable patterns in data has been given the variation of names, knowledge extraction, information discovery, data archaeology, etc. Statisticians have mostly use the term data mining (U. Fayyad, Piatetsky-Shapiro, & Smyth, 1996). However, the term of knowledge discovery in databases is (KDD) firstly used in the IJCAI-89 workshop (Piatetsky-Shapiro, 1990).

KDD continues to improve from the junction of different research fields such as pattern recognition, machine learning, statistics and AI. Data mining is one of the steps of KDD process. This fundamental element relies on known techniques from statistics, machine learning, and pattern recognition.

KDD targets on the overall process of knowledge discovery from data, this means how it is stored and accessed efficiently also how results can be visualized. On the other hand, data mining is a result of a natural development of information technology.

Considerable functionalities participate in information technology since it begins data collection and database creation, data management and advanced data analysis.

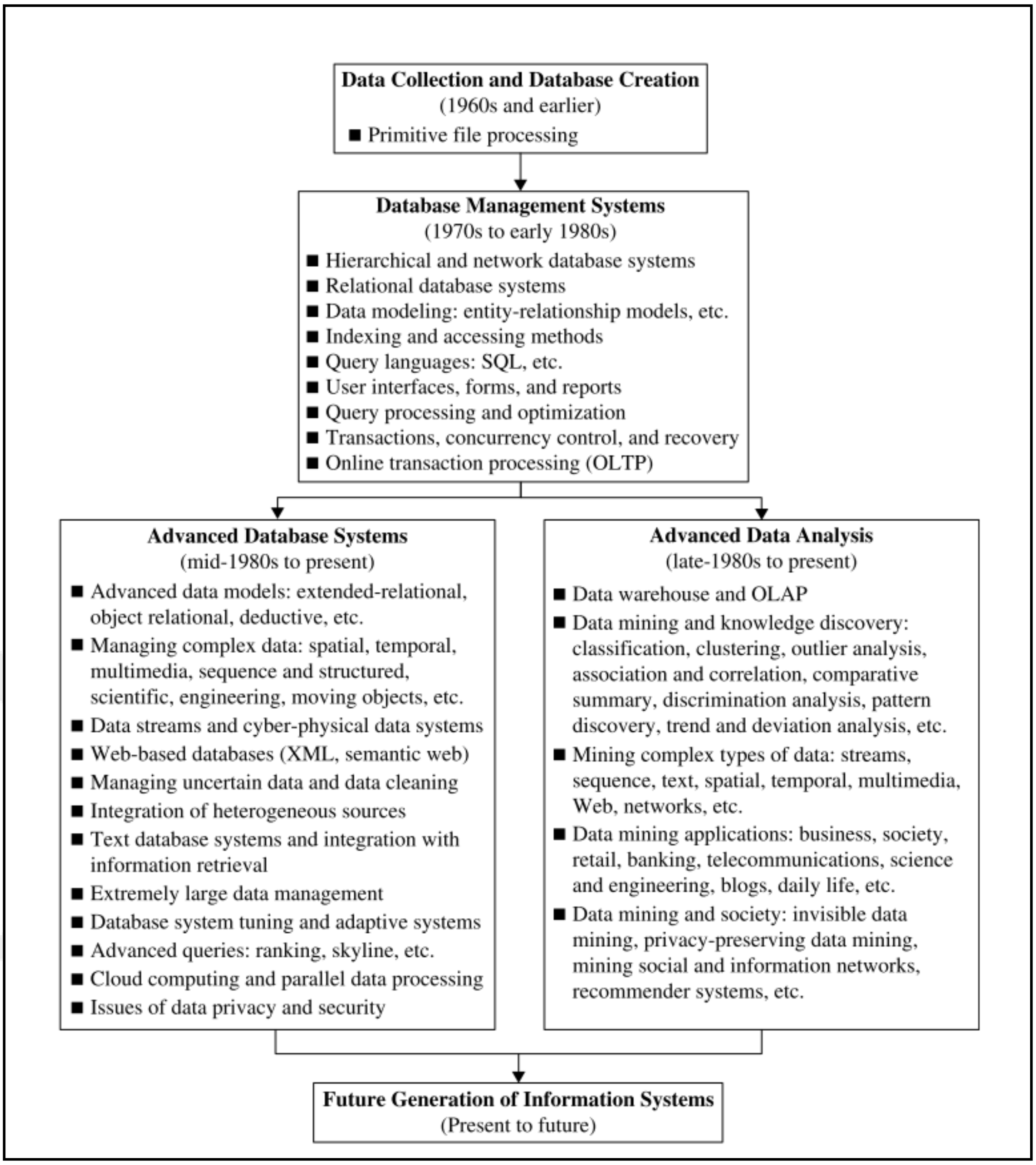

Data Collection and Database Creation
(1960s and earlier)
■ Primitive file processing
Database Management Systems
(1970s to early 1980s)
■ Hierarchical and network database systems
■ Relational database systems
■ Data modeling: entity-relationship models, etc.
■ Indexing and accessing methods
■ Query languages: SQL, etc.
■ User interfaces, forms, and reports
■ Query processing and optimization
■ Transactions, concurrency control, and recovery
■ Online transaction processing (OLTP)
Advanced Database Systems
(mid-1980s to present)
■ Advanced data models: extended-relational, object relational, deductive, etc.
■ Managing complex data: spatial, temporal, multimedia, sequence and structured, scientific, engineering, moving objects, etc.
■ Data streams and cyber-physical data systems
■ Web-based databases (XML, semantic web)
■ Managing uncertain data and data cleaning
■ Integration of heterogeneous sources
■ Text database systems and integration with information retrieval
■ Extremely large data management
■ Database system tuning and adaptive systems
■ Advanced queries: ranking, skyline, etc.
■ Cloud computing and parallel data processing
■ Issues of data privacy and security
Advanced Data Analysis
(late-1980s to present)
■ Data warehouse and OLAP
■ Data mining and knowledge discovery: classification, clustering, outlier analysis, association and correlation, comparative summary, discrimination analysis, pattern discovery, trend and deviation analysis, etc.
■ Mining complex types of data: streams, sequence, text, spatial, temporal, multimedia, Web, networks, etc.
■ Data mining applications: business, society, retail, banking, telecommunications, science and engineering, blogs, daily life, etc.
■ Data mining and society: invisible data mining, privacy-preserving data mining, mining social and information networks, recommender systems, etc.
Future Generation of Information Systems
(Present to future)

Figure 3 Evolution of Data Science

While evolving continues, all developments improved in the chain reaction. Researches on data collection and database creation methodologies serve as an essential for subsequent discoveries like data storage, data retrieval, query and transaction processing.

First, simple file processing systems derive to practical and authoritative database systems since the 1960s. Second, database systems evolve hierarchical and network database systems to relational database systems. In addition, data indexing and modeling tools and accessing methods flourish with query languages. After creation of database management systems, data warehousing and data mining become popular. Online analytical processing (OLAP) tools are useful for summarizing information from different perspectives. During the 1990s, the World Wide Web and web-based databases appeared. Internet-based global information bases like various kinds of interconnected databases play a critical role in the industry. Meanwhile the word “The world is data rich but information poor” became meaningful. Today, a huge amount of stored data become “data tombs”. The enlarging gap between data and information must be reduced via data mining tools.

3.2. Definition of Data Mining

Data mining is a way of analyzing patterns and getting the information from the size of the databases. Data mining is an interdisciplinary field of research that is part of many areas including information discovery. All the way to understand, analyze and use the data in a large quantity passes through data mining (Fu, 1997).

All kind of stored data is an inconsequential data without knowledge discovery. It is the essence of data mining. Discovery of information is a domain of need for computing theories and tools to help people extract valuable information from ever-increasing quantities of digital data. This field is known as knowledge discovery in databases (KDD). KDD is the method for dealing with making data more understandable. The challenge of KDD is a process for mapping low-level data (any kinds of the report or monitored data of production line). KDD is the process of characterized valid, peculiar, useful and understandable patterns in data (U. Fayyad et al., 1996). The KDD process may apply the following steps; data selection, data cleaning, data transformation, data mining, finding presentation, finding interpretation and finding evaluation.

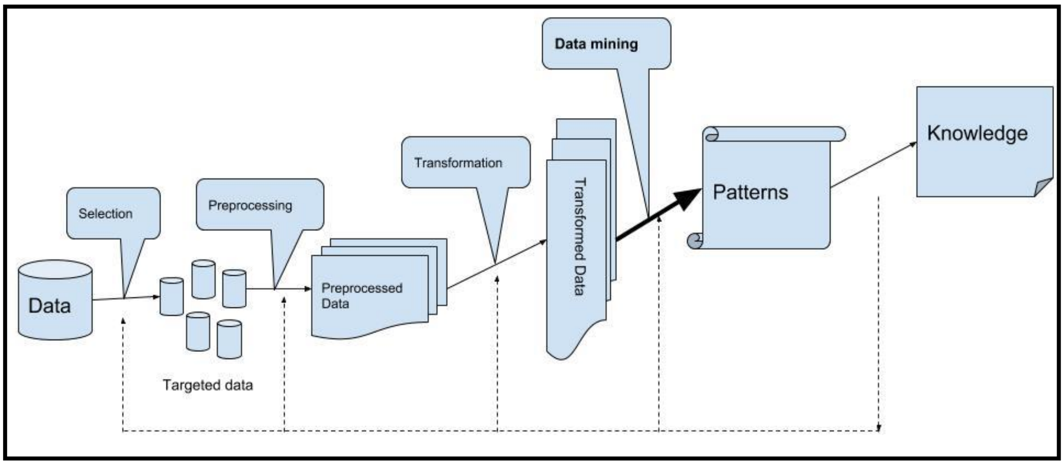


Figure 4 KDD Process

First visualization of KDD process produced by (Brachman & Anand, 1994)

1. First step is applying KDD methodology to the system is understanding the data and identifying the goal of process.

2. Second step is focusing targets; data samples, variables.

3. Third is preprocessing the data; this step includes data cleaning like removing noise, collecting necessary information and deciding strategies for data fields.

4. Forth is data contraction and estimation, discovering valuable features for goal of the task.

5. Fifth is reaching the goals of KDD process to a peculiar data-mining method.

6. Sixth is choosing which data mining techniques can be applicable for existing material. Searching for data patterns and defining characterization of process is the main point of this step.

7. Seventh is defining for patterns of interest in a particular representational form.

8. Eighth is visualization of processed patterns. This step can cause to start the process again with respect to defined iterations.

9. Ninth is defining behavior for discovered knowledge using the outputs.

KDD process includes flexible application methodologies. Operators can execute loops between any two steps. For successful KDD process, all steps should have been implemented carefully.

3.3. Fundamentals of Data Mining

Data mining is a profound topic that share interests with many other research areas like neural networks and machine learning. However, data mining has distinct methodological application tasks. These are summarization, classification, clustering, association and trend analysis (Chen, Hun, & Yu, 1996).

- **Summarization**

Summarization is the generalization of data. It is an application dependable process. This task is used for resulting a smaller set which gives a general overview of data. It includes methods for discovering a solid description for a subset of data. Summarization rules are characterized for derivation of rules (Zaki, Parthasarathy, Ogihara, & Li, 1997). Different combinations of abstraction levels and dimensions acknowledge various kinds of patterns and consistency.

- **Classification**

Specialization of model or function, which determines the class of objects, based on its attributes. Classification is learning behavior that maps a data item to several predefined parts (U. M. Fayyad, Djorgovski, & Weir, 1996). A classification function is produced by analyzing the relation between the attributes and the classes in training set. These objects can be used in classifying future objects and improve a better understanding of the classes in the database. Examples of classification methods used as part of knowledge discovery applications include the identification of trends in financial markets (Apté & Hong, 1994) and the automated identification of objects of interest in large image databases.

- **Association**

Association is the searching for closeness and relation of objects. Such kind of connection is called as association rule. An association rule explains the associative contact among the objects. Association techniques are often applied to interactive data analysis. Association rules in industrial processes can provide useful knowledge to explain industrial fault (Martínez-De-Pisón, Sanz, Martínez-De-Pisón, Jiménez, & Conti, 2012).

- **Clustering**

Clustering is basically grouping the objects that have same specialties. Clusters are specified by considering their similar features. Each cluster has relation analogous. Identification of clusters helps the executives to understand similar situations better because same results will be in the same cluster. There will be services which are more advisable. Defined categories can be mutually exclusive and exhaustive or consist of a richer representation, such as ranked or overlapping categories.

- **Trend analysis**

Time series data means that data is evaluated with time, like credit card transactions or stock prices. Most of the data records are recorded with time stamps. Such kind of data can be seen as objects with a variable time, and the objects are the snapshots of data with values that changes over time. Finding patterns and regularities is an important action in the data evolutions along the time. Trend analysis discovers impressive patterns in the progression history of the objects. Object's evolution is an important topic in trend analysis. Once up, down, peak, etc. are defined in the data next step is predicting the future behaviors of the object.

Another topic in trend analysis is focusing on changing trends in data with increasing, decreasing streaks, etc. By comparing objects with looking at their time series events with similar or different trends can be discovered and it definitely helps researchers to understand behaviors of the objects.

3.4. Data Mining Approaches

There are four data mining techniques with different approaches. These are, statistical approaches, machine learning approaches, database-oriented approaches and neural network approaches.

3.4.1. Statistical Approaches

Bayesian network, regression analysis, correlation analysis and cluster analysis are the base statistical research areas which are related to data mining. Most of the time statistical approach processes are developed from a set of training data. At first, an optimal model based on measures is developed in the hypothesis stage. Rules, patterns and regularities are extracted from the optimal model. The Bayesian probability theorem is used in a Bayesian network. It is a directed graph that represents the relationships between the defined variables (Bouckaert, 2004). The derived regression is a function, which points a set of variables of objects to an output variable. Correlation analysis examines the correspondence of variables to each other. Cluster analysis investigates groups which are a set of measurable objects. In the graph nodes in a Bayesian network shows variables or states while edges represent the relations between nodes, directed from the cause to the effect.

3.4.2. Machine Learning Approaches

Machine learning methods search for the best model that is related with the processed data like statistical methods. The difference from the statistical approaches is that the searching space in machine learning approaches is a cognitive space of n attributes instead of a vector space of n dimensions. Moreover, most machine-learning methods use heuristics in the searching. Decision tree induction, inductive concept learning and conceptual clustering are most common machine learning methods used for data mining. A decision tree is a characterization tree, which specifies an object’s class by tracing the path from the root to a leaf node, branch-choosing process performing to the attribute values of the object. Decision trees are analyzed from the training set and classification rules can be executed from the decision trees. Inductive concept learning is the task of learning to assign cases to a discrete set of classes. Groups or clusters in a set of objects can be found by conceptual closeness between objects based on conceptual clustering.

3.4.3. Database-Oriented Approaches

Database-oriented approaches are different from the other two approaches. Data model or database specific heuristics are used to discover the characteristics of the data stored. The attribute-oriented induction, the iterative database scanning for frequent item sets, and the attribute focusing, are classics of the database-oriented methods.

In attribute-oriented induction, classic data are generalized into high-level data using theoretical hierarchies. The iterative database scanning method searches frequent item sets in a transactional database. The association rules are then collected from these frequent item sets. The attribute orientation method searches for patterns with unusual possibilities by adding attributes to attributes, depending on the selection.

3.4.4. Neural Network Approaches

A neural network is a set of inter connected nodes, called neurons. A neuron is a simple computing device that executes a function of its inputs, which can be outputs of other neurons or attribute values of an object. By processing the connection and the parameters of the neurons, a neural network can be trained to model the relationship between a set of input attributes and an output attribute. It can be used in classification.

3.5. Implemented Algorithms

Algorithms which are applied to 20 datasets are discovered in this part.

- **J48 - Decision Tree**

Decision Tree Algorithm is trying to find out the way the characteristics of vector behaves for several instances. J48 is an enlargement of Iterative Dichotomiser 3 (ID3) algorithm. Decision trees pruning, continuous attribute value ranges, derivation of rules and accounting for missing values are additional features of J48. This algorithm can be found as open source code in WEKA.

Steps of the algorithm:

1. First, the instance belongs to the same class, the tree represents a leaf, so it is returned by labeling with the same class.

2. Second, there is a potential information calculation for every character, given by a test on the attribute.

3. Finally, the best attribute is found based on the present selection criterion and that attribute is selected for branching.(Kaur & Chhabra, 2014)

- **Random Tree & Random Forest**

Random trees are first published by Leo Breiman and Adele Cutler (Breiman, 1999). Random tree algorithm can handle both regression and classification problems. Random forest is an ensemble of tree predictors that is called forest. The classification works as follows; the random tree classifier takes the vector of input properties, classifies it with every tree in the forest, and extracts the class label that takes the majority of votes. In case of a regression, the classifier response is the average of the responses over all the trees in the forest. In random trees, there is no need for any accuracy estimation procedures, such as cross validation or bootstrap, or a separate test set to get an estimate of the training error. The error is estimated internally during the training.

- **HoeffdingTree**

There are few predefined agents for this algorithm. The agents are defined in the following list:

1. Agent Miner (AM) is an atomic unit that implements a data-mining algorithm and trains a model on a local data source.

2. The Mobile Agent Resource Discoverer (MRD) searches the network for data streams and available AMs relevant for the data-mining task.

3. The Mobile Agent Decision Maker (MADM) select AMs that have been discovered in previous step and retrieves information about the data-mining task.

The Hoeffding tree algorithm has been designed for classifying high-speed data streams. AM runs a Hoeffding tree algorithm and the MADMs are used to collect the classification results. The owner of each AM may have subscribed to a certain subset of the data stream, or to different features of the same data.

The basic scenario is given in the following steps:

1. Start the MRD agent to discover classifier AMs with a for the classification task-relevant data stream.

2. When the MRD agent returns, an MADM is started that loads the unlabeled data instance for classification.

3. On each visited AM, the MADM asks the AM to predict the classification of its unlabeled instances and retrieves the AM's weight or estimated accuracy.

4. The MADM returns to the task initiator and performs a weighted majority voting for each unlabeled data instance using each AM's prediction and weight to give the final prediction (Stahl, Gaber, Bramer, & Yu, 2011).

- **Naïve Bayes**

Naïve Bayes is the probabilistic classifier based on applying Bayes Theorem with strong independence assumptions, which assumes all of the features are equally independent. It uses a Bayesian algorithm for the total probability procedure. (Yuan, 2010). Naive Bayes is useful for very large data sets. Naive Bayes classifiers works quite well in many real-world situations such as document classification and spam filtering.

- **Bayes Network**

A Bayesian network is a structure that shows conditional dependencies between field variables and shows probabilistic relationships between field variables. The Bayesian network consists of directed acyclic graphs and probability tables. Network nodes represent domain variables, and an arc between two nodes indicates the existence of a basic relationship or dependency between these two nodes. (Cayci & Eibe, 2011).

- **Naïve Bayes Multinomial Text**

Multinomial Bayes classifier is a specific instance of a Naïve Bayes classifier which uses a multinomial distribution for each of the features.

Multinomial Naïve Bayes for text data operates (and only) on String attributes. Other types of input attributes are accepted but ignored during training and classification.

- **JRIP(Ripper)**

JRip (RIPPER) is one of the basic and most popular algorithms. Classes are examined in growing size and an initial set of rules for the class is generated using incremental reduced error JRip (RIPPER) proceeds by treating all the examples of a particular decision in the training data as a class and finding a set of rules that cover all the members of that class. Thereafter it proceeds to the next class and does the same, repeating this until all classes have been covered (Rajput, Aharwal, Dubey, Saxena, & Raghuvanshi, 2011).

- **Decision Table**

A decision table is used in both test and requirement management. It is a structured exercise to plan requirements while dealing with complex data mining rules. Decision tables are used to model complicated logic. They can make it easy to see that all combinations of conditions have been considered and when conditions are missed, it is easy to see this.

Decision tables are similar to decision trees. Decision tables will always have the same number of conditions that need to be tested and actions that must be performed even if the set of branches being analyzed is resolved to true. A decision tree can have one branch with more conditions that need to be tested than other branches of the tree (Pollack, 1972).

- **SVM (Support Vector Machine)**

Classification, regression and outlier detection can be handled by Support vector machine's (SVM) supervised learning methods. There are some advantages and disadvantages of support vector machines which are;

Advantages are, suitable for high dimensional spaces where the number of dimensions is greater than the number of samples, memory efficient and different Kernel functions can be described for the decision function. Disadvantages are, number of samples is lower than number of features, choosing kernel functions and regularization term needs to be avoided, SVMs do not provide probability estimates. (Chang & Lin, 2013)

- **AdaBoostM1**

Boosting is an approach to machine learning based on creating an accurate prediction rule by combining many weak and inaccurate rules. The AdaBoost algorithm of Freund and Schapire was the first practical boosting algorithm, and remains one of the most used and studied, with applications in many fields. (Freund & Schapire, 1995).

- **Multi Class Classifier**

Supervised learning is a machine-learning task of inferring a function of certain variables to predict other variables. Classification problems are supervised learning problems for predicting variables that comprise a finite number of categories called classes. Of these problems, when the number of classes is larger than two, the classification problems are called multi-class classification problems. Most real-world scenarios, such as handwritten digit recognition, text categorization, and face recognition, correspond to multi-class classification problems.(Kang, Cho, & Kang, 2015).

- **LogitBoost**

AdaBoost algorithm is vulnerable while handling noisy data. To solve the problem, Friedman et al. (2000) proposed a binomial log-likelihood loss function, which changes linearly with the classification error and turns out to be less sensitive to noise and outliers. The optimization can be achieved by using Newton steps to fit an additive symmetric logistic model. LogitBoost algorithm is optimizing log loss instead of the exponential loss, due to that, it could be less sensitive to outlier (Cai, Feng, Lu, & Chou, 2006).

- **Bagging**

Bootstrap Aggregation (Bagging) is a technique that combines the predictions from multiple machine learning algorithms together to make predictions more accurate. Bootstrap Aggregation is a general procedure that reduces the variance for algorithms that have high variance.

- **Classification via Regression**

Executing classification by using regression.

- **Locally Weighted Learning (LWL)**

Lazy learning methods, delaying the processing of training data as much as an inquiry should be answered. This usually involves storing the training data in memory and finding relevant data in the database to answer a particular query. This type of learning is also referred to as memory-based learning. Relevance is often measured using a distance function. One form of Lazy Learning finds a set of nearest neighbors and selects or votes on the predictions made by each of the stored points. This is known as locally weighted learning. In most learning methods, a single global model is used to fit all of the training data. The query which is answered, is known during processing of training data (Englert, 2012).

- **Instance Based Learning (IBK)**

IBL finds the closest training instance in Euclidean distance to the test and predicts the same class as this training distance. If several instances qualify as the closest, the first one found is used. IBL algorithms do not create expansion concept descriptions. Concept descriptions are determined by how the selected similarity and classification function of the IBL algorithm uses the existing set of recorded distances. These functions are two of the three components in the following framework that describes all IBL algorithms:

1. Similarity Function: This calculates the similarity between training instances i and the instances in the concept depiction. Similarities are numeric valued.

2. Classification Function: This gets the similarity function’s results and the classification performance records of the instances in the concept description. It returns a classification for i.

Concept Description Updater keeps records on classification performance and decides which instances to include in the concept description. Inputs include “i”, the classification results, the similarity results, and a current concept description. It returns the changed concept description (Vijayarani & Muthulakshmi, 2013).

- **MultiScheme**

This algorithm can be efficient for selecting classifier from among several using cross validation on the training data or the performance on training data. Performance is calculated for the rate of correct classification or mean-square regression error. The algorithm has several options in WEKA implementation; cross validation for model selection can use the given number of folds, and classifier specification.

- **NaïveBayesUpdateable**

The algorithm class is defined for Naïve Bayes classifier by using estimator classes. This is the updatable version of Naïve Bayes. The classifier will use a default precision of 0.1 for numeric attributes when Classifier is called with zero training instances. Several implementation options can be found; using kernel density estimator or normal distribution for numeric attributes, using supervised discretization to process numeric attributes.

- **Vote**

The algorithm class combining other classifiers. Different combinations of probability estimates for classification are available in implementation. Given combination rule and classifier specification defines the way of algorithm.

3.6. WEKA Program

The Waikato Environment for Knowledge Analysis (WEKA) is a framework that allows researchers working with the most advanced level techniques in machine learning. WEKA workbench is implemented in JAVA programming language. WEKA project aims to provide extensive collection of data processing tools and machine learning algorithms to practitioners. WEKA is a milestone in the history of the data mining and machine learning research communities by adopting widespread algorithms inside. WEKA implements algorithms for data pre-processing, classification, regression, clustering and association rules; it also includes visualization tools. WEKA also provides an application-programming interface (API) and plug-in mechanisms for researchers implement their state of the art algorithms to study and test in the program.

CHAPTER 4
RESEARCH

3.7. Factory Information

The company is a Turkey-based company engaged in the manufacturing of packaging products. It produces printed, unprinted and laminated flexible packaging for various sectors in nine categories: Snack Foods Packaging, such as crispy potato chips, corn chips, extruded snacks and peanuts; Biscuits Packaging, including all kinds of biscuits, cookies, cracker, rusky bread and cakes; Sugar and Chocolate Confectionary encompassing chocolates, chocolate covered bars, wafers, cereal bars, candies and ice cream; Bread and Fresh Food Packaging, such as many different types of bread and bakery products, fresh vegetables; Instant Food & Drinks, including instant soup, boullion, pudding, powdered, drinks, baking powder, milk powder, ketchup, mayonnaise, various sauces; Dried Foods & Pasta Packaging, including legumes, dried fruits, pasta, macaroni, noodles, species, herbs, salt and sugar; Coffee & Tea Products, such as granule or roast coffee, tea and yeast; Chemicals & Hygiene Products, comprising detergent, fertilizers and diverse chemical products. Wet pipes, refreshing towels, napkins to pack in single, double and three-layer films and Other Product Packaging such as Frozen Food, Beverages, Pet Food, Meat and Dairy Products.

Figure 5 Sample Products

3.8. Manufacturing Process

The company produces films on customer order. A production order is created for each order. There are main pieces of information of order. Information such as customer name, product description, amount of order, amount of production, type of work, type of structure, number of layers product has, film types, number of colors, net pressure, bobbin width, number of cylinders / plate, which operations will be used to produce a product, which machine will be used to perform these operations, which operator will work in the operation, the amount of material consumed for the operation, and the amount of waste are the information of the production order.

The main materials are BOPP Films which can be used as HS, plain, clear, metallized, dull, solid white, cavitated white, coated; PET Films which are a variety of clear, metallized, coated, shrinkable, twistable; PE Films which are clear, white, LD, LLD, HD, EVOH, peel, paper like, twistable, antifog; PVC Films which can be used as clear, white, twistable, shrinkable; other films like CPP, BOPA, aluminum, paper etc.

Rotogravure printing, flexo printing, adhesive lamination, slitting – rewinding, micro-perforation, macro-perforation and bag making are the basic processes of production. These operations can be repeated in the production process according to the product to be produced. For example, if the product is different from one-layer work, the lamination operation is applied between each film layer. In addition, auxiliary operations may be included among these operations. In addition, the printing operation can be varied according to the product to be produced.

Cold seal, heat seal lacquer, matt lacquer, PVDC, acrylic, LTS, anti-fog, easy peel - burst peel, easy tear, high barrier combinations, dead fold and digital coding are the special features of production.

Depending on the product to be produced, it is decided which films will be used, which colors will be included, which features will be used, which machine will be operated, and which operations will be performed.

Cylinders or plates contain the design of the product. Cylinder is used in flexo-printing works, and plate is used in gravure printing works for the same purposes. Both are used for processing colors and other features.

A product can be printed or unprinted. If it is not printed, no cylinder or plate is used for color. In case of a printed job, the cylinder or plate is immersed in the paint box. If special features are applied, one of them is used depending on printing specialty. After immersing process, it is connected to the machine. The film to be used is also attached to the machine. Processing of the films is performed using all the components required for production. According to the structure of the product, the films are added to each other to obtain the final product. The structure of the final product shows all films which are used.

Figure 6 Production Line Sample

3.9. Explanation of Data

Research data, which is collected from production line, is described in this section. Data is stored in WEKA files (.arff). The data was modified for study. Final data set in the WEKA file has 23 columns and 2936 rows and it contains all the information of manufacturing process.

The data includes printed works wastage amount of 2016. These works are 2-layer productions.

Meanings of Column Labels:

MATERIAL: The material is used to describe the product, product code to be produced.

WORKCENTER1: It indicates on which machine the product transfer operation is performed.

WORKCENTER2: The product indicates on which machine the printing operation is carried out.

WORKCENTER3: Shows on which machine the slitting operation is performed on.

WORKCENTER4: The product indicates on which machine the lamination operation is carried out.

WORKCENTER5: The product indicates on which machine the packaging operation is performed.

TONNAGEGROUP: The tonnage group represents the grouped values of the work order quantities. If the amount of work order is 0-250, the tonnage group is 250. If the amount of work order is 250-500, the tonnage group is 500. If the amount of work order is 5000-1000, the tonnage group is 1000. If the amount of work order is 1000-2500, the tonnage group is 2500. If the amount of work order is 2500-5000, the tonnage group is 5000. If the amount of work order is 5000 denier, the tonnage group is expressed as 5001.

CUSTOMER: Customer is used to define which product belongs to which customer

WORKTYPE: There are two types of jobs. These are 0 and 1. 0 is used to define normal jobs. 1 is used to describe unified work orders. A combined work order states that more than one material from a client is produced at the same time.

MONTH: Specifies the month the production is completed.

FILMGROUP: Materials were grouped according to their structure and 3 main parts were taken as the basis for this study.

Table 1 Production Groups

Data	**Group name**
≤20 OPP_TRN + ≤20 OPP_MET	E
>20 OPP_TRN + ≤20 OPP_MET	F
≤20 OPP_MAT + ≤20 OPP_MET	G

FILMTYPE1: It shows that first film type used in production. It is first part of film group. Ex: OPP_TRN, OPP_MAT

WIDTH1: It shows the microns of the product in the FILMTYPE1

FILMTYPE2: It shows that second film type used in production. It is first part of film group. Ex: OPP_MET

WIDTH2: It shows the microns of the product in the FILMTYPE2

LAC: Indicates use of lac in production (1 used, 0 not used)

COLDSEAL: Indicates use of cold seal in production (1 used, 0 not used)

PRINTWIDTH: Net print is used to define the area to be printed on the minimum print size. A certain amount of film is always considered as waste.

REELWIDTH: Indicates the coil width

COLOR (Number of Colors): The number of colors is used to indicate how many different colors are used in the production of the product.

CYLINDER & PLATE (Number of Cylinder and Plate): Cylinder and Plate are used to include liquids to the production. Liquids can be color, liquid, lac, cold seal, etc. Cylinder is used in flexo printing works, and plate is used in gravure printing works**.**

Two data sets are created by using 23 columns which are defined above. GROUP1 column is added to dataset1. GROUP2 column is added to dataset2. The wastage rate distinguishes two groups from each other (WR: wastage rate).

Group1 has yes and no values. If WR is smaller than 0.15, value is “NO” otherwise “YES”.

Group2 has S, M, L values. If WR is smaller than 0.14, value is “S”. If WR is between 0.14 and 0.16, value is “M”. Finally, if WR is bigger than 0.16, value is “L”.

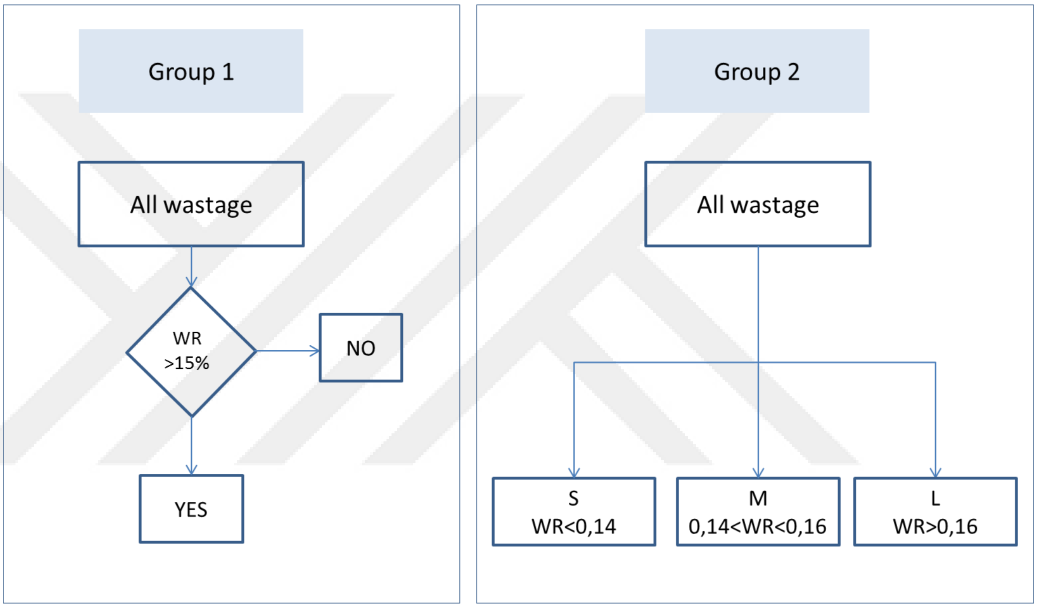
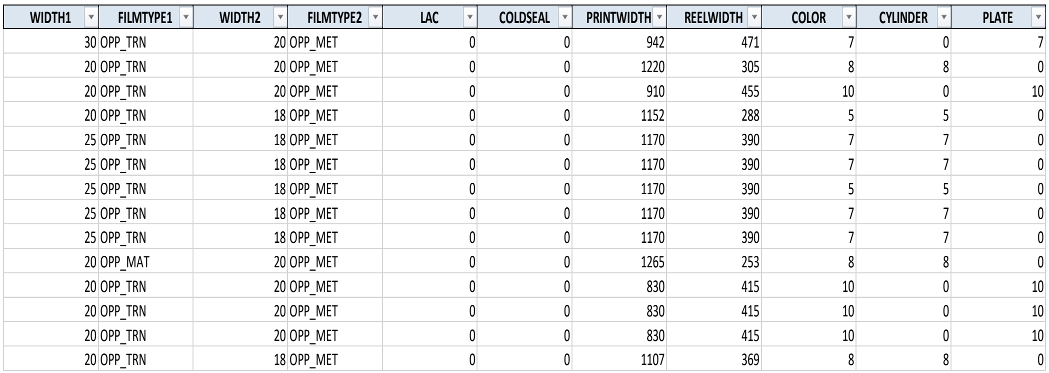


Figure 7 Data Set Explanation

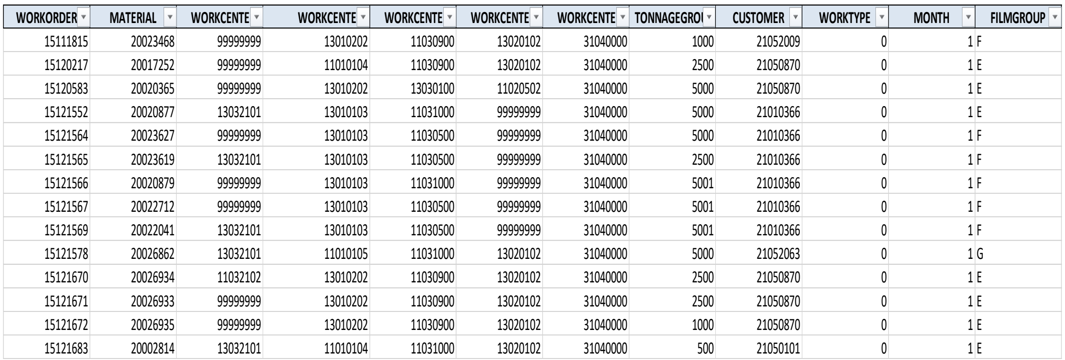
Data set example is showed in below.

Table 2 Data Set Example Part 1



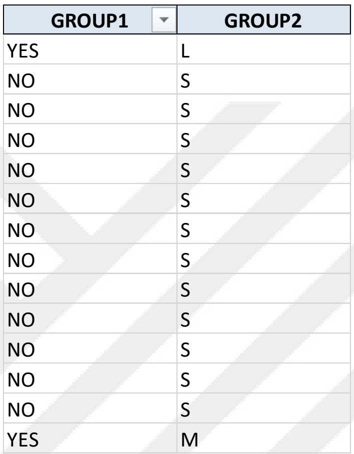
WIDTH1	FILMTYPE1	WIDTH2	FILMTYPE2	LAC	COLDSEAL	PRINTWIDTH	REELWIDTH	COLOR	CYLINDER	PLATE
30	OPP_TRN	20	OPP_MET	0	0	942	471	7	0	7
20	OPP_TRN	20	OPP_MET	0	0	1220	305	8	8	0
20	OPP_TRN	20	OPP_MET	0	0	910	455	10	0	10
20	OPP_TRN	18	OPP_MET	0	0	1152	288	5	5	0
25	OPP_TRN	18	OPP_MET	0	0	1170	390	7	7	0
25	OPP_TRN	18	OPP_MET	0	0	1170	390	7	7	0
25	OPP_TRN	18	OPP_MET	0	0	1170	390	5	5	0
25	OPP_TRN	18	OPP_MET	0	0	1170	390	7	7	0
25	OPP_TRN	18	OPP_MET	0	0	1170	390	7	7	0
20	OPP_MAT	20	OPP_MET	0	0	1265	253	8	8	0
20	OPP_TRN	20	OPP_MET	0	0	830	415	10	0	10
20	OPP_TRN	20	OPP_MET	0	0	830	415	10	0	10
20	OPP_TRN	20	OPP_MET	0	0	830	415	10	0	10
20	OPP_TRN	18	OPP_MET	0	0	1107	369	8	8	0

Table 3 Data Set Example Part 2



WORKORDER	MATERIAL	WORKCENTE	WORKCENTE	WORKCENTE	WORKCENTE	WORKCENTE	TONNAGEGROI	CUSTOMER	WORKTYPE	MONTH	FILMGROUP
15111815	20023468	99999999	13010202	11030900	13020102	31040000	1000	21052009	0	1	F
15120217	20017252	99999999	11010104	11030900	13020102	31040000	2500	21050870	0	1	E
15120583	20020365	99999999	13010202	13030100	11020502	31040000	5000	21050870	0	1	E
15121552	20020877	13032101	13010103	11031000	99999999	31040000	5000	21010366	0	1	E
15121564	20023627	99999999	13010103	11030500	99999999	31040000	5000	21010366	0	1	F
15121565	20023619	13032101	13010103	11030500	99999999	31040000	2500	21010366	0	1	F
15121566	20020879	99999999	13010103	11031000	99999999	31040000	5001	21010366	0	1	F
15121567	20022712	99999999	13010103	11030500	99999999	31040000	5001	21010366	0	1	F
15121569	20022041	13032101	13010103	11030500	99999999	31040000	5001	21010366	0	1	F
15121578	20026862	13032101	11010105	11031000	13020102	31040000	5000	21052063	0	1	G
15121670	20026934	11032102	13010202	11030900	13020102	31040000	2500	21050870	0	1	E
15121671	20026933	99999999	13010202	11030900	13020102	31040000	2500	21050870	0	1	E
15121672	20026935	99999999	13010202	11030900	13020102	31040000	1000	21050870	0	1	E
15121683	20002814	13032101	11010104	11031000	13020102	31040000	500	21050101	0	1	E

Table 4 Data Set Example Part 3



GROUP1	GROUP2
YES	L
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
NO	S
YES	M

The data counts for Group 1 and Group 2 are detailed below. There are 974 yes values, 1962 no values in Group1 and there are 1848 small values, 200 medium values, 888 large values in Group2.

Table 5 Data Distribution of Group 1

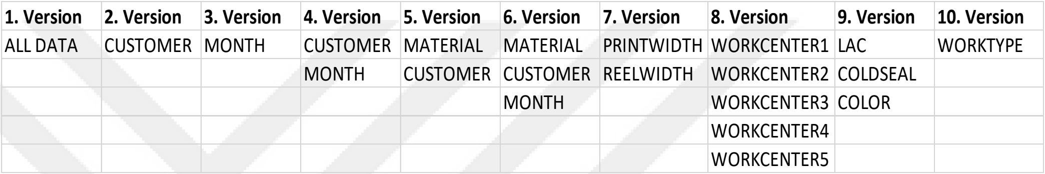
Group 1	
Yes	974
No	1962

Table 6 Data Distribution of Group 2

Group 2	
S	1848
M	200
L	888

Ten versions were created for each group. Each version is created by selecting different columns. All columns are covered in first version. Customer column is removed in the second version. Month column is removed in third version. Fourth version is created removing customer and month columns together. Material and customer columns are deducted in fifth version. Sixth version has data without material, customer and month columns. Print width and reel width columns are removed from seventh version. All work center columns are eliminated in eighth version. Lac, cold seal and color columns are removed from ninth version and finally work type columns is deducted in tenth version.

Table 7 Version Creation



1. Version	2. Version	3. Version	4. Version	5. Version	6. Version	7. Version	8. Version	9. Version	10. Version
ALL DATA	CUSTOMER	MONTH	CUSTOMER	MATERIAL	MATERIAL	PRINTWIDTH	WORKCENTER1	LAC	WORKTYPE
			MONTH	CUSTOMER	CUSTOMER	REELWIDTH	WORKCENTER2	COLDSEAL	
					MONTH		WORKCENTER3	COLOR	
							WORKCENTER4		
							WORKCENTER5		

Each version of groups has same columns. For example, version one has all columns. Group one and version one includes all columns and result sets which are “yes”, “no”. Group two and version one includes all columns and result sets which are “S”, “M”,” L”. Customer column was deleted from data set and it has named as version two. Group one and version two includes all columns except customer and result sets which are yes, no. Group two and version two includes all columns except customer and result sets which are “S”, “M”, “L”.

3.10. Implementation

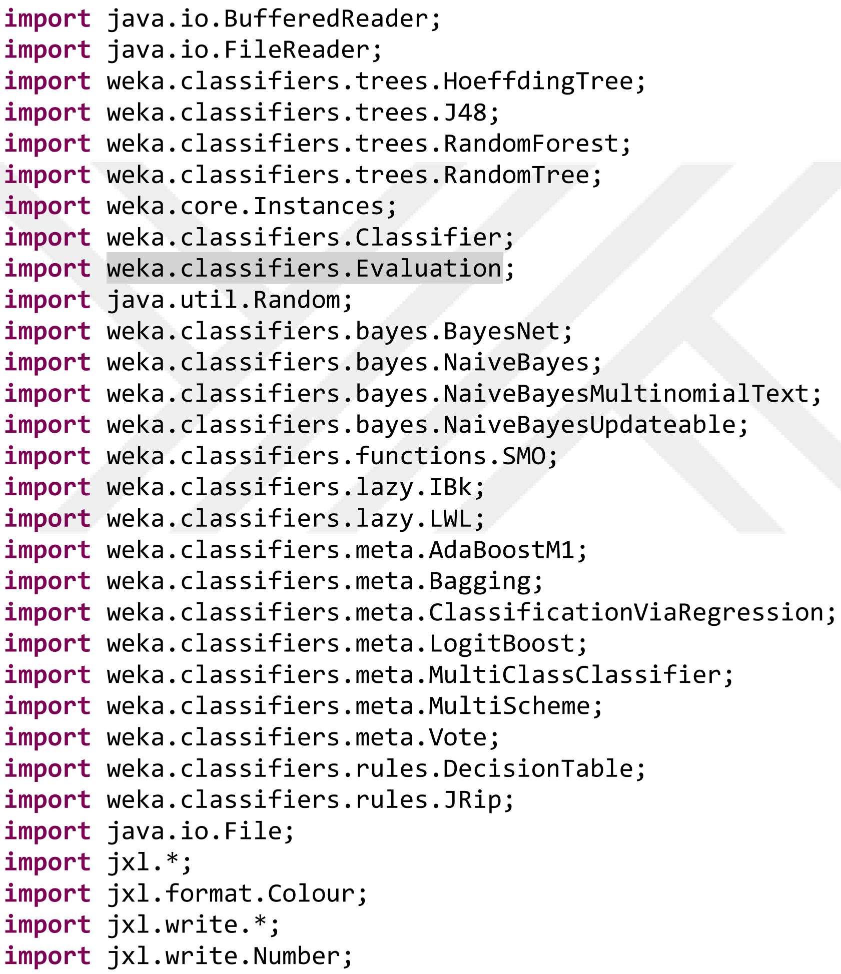
Twenty algorithms which are described above are applied to two data set groups and ten versions by using WEKA. Analyze is explained in evaluation part.

3.11. Evaluation

As it is explained in below chapters, wastage is one of the most important factors affecting production efficiency. It is very essential to know the variables affecting the rate of waste and to estimate effects of these variables to production. The study investigates the wastage exceeds fifteen percentage.

WEKA algorithms are applied to twenty data sets. To generate results Java is used as a programming language and Eclipse is used as IDE. WEKA API is included in Eclipse. All results are extracted to excel file based on algorithms and data sets.

Weka.jar and jxl.jar are added to the Eclipse library. To run algorithms in WEKA, required packages are imported which are shown below. All java code is in Appendix C.



```java
import java.io.BufferedReader;
import java.io.FileReader;
import weka.classifiers.trees.HoeffdingTree;
import weka.classifiers.trees.J48;
import weka.classifiers.trees.RandomForest;
import weka.classifiers.trees.RandomTree;
import weka.core.Instances;
import weka.classifiers.Classifier;
import weka.classifiers.Evaluation;
import java.util.Random;
import weka.classifiers.bayes.BayesNet;
import weka.classifiers.bayes.NaiveBayes;
import weka.classifiers.bayes.NaiveBayesMultinomialText;
import weka.classifiers.bayes.NaiveBayesUpdateable;
import weka.classifiers.functions.SMO;
import weka.classifiers.lazy.IBk;
import weka.classifiers.lazy.LWL;
import weka.classifiers.meta.AdaBoostM1;
import weka.classifiers.meta.Bagging;
import weka.classifiers.meta.ClassificationViaRegression;
import weka.classifiers.meta.LogitBoost;
import weka.classifiers.meta.MultiClassClassifier;
import weka.classifiers.meta.MultiScheme;
import weka.classifiers.meta.Vote;
import weka.classifiers.rules.DecisionTable;
import weka.classifiers.rules.JRip;
import java.io.File;
import jxl.*;
import jxl.format.Colour;
import jxl.write.*;
import jxl.write.Number;
```

Figure 8 Java Imported Packages

First, total accuracy, true positive, true negative and ROC curve are calculated and considered.

Table 8 Confusion Matrix

		Classified as	
		Yes	No
Actual	Yes	(TP)	(FP)
	No	(FN)	(TN)

True Positive (TP): Number of examples classified positive that are actually positive.

True Negative (TN): Number of examples classified negative that are actually negative.

Accuracy: The ratio of correctly classified instances to total number of instances.

Accuracy = (TP + TN)/(TP+TN+FP+FN)

ROC Curve: Receiver Operating Characteristic (ROC) curve shows the portion of true positive rate in false positive rate for each cut off point which shows sensitivity/specificity ratio belongs to specific decision threshold. 100% sensitivity, 100% specificity shows no overlap in the distributions which is represented through the upper left corner of the ROC curve.

Twenty algorithm results for Group1 results in Appendix C and twenty algorithm results for Group2 results in Appendix D.

While using pivot charts, results are compared for twenty data sets based on twenty algorithms. Most successful three algorithms are selected for Group1 and Group2. Precision and recall calculations are added to result sets for these algorithms.

Table 9 Confusion Matrix

		Classified as	
		Yes	No
Actual	Yes	(TP)	(FP)
	No	(FN)	(TN)

Precision: The ratio of true positive to predicted positive.

Precision = TP/(TP+FP)

Recall: The TP rate

Recall = TP/(TP+FN)

NaiveBayes, BayesNet and LogitBoost algorithms have best results for Group1 datasets. All results for 10 versions of Group1 are specified below.

Table 10 Group1 Best Algorithms Results

V1	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7646	0,7043	0,7946	0,8177	0,63	0,704
BayesNet	0,7663	0,7064	0,7961	0,8194	0,632	0,706
LogitBoost	0,8082	0,6715	0,8761	0,8560	0,729	0,671

V2	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7783	0,6982	0,8180	0,8213	0,656	0,698
BayesNet	0,7772	0,6899	0,8206	0,8232	0,656	0,69
LogitBoost	0,8052	0,6345	0,8899	0,8524	0,741	0,634

V3	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7636	0,7094	0,7905	0,8176	0,627042	0,709446
BayesNet	0,7640	0,7064	0,7926	0,8191	0,628311	0,706366
LogitBoost	0,8082	0,6715	0,8761	0,8560	0,729097	0,671458

V4	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7762	0,6992	0,8145	0,8211	0,651675	0,699179
BayesNet	0,7738	0,6858	0,8175	0,8228	0,651072	0,685832
LogitBoost	0,8052	0,6345	0,8899	0,8524	0,741007	0,634497

V5	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7752	0,6930	0,8160	0,8131	0,651544	0,693018
BayesNet	0,7725	0,6982	0,8094	0,8150	0,645161	0,698152
LogitBoost	0,8052	0,6345	0,8899	0,8524	0,741	0,634

V6	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7752	0,6920	0,8165	0,8130	0,651838	0,691992
BayesNet	0,7732	0,6940	0,8124	0,8150	0,64751	0,694045
LogitBoost	0,8052	0,6345	0,8899	0,8524	0,741007	0,634497

V7	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7606	0,6992	0,7910	0,8177	0,624198	0,699179
BayesNet	0,7653	0,7012	0,7971	0,8189	0,631822	0,701232
LogitBoost	0,7987	0,6037	0,8955	0,8530	0,741	0,604

V8	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7745	0,6920	0,8155	0,8223	0,650579	0,691992
BayesNet	0,7728	0,6828	0,8175	0,8235	0,650049	0,682752
LogitBoost	0,8059	0,6088	0,9037	0,8531	0,758	0,609

V9	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7677	0,7115	0,7956	0,8202	0,633455	0,711499
BayesNet	0,7694	0,7074	0,8002	0,8215	0,637373	0,707392
LogitBoost	0,8082	0,6715	0,8761	0,8560	0,729097	0,671458

V10	TA	TP	TN	ROC	Precision	Recall
NaiveBayes	0,7643	0,7043	0,7941	0,8177	0,629358	0,704312
BayesNet	0,7667	0,7053	0,7971	0,8192	0,63318	0,705339
LogitBoost	0,8082	0,6715	0,8761	0,8560	0,729097	0,671458

RandomForest, NaiveBayes and BayesNet algorithms have best results for Group2 datasets. All results for 10 versions of Group2 are defined below.

Table 11 Group2 Best Algorithms Results

V1	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,734332	0,734332	0,671406	0,808027	0,694413	0,734332
NaiveBayes	0,718324	0,718324	0,75499	0,809371	0,706168	0,718324
BayesNet	0,708787	0,708787	0,756876	0,809385	0,706809	0,708787
V2	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,737738	0,737738	0,669992	0,809911	0,698532	0,737738
NaiveBayes	0,735014	0,735014	0,752752	0,81466	0,712226	0,735014
BayesNet	0,726499	0,726499	0,75459	0,814043	0,712723	0,726499
V3	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,729223	0,729223	0,684324	0,80533	0,699899	0,729223
NaiveBayes	0,717302	0,717302	0,754116	0,809777	0,706823	0,717302
BayesNet	0,708447	0,708447	0,75882	0,809563	0,707846	0,708447
V4	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,728542	0,728542	0,679352	0,804248	0,700198	0,728542
NaiveBayes	0,733992	0,733992	0,754278	0,815129	0,713452	0,733992
BayesNet	0,723433	0,723433	0,757442	0,814479	0,714771	0,723433
V5	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,743529	0,743529	0,688872	0,813634	0,70271	0,743529
NaiveBayes	0,734332	0,734332	0,736098	0,806638	0,700603	0,734332
BayesNet	0,727861	0,727861	0,737108	0,806629	0,697252	0,727861
V6	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,736376	0,736376	0,692477	0,806427	0,703421	0,736376
NaiveBayes	0,731948	0,731948	0,734819	0,807454	0,694539	0,731948
BayesNet	0,728883	0,728883	0,737613	0,807401	0,693991	0,728883
V7	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,735695	0,735695	0,668182	0,812973	0,697803	0,735695
NaiveBayes	0,718324	0,718324	0,753267	0,809203	0,707046	0,718324
BayesNet	0,717643	0,717643	0,759608	0,810826	0,712993	0,717643
V8	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,731608	0,731608	0,68548	0,803712	0,701052	0,731608
NaiveBayes	0,728542	0,728542	0,748775	0,81535	0,707807	0,728542
BayesNet	0,718324	0,718324	0,750921	0,814004	0,70741	0,718324
V9	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,736717	0,736717	0,677239	0,811048	0,699606	0,736717
NaiveBayes	0,722752	0,722752	0,755251	0,811612	0,707654	0,722752
BayesNet	0,71015	0,71015	0,75673	0,811104	0,706739	0,71015
V10	TA	TP	TN	ROC	Precision	Recall
RandomForest	0,734332	0,734332	0,669745	0,807868	0,696231	0,734332
NaiveBayes	0,718665	0,718665	0,755691	0,809191	0,706626	0,718665
BayesNet	0,708447	0,708447	0,756851	0,809111	0,706636	0,708447

When all results are examined, NaiveBayes algorithm for version 4 of Group1has the best result with 0,78 total accuracy, 0,70 true positive, 0,81 true negative, 0,82 Roc curve, 0,65 precision and 0,70 recall values.

Additionally, NaiveBayes algorithm for version 4 of Group2 has also the best result with 0,73 total accuracy, 0,73 true positive, 0,75 true negative, 0,81 Roc curve, 0,71 precision and 0,73 recall values.

Table 12 Results for Group1 and Group2

	Group	**Version**	**TA**	**TP**	**TN**	**ROC**	**Precision**	**Recall**
NaiveBayes	G1	V4	0,77623	0,69918	0,81448	0,82112	0,65167	0,699178
NaiveBayes	G2	V4	0,73399	0,73399	0,75427	0,81512	0,71345	0,733992

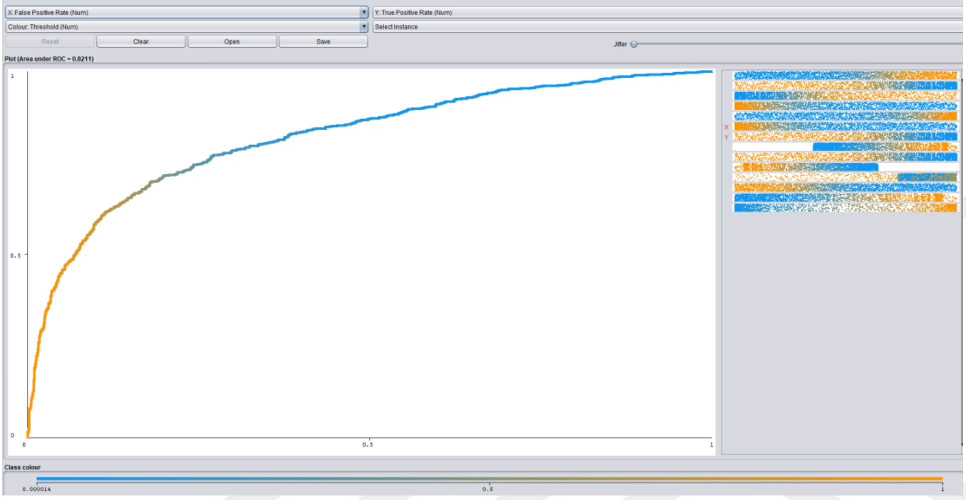

X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
Plot (Area under ROC = 0.8211)

Figure 9 Group 1 Version 4 NaiveBayes - ROC Curve

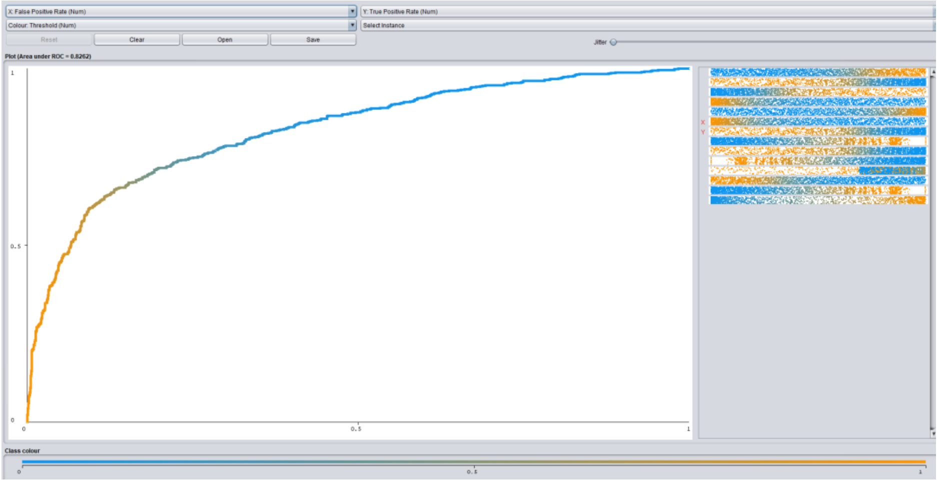

X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour: Threshold (Num)
Select Instance
Plot (Area under ROC = 0.8262)

Figure 10 Group 1 Version 4 NaiveBayes - ROC Curve

Either we take the range limit as 15% or in the range of 14% and 16% same results are obtained. NaiveBayes algorithm with version 4 has the best result for Group1 and Group2.

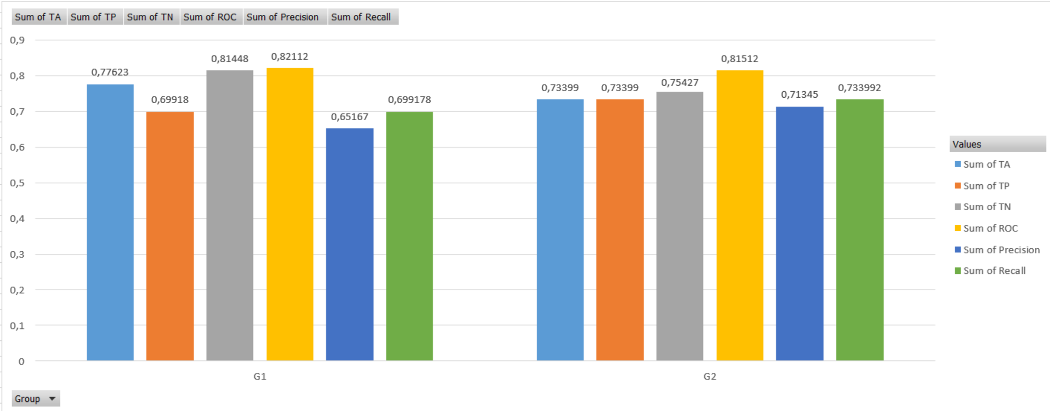


Figure 11 Group 1 and Group 2 Version 4 NaiveBayes Results

However, the results can be improved the information which are explained in conclusion and future work chapter.

CHAPTER 5
CONCLUSION AND FUTURE WORK

Wastage is one of the most important factors affecting production efficiency. It is very essential to know the variables affecting the rate of waste and to estimate effects of these variables to production. In addition, defining variables could increase the production efficiency and process flows. Actions should be taken to reduce the rate of wastage. Reducing the rate of wastage means increasing efficiency in production.

In this study, in case of using Group1 dataset, Naive Bayes algorithm has the best estimation by using the following attributes; MATERIAL, WORKCENTER1, WORKCENTER2, WORKCENTER3, WORKCENTER4, WORKCENTER5, TONNAGEGROUP, WORKTYPE, FILMGROUP, WIDTH1, FILMTYPE1, WIDTH2, FILMTYPE2, LAC, COLDSEAL, PRINTWIDTH, REELWIDTH, COLOR, CYLINDER, PLATE, GROUP1. While analyzing these attributes, the wastage rate above 15 percent is estimated 69 percent with 77.6 percent accuracy. The wastage rate below 15 percent is estimated 81 percent with 77.6 percent accuracy.

In case of using Group2 dataset, Naive Bayes algorithm has the best estimation by using the following attributes; MATERIAL, WORKCENTER1, WORKCENTER2, WORKCENTER3, WORKCENTER4, WORKCENTER5, TONNAGEGROUP, WORKTYPE, FILMGROUP, WIDTH1, FILMTYPE1, WIDTH2, FILMTYPE2, LAC, COLDSEAL, PRINTWIDTH, REELWIDTH, COLOR, CYLINDER, PLATE, GROUP2. While analyzing these attributes, the wastage rate above 16 percent is estimated 73 percent with 73 percent accuracy.

This study shows that when month and customer are removed together, the result is the best. Time and customer information don’t affect wastage rate.

Additionally, Naïve Bayes is not an algorithm which provides rules or trees like Decision Tree, Random Tree, JRip etc. Rule and tree algorithms are also investigated during the study. But these algorithms don’t give valuable results. Generated trees and rules are not valid and useful. The chosen algorithm produces results with over 70 percent accurate. Bu, it is not possible to make interpretation about the Naïve Bayes algorithm.

This study can be taken as a starting point to conduct further research on this topic. First of all, the ERP system of the company is not totally trusted to determine an accurate wastage amount. Therefore, some part of data is collected from operators manually, not from the system, which means data can be manipulated. In order to get more accurate results, the system should be improved to provide data reliability.

Additionally, data sets used in this study comprises of data belonging to one-year period. Therefore, it would be beneficial to run a study with more number of data.

23 attributes used for this study. On the other hand other criterias (i.e operator impact) affect the wastage amount should be investigated in detail to get more accurate results.

This study contains 2-layer products data with chosen film types. The variety of products to be analyzed can be increased.

Moreover, classifier evaluation options in WEKA should be changed to improve the accuracy.

Specific rules could be generated by using algorithms. The rules show the attributes and their impacts. To decrease wastage rate, attributes in the rules should be investigated further in company. Additionally, it would be worthwhile to integrate the rules into company ERP system. During production planning, system would suggest new machine, operator etc. by considering wastage. The system would predict as decision support system.

REFERENCES

Apté, C., & Hong, S. J. (1994). Predicting Equity Returns from Securities Data with Minimal Rule Generation. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Workshop*, 407–418.

Ben-gal, I. (2005). Outlier Detection. *Data Mining and Knowledge Discovery Handbook*, 131–146. https://doi.org/10.1007/0-387-25465-x_7

Brachman, R. J., & Anand, T. (1994). The Process of Knowledge Discovery in Databases: A First Sketch. *KDD Workshop*, 1–11. Retrieved from http://www.aaai.org/Papers/Workshops/1994/WS-94-03/WS94-03-001.pdf

Breiman, L. (1999). RANDOM FORESTS--RANDOM FEATURES. *Machine Learning*, 1–29. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118445112.stat06520

Cai, Y. D., Feng, K. Y., Lu, W. C., & Chou, K. C. (2006). Using LogitBoost classifier to predict protein structural classes. *Journal of Theoretical Biology*, *238*(1), 172–176. https://doi.org/10.1016/j.jtbi.2005.05.034

Cayci, A., & Eibe, S. (2011). *Bayesian Networks to Predict Data Mining Algorithm Behavior in Ubiquitous Environments*. (M. Atzmueller, A. and Hotho, M. and Strohmaier, & A. and Chin, Eds.), *Mining Ubiquitous and …*. Springer Berlin Heidelberg. Retrieved from http://www.kde.cs.uni-kassel.de/ws/muse2010/proceedings.pdf#page=23

Chang, C., & Lin, C. (2013). LIBSVM : A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*, 1–39. https://doi.org/10.1145/1961189.1961199

Chen, M.-S., Hun, J., & Yu, P. S. (1996). Data Mining: An Overview from Database Perspective. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, *8*, 866–883.

Choudhary, A. K., Harding, J. A., & Tiwari, M. K. (2009). Data mining in manufacturing: A review based on the kind of knowledge. *Journal of Intelligent Manufacturing*, *20*(5), 501–521. https://doi.org/10.1007/s10845-008-0145-x

Cunha, C. Da, Agard, B., & Kusiak, A. (2010). quality r P Fo r R w On ly.

Deng, Y., Zhou, N., Liu, X., & Lu, Q. (2018). Research on Fault Diagnosis of Flexible Material R2R Manufacturing System Based on Quality Control Chart and SoV. *Mathematical Problems in Engineering*, *2018*, 1–8. https://doi.org/10.1155/2018/6350380

Englert, P. (2012). Locally Weighted Learning, 1–9. Retrieved from http://www.ias.informatik.tu-darmstadt.de/uploads/Teaching/AutonomousLearningSystems/Englert_ALS_2012.pdf

Fayyad, U. M., Djorgovski, S. G., & Weir, N. (1996). From Digitized Images to Online Catalogs. Data Mining a Sky Survey. *AI Magazine*, *17*(2), 51–66.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, *17*(3), 37. https://doi.org/10.1609/aimag.v17i3.1230

Freund, Y., & Schapire, R. E. (1995). A desicion-theoretic generalization of on-line learning and an application to boosting, *139*, 23–37. https://doi.org/10.1007/3-540-59119-2_166

Fu, Y. (1997). Data mining. *IEEE Potentials*, *16*(4), 18–20. https://doi.org/10.1109/45.624335

Hall, M. A., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *SIGKDD Explorations*, *11*(1), 10–18. https://doi.org/10.1145/1656274.1656278

Harding, J. A., Shahbaz, M., Srinivas, & Kusiak, A. (2006). Data Mining in Manufacturing: A Review. *Journal of Manufacturing Science and Engineering*, *128*(4), 969. https://doi.org/10.1115/1.2194554

He, S. G., Zhen, H., Wang, A., & Li, L. (2009). Quality Improvement using Data Mining in Manufacturing Processes. *Data Mining and Knowledge Discovery in Real Life Applications*, (February), 436. https://doi.org/10.5772/6459

Kang, S., Cho, S., & Kang, P. (2015). Constructing a multi-class classifier using one-against-one approach with different binary classifiers. *Neurocomputing*, *149*(PB), 677–682. https://doi.org/10.1016/j.neucom.2014.08.006

Kaur, G., & Chhabra, A. (2014). Improved J48 Classification Algorithm for the Prediction of Diabetes. *International Journal of Computer Applications*, *98*(22), 13–17. https://doi.org/10.5120/17314-7433

Kusiak, A. (2000). Decomposition in data mining: an industrial case study. *IEEE Transactions on Electronics Packaging Manufacturing*, *23*(4), 238–238. https://doi.org/10.1109/TEPM.2000.895066

Mahmood, K., Karaulova, T., Otto, T., & Shevtshenko, E. (2017). Performance Analysis of a Flexible Manufacturing System (FMS). *Procedia CIRP*, *63*, 424–429. https://doi.org/10.1016/j.procir.2017.03.123

Martínez-De-Pisón, F. J., Sanz, A., Martínez-De-Pisón, E., Jiménez, E., & Conti, D. (2012). Mining association rules from time series to explain failures in a hot-dip galvanizing steel line. *Computers and Industrial Engineering*, *63*(1), 22–36. https://doi.org/10.1016/j.cie.2012.01.013

Menon, R., Tong, L. H., Sathiyakeerthi, S., Brombacher, A., & Leong, C. (2004). The needs and benefits of applying textual data mining within the product development process. *Quality and Reliability Engineering International*, *20*(1), 1–15. https://doi.org/10.1002/qre.536

Niaki, S. T. A., & Abbasi, B. (2005). Fault diagnosis in multivariate control charts using artificial neural networks. *Quality and Reliability Engineering International*, *21*(8), 825–840. https://doi.org/10.1002/qre.689

Piatetsky-Shapiro, G. (1990). Knowledge Discovery in Real Databases: A Report on the IJCAI-89 Workshop. *AI Magazine*, *11*(4), 68. https://doi.org/10.1609/aimag.v11i4.873

Pollack, S. L. (1972). Decision Tables, (May).

Rajput, a., Aharwal, R. P., Dubey, M., Saxena, S. P., & Raghuvanshi, M. (2011). J48 and JRIP rules for e-governance data. *International Journal of Computer Science and Security, 5*(2), 201–207.

Rokach, L., & Maimon, O. (2006). Data Mining for Improving Manufacturing'S Quality: a Feature Set Decomposition Approach, 1–33.

Stahl, F., Gaber, M. M., Bramer, M., & Yu, P. S. (2011). Distributed hoeffding trees for pocket data mining. *Proceedings of the 2011 International Conference on High Performance Computing and Simulation, HPCS 2011*, (May 2014), 686–692. https://doi.org/10.1109/HPCSim.2011.5999893

Vijayarani, S., & Muthulakshmi, M. (2013). Comparative Analysis of Bayes and Lazy Classification Algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, *2*(8), 3118–3124.

Witten, I. H., Frank, E., & Hall, M. a. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. *Complementary literature None*. https://doi.org/0120884070, 9780120884070

Yuan, L. (2010). An Improved Naive Bayes Text Classification Algorithm In Chinese Information Processing. *Proceedings of the Third International Symposium on Computer Science and Computational Technology(ISCSCT '10)*, (15), 267–269.

Zaki, M. J., Parthasarathy, S., Ogihara, M., & Li, W. (1997). New Algorithms for Fast Discovery of Association Rules. *3rd Intl Conf on Knowledge Discovery and Data Mining*, *20*(651), 283–286. https://doi.org/10.1.1.42.5143

Zhao, K., Liu, B., Street, S. M., Tirpak, T. M., & Xiao, W. (2005). A Visual Data Mining Framework for Convenient Identification. *The Fifth IEEE International Conference on Data Mining (ICDM-2005)*, 27–30. https://doi.org/10.1109/ICDM.2005.16

APPENDIX A – GROUP1 ALGORITHM RESULTS

Table 13 Group 1 Algorithm Results

AdaBoostM1	Accuracy	TP	TN	ROC
V1	0,787807	0,523614	0,91896	0,851383
V2	0,787807	0,523614	0,91896	0,851383
V3	0,787807	0,523614	0,91896	0,851383
V4	0,787807	0,523614	0,91896	0,851383
V5	0,787807	0,523614	0,91896	0,851383
V6	0,787807	0,523614	0,91896	0,851383
V7	0,787807	0,523614	0,91896	0,851383
V8	0,788488	0,560575	0,901631	0,846927
V9	0,787807	0,523614	0,91896	0,851383
V10	0,787807	0,523614	0,91896	0,851383

Bagging	Accuracy	TP	TN	ROC
V1	0,718324	0,444559	0,85423	0,74326
V2	0,718324	0,444559	0,85423	0,743272
V3	0,718324	0,443532	0,85474	0,743264
V4	0,718324	0,443532	0,85474	0,743264
V5	0,798025	0,617043	0,88787	0,852277
V6	0,805858	0,629363	0,893476	0,857967
V7	0,718324	0,444559	0,85423	0,74326
V8	0,719005	0,445585	0,85474	0,743451
V9	0,718324	0,444559	0,85423	0,74326
V10	0,718324	0,444559	0,85423	0,74326

BayesNet	Accuracy	TP	TN	ROC
V1	0,766349	0,706366	0,796126	0,819355
V2	0,777248	0,689938	0,820591	0,823182
V3	0,763965	0,706366	0,792559	0,819091
V4	0,773842	0,685832	0,817533	0,822835
V5	0,77248	0,698152	0,809378	0,815039
V6	0,773161	0,694045	0,812436	0,815027
V7	0,765327	0,701232	0,797146	0,818877
V8	0,77282	0,682752	0,817533	0,823507
V9	0,769414	0,707392	0,800204	0,821471
V10	0,766689	0,705339	0,797146	0,819218

ClassificationViaRegression	Accuracy	TP	TN	ROC
V1	0,746935	0,517454	0,860856	0,78312
V2	0,751362	0,528747	0,861876	0,786104
V3	0,747956	0,51232	0,864934	0,786581
V4	0,747956	0,520534	0,860856	0,783052
V5	0,807221	0,637577	0,891437	0,864168
V6	0,804496	0,631417	0,890418	0,86527
V7	0,742507	0,522587	0,851682	0,779379
V8	0,748978	0,523614	0,860856	0,786533
V9	0,746935	0,526694	0,856269	0,784584
V10	0,746935	0,517454	0,860856	0,78312

DecisionTable	Accuracy	TP	TN	ROC
V1	0,801431	0,606776	0,898063	0,855836
V2	0,801431	0,606776	0,898063	0,855836
V3	0,801431	0,606776	0,898063	0,855836
V4	0,801431	0,606776	0,898063	0,855836
V5	0,801431	0,606776	0,898063	0,855836
V6	0,801431	0,606776	0,898063	0,855836
V7	0,799387	0,598563	0,899083	0,857196
V8	0,800409	0,61191	0,893986	0,847156
V9	0,802452	0,60883	0,898573	0,856314
V10	0,801431	0,606776	0,898063	0,855836

HoeffdingTree	Accuracy	TP	TN	ROC
V1	0,718324	0,507187	0,82314	0,733313
V2	0,718324	0,507187	0,82314	0,733313
V3	0,718324	0,507187	0,82314	0,733488
V4	0,718324	0,507187	0,82314	0,733458
V5	0,808243	0,601643	0,910805	0,822081
V6	0,809605	0,62423	0,901631	0,827166
V7	0,718324	0,507187	0,82314	0,733229
V8	0,717984	0,510267	0,821101	0,732379
V9	0,718324	0,507187	0,82314	0,73326
V10	0,718324	0,507187	0,82314	0,733387

IBK	Accuracy	TP	TN	ROC
V1	0,727861	0,599589	0,791539	0,718601
V2	0,72718	0,597536	0,791539	0,713283
V3	0,727861	0,591376	0,795617	0,727113
V4	0,728202	0,591376	0,796126	0,724364
V5	0,72173	0,594456	0,784913	0,707772
V6	0,732629	0,599589	0,798675	0,724551
V7	0,731267	0,607803	0,792559	0,735565
V8	0,742166	0,64271	0,791539	0,753648
V9	0,735354	0,61499	0,795107	0,727098
V10	0,72752	0,599589	0,79103	0,718254

J48	Accuracy	TP	TN	ROC
V1	0,797684	0,667351	0,862385	0,840596
V2	0,797003	0,668378	0,860856	0,840644
V3	0,797684	0,667351	0,862385	0,840596
V4	0,797003	0,668378	0,860856	0,840644
V5	0,800068	0,610883	0,893986	0,837788
V6	0,798706	0,623203	0,885831	0,838622
V7	0,797684	0,667351	0,862385	0,840596
V8	0,797003	0,668378	0,860856	0,840644
V9	0,797684	0,667351	0,862385	0,840596
V10	0,797684	0,667351	0,862385	0,840596

JRip	Accuracy	TP	TN	ROC
V1	0,806199	0,629363	0,893986	0,780722
V2	0,802112	0,63655	0,884302	0,777787
V3	0,808243	0,617043	0,90316	0,769491
V4	0,807561	0,641684	0,889908	0,780305
V5	0,806199	0,652977	0,882263	0,780611
V6	0,810627	0,655031	0,88787	0,782857
V7	0,808243	0,637577	0,892966	0,778062
V8	0,803474	0,626283	0,891437	0,775399
V9	0,810286	0,648871	0,890418	0,778109
V10	0,801431	0,632444	0,885321	0,77031

LogitBoost	Accuracy	TP	TN	ROC
V1	0,808243	0,671458	0,876147	0,856039
V2	0,805177	0,634497	0,889908	0,852422
V3	0,808243	0,671458	0,876147	0,856039
V4	0,805177	0,634497	0,889908	0,852422
V5	0,805177	0,634497	0,889908	0,852422
V6	0,805177	0,634497	0,889908	0,852422
V7	0,798706	0,603696	0,895515	0,852972
V8	0,805858	0,60883	0,90367	0,853066
V9	0,808243	0,671458	0,876147	0,856039
V10	0,808243	0,671458	0,876147	0,856039

LWL	Accuracy	TP	TN	ROC
V1	0,77827	0,401437	0,965341	0,815689
V2	0,77827	0,401437	0,965341	0,820262
V3	0,77827	0,401437	0,965341	0,815652
V4	0,77827	0,401437	0,965341	0,820332
V5	0,77827	0,401437	0,965341	0,820193
V6	0,77827	0,401437	0,965341	0,820262
V7	0,77827	0,401437	0,965341	0,815418
V8	0,77827	0,401437	0,965341	0,826965
V9	0,77827	0,401437	0,965341	0,817409
V10	0,77827	0,401437	0,965341	0,815475

MultiClassClassifier	Accuracy	TP	TN	ROC
V1	0,747275	0,554415	0,843017	0,754301
V2	0,750341	0,566735	0,841488	0,754429
V3	0,745232	0,560575	0,836901	0,750536
V4	0,745232	0,566735	0,833843	0,753707
V5	0,801431	0,645791	0,878695	0,862508
V6	0,804155	0,646817	0,882263	0,862938
V7	0,747275	0,550308	0,845056	0,752199
V8	0,749659	0,556468	0,845566	0,75062
V9	0,747956	0,553388	0,844546	0,748436
V10	0,746935	0,554415	0,842508	0,752476

MultiScheme	Accuracy	TP	TN	ROC
V1	0,668256	0	1	0,498361
V2	0,668256	0	1	0,498361
V3	0,668256	0	1	0,498361
V4	0,668256	0	1	0,498361
V5	0,668256	0	1	0,498361
V6	0,668256	0	1	0,498361
V7	0,668256	0	1	0,498361
V8	0,668256	0	1	0,498361
V9	0,668256	0	1	0,498361
V10	0,668256	0	1	0,498361

NaiveBayes	Accuracy	TP	TN	ROC
V1	0,764646	0,704312	0,794597	0,817714
V2	0,77827	0,698152	0,818043	0,821323
V3	0,763624	0,709446	0,79052	0,817579
V4	0,776226	0,699179	0,814475	0,82112
V5	0,775204	0,693018	0,816004	0,813065
V6	0,775204	0,691992	0,816514	0,812959
V7	0,760559	0,699179	0,79103	0,81773
V8	0,774523	0,691992	0,815494	0,822301
V9	0,767711	0,711499	0,795617	0,820192
V10	0,764305	0,704312	0,794088	0,817663

NaiveBayesMultinomialText	Accuracy	TP	TN	ROC
V1	0,668256	0	1	0,498361
V2	0,668256	0	1	0,498361
V3	0,668256	0	1	0,498361
V4	0,668256	0	1	0,498361
V5	0,668256	0	1	0,498361
V6	0,668256	0	1	0,498361
V7	0,668256	0	1	0,498361
V8	0,668256	0	1	0,498361
V9	0,668256	0	1	0,498361
V10	0,668256	0	1	0,498361

NaiveBayesUpdateable	Accuracy	TP	TN	ROC
V1	0,764646	0,704312	0,794597	0,817714
V2	0,77827	0,698152	0,818043	0,821323
V3	0,763624	0,709446	0,79052	0,817579
V4	0,776226	0,699179	0,814475	0,82112
V5	0,775204	0,693018	0,816004	0,813065
V6	0,775204	0,691992	0,816514	0,812959
V7	0,760559	0,699179	0,79103	0,81773
V8	0,774523	0,691992	0,815494	0,822301
V9	0,767711	0,711499	0,795617	0,820192
V10	0,764305	0,704312	0,794088	0,817663

RandomForest	Accuracy	TP	TN	ROC
V1	0,778951	0,553388	0,890928	0,81855
V2	0,775886	0,539014	0,893476	0,820032
V3	0,768052	0,546201	0,878186	0,809511
V4	0,762262	0,529774	0,877676	0,80931
V5	0,787466	0,581109	0,889908	0,833005
V6	0,786444	0,597536	0,880224	0,824598
V7	0,776226	0,544148	0,891437	0,817979
V8	0,76703	0,550308	0,874618	0,807196
V9	0,771458	0,541068	0,885831	0,818174
V10	0,767371	0,534908	0,882773	0,81616

RandomTree	Accuracy	TP	TN	ROC
V1	0,732289	0,568789	0,813456	0,722886
V2	0,72718	0,567762	0,80632	0,722044
V3	0,723093	0,555441	0,80632	0,716002
V4	0,732629	0,558522	0,819062	0,728788
V5	0,741485	0,616016	0,803772	0,724776
V6	0,722071	0,60883	0,778287	0,707264
V7	0,735354	0,604723	0,800204	0,728231
V8	0,752384	0,623203	0,816514	0,760089
V9	0,705041	0,516427	0,798675	0,702088
V10	0,713896	0,539014	0,800714	0,707767

SMO	Accuracy	TP	TN	ROC
V1	0,789169	0,62423	0,87105	0,74764
V2	0,78951	0,626283	0,87054	0,748412
V3	0,787466	0,62423	0,868502	0,746366
V4	0,783379	0,617043	0,865953	0,741498
V5	0,800409	0,676591	0,861876	0,769234
V6	0,800409	0,676591	0,861876	0,769234
V7	0,790872	0,629363	0,87105	0,750207
V8	0,785082	0,620123	0,866972	0,743548
V9	0,790872	0,628337	0,87156	0,749948
V10	0,789169	0,62423	0,87105	0,74764

Vote	Accuracy	TP	TN	ROC
V1	0,668256	0	1	0,498361
V2	0,668256	0	1	0,498361
V3	0,668256	0	1	0,498361
V4	0,668256	0	1	0,498361
V5	0,668256	0	1	0,498361
V6	0,668256	0	1	0,498361
V7	0,668256	0	1	0,498361
V8	0,668256	0	1	0,498361
V9	0,668256	0	1	0,498361
V10	0,668256	0	1	0,498361

APPENDIX B – GROUP2 ALGORITHM RESULTS

Table 14 Group 2 Algorithm Result

AdaBoostM1	Accuracy	TP	TN	ROC
V1	0,739441	0,739441	0,591470	0,777604
V2	0,739441	0,739441	0,591470	0,777604
V3	0,739441	0,739441	0,591470	0,777604
V4	0,739441	0,739441	0,591470	0,777604
V5	0,739441	0,739441	0,591470	0,777604
V6	0,739441	0,739441	0,591470	0,777604
V7	0,739441	0,739441	0,591470	0,777604
V8	0,739441	0,739441	0,591470	0,777604
V9	0,739441	0,739441	0,591470	0,777604
V10	0,739441	0,739441	0,591470	0,777604

Bagging	Accuracy	TP	TN	ROC
V1	0,683583	0,683583	0,634553	0,732744
V2	0,683583	0,683583	0,634553	0,732791
V3	0,683924	0,683924	0,634700	0,732677
V4	0,683924	0,683924	0,634700	0,732724
V5	0,758174	0,758174	0,710802	0,837526
V6	0,762262	0,762262	0,717746	0,841520
V7	0,683583	0,683583	0,634553	0,732744
V8	0,683243	0,683243	0,632867	0,733055
V9	0,683583	0,683583	0,634553	0,732744
V10	0,683583	0,683583	0,634553	0,732744

BayesNet	Accuracy	TP	TN	ROC
V1	0,708787	0,708787	0,756876	0,809385
V2	0,726499	0,726499	0,754590	0,814043
V3	0,708447	0,708447	0,758820	0,809563
V4	0,723433	0,723433	0,757442	0,814479
V5	0,727861	0,727861	0,737108	0,806629
V6	0,728883	0,728883	0,737613	0,807401
V7	0,717643	0,717643	0,759608	0,810826
V8	0,718324	0,718324	0,750921	0,814004
V9	0,710150	0,710150	0,756730	0,811104
V10	0,708447	0,708447	0,756851	0,809111

ClassificationViaRegression	Accuracy	TP	TN	ROC
V1	0,710150	0,710150	0,682012	0,776075
V2	0,704019	0,704019	0,673322	0,773913
V3	0,719005	0,719005	0,692620	0,780626
V4	0,717984	0,717984	0,691686	0,777856
V5	0,774864	0,774864	0,728324	0,853016
V6	0,773842	0,773842	0,727020	0,854350
V7	0,715940	0,715940	0,690800	0,775161
V8	0,709469	0,709469	0,675747	0,774444
V9	0,717984	0,717984	0,689719	0,775821
V10	0,710150	0,710150	0,682012	0,776075

DecisionTable	Accuracy	TP	TN	ROC
V1	0,768052	0,768052	0,704383	0,845341
V2	0,768052	0,768052	0,704383	0,845341
V3	0,768052	0,768052	0,704383	0,845341
V4	0,768052	0,768052	0,704383	0,845341
V5	0,768052	0,768052	0,704383	0,845341
V6	0,768052	0,768052	0,704383	0,845341
V7	0,762943	0,762943	0,694720	0,844443
V8	0,766689	0,766689	0,708531	0,841639
V9	0,768733	0,768733	0,704247	0,845660
V10	0,768052	0,768052	0,704383	0,845341

HoeffdingTree	Accuracy	TP	TN	ROC
V1	0,682902	0,682902	0,602869	0,710401
V2	0,682902	0,682902	0,602869	0,710384
V3	0,683243	0,683243	0,602893	0,710708
V4	0,683243	0,683243	0,602893	0,710693
V5	0,770777	0,770777	0,715529	0,813072
V6	0,770095	0,770095	0,710925	0,814045
V7	0,682902	0,682902	0,602869	0,710996
V8	0,683583	0,683583	0,601749	0,710320
V9	0,682902	0,682902	0,602869	0,710324
V10	0,682902	0,682902	0,602869	0,710345

IBK	Accuracy	TP	TN	ROC
V1	0,674046	0,674046	0,642716	0,704768
V2	0,672684	0,672684	0,641326	0,702288
V3	0,695504	0,695504	0,660393	0,722000
V4	0,694142	0,694142	0,659742	0,717937
V5	0,667575	0,667575	0,639724	0,689037
V6	0,691417	0,691417	0,657270	0,711808
V7	0,684264	0,684264	0,638107	0,720662
V8	0,708106	0,708106	0,674474	0,751311
V9	0,676090	0,676090	0,647787	0,711743
V10	0,673706	0,673706	0,642568	0,704439

J48	Accuracy	TP	TN	ROC
V1	0,764986	0,764986	0,731366	0,831731
V2	0,764986	0,764986	0,731366	0,831731
V3	0,764986	0,764986	0,731366	0,831731
V4	0,764986	0,764986	0,731366	0,831731
V5	0,762602	0,762602	0,712847	0,824563
V6	0,762943	0,762943	0,716381	0,824296
V7	0,764986	0,764986	0,731366	0,831731
V8	0,764986	0,764986	0,731366	0,831731
V9	0,764986	0,764986	0,731366	0,831731
V10	0,764986	0,764986	0,731366	0,831731

JRip	Accuracy	TP	TN	ROC
V1	0,771117	0,771117	0,697403	0,742113
V2	0,763624	0,763624	0,678213	0,731067
V3	0,765327	0,765327	0,687999	0,746214
V4	0,770436	0,770436	0,689783	0,740039
V5	0,765668	0,765668	0,690301	0,741312
V6	0,765327	0,765327	0,681106	0,737869
V7	0,766689	0,766689	0,693760	0,744968
V8	0,773501	0,773501	0,693698	0,737117
V9	0,768733	0,768733	0,693784	0,742352
V10	0,763624	0,763624	0,680367	0,739054

LogitBoost	Accuracy	TP	TN	ROC
V1	0,770436	0,770436	0,729604	0,850777
V2	0,770095	0,770095	0,729887	0,847898
V3	0,770436	0,770436	0,729604	0,850777
V4	0,770095	0,770095	0,729887	0,847898
V5	0,768052	0,768052	0,720201	0,849721
V6	0,768392	0,768392	0,720779	0,850354
V7	0,772480	0,772480	0,732644	0,850918
V8	0,772480	0,772480	0,727043	0,848522
V9	0,770436	0,770436	0,729481	0,851001
V10	0,770436	0,770436	0,729604	0,850777

LWL	Accuracy	TP	TN	ROC
V1	0,739441	0,739441	0,591470	0,805830
V2	0,739441	0,739441	0,591470	0,810564
V3	0,739441	0,739441	0,591470	0,806025
V4	0,739441	0,739441	0,591470	0,810815
V5	0,739441	0,739441	0,591470	0,810347
V6	0,739441	0,739441	0,591470	0,810623
V7	0,739441	0,739441	0,591470	0,805617
V8	0,739441	0,739441	0,591470	0,817036
V9	0,739441	0,739441	0,591470	0,807751
V10	0,739441	0,739441	0,591470	0,805218

MultiClassClassifier	Accuracy	TP	TN	ROC
V1	0,699591	0,699591	0,698163	0,746486
V2	0,701975	0,701975	0,702952	0,744130
V3	0,698569	0,698569	0,699073	0,746084
V4	0,697888	0,697888	0,702528	0,741664
V5	0,764646	0,764646	0,721124	0,849141
V6	0,765327	0,765327	0,722712	0,850567
V7	0,699251	0,699251	0,694448	0,745042
V8	0,702316	0,702316	0,706482	0,741860
V9	0,694142	0,694142	0,697151	0,744526
V10	0,697888	0,697888	0,696809	0,746267

MultiScheme	Accuracy	TP	TN	ROC
V1	0,629428	0,629428	0,370572	0,498844
V2	0,629428	0,629428	0,370572	0,498844
V3	0,629428	0,629428	0,370572	0,498844
V4	0,629428	0,629428	0,370572	0,498844
V5	0,629428	0,629428	0,370572	0,498844
V6	0,629428	0,629428	0,370572	0,498844
V7	0,629428	0,629428	0,370572	0,498844
V8	0,629428	0,629428	0,370572	0,498844
V9	0,629428	0,629428	0,370572	0,498844
V10	0,629428	0,629428	0,370572	0,498844

NaiveBayes	Accuracy	TP	TN	ROC
V1	0,718324	0,718324	0,754990	0,809371
V2	0,735014	0,735014	0,752752	0,814660
V3	0,717302	0,717302	0,754116	0,809777
V4	0,733992	0,733992	0,754278	0,815129
V5	0,734332	0,734332	0,736098	0,806638
V6	0,731948	0,731948	0,734819	0,807454
V7	0,718324	0,718324	0,753267	0,809203
V8	0,728542	0,728542	0,748775	0,815350
V9	0,722752	0,722752	0,755251	0,811612
V10	0,718665	0,718665	0,755691	0,809191

NaiveBayesMultinomialText	Accuracy	TP	TN	ROC
V1	0,629428	0,629428	0,370572	0,498844
V2	0,629428	0,629428	0,370572	0,498844
V3	0,629428	0,629428	0,370572	0,498844
V4	0,629428	0,629428	0,370572	0,498844
V5	0,629428	0,629428	0,370572	0,498844
V6	0,629428	0,629428	0,370572	0,498844
V7	0,629428	0,629428	0,370572	0,498844
V8	0,629428	0,629428	0,370572	0,498844
V9	0,629428	0,629428	0,370572	0,498844
V10	0,629428	0,629428	0,370572	0,498844

NaiveBayesUpdateable	Accuracy	TP	TN	ROC
V1	0,718324	0,718324	0,754990	0,809371
V2	0,735014	0,735014	0,752752	0,814660
V3	0,717302	0,717302	0,754116	0,809777
V4	0,733992	0,733992	0,754278	0,815129
V5	0,734332	0,734332	0,736098	0,806638
V6	0,731948	0,731948	0,734819	0,807454
V7	0,718324	0,718324	0,753267	0,809203
V8	0,728542	0,728542	0,748775	0,815350
V9	0,722752	0,722752	0,755251	0,811612
V10	0,718665	0,718665	0,755691	0,809191

RandomForest	Accuracy	TP	TN	ROC
V1	0,734332	0,734332	0,671406	0,808027
V2	0,737738	0,737738	0,669992	0,809911
V3	0,729223	0,729223	0,684324	0,805330
V4	0,728542	0,728542	0,679352	0,804248
V5	0,743529	0,743529	0,688872	0,813634
V6	0,736376	0,736376	0,692477	0,806427
V7	0,735695	0,735695	0,668182	0,812973
V8	0,731608	0,731608	0,685480	0,803712
V9	0,736717	0,736717	0,677239	0,811048
V10	0,734332	0,734332	0,669745	0,807868

RandomTree	Accuracy	TP	TN	ROC
V1	0,671662	0,671662	0,638181	0,704545
V2	0,687330	0,687330	0,633779	0,712182
V3	0,681199	0,681199	0,637579	0,717361
V4	0,686989	0,686989	0,650615	0,725343
V5	0,660422	0,660422	0,663755	0,691115
V6	0,680858	0,680858	0,669303	0,698665
V7	0,662125	0,662125	0,604937	0,703717
V8	0,701975	0,701975	0,656560	0,748441
V9	0,693460	0,693460	0,654774	0,724123
V10	0,677112	0,677112	0,630142	0,718090

SMO	Accuracy	TP	TN	ROC
V1	0,755790	0,755790	0,722872	0,740747
V2	0,756471	0,756471	0,725752	0,742573
V3	0,752384	0,752384	0,723176	0,740035
V4	0,755450	0,755450	0,724753	0,742634
V5	0,767371	0,767371	0,735416	0,752992
V6	0,767371	0,767371	0,735416	0,752992
V7	0,755790	0,755790	0,725086	0,742072
V8	0,752725	0,752725	0,717478	0,738597
V9	0,757493	0,757493	0,725764	0,743284
V10	0,756471	0,756471	0,723290	0,741215

Vote	Accuracy	TP	TN	ROC
V1	0,629428	0,629428	0,370572	0,498844
V2	0,629428	0,629428	0,370572	0,498844
V3	0,629428	0,629428	0,370572	0,498844
V4	0,629428	0,629428	0,370572	0,498844
V5	0,629428	0,629428	0,370572	0,498844
V6	0,629428	0,629428	0,370572	0,498844
V7	0,629428	0,629428	0,370572	0,498844
V8	0,629428	0,629428	0,370572	0,498844
V9	0,629428	0,629428	0,370572	0,498844
V10	0,629428	0,629428	0,370572	0,498844

APPENDIX C – JAVA CODE

```
package data;

import java.io.BufferedReader;
import java.io.FileReader;
import java.lang.reflect.Array;

import weka.classifiers.trees.HoeffdingTree;
import weka.classifiers.trees.J48;
import weka.classifiers.trees.RandomForest;
import weka.classifiers.trees.RandomTree;
import weka.core.Instances;
import weka.classifiers.Classifier;
import weka.classifiers.Evaluation;
import java.util.Random;
import weka.classifiers.bayes.BayesNet;
import weka.classifiers.bayes.NaiveBayes;
import weka.classifiers.bayes.NaiveBayesMultinomialText;
import weka.classifiers.bayes.NaiveBayesUpdateable;
import weka.classifiers.functions.Logistic;
import weka.classifiers.functions.MultilayerPerceptron;
import weka.classifiers.functions.SMO;
import weka.classifiers.functions.SimpleLogistic;
import weka.classifiers.lazy.IBk;
import weka.classifiers.lazy.LWL;
import weka.classifiers.meta.AdaBoostM1;
import weka.classifiers.meta.Bagging;
import weka.classifiers.meta.ClassificationViaRegression;
import weka.classifiers.meta.LogitBoost;
import weka.classifiers.meta.MultiClassClassifier;
import weka.classifiers.meta.MultiScheme;
import weka.classifiers.meta.Vote;
import weka.classifiers.rules.DecisionTable;
import weka.classifiers.rules.JRip;
import java.io.File;
import jxl.*;
import jxl.format.Colour;
import jxl.write.*;
import jxl.write.Number;
public class run {

        public static void main(String[] args) throws Exception {

                BufferedReader reader;

                Instances traindata;

                WritableWorkbook workbook = null;
```

```
                    WritableSheet sheet = null;

                    WritableFont arial10font = new WritableFont(WritableFont.ARIAL, 10,
WritableFont.BOLD, false);

                    arial10font.setColour(Colour.RED);

                    WritableCellFormat arial10format_USER = new WritableCellFormat
(arial10font);

                    arial10format_USER.setWrap(true);

                    WritableFont arial10font_3 = new WritableFont(WritableFont.ARIAL,
10, WritableFont.BOLD, false);

                    WritableCellFormat arial10format_3 = new WritableCellFormat
(arial10font_3);

                    arial10format_3.setShrinkToFit(true);

                    WritableFont arial10font_DETAIL = new
WritableFont(WritableFont.ARIAL, 10, WritableFont.BOLD, false);

                    arial10font_DETAIL.setColour(Colour.BLUE);

                    WritableFont arial10format_DETAIL = new WritableFont
(arial10font_DETAIL);

                    NumberFormat fourdps = new NumberFormat("0.####");

                    WritableCellFormat fourdpsFormat = new
WritableCellFormat(arial10format_DETAIL,fourdps);

                    fourdpsFormat.setWrap(true);

                    String filePath ="C:\\Users\\Hibizz\\Desktop\\TEZ\\weka\\fixed file\\";
                    workbook = Workbook.createWorkbook(new
File(filePath+"results.xls"));

                    String sheetName ="";

                    Classifier cls = null;

                    //for algorithms

                    for(int k=0;k<20;k++)
                    {
                              switch (k) {

                              case 0:
                                        sheetName = "Vote";
                                        cls = new Vote();
                                        break;
                              case 1:
```

```
        sheetName = "NaiveBayesUpdateable";
        cls = new NaiveBayesUpdateable();
        break;
case 2:
        sheetName = "MultiScheme";
        cls = new MultiScheme();
        break;
case 3:
        sheetName = "NaiveBayes";
        cls = new NaiveBayes();
        break;
case 4:
        sheetName = "RandomTree";
        cls = new RandomTree();
        break;
case 5:
        sheetName = "AdaBoostM1";
        cls = new AdaBoostM1();
        break;
case 6:
        sheetName = "NaiveBayesMultinomialText";
        cls = new NaiveBayesMultinomialText();
        break;
case 7:
        sheetName = "j48";
        cls = new J48();
        break;
case 8:
        sheetName = "RandomForest";
        cls = new RandomForest();
        break;
case 9:
        sheetName = "SMO";
        cls = new SMO();
        break;
case 10:
        sheetName = "MultiClassClassifier";
        cls = new MultiClassClassifier();
        break;
case 11:
        sheetName = "JRip";
        cls = new JRip();
        break;
case 12:
        sheetName = "LWL";
        cls = new LWL();
        break;
case 13:
        sheetName = "LogitBoost";
        cls = new LogitBoost();
        break;
case 14:
        sheetName = "DecisionTable";
        cls = new DecisionTable();
        break;
```

```
                case 15:
                        sheetName = "BayesNet";
                        cls = new BayesNet();
                        break;
                case 16:
                        sheetName = "IBk";
                        cls = new IBk();
                        break;
                case 17:
                        sheetName = "Bagging";
                        cls = new Bagging();
                        break;
                case 18:
                        sheetName = "ClassificationViaRegression";
                        cls = new ClassificationViaRegression();
                        break;
                case 19:
                        sheetName = "HoeffdingTree";
                        cls = new HoeffdingTree();
                        break;

                }
                sheet = workbook.createSheet(sheetName, k);

                Label label = new Label(0, 0, "DATA
SETS",arial10format_USER);

                sheet.addCell(label);

                label = new Label(0, 1, "Accuracy",arial10format_3);

                sheet.addCell(label);

                label = new Label(0, 2, "TP",arial10format_3);

                sheet.addCell(label);

                label = new Label(0, 3, "TN",arial10format_3);

                sheet.addCell(label);

                label = new Label(0, 4, "ROC",arial10format_3);

                sheet.addCell(label);

                label = new Label(0, 5, "Precision",arial10format_3);

                sheet.addCell(label);

                label = new Label(0, 6, "Recall",arial10format_3);

                sheet.addCell(label);


                int versiyon = 1;
```

```
            int dataSetCount = 1;
            String wekaFilePath = "";

            for(int j=1;j<41;j++)
            {
                System.out.println(sheetName + " "
+dataSetCount+".dataset " + versiyon +".versiyon");
                if(dataSetCount == 1)
                {
                    switch (versiyon) {
                    case 1:
                        wekaFilePath =
filePath+"DATA1_v1.arff";
                        break;
                    case 2:
                        wekaFilePath =
filePath+"DATA1_v2.arff";
                        break;
                    case 3:
                        wekaFilePath =
filePath+"DATA1_v3.arff";
                        break;
                    case 4:
                        wekaFilePath =
filePath+"DATA1_v4.arff";
                        break;
                    case 5:
                        wekaFilePath =
filePath+"DATA1_v5.arff";
                        break;
                    case 6:
                        wekaFilePath =
filePath+"DATA1_v6.arff";
                        break;
                    case 7:
                        wekaFilePath =
filePath+"DATA1_v7.arff";
                        break;
                    case 8:
                        wekaFilePath =
filePath+"DATA1_v8.arff";
                        break;
                    case 9:
                        wekaFilePath =
filePath+"DATA1_v9.arff";
                        break;
                    case 10:
                        wekaFilePath =
filePath+"DATA1_v10.arff";
                        break;
                    }
                }

                if(dataSetCount == 2)
                {
```

```
                              switch (versiyon) {
                              case 1:
                                    wekaFilePath =
filePath+"DATA2_v1.arff";
                                    break;
                              case 2:
                                    wekaFilePath =
filePath+"DATA2_v2.arff";
                                    break;
                              case 3:
                                    wekaFilePath =
filePath+"DATA2_v3.arff";
                                    break;
                              case 4:
                                    wekaFilePath =
filePath+"DATA2_v4.arff";
                                    break;
                              case 5:
                                    wekaFilePath =
filePath+"DATA2_v5.arff";
                                    break;
                              case 6:
                                    wekaFilePath =
filePath+"DATA2_v6.arff";
                                    break;
                              case 7:
                                    wekaFilePath =
filePath+"DATA2_v7.arff";
                                    break;
                              case 8:
                                    wekaFilePath =
filePath+"DATA2_v8.arff";
                                    break;
                              case 9:
                                    wekaFilePath =
filePath+"DATA2_v9.arff";
                                    break;
                              case 10:
                                    wekaFilePath =
filePath+"DATA2_v10.arff";
                                    break;
                              }
                        }

                        if(dataSetCount == 3)
                        {
                              switch (versiyon) {
                              case 1:
                                    wekaFilePath =
filePath+"DATA3_v1.arff";
                                    break;
                              case 2:
                                    wekaFilePath =
filePath+"DATA3_v2.arff";
                                    break;
```

```
                    case 3:
                        wekaFilePath =
filePath+"DATA3_v3.arff";
                        break;
                    case 4:
                        wekaFilePath =
filePath+"DATA3_v4.arff";
                        break;
                    case 5:
                        wekaFilePath =
filePath+"DATA3_v5.arff";
                        break;
                    case 6:
                        wekaFilePath =
filePath+"DATA3_v6.arff";
                        break;
                    case 7:
                        wekaFilePath =
filePath+"DATA3_v7.arff";
                        break;
                    case 8:
                        wekaFilePath =
filePath+"DATA3_v8.arff";
                        break;
                    case 9:
                        wekaFilePath =
filePath+"DATA3_v9.arff";
                        break;
                    case 10:
                        wekaFilePath =
filePath+"DATA3_v10.arff";
                        break;
                    }
                }

                if(dataSetCount == 4)
                {
                    switch (versiyon) {
                    case 1:
                        wekaFilePath =
filePath+"DATA4_v1.arff";
                        break;
                    case 2:
                        wekaFilePath =
filePath+"DATA4_v2.arff";
                        break;
                    case 3:
                        wekaFilePath =
filePath+"DATA4_v3.arff";
                        break;
                    case 4:
                        wekaFilePath =
filePath+"DATA4_v4.arff";
                        break;
                    case 5:
                        wekaFilePath =
filePath+"DATA4_v5.arff";
                        break;
                    case 6:
                        wekaFilePath =
filePath+"DATA4_v6.arff";
                        break;
                    case 7:
                        wekaFilePath =
filePath+"DATA4_v7.arff";
                        break;
                    case 8:
                        wekaFilePath =
filePath+"DATA4_v8.arff";
                        break;
                    case 9:
                        wekaFilePath =
filePath+"DATA4_v9.arff";
                        break;
                    case 10:
                        wekaFilePath =
filePath+"DATA4_v10.arff";
                        break;
                    }
                }

                reader = new BufferedReader(new
FileReader(wekaFilePath));

                traindata = new Instances(reader);

                reader.close();

                // setting class attribute

                traindata.setClassIndex(traindata.numAttributes() - 1);

                Evaluation eval = new Evaluation(traindata);
                System.out.println("eval tanımlandı");
                eval.crossValidateModel(cls, traindata, 10, new
Random(1));
                System.out.println("cross validate yapıldı");
                //System.out.println(eval.toMatrixString());
                label = new Label(j, 0, "DATA" + dataSetCount +
"_V"+versiyon,arial10format_USER);

                sheet.addCell(label);

                Number number = new Number(j, 1,
eval.pctCorrect()/100,fourdpsFormat); //Correctly classified instance

                sheet.addCell(number);

                number = new Number(j, 2,
eval.truePositiveRate(0),fourdpsFormat); //TP
```

```
                sheet.addCell(number);

                number = new Number(j, 3,
eval.trueNegativeRate(0),fourdpsFormat); //TN

                sheet.addCell(number);

                number = new Number(j, 4,
eval.areaUnderROC(1),fourdpsFormat); //ROC

                sheet.addCell(number);

                versiyon = versiyon + 1;
                if(versiyon ==11) {
                    versiyon =1;
                }

                if(j == 10 || j==20 || j==30 || j==40)
                {
                    dataSetCount = dataSetCount+1;
                }

            }
        }

        workbook.write();

        workbook.close();

    }

}
```

Figure 12 Java Code