T.C.
BAHÇEŞEHİR UNIVERSITY

AN APPLICATION OF ADAPTIVE-NETWORK-BASED FUZZY INFERENCE SYSTEM ON AUTOMATED TELLER MACHINE DATA AND COMPARISON OF DIFFERENT DATA MINING ALGORITHMS

M.S. THESIS

Mustafa KARA

Istanbul, 2011

T.C.

BAHÇEŞEHİR ÜNİVERSİTESİ

The Graduate School of Natural and Applied Sciences

Computer Engineering

AN APPLICATION OF ADAPTIVE-NETWORK-BASED FUZZY INFERENCE SYSTEM ON AUTOMATED TELLER MACHINE DATA AND COMPARISON OF DIFFERENT DATA MINING ALGORITHMS

M.S. THESIS

Mustafa KARA

Supervisor: Assoc. Prof. Dr. Adem KARAHOCA

Istanbul, 2011

T.C

BAHÇEŞEHİR ÜNİVERSİTESİ

The Graduate School of Natural and Applied Sciences

Computer Engineering

Title of the Master's Thesis	: An Application of Adaptive-Network-Based Fuzzy Inference System on Automated Teller Machine Data And Comparison of Different Data Mining Algorithms
Name/Last Name of the Student	: Mustafa KARA
Date of Thesis Defense	: 9.9.2011

The thesis has been approved by the Graduate School of Natural and Applied Sciences.

Signature

Assoc. Prof. F. Tunç BOZBURA

Acting Director

This is to certify that we have read this thesis and that we find it fully adequate in scope, quality and content, as a thesis for the degree of Master of Science.

Examining Committee Members:

Assoc. Prof. Adem Karahoca (Supervisor)	: Signature
Asst. Prof. M. Alper Tunga	: Signature
Asst. Prof. Yalçın Çekiç	: Signature

ACKNOWLEDGMENTS

It is a pleasure to thank the many people who made this thesis possible.

It is difficult to overstate my gratitude to my supervisor, **Assoc. Dr. Adem Karahoca**. With his enthusiasm, his inspiration, and his great efforts to explain things clearly and simply, he helped to make the subject fun for me. During my thesis-writing period, he provided encouragement, sound advice, good teaching, good company, and lots of good ideas. I would have been lost without him support.

I also wish to thank instructor **Dilek Karahoca**, who helped me on various topics in the area of technology acceptance theory.

I wish to thank my entire extended **family** for providing a loving environment for me.

Lastly, and most importantly, I wish to thank my **parents**. They bore me, raised me, supported me, taught me, and loved me. To them I dedicate this thesis.

9.9.2011 Mustafa KARA

ABSTRACT

AN APPLICATION OF ADAPTIVE-NETWORK-BASED FUZZY INFERENCE SYSTEM ON AUTOMATED TELLER MACHINE DATA AND COMPARISON OF DIFFERENT DATA MINING ALGORITHMS

KARA, Mustafa

Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Adem Karahoca

September 2011, 66 Pages

Data mining applications have been shown to be highly effective in addressing many important business problems. Data mining in various forms is becoming a major component of business operations. Almost every business process today involves some form data mining.

In this study, data mining techniques used for prediction of drawn amount in Automated Teller Machine (ATM).At the end of the study predictions of different data mining algorithms are compared to each other to see which method is better and efficient on large amount of datasets.

Keywords: Data mining, ATM, Estimation

ÖZET

BİR BANKAMATİK VERİLERİNİN FARKLI VERİ MADENCİLİĞİ TEKNİKLERİ KULLANARAK KARARLAŞTIRILMASI

Kara, Mustafa

Bilgisayar Mühendisliği Bölümü

Tez Danışmanı: Doç. Dr. Adem Karahoca

Eylül 2011, 66 Sayfa

İş dünyasında karşılaşılan önemli problemlerde ve çözümlerinde veri madenciliği uygulamaları oldukça etkili bir biçimde yol göstermektedir. İş uygulamalarının ana öğelerinde veri madenciliği farklı bir biçimlerde yer almaktadır.

Bu çalışmada veri madenciliği teknikleri kullanılarak ATMlerden çekilen para miktarı tahmin edilmiştir.Çalışmanın sonucunda çeşitli veri madenciliği algoritmaları karşılaştırılmış ve bu tür veriler için en uygun yöntem belirlenmiştir.

Anahtar Kelimler: Veri madenciliği, ATM, Tahmin

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

ABBRIVIATIONS

Adaptive-Network-Based Fuzzy Inference System	: ANFIS
Automated Teller Machine	: ATM
Customer Relationship Management	: CRM
Iterative Dichotomiser 3	: ID3
False Positive	: FP
False Negative	: FN
True Positive	: TP
True Negative	: TN

1. INTRODUCTION

ATMs have become an integral part of the banking industry over the past years, effectively managing cash stocks in potentially thousands of diverse locations with wildly diverse cash needs. The failure to reliably manage cash inventories results in ATMs overstocked with cash that could otherwise be an invested asset or run the risk of lost fees and costly emergency cash deliveries.

Data mining is the process of sorting through large amounts of data and picking out relevant information. It is usually used by business intelligence organizations, and financial analysts, but is increasingly being used in the sciences to extract information from the enormous data sets generated by modern experimental and observational methods. It has been described as "the nontrivial extraction of implicit, previously unknown, and potentially useful information from data" and "the science of extracting useful information from large data sets or databases." Data mining in relation to enterprise resource planning is the statistical and logical analysis of large sets of transaction data, looking for patterns that can aid decision making(Peng, Y., Kou, G., Shi, Y. and Chen, Z, 2006).

Data mining is largely used in several applications such as understanding consumer research marketing, product analysis, demand and supply analysis, e-commerce, investment trend in stocks & real estates, telecommunications and so on. Data mining is based on mathematical algorithm and analytical skills to drive the desired results from the huge database collection.

Data mining has great importance in today’s highly competitive business environment. A new concept of Business Intelligence data mining has evolved now, which is widely used by leading corporate houses to stay ahead of their competitors. Business Intelligence (BI) can help in providing latest information and used for competition analysis, market research, economical trends, consume behavior, industry research, geographical information analysis and so on. Business Intelligence Data Mining helps in decision-making(EibeFrank,2005).

Data mining applications are widely used in direct marketing, health industry, e-commerce, customer relationship management (CRM), FMCG industry, telecommunication industry and financial sector. Data mining is available in various forms like text mining, web mining, audio & video data mining, pictorial data mining, relational databases, and social networks data mining.

Data mining, however, is a crucial process and requires lots of time and patience in collecting desired data due to complexity and of the databases. This could also be possible that you need to look for help from outsourcing companies. These outsourcing companies are specialized in extracting or mining the data, filtering it and then keeping them in order for analysis. Data Mining has been used in different context but is being commonly used for business and organizational needs for analytical purposes.

As stated reasons above, in this study, a model for prediction of drawn money from ATMs is developed with data mining. Also different approaches and algorithms are evaluated and compared on same dataset.

2. BACKGROUND

In the study, ATM Cash Management data between 06.04.2010 and 15.06.2010 of a bank were used. The total numbers of data row were 14441 after the pruning phase. These data were collected from 265 ATM (165 ON SITE, 100 OFF SITE). The number of payday between the 06.04.2010 and 15.06.2010 were 443 and the numbers of other days were 13998.

Everyday, each ATM data was recorded. The recorded data were the following; day, ATM no, location (ON SITE / OFF SITE), payday or not payday, drawn amount. The mean and range of the ATM parameters are presented in the Table 2.1

Table 2.1 ATM Dataset Parameters

Parameters	Range
Day	(1-7)
ATM No	(1-265)
Location	On Site 165 , Off Site 100
Payday	Payday 443, Not Payday 13998
Drawn Money	(1000TL-40000TL):9630TL

The optimization of the amount of kept in ATM is a very important problem for financial institutions, especially for banks.The kept amount of money in an ATM,would cause insatisfation for ATM users.Although basically the purposes of foundation of companies to make profits for this purpose, customer satisfaction, timely manner to provide the requested service,and the economic planning obligations as are required to provide.As mentioned, out all the obligations that is basically done in the best way of cash management raises the need to automate.One of the best methods to be used for this purpose, is the data minig. Data mining is the maximum yield with minimum cost will be.

The main purpose of the foundations is earning much money so they have to provide satisfaction of their workers and customers, also they have to supply the products and services according to the criterion required by customer and on time with using the most economic way. It clearly seems that it is essential to atomize the cash management. The one of the best way of gain this purpose is using Data Mining. Data Mining can supply the maximum gain with minimum loss.

In this study, data mining techniques used for prediction of drawn amount in Automated Teller Machine (ATM). At the end of the study predictions of different data mining algorithms are compared to each other to see which method is better and efficient on large amount of datasets.

2.1. DATA CLEANING PHASE

One of the first and most important steps in any data processing task is to verify that your data values are correct or, at the very least, conform to some a set of rules. For example, a variable called "Location" would be expected to have only two values; whether ATM is located inside the bank or offsite. Some critical applications require a double entry and verification process of data entry. Whether this is done or not, it is still useful to run your data through a series of data checking operations.

At the beginning of the study, each attribute were checked against noise with the help of scripts. Anomalous conditions were corrected.

Study is based on 3 months-dataset. Due to this condition, some unnecessary attributes were removed from dataset. To set the data in suitable format for algorithms, the date format converted into separate days and months column. After that, basic data mining approaches performed on updated datasets, but still results were undesirable. If there was a dataset of a whole year it would be useful to use month attribute, however because of the 3 months-dataset it is appeared that month column is unnecessary. By removing the month attribute and coding the day attribute into 7-valued week system, better estimates and better time performances obtained.

To have more consistent results, rows with the excessive high and low Drawn Money attributes were removed. Additionally a new column is created for grouping Drawn Money attribute. After this cleaning process, desired results were obtained.

2.2. FACTORS THAT AFFECTS ESTIMATION RESULTS

The most important reason is disorder of salary days. Besides, breakdown of ATMs affects anomaly on the recorded data. Also duplicate use of ATM numbers for different ATMs cause inconsistence data record. Duplicate numbered ATMs are corrected. In every place replacement of ATMs a new, unique ATM number must be assigned.

3. ADAPTIVE-NETWORK-BASED FUZZY INFERENCE SYSTEM (ANFIS)

In the literature sources, we can find different kinds of justification for fuzzy systems theory. Human knowledge nowadays becomes increasingly important, we gain it from experiencing the world within which we live and use our ability to reason to create order in the mass of information. Since we are all limited in our ability to perceive the world and to profound reasoning, we find ourselves everywhere confronted by uncertainty which is a result of lack of information (lexical impression, incompleteness), in particular, inaccuracy of measurements. The other limitation factor in our desire for precision is a natural language used for describing/sharing knowledge, communication, etc. We understand core meanings of word and are able to communicate accurately to an acceptable degree, but generally we cannot precisely agree among ourselves on the single word or terms of common sense meaning. In short, natural languages are vague(J.-S. Roger Jang, 2004).

Our perception of the real world is pervaded by concepts which do not have sharply defined boundaries; for example, many, tall, much larger than, young, etc. are true only to some degree and they are false to some degree as well. These concepts can be called fuzzy or gray (vague) concept, a human brain works with them, while computers may not do it. Natural languages, which are much higher in level than programming languages, are fuzzy whereas programming languages are not. The door to the development of fuzzy computers was opened in 1985 by the design of the first logic chip by Masaki Togai and Hiroyuki Watanabe at Bell Telephone Laboratories. In the years to come fuzzy computers will employ both fuzzy hardware and fuzzy software, and they will be much closer in structure to the human brain than the present-day computers are. The entire real world is complex; it is found that the complexity arises from uncertainty in the form of ambiguity.

The Fuzzy Logic tool was introduced in 1965, also by Lotfi Zadeh, and is a mathematical tool for dealing with uncertainty. It offers to a soft computing partnership the important concept of computing with words'. It provides a technique to deal with imprecision and information granularity. The fuzzy theory provides a mechanism for representing linguistic constructs such as “many,” “low,” “medium,” “often,” “few.” In general, the fuzzy logic provides an inference structure that enables appropriate human reasoning capabilities(Zadeh, L.A., 1968).

On the contrary, the traditional binary set theory describes crisp events, events that either do or do not occur. It uses probability theory to explain if an event will occur, measuring the chance with which a given event is expected to occur. The theory of fuzzy logic is based upon the notion of relative graded membership and so are the functions of mentation and cognitive processes. The utility of fuzzy sets lies in their ability to model uncertain or ambiguous data, Figure 3.1. so often encountered in real life.

It is important to observe that there is an intimate connection between Fuzziness and Complexity. As the complexity of a task (problem), or of a system for performing that task, exceeds a certain threshold, the system must necessarily become fuzzy in nature. Zadeh, originally an engineer and systems scientist, was concerned with the rapid decline in information afforded by traditional mathematical models as the complexity of the target system increased. As he stressed, with the increasing of complexity our ability to make precise and yet significant statements about its behavior diminishes. Real world problems (situations) are too complex, and the complexity involves the degree of uncertainty; as uncertainty increases, so does the complexity of the problem. Traditional system modeling and analysis techniques are too precise for such problems (systems), and in order to make complexity less daunting we introduce appropriate simplifications, assumptions, etc. (i.e., degree of uncertainty or Fuzziness) to achieve a satisfactory compromise between the information we have and the amount of uncertainty we are willing to accept. In this aspect, fuzzy systems theory is similar to other engineering theories, because almost all of them characterize the real world in an approximate manner(Von Altrock, 1995).

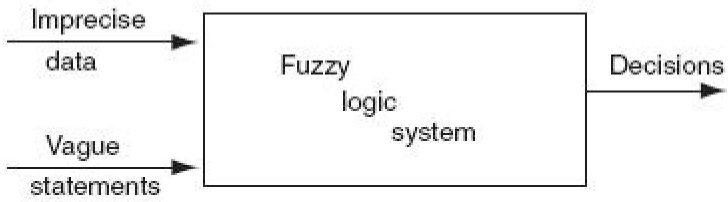

Figure 3.1: A Fuzzy Logic System

In practice fuzzy logic means computation of words. Since computation with words is possible, computerized systems can be built by embedding human expertise articulated in daily language. Also called a fuzzy inference engine or fuzzy rule-base, such a system can perform approximate reasoning somewhat similar to but much more primitive than that of the human brain. Computing with words seems to be a slightly futuristic phrase today since only certain aspects of natural language can be represented by the calculus of fuzzy sets, but still fuzzy logic remains one of the most practical ways to mimic human expertise in a realistic manner. The fuzzy approach uses a premise that humans do not represent classes of as fully disjoint but rather as sets in which there may be grades of membership intermediate between full membership and non-membership. Thus, a fuzzy set works as a concept that makes it possible to treat fuzziness in a quantitative manner(Von Altrock, 1995).

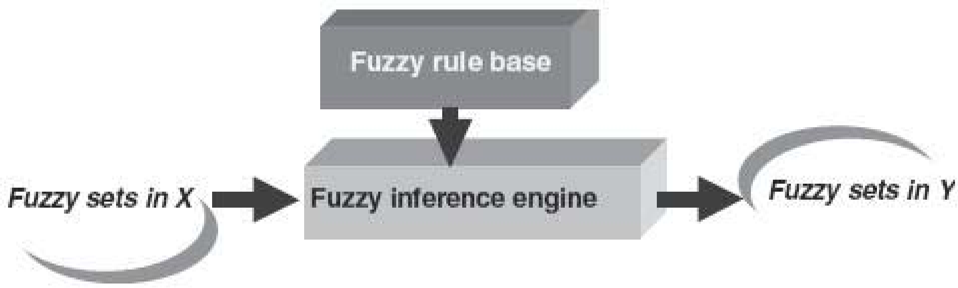

Figure 3.2: Configuration of a Pure Fuzzy System

Fuzzy sets form the building blocks for fuzzy IF–THEN rules which have the general form "IF *X* is *A* THEN *Y* is *B*," where *A* and *B* are fuzzy sets. The term "fuzzy systems" refers mostly to systems that are governed by fuzzy IF–THEN rules. The IF part of an implication is called the antecedent whereas the second, THEN part is a consequent. A fuzzy system is a set of fuzzy rules that converts inputs to outputs. The basic configuration of a pure fuzzy system is shown in Figure 3.2 The fuzzy inference engine (algorithm) combines fuzzy IF–THEN rules into a mapping from fuzzy sets in the input space *X* to fuzzy sets in the output space *Y* based on fuzzy logic principles. From a knowledge representation viewpoint, a fuzzy IF–THEN rule is a scheme for capturing knowledge that involves imprecision. The main feature of reasoning using these rules is its partial matching capability, which enables an inference to be made from a fuzzy rule even when the rule's condition is only partially satisfied.

Fuzzy systems, on one hand, are rule-based systems that are constructed from a collection of linguistic rules; on the other hand, fuzzy systems are nonlinear mappings of inputs to outputs, i.e., certain types of fuzzy systems can be written as compact nonlinear formulas. The inputs and outputs can be numbers or vectors of numbers. These rule-based systems in theory model represent any system with arbitrary accuracy, i.e., they work as universal approximations.

The vital point of a fuzzy system is its rules; smart rules give smart systems and other rules give smart systems and other rules give less smart or even dumb systems. The number of rules increases exponentially with the dimension of the input space (number of system variables). This rule explosion is called the principle of dimensionality and is a general problem for mathematical models. For the last five years several approaches based on decomposition (cluster) merging and fusing have been proposed to overcome this problem. Hence, Fuzzy models are not replacements for probability models. The fuzzy models sometimes found to work better and sometimes they do not. But mostly fuzzy is evidently proved that it provides better solutions for complex problems. (Mamdami, E. H, 2000).

3.1. MEMBERSHIP FUNCTIONS

Fuzziness in a fuzzy set is characterized by its membership functions. It classifies the element in the set, whether it is discrete or continuous. The membership functions can also be formed by graphical representations. The graphical representations may include different shapes. There are certain restrictions regarding the shapes used. The rules formed to represent the fuzziness in an application are also fuzzy. The "shape" of the membership function is an important criterion that has to be considered. There are different methods to form membership functions. This chapter discusses on the features and the various methods of arriving membership functions. (Mamdami, E. H, 2000).

3.1.1. FEATURES OF MEMBERSHIP FUNCTIONS

The feature of the membership function is defined by three properties. They are:
(A) Core
(B) Support
(C) Boundary

The Fig. 4.1 shown below defines the properties listed above.
The membership can take value between 0 and 1.

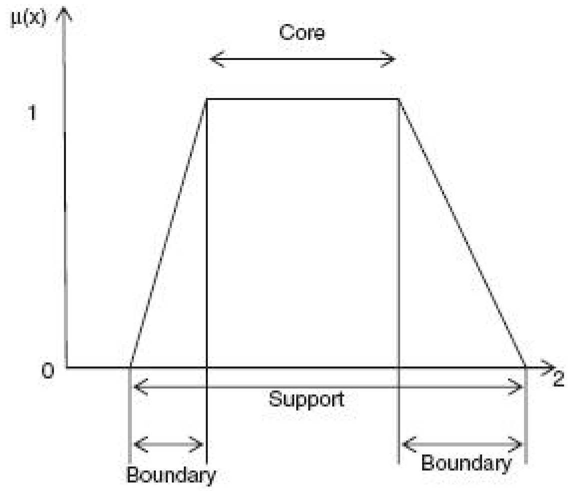

Figure 3.3: Features of Membership Functions

(A) Core

If the region of universe is characterized by full membership in the set *A* then this gives the core of the membership function of fuzzy at *A*.
The elements, which have the membership function as 1, are the elements of the core, i.e., here $\mu A(x) = 1$.

(B) Support

If the region of universe is characterized by nonzero membership in the set A, this defines the support of a membership function for fuzzy set A.

The support has the elements whose membership is greater than 0. $\mu A(x) > 0$.

(C) Boundary

If the region of universe has a nonzero membership but not full membership, this defines the boundary of a membership; this defines the boundary of a membership function for fuzzy set A:

The boundary has the elements whose membership is between 0 and 1, $0 < \mu A(x) < 1$(Mamdami, E. H.).

3.2. CLASSIFICATION OF FUZZY SETS

The fuzzy sets can be classified based on the membership functions. They are:

Normal fuzzy set: If the membership function has at least one element in the universe whose value is equal to 1, then that set is called as normal fuzzy set.

Subnormal fuzzy set: If the membership function has the membership values less than 1, then that set is called as subnormal fuzzy set.

These two sets are shown in Figure 3.4

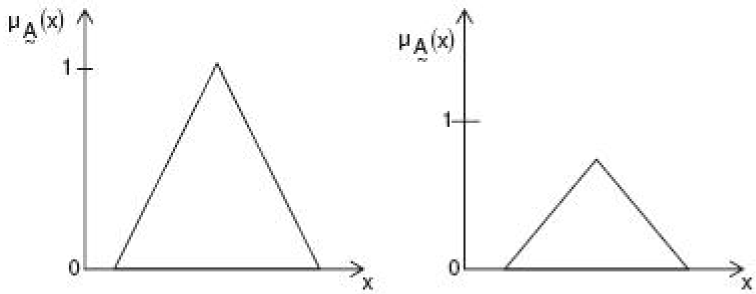

Figure 3.4: Normal and Subnormal Fuzzy Sets

Convex fuzzy set: If the membership function has membership values those are monotonically increasing, or, monotonically decreasing, or they are monotonically increasing and decreasing with the increasing values for elements in the universe, those fuzzy set A is called convex fuzzy set.

Nonconvex fuzzy set: If the membership function has membership values which are not strictly monotonically increasing or monotonically decreasing or both monotonically increasing and decreasing with increasing values for elements in the universe, then this is called as nonconvex fuzzy set.
Figure 3.5 shows convex and nonconvex fuzzy set (Mamdami, E. H).

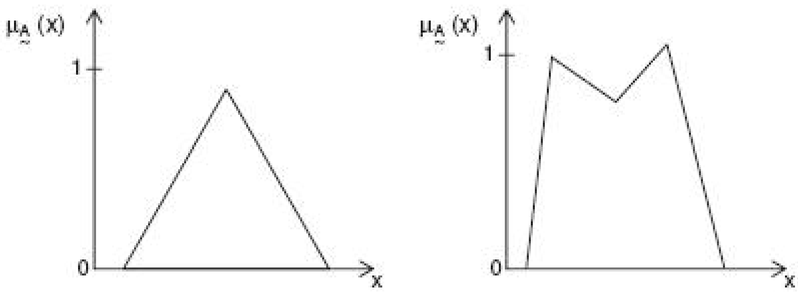

Figure 3.5: Convex and Nonconvex Fuzzy Sets

3.3. FUZZIFICATION

Fuzzification is an important concept in the fuzzy logic theory. Fuzzification is the process where the crisp quantities are converted to fuzzy (crisp to fuzzy). By identifying some of the uncertainties present in the crisp values, we form the fuzzy values. The conversion of fuzzy values is represented by the membership functions.

In any practical applications, in industries, etc., measurement of voltage, current, temperature, etc., there might be a negligible error. This causes imprecision in the data. This imprecision can be represented by the membership functions. Hence fuzzification is performed. Thus fuzzification process may involve assigning membership values for the given crisp quantities.(Mamdami, E. H.).

3.4. FUZZY RULE-BASED SYSTEM

Rules form the basis for the fuzzy logic to obtain the fuzzy output. The rule-based system is different from the expert system in the manner that the rule comprising the rule-based system originates from sources other than that of human experts and hence is different from expert systems. The rule-based form uses linguistic variables as its antecedents and consequents. The antecedents express an inference or the inequality, which should be satisfied. The consequents are those, which we can infer, and is the output if the antecedent inequality is satisfied(J.-S. R. JANG, C.-T. SUN,1993).

3.4.1 FUZZY INFERENCE SYSTEM

Fuzzy inference systems (FISs) are also known as fuzzy rule-based systems, fuzzy model, fuzzy expert system, and fuzzy associative memory. This is a major unit of a fuzzy logic system. The decision-making is an important part in the entire system. The FIS formulates suitable rules and based upon the rules the decision is made. This is mainly based on the concepts of the fuzzy set theory, fuzzy IF–THEN rules, and fuzzy reasoning. FIS uses “IF. . . THEN. . . ” statements, and the connectors present in the rule statement are “OR” or “AND” to make the necessary decision rules. The basic FIS can take either fuzzy inputs or crisp inputs, but the outputs it produces are almost always fuzzy sets. When the FIS is used as a controller, it is necessary to have a crisp output. Therefore in this case defuzzification method is adopted to best extract a crisp value that best represents a fuzzy set. The whole FIS is discussed in detail in the following subsections. (J.-S. R. JANG, C.-T. SUN,1993).

4. METHODS FOR MATLAB

ATM Cash Management data between 06.04.2010 and 15.06.2010 of a bank were used. The total numbers of data row were 14441 after the pruning phase. These data were collected from 265 ATM (165 ON SITE, 100 OFF SITE). The number of payday between the 06.04.2010 and 15.06.2010 were 443 and the numbers of other days were 13998.

Everyday, each ATM data was recorded. The recorded data were the following; day, ATM no, location (ON SITE / OFF SITE), payday or not payday, drawn amount.

After applying the cleaning steps explained in 2.1. a part of optimized data set is shown in Figure 3.6 After all these processes dataset is ready to be applied in ANFIS.

DAY_CODING	ATM_NO	OFFSITE 0/INSITE 1	wage 1 /Oth. 0	Cekilen_Grouped	Gercek_Cekilen
7	222	0	0	4	20080
4	7	1	0	4	20080
2	261	1	0	4	20100
3	200	1	0	4	20110
3	118	1	0	4	20120
7	2	1	0	4	20120
2	209	1	0	4	20120
4	7	1	0	4	20120
1	208	1	1	4	20120
7	7	1	0	4	20130
2	170	1	0	4	20140
3	228	1	0	4	20140
3	7	1	0	4	20150
2	92	1	0	4	20150
1	46	1	0	4	20160

Figure 3.6: Partial View of Optimized Dataset

4.1. 10-FOLD CROSS VALIDATION

After aquiring the optimized dataset it is splitted into two parts which a 60 percent of training and a 40 percent of checking data to run in ANFIS. This operation is repeated randomly ten times to have ten different datasets. When the whole procedure is applied to all ten randomly created dataset, mean values of these datasets are the most accurate results.

4.2. GRID PARTITIONING METHOD

For the five input value of the dataset, number of membership functions (MF) for the best results are as follows:

Table 4.1 Number of Membership Functions of Attributes for Grid Partitioning

Attributes	Number of Membership Functions
Day	2
ATM No	2
Location	1
Payday/Not Payday	1
Drawn Grouping	2

Gaussian curve built-in membership function type is used for grid partition method and output membership function type is linear as shown in Figure 4.1

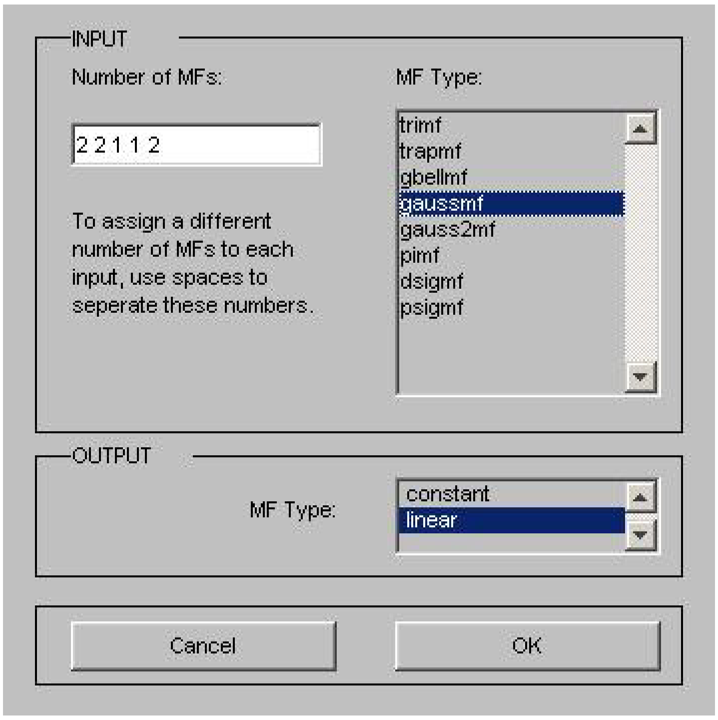

Figure 4.1 MF properties for FIS Generation

After training FIS with the epoch value 10 and the hybrid optimization method, training error graphic is as in Figure 4.2. The mean error after 10 epochs is 1929.9499. Detailed results of each epochs values are shown in Table 4.2

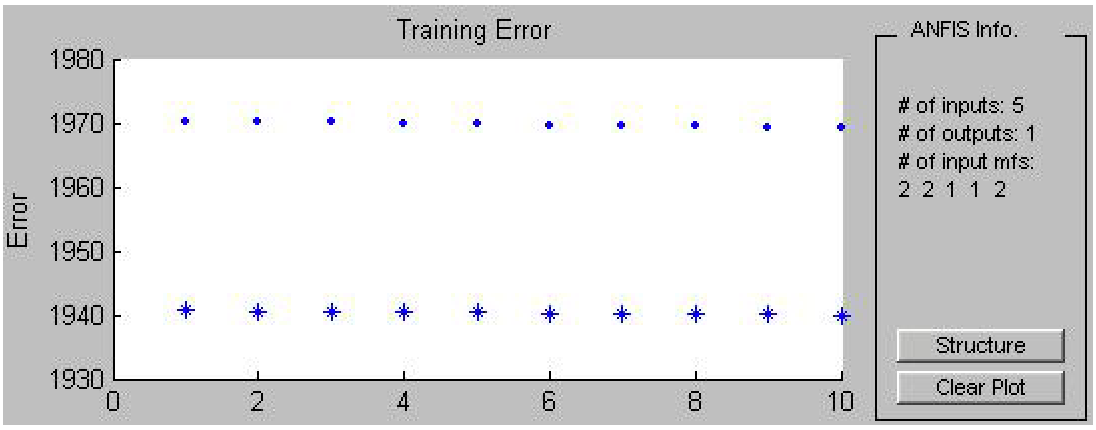

Figure 4.2: Training error graphic with 10 epochs

Table 4.2 Grid Partitioning Results

```
ANFIS info:

  Number of nodes: 40

  Number of linear parameters: 48

  Number of nonlinear parameters: 16

  Total number of parameters: 64

  Number of training data pairs: 8664

  Number of checking data pairs: 5777

  Number of fuzzy rules: 8

Start training ANFIS ...

  1        1940.7        1970.24

  2        1940.61       1970.12

Designated epoch number reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

  1        1940.61       1970.12

  2        1940.52       1970

Designated epoch number reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

  1        1940.52       1970

  2        1940.44       1969.88

Designated epoch numbers reached -> ANFIS training completed at epoch 2.

Start training ANFIS ...

  1        1940.44       1969.88
```

2 1940.35 1969.77

Designated epoch number reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

1 1940.35 1969.77

2 1940.27 1969.65

Designated epoch numbers reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

1 1940.27 1969.65

2 1940.19 1969.54

Designated epoch numbers reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

1 1940.19 1969.54

2 1940.1 1969.43

Designated epoch numbers reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

1 1940.1 1969.43

2 1940.03 1969.31

Designated epoch number reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

1 1940.03 1969.31

2 1939.95 1969.2

Designated epoch number reached --> ANFIS training completed at epoch 2.

Start training ANFIS ...

1	1939.95	1969.2
2	1939.88	1969.1

Designated epoch number reached --> ANFIS training completed at epoch 2.

Avarage testing error according to training data: 1939.8758

Avarage testing error according to checking data: 1969.0967

There are 6 rules according to the number of membership functions. Model structure of the grid partitioning method is shown in Figure 4.3.

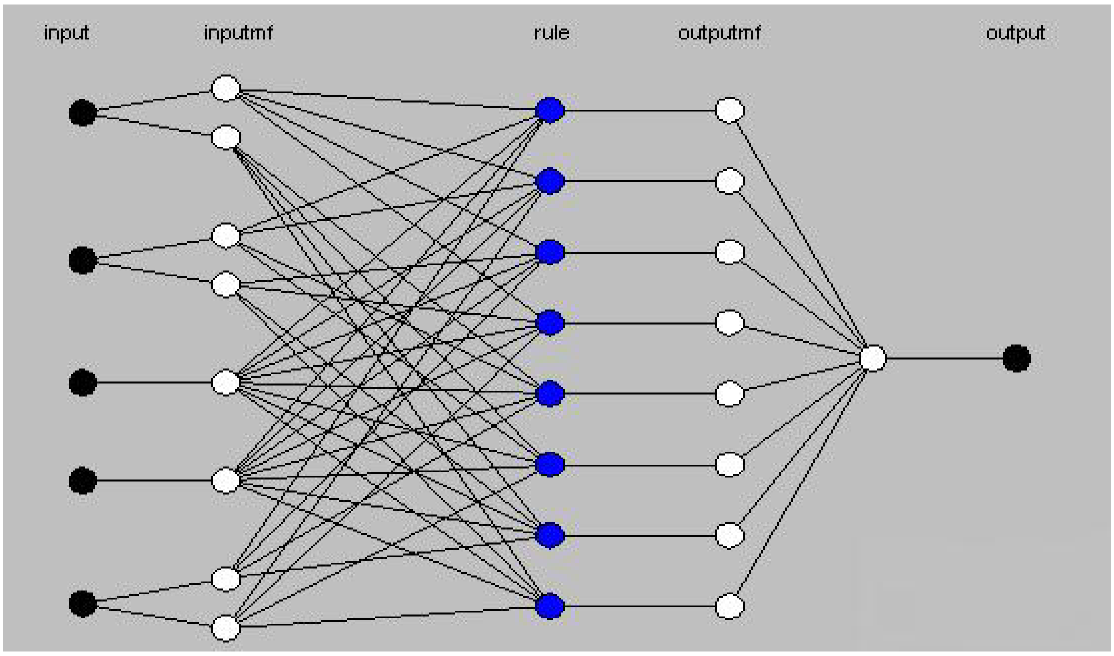

Figure 4.3 : ANFIS Model Structure for Grid Partitioning

After testing FIS file with Grid Partitioning method, estimated outputs with the red dots, as can be seen in Figure 4.4 for training data and Figure 4.5 for checking data, are the whole results. But it is not a useful graphic to see the estimated drawn money for a specific ATM. To make it possible to see specific estimated drawn money *evalfis* function can be used. For example to see if what is the estimated drawn money for the ATM No:78, on Friday, the required function is as follows:

```
predicted(1,:)=evalfis([5 78 0 0 1],outfis);
actual(1,:)=1000;
```

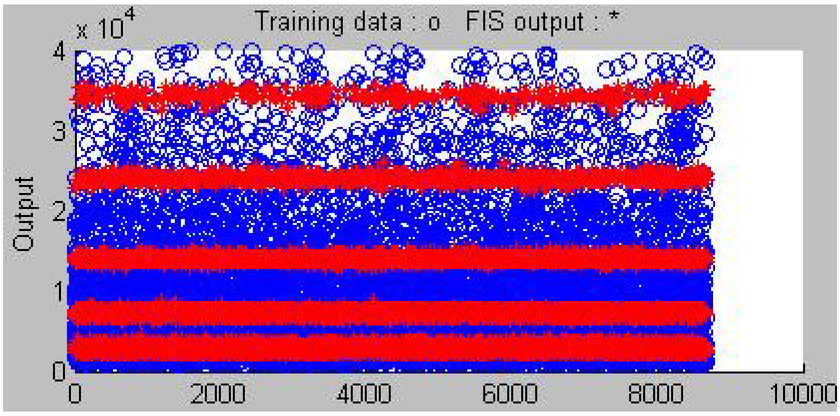

Figure 4.4: Testing FIS (Plot against training data for Grid Partitioning)

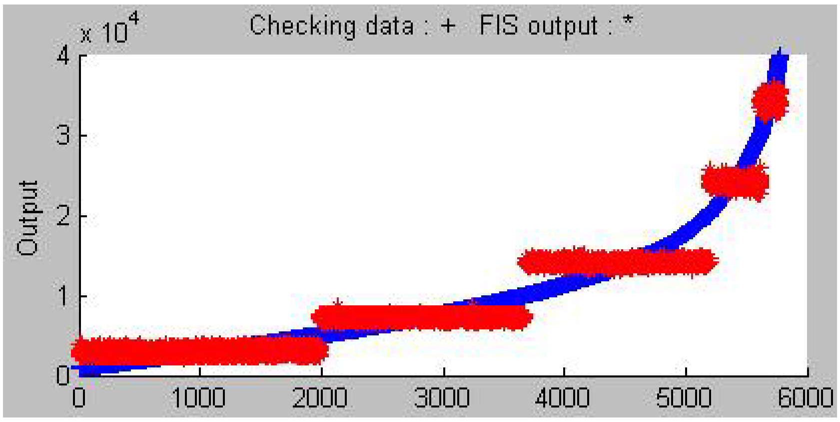

Figure 4.5 Testing FIS (Plot against checking data for Grid Partitioning)

After evaluating the evalfis function on every single data, actual versus estimated values on graphic is shown in Figure 4.6.

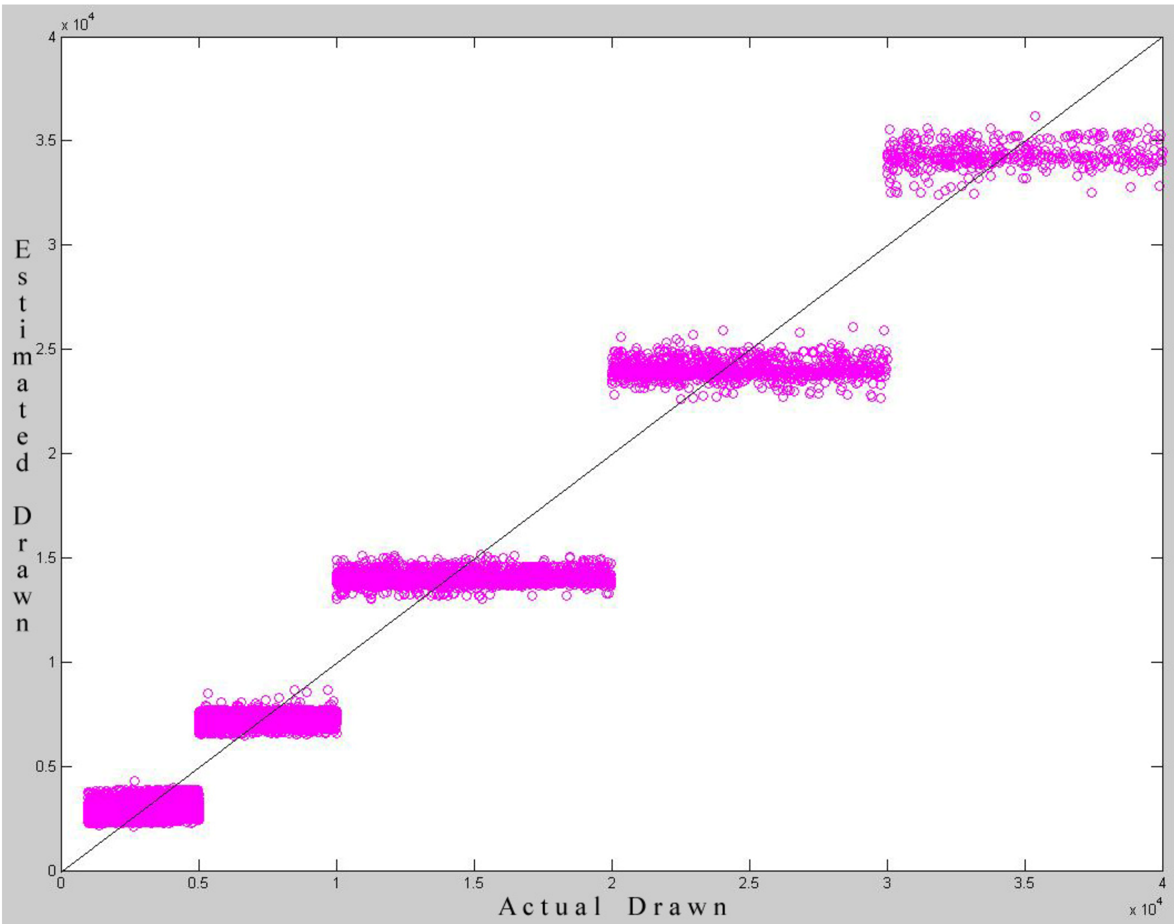

Figure 4.6: Actual Drawn versus Estimated Drawn Graphic for Grid Partitioning Method

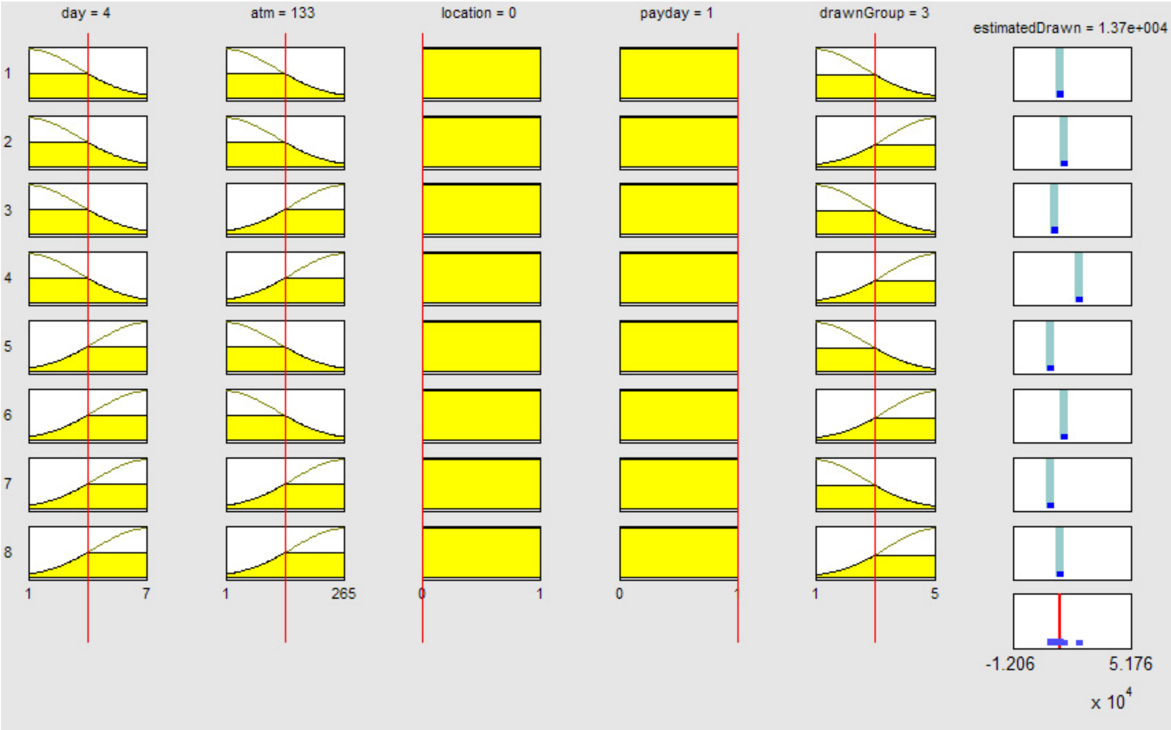

Figure 4.7 Rule View of Grid Partitioning Method

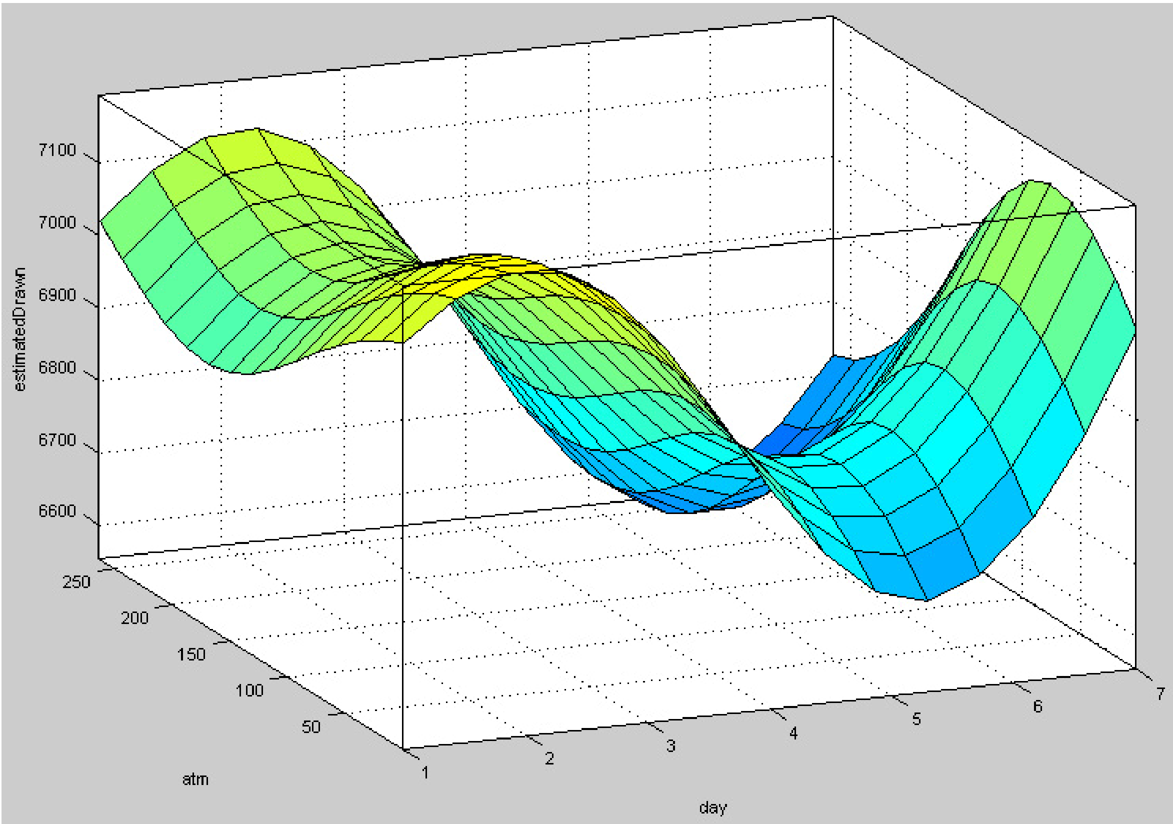

Figure 4.8: Surface Graphic for Estimated Drawn Money according to ATM-Day

As you see in Figure 4.8, this is a generalized view of drawn group *2* for all ATMs, where ATMs are all considered offsite and not on a payday. If it is not a payday, mostly drawn money from ATMs are on Monday, Tuesday and Saturday, Sunday. This is an ordinary situation because people mostly draw their money in the beginning and the end of the week. This is because of to use money for basic needs in the beginning of the week and for entertainment and holiday expense on weekend. In Figure 4.9, *drawn group* is 2, that means estimation should be in the interval of 5.000-10.000 TL. As seen in Figure 4.8, estimation results are between 6.000 and 7.100 TL. That shows grid partitioning gives consistent results. In Figure 4.9, fixed attributes are 'onsite' for *location,* 'payday' and *drawn group* 4. In this dataset, paydays are usually come across on Mondays, just like in graphic. People ordinarily draw their salary on paydays, so this is why ATMs are used more frequently on Mondays. Additionally, attribute location is set to 'onsite',banks are closed, so that is why ATMs are used less on weekend. Drawn group is 4, that means estimations are between 20.000-30.000 TL, like in Figure 4.9.

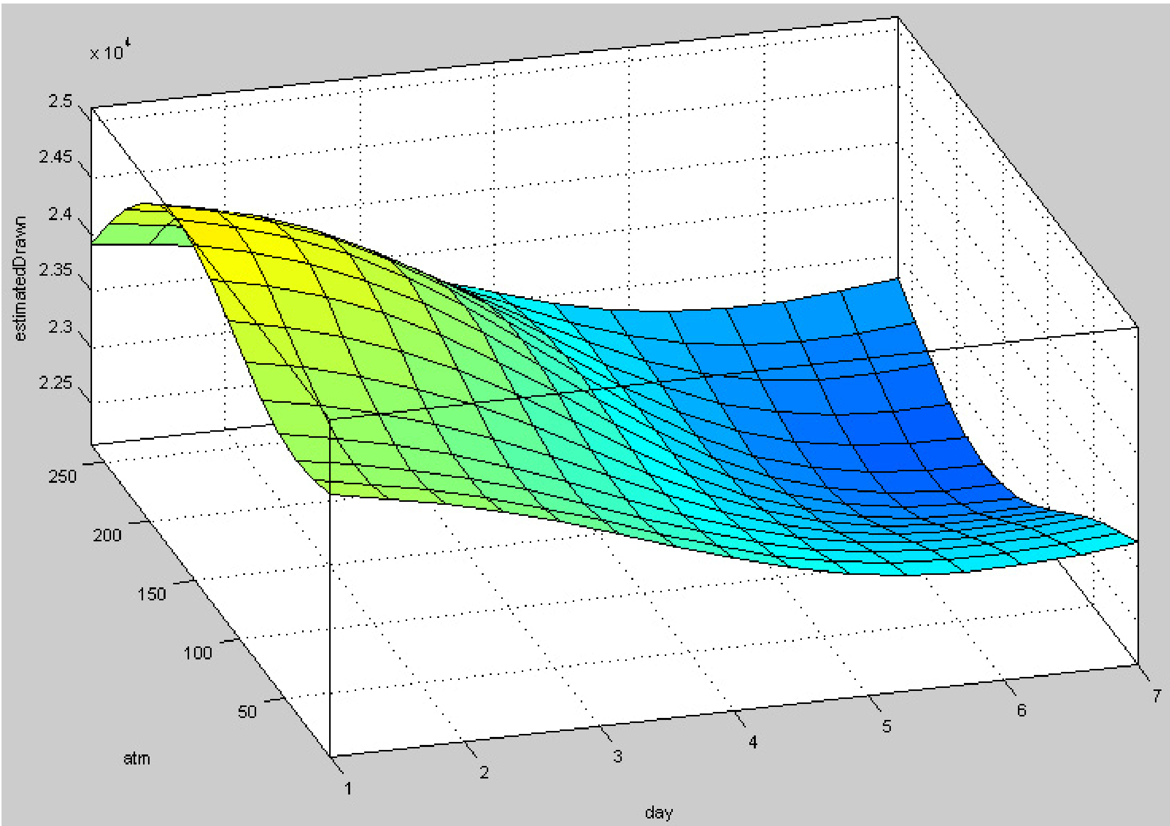

Figure 4.9: Surface Graphic for Estimated Drawn Money according to Day-Location

(Fixed attributes: location(onsite), payday(yes), drawnGroup(4)**)**

4.3. SUBTRACTIVE CLUSTERING

For the five input value of the dataset, number of membership functions (MF) created by ANFIS are as follows:

Table 4.3: Number of Membership Functions of Attributes for Sub. Clustering

Attributes	Number of Membership Functions
Day	18
ATM No	18
Location	18
Payday/Not Payday	18
Drawn Grouping	18

Subtractive clustering is a technique for automatically generating fuzzy inference systems by detecting clusters in input-output training data.Parameters for clustering genfis are shown in Figure 4.9.

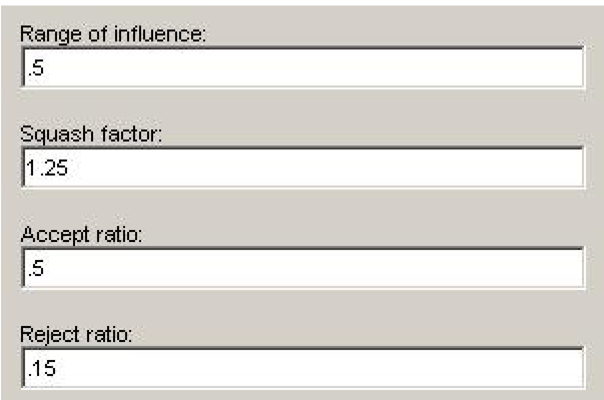

Figure 4.10: Parameters for Subtractive Clustering

After training FIS with the epoch value 10 and the hybrid optimization method, training error graphic is as in Figure 4.10 The mean error after 10 epochs is 1947.1933. Detailed results of each epochs values are shown in Table 4.4

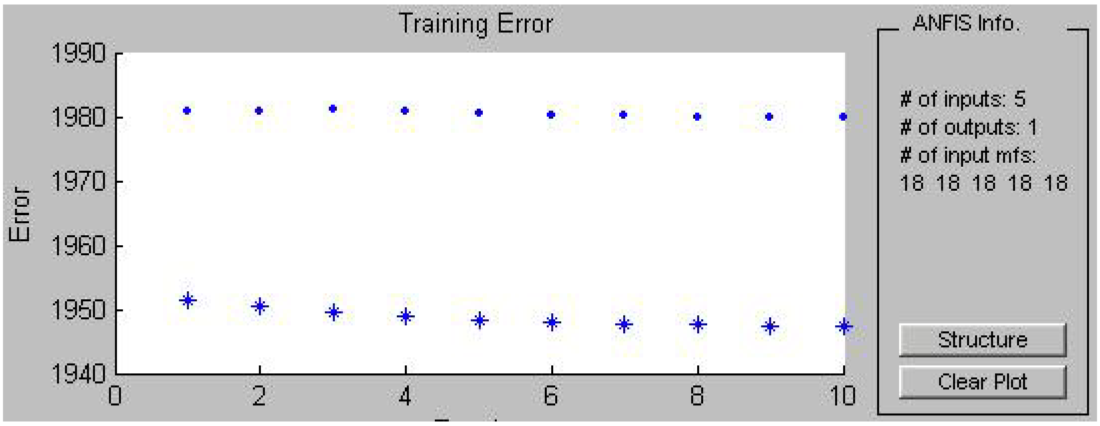

Figure 4.11: Training error graphic with 10 epochs

Table 4.4 Subtractive Clustering Results

```
ANFIS info:

  Number of nodes: 224

  Number of linear parameters: 108

  Number of nonlinear parameters: 180

  Total number of parameters: 288

  Number of training data pairs: 8664

  Number of checking data pairs: 5777

  Number of fuzzy rules: 18

Start training ANFIS ...

  1       1951.27       1980.69

  2       1950.36       1980.86

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1       1950.36       1980.86
```

```
  2          1949.64          1981.01

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1          1949.64          1981.01

  2          1948.83          1980.68

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1          1948.83          1980.68

  2          1948.22          1980.54

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1          1948.22          1980.54

  2          1947.92          1980.04

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1          1947.92          1980.04

  2          1947.8           1980.3

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...
```

```
  1         1947.8          1980.3

  2         1947.56         1979.91

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1         1947.56         1979.91

  2         1947.33         1979.9

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1         1947.33         1979.9

  2         1947.19         1979.84

Designated epoch number reached --> ANFIS training completed at epoch 2.



Start training ANFIS ...

  1         1947.19         1979.84

  2         1946.87         1979.61

Designated epoch number reached --> ANFIS training completed at epoch 2.
```

Avarage testing error according to training data: 1946.867

Avarage testing error according to checking data: 1979.6054

There are 18 rules according to the number of membership functions, shown in Figure 4.11

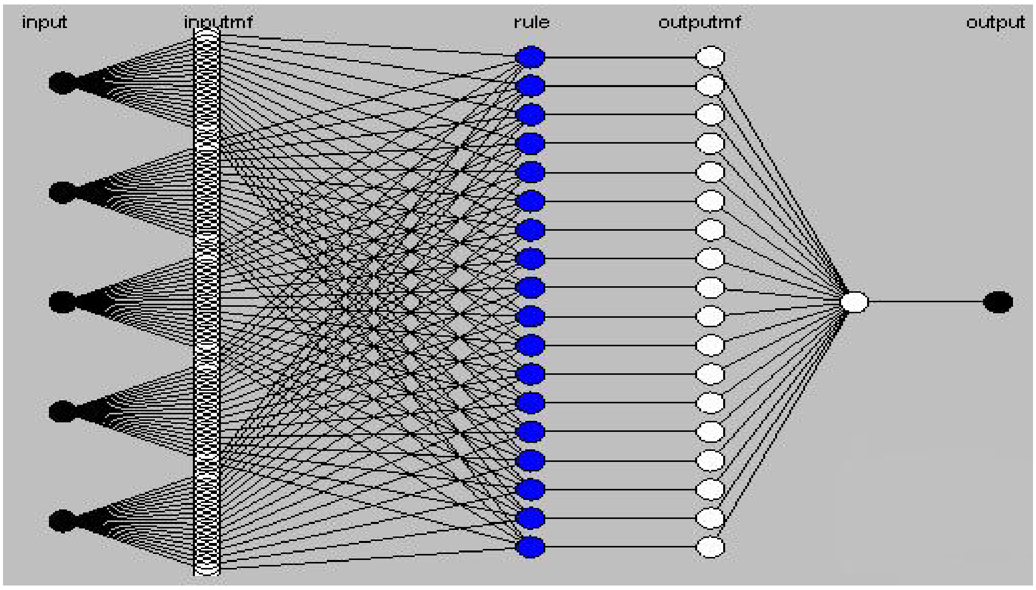

Figure 4.12: ANFIS Model Structure for Subtractive Clustering

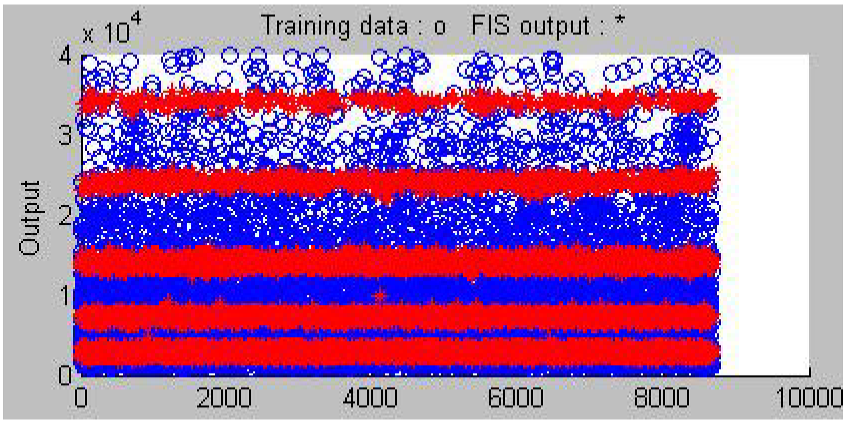

Figure 4.13: Testing FIS (Plot against training data for Subtractive Clustering)

Subtractive Clustering method results nearly the same like Grid Partitioning method.

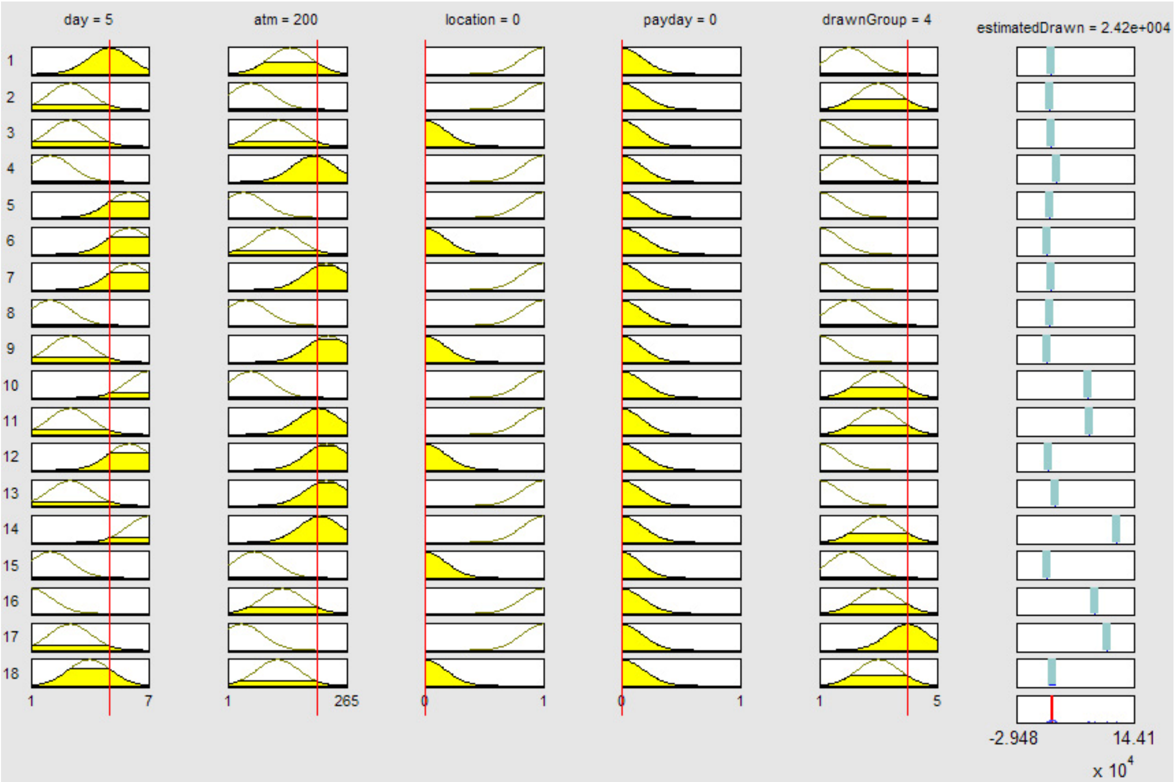

Figure 4.14: Rule View of Subtractive Clustering Method

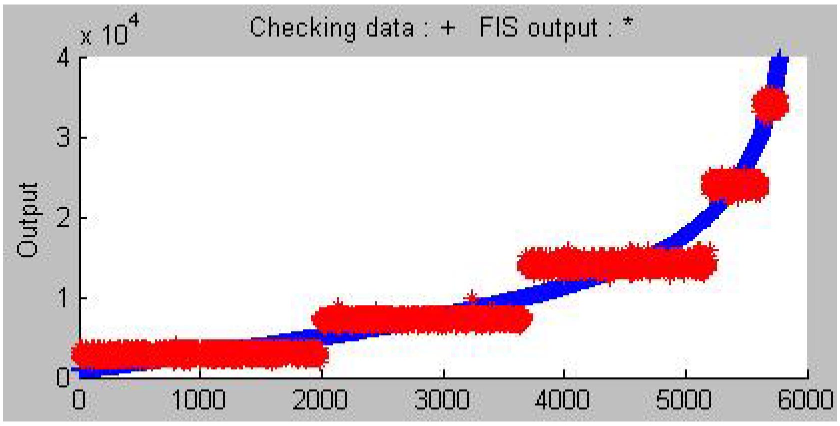

Figure 4.15: Testing FIS (Plot against checking data for Subtractive Clustering)

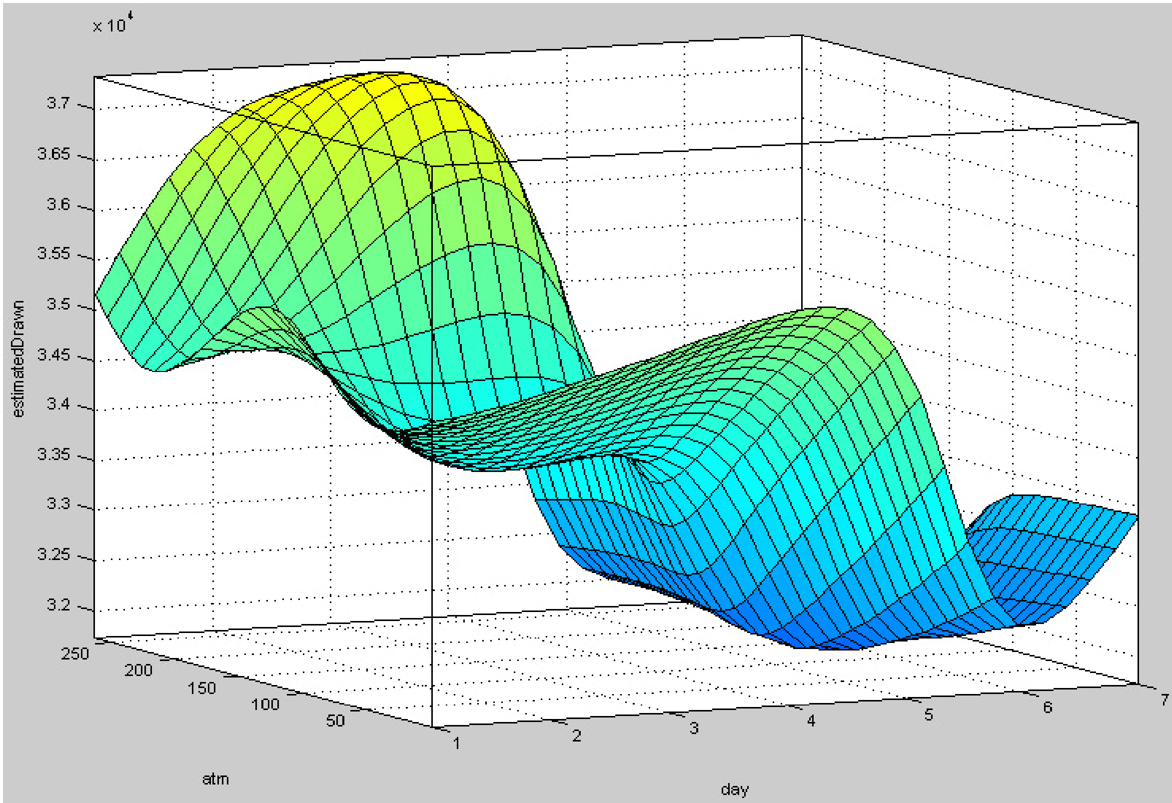

Figure 4.16: Surface Graphic for Estimated Drawn Money according to ATM-Day
(Fixed attributes: location(onsite), payday(yes), drawnGroup(5)**)**

Dataset is split into 5 parts according to the withdrawn money. The groups in order to 1 to 5 are 1.000TL-5.000TL, 5.001TL-10.000TL, 10.001TL-20.000TL, 20.001TL-30.000TL and 30.001TL-40.000TL. Due to this grouping, most frequently used ATMs are in group 5 and the less frequently used ones are in group 1.

In Figure 4.16, *drawn group* is set to 5. That shows the most frequently used ATMs where the highest money is drawn back. Because of the *drawn group* value is 5 and *location* is onsite in Figure 4.16, weekend use of ATMs are less than weekdays. Thats because of banks are closed on weekend and some of the ATMs are inside the bank. Additionally, estimated results are in between 32.000TL and 37.000 TL.

Furthermore, in the 3-month dataset paydays was in general on Mondays and Thursdays. If it is payday like in Figure 4.16, it can be easily seen that most withdrawn money from ATMs are either Monday, Thursday or the following day.

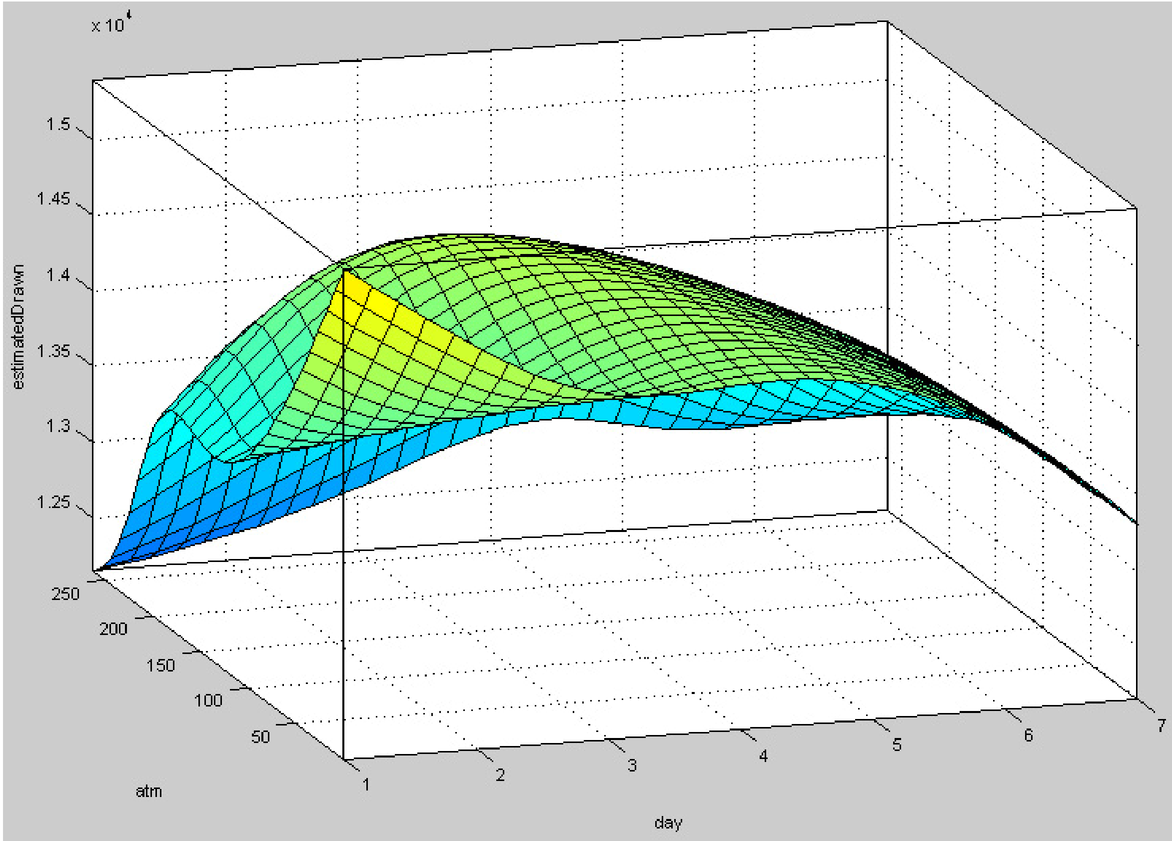

Figure 4.17: Surface Graphic for Estimated Drawn Money according to ATM-Day
(Fixed attributes: location(offsite), payday(no), drawnGroup(3)**)**

In Figure 4.17 drawn group is 3, and it is the middle of the grouping. So, differences of drawn money amount distribution in between ATMs are not too much like in Figure 4.16 . Estimated outputs are between 12.000TL and 15.000TL just have to be in drawn group 3.

Before grouping the dataset according to drawn money amount, estimated results were too bad to use this model as can be seen in Figure 4.18 . But after grouping the data, model works better times and times.

Differences before grouping and after grouping can be seen in Figure 4.18 and Figure 4.19 . In Figure 4.19 results are exceedingly consistent than in Figure 4.18 .
Better results can be obtained if dataset is split into more fragments, but it will be harder and time-consuming.

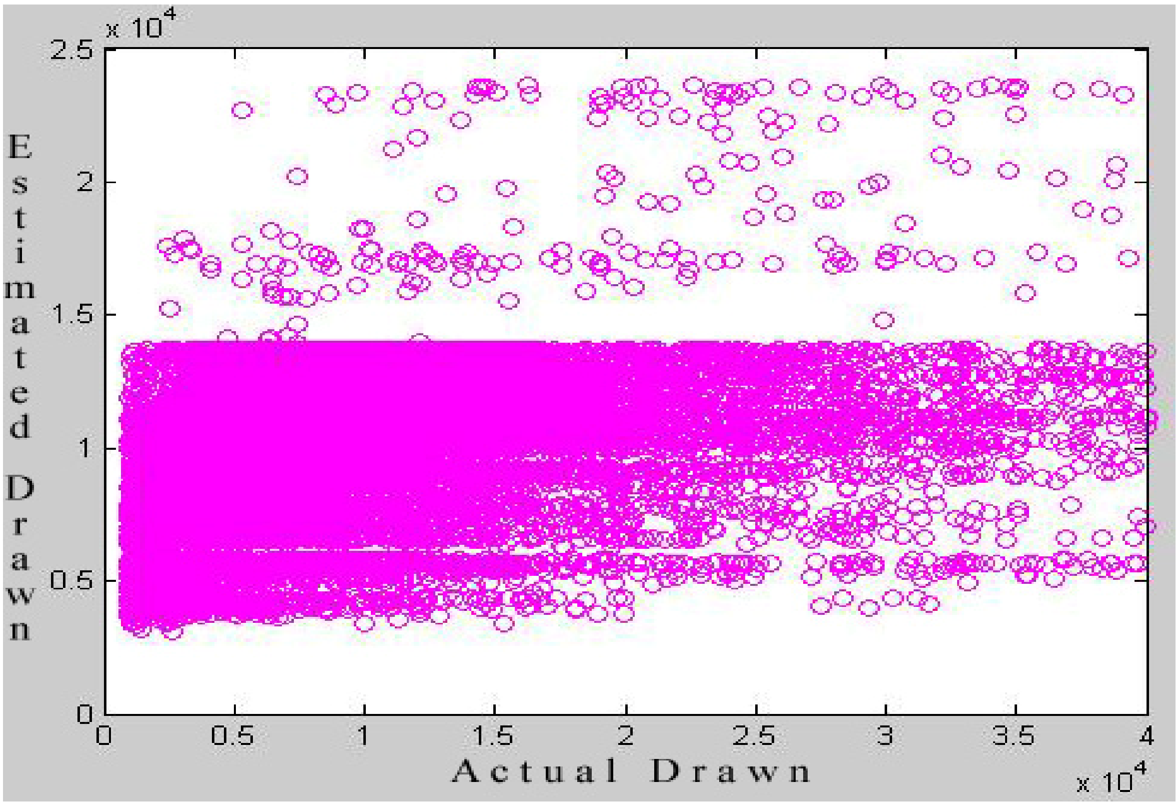

Figure 4.18: Estimated Drawn versus Actual Drawn Before Grouping Due to Drawn Money

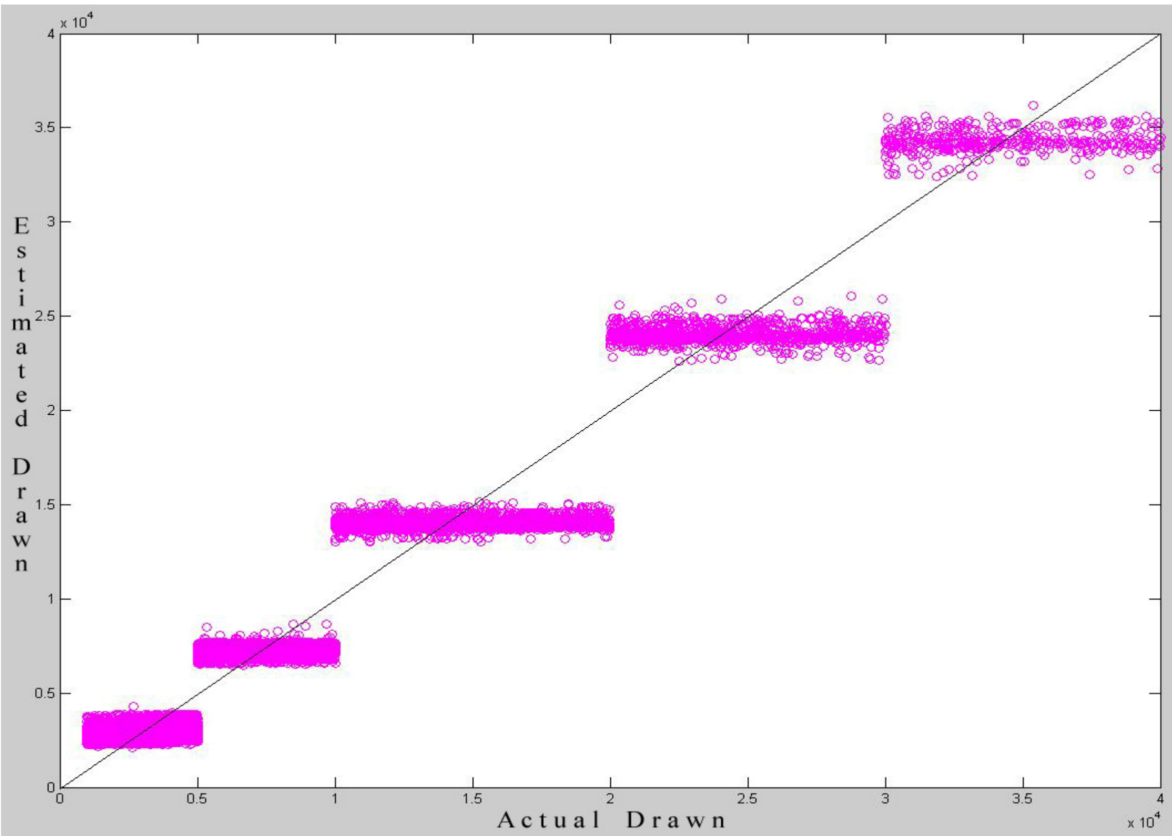

Figure 4.19: Estimated Drawn versus Actual Drawn Money

5. WEKA

5.1 ITERATIVE DICHOTOMISER 3 (ID3)

ID3 (Iterative Dichotomiser 3) is an algorithm used to generate a decision tree invented by Ross Quinlan.

John Ross Quinlan is a pioneering researcher in data mining and decision theory. He has contributed extensively to the development of decision tree algorithms, including inventing the canonical C4.5 and ID3 algorithms. (G. Holmes,1994)
Ross Quinlan invented the Iterative Dichotomiser 3 (ID3) algorithm which is used to generate decision trees. ID3 follows the principal of Occam's razor in attempting to create the smallest decision tree possible. He then expanded upon the principals used in ID3 to create C4.5.

The algorithm is based on Occam's razor: it prefers smaller decision trees (simpler theories) over larger ones. However, it does not always produce the smallest tree, and is therefore a heuristic. Occam's razor is formalized using the concept of information entropy:

$$I_E(i) = -\sum_{j=1}^{m} f(i,j) \log f(i,j)$$

The ID3 algorithm can be summarized as follows:
Take all unused attributes and count their entropy concerning test samples
Choose attribute for which entropy is minimum
Make node containing that attribute
An explanation of the implementation of ID3 can be found at C4.5 algorithm, which is an extended version of ID3(Ian H. Witten; Eibe Frank, Len Trigg, Mark Hall, Geoffrey Holmes, and Sally Jo Cunningham,1999).

Algorithm 1. ID3

```
ID3(in T : table; C : classification attribute)
            return decision tree

{ if (T is empty) then return(null); /* Base case 0 */
   N := a new node;
   if (there are no predictive attributes in T) /* Base case 1 */
then label N with most common value of C in T (deterministic tree)
or with frequencies of C in T (probabilistic tree)
   else if (all instances in T have the same value V of C) /* Base case 2 */
          then label N, "X.C=V with probability 1"
   else { for each attribute A in T compute AVG ENTROPY(A,C,T);
AS := the attribute for which AVG ENTROPY(A,C,T) is minimal;
if (AVG ENTROPY(AS,C,T) is not substantially smaller than ENTROPY(C,T)) /* Base
case 3 */
then label N with most common value of C in T (deterministic tree)
or with frequencies of C in T (probabilistic tree).
else {
             label N with AS;
   for each value V of AS do {
N1 := ID3(SUBTABLE(T,A,V),C) /* Recursive call */
if (N1 != null) then make an arc from N to N1 labelled V;
}
   } }
   return N;
}

SUBTABLE(in T : table; A : predictive attribute; V : value) return table;
{  T1 := the set of instance X in T such that X.A = V;
   T1 := delete column A from T1;
   return T1
}
```

```
ENTROPY(in C : classification attribute; T : table) return real number;
{ for each value V of C, let p(V) := FREQUENCY(C,V,T);
  return −PV p(V ) log2(p(V )) /* By convention, we consider 0 · log2(0) to be 0. */
}

AVG ENTROPY(in A: predictive attribute; C : classification attribute; T : table)
   return real number;
{ return PV FREQUENCY(A,V,T) · ENTROPY(C,SUBTABLE(T,A,V)) }

FREQUENCY(in B : attribute; V : value; T : table) return real number;
{ return #{ X in T | X.B=V } / size(T); } (G. Holmes).
```

5.2 NAIVE BAYES

Bayesian networks are a popular medium for graphically representing and manipulating attribute interdependencies and represent a joint probability distribution over a set of discrete, stochastic variables. Bayesian classification has been widely used in many machine learning applications, also in medical diagnosis. The Bayesian approach searches in a model space for the “best” class descriptions. A best classification optimally trades off predictive accuracy against the complexity of the classes, and so does not overfit the data. Such classes are also fuzzy, instead of each case being assigned to a class, a case has a probability of being a member of each of the different classes. (Kattan M, & Zupan B 2004)

Bayesian networks have several advantages for data analysis. Firstly, since the model encodes dependencies among all variables, it readily handles situations where some data entries are missing. Secondly, a Bayesian network can be used to learn causal relationships and hence can be used to gain understanding about a problem domain and to predict the consequences. Thirdly, because the model includes both causal and probabilistic semantics, it is an ideal representation for combining prior knowledge, which often comes in causal form and data. Finally, bayesian statistical methods in conjunction with bayesian networks offer an efficient and widely recognized approach for avoiding the over-fitting of data. (Demsar J 2004)

The Naive Bayesian Classifier is one of the most computationally efficient algorithm for machine learning and data mining. The naive Bayesian classifier is a bayesian network, used for classification. It is a probabilistic approach to classification. Compared with neural networks, decision trees, clustering and regression, the naive bayesian classifier is a simple and, effective classifier. For example, in medical area, mineral potential mapping is used as naive bayesian classifier. Belonging to the bayesian network classifier, bayesian classifier predicts a class C for a pattern x. The expression of the Bayesian Classification is shown in equation. (Mozina M,2004)

$$P(C|X = x) = \frac{p(X = x|C) \bullet P(C)}{p(X = x)}$$

5.3 CONJUNCTIVE RULE

This class implements a single conjunctive rule learner that can predict for numeric and nominal class labels.

A rule consists of antecedents "AND"ed together and the consequent (class value) for the classification/regression. In this case, the consequent is the distribution of the available classes (or mean for a numeric value) in the dataset. If the test instance is not covered by this rule, then it's predicted using the default class distributions/value of the data not covered by the rule in the training data.This learner selects an antecedent by computing the Information Gain of each antecendent and prunes the generated rule using Reduced Error Prunning (REP) or simple pre-pruning based on the number of antecedents.(I.H. Witten,1994)

For classification, the Information of one antecedent is the weighted average of the entropies of both the data covered and not covered by the rule.
For regression, the Information is the weighted average of the mean-squared errors of both the data covered and not covered by the rule.

6. METHODS FOR WEKA

6.1 DECISION TREE LEARNING

Decision tree learning, used in data mining and machine learning, uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value. More descriptive names for such tree models are classification trees or regression trees. In these tree structures, leaves represent classifications and branches represent conjunctions of features that lead to those classifications.

In decision theory and decision analysis, a decision tree is a graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It can be used to create a plan to reach a goal. Decision trees are constructed in order to help with making decisions. A decision tree is a special form of tree structure. Another use of trees is as a descriptive means for calculating conditional probabilities.

In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data but not decisions; rather the resulting classification tree can be an input for decision making.

Practical example

Our friend David is the manager of a famous golf club. Sadly, he is having some trouble with his customer attendance. There are days when everyone wants to play golf and the staff are overworked. On other days, for no apparent reason, no one plays golf and staff have too much slack time. David’s objective is to optimise staff availability by trying to predict when people will play golf. To accomplish that he needs to understand the reason people decide to play and if there is any explanation for that. He assumes that weather must be an important underlying factor, so he decides to use the weather forecast for the upcoming week. So during two weeks he has been recording:

The outlook, whether it was sunny, overcast or raining.

The temperature (in degrees Fahrenheit).

The relative humidity in percent.

Whether it was windy or not.

Whether people attended the golf club on that day.

David compiled this dataset into a table containing 14 rows and 5 columns as shown below (G. Holmes; A. Donkin and I.H. Witten).

Table 6.1 Play Golf Dataset

Play golf dataset

Independent variables				Dep. var
OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
sunny	85	85	FALSE	Don't Play
sunny	80	90	TRUE	Don't Play
overcast	83	78	FALSE	Play
rain	70	96	FALSE	Play
rain	68	80	FALSE	Play
rain	65	70	TRUE	Don't Play
overcast	64	65	TRUE	Play
sunny	72	95	FALSE	Don't Play
sunny	69	70	FALSE	Play
rain	75	80	FALSE	Play
sunny	75	70	TRUE	Play
overcast	72	90	TRUE	Play
overcast	81	75	FALSE	Play
rain	71	80	TRUE	Don't Play

He then applied a decision tree model to solve his problem.

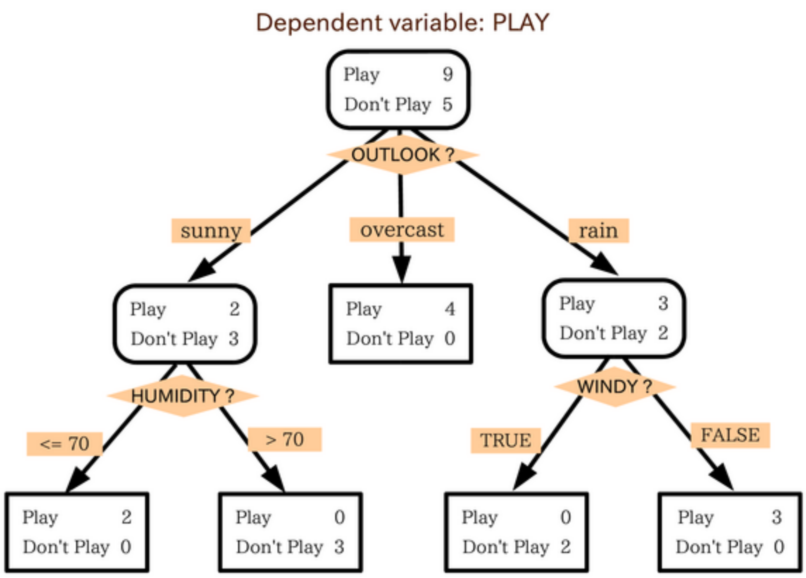

Figure 6.1 Play Golf Decision Tree

A decision tree is a model of the data that encodes the distribution of the class label (again the Y) in terms of the predictor attributes. It is a directed acyclic graph in form of a tree. The top node represents all the data. The classification tree algorithm concludes that the best way to explain the dependent variable, play, is by using the variable "outlook". Using the categories of the variable outlook, three different groups were found:

One that plays golf when the weather is sunny,

One that plays when the weather is cloudy, and

One that plays when it's raining.

David's first conclusion: if the outlook is overcast people always play golf, and there are some fanatics who play golf even in the rain. Then he divided the sunny group in two. He realised that people don't like to play golf if the humidity is higher than seventy percent.

Finally, he divided the rain category in two and found that people will also not play golf if it is windy. (G. Holmes; A. Donkin and I.H. Witten,1994)

And lastly, here is the short solution of the problem given by the classification tree: David dismisses most of the staff on days that are sunny and humid or on rainy days that are windy, because almost no one is going to play golf on those days. On days when a lot of people will play golf, he hires extra staff. The conclusion is that the decision tree helped David turn a complex data representation into a much easier structure (parsimonious).

6.1.1 DECISION TREE ADVANTAGES

Amongst other data mining methods, decision trees have several advantages:

Simple to understand and interpret. People are able to understand decision tree models after a brief explanation.

Requires little data preparation. Other techniques often require data normalization, dummy variables need to be created and blank values to be removed.

Able to handle both numerical and categorical data. Other techniques are usually specialized in analyzing datasets that have only one type of variable. Ex: relation rules can be only used with nominal variables while neural networks can be used only with numerical variables.

Use a white box model. If a given situation is observable in a model the explanation for the condition is easily explained by Boolean logic. An example of a black box model is an artificial neural network since the explanation for the results is excessively complex to be comprehended.
Possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model.

Robust, perform well with large data in a short time. Large amounts of data can be analyzed using personal computers in a time short enough to enable stakeholders to take decisions based on its analysis (G. Holmes; A. Donkin and I.H. Witten).

Table 6.2 ATM Cash Management Dataset ID3 Decision Tree (Just for first 3 ATMs)

```
atm = 1
|  day = 1
|  |  group = 1: null
|  |  group = 2: 8960
|  |  group = 3
|  |  |  wage = 0: 10780
|  |  |  wage = 1: 16310
|  |  group = 4: null
|  |  group = 5: null
|  day = 2
|  |  group = 1: null
|  |  group = 2: 7400
|  |  group = 3: 10440
|  |  group = 4: 27660
|  |  group = 5: null
|  day = 3
|  |  group = 1: null
|  |  group = 2: 6950
|  |  group = 3: 10060
|  |  group = 4: 20200
|  |  group = 5: null
|  day = 4
|  |  group = 1: null
|  |  group = 2: 9450
|  |  group = 3: 11230
```

```
atm = 2
|  day = 1
|  |  group = 1: null
|  |  group = 2: 8470
|  |  group = 3: 13670
|  |  group = 4: 20490
|  |  group = 5: 34100
|  day = 2
|  |  group = 1: null
|  |  group = 2: 7750
|  |  group = 3: 11540
|  |  group = 4: 21800
|  |  group = 5: 31080
|  day = 3
|  |  group = 1: null
|  |  group = 2: 5240
|  |  group = 3: 10070
|  |  group = 4: null
|  |  group = 5: 30020
|  day = 4
|  |  group = 1: null
|  |  group = 2: null
|  |  group = 3: 12070
|  |  group = 4: 28500
|  |  group = 5: 32350
```

```
atm = 3
|  day = 1
|  |  group = 1: 4990
|  |  group = 2: null
|  |  group = 3: 12550
|  |  group = 4
|  |  |  wage = 0: 24470
|  |  |  wage = 1: 29790
|  |  group = 5: null
|  day = 2
|  |  group = 1: null
|  |  group = 2: 8060
|  |  group = 3: 10120
|  |  group = 4: 21930
|  |  group = 5: null
|  day = 3
|  |  group = 1: 4780
|  |  group = 2: 8350
|  |  group = 3: 10270
|  |  group = 4: 28880
|  |  group = 5: null
|  day = 4
|  |  group = 1: null
|  |  group = 2: null
```

```
| | group = 4: 23390
| | group = 5: 31220
| day = 5
| | wage = 0: 11000
| | wage = 1: 10270
| day = 6
| | group = 1: 1940
| | group = 2: 9200
| | group = 3: 14250
| | group = 4: null
| | group = 5: null
| day = 7
| | group = 1: 1460
| | group = 2: 6580
| | group = 3: 10470
| | group = 4: 24980
| | group = 5: null
```

```
| day = 5
| | group = 1: null
| | group = 2: 6420
| | group = 3
| | | wage = 0: 10610
| | | wage = 1: 11820
| | group = 4: null
| | group = 5: null
| day = 6
| | group = 1: 2960
| | group = 2: 6170
| | group = 3: null
| | group = 4: null
| | group = 5: 34480
| day = 7
| | group = 1: null
| | group = 2: null
| | group = 3: 12130
| | group = 4: 20120
| | group = 5: 37840
```

```
| | group = 3: 12970
| | group = 4: null
| | group = 5: 36200
| day = 5
| | group = 1
| | | wage = 0:
2770
| | | wage = 1:
3130
| | group = 2: 5150
| | group = 3: 10730
| | group = 4: null
| | group = 5: null
| day = 6
| | group = 1: 1450
| | group = 2: null
| | group = 3: null
| | group = 4: 21980
| | group = 5: null
| day = 7
| | group = 1: null
| | group = 2: null
| | group = 3: 12860
| | group = 4: 28300
| | group = 5: 32110
```

6.2 COMPARISON OF DATA MINING TECHNIQUES

Data mining techniques have been classified with respect to sensitivity and specificity, as given below.

TP, TN, FP, FN, Sensitivity, Specificity, TP Rate, FP Rate, Precision

TP means True Positive items correctly classified. TN means True Negative items correctly classified. FP means False Positive items correctly classified and FN means False Negative items correctly classified. Sensitivity is the fraction of positive instances. It is also defined as TP Rate. TP Rate is defined as Sensitivity and evaluated as

$$\text{TP RATE} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Specificity is the fraction of the samples predicted as positives which are truly positives. It is also defined as 1- FP Rate. Specificity is computed as

$$\text{SPECIFICITY} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

FP Rate is evaluated as

$$\text{FP RATE} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

Precision or accuracy is the fraction of correctly classified samples. It is computed as (G. Holmes; A. Donkin and I.H. Witten).

$$\text{PRECISION} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{TP} + \text{FN} + \text{FP}}$$

6.3 RECEIVER OPERATING CHARACTERISTIC CURVE

Receiver Operating Characteristic (ROC) curve is used to view the performance of classification. ROC curve is drawn as FP Rate (1-specificity) vs TP Rate (sensitivity). The perfect model for ROC curve is a line between the points [0, 0], [0, 1] and a line [0, 1], [1, 1]. Any signal which is close as those lines gives more reliable information. The area under the ROC curve (AUC) is a scalar measure gauging one facet of performance. In this note, five idealized models are utilized to relate the shape of the ROC curve, and the area under it, to features of the underlying distribution of forecasts. This allows for an interpretation of the former in terms of the latter. The analysis is pedagogical in that many of the findings are already known in more general (and more realistic) settings; however, the simplicity of the models considered here allows for a clear exposition of the relation. For example, although in general there are many reasons for an asymmetric ROC curve, the models considered here clearly illustrate that an asymmetry in the ROC curve can be attributed to unequal widths of the distributions. Furthermore, it is shown that AUC discriminates well between “good” and “bad” models, but not between “good” models.(Fawcett T. 2006)

ROC analysis provides tools to select possibly optimal models and to discard suboptimal ones independently from (and prior to specifying) the cost context or the class distribution. ROC analysis is related in a direct and natural way to cost/benefit analysis of diagnostic decision making. Widely used in medicine, radiology, psychology and other areas for many decades, it has been introduced relatively recently in other areas like machine learning and data mining.(Fawcett, T. 2006)

Properties of ROC

• ROC Area:

– 1.0: perfect prediction

– 0.9: excellent prediction

– 0.8: good prediction

– 0.7: mediocre prediction

– 0.6: poor prediction

– 0.5: random prediction

– <0.5: something wrong!

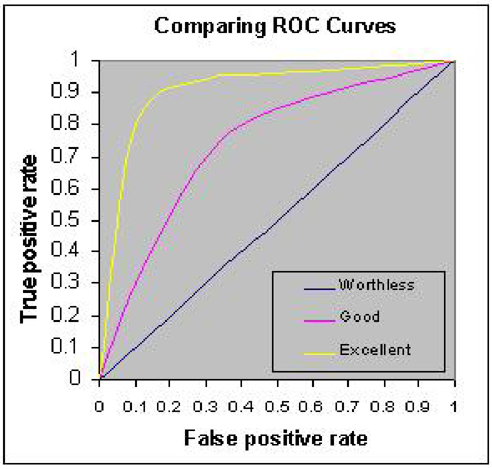

Figure 6.2: Example of ROC Curves

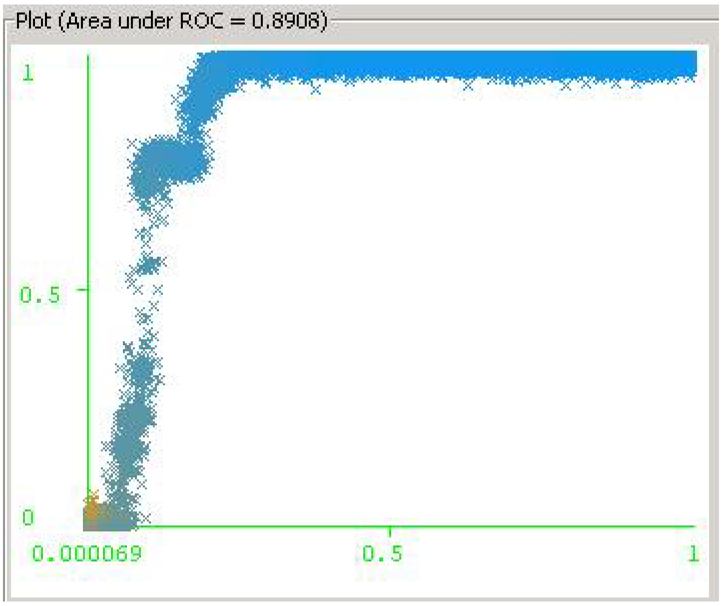

Figure 6.3: ROC Area with using Naïve Bayes

7. RESULTS

In this study, I performed some classification method on this data set. These are ID3, Naïve Bayes and Conjunctive Rule. At the preparation stage for setting the data, 10 fold cross validation method is used.
10-fold Cross Validation
Break data into 10 sets of size n/10.
Train on 9 datasets and test on 1.
Repeat 10 times and take a mean accuracy.

At the comparison stage, we can compare results in few ways. Firstly we can consider Roc Areas. After running the software, Weka found ROC areas, and prediction values for each drawn value. For example; For the Drawn 1330; Naïve Bayes Roc Area is 0.859, ID3 Roc Area is 0.597 and the Conjunctive Rule Roc Area is 827. If we consider all dataset, Naïve bayes had the best Roc Area values, secondly ID3 and the Conjunctive Rule had the worst Roc Area values.

Table 7.1 Example of ROC Area

Drawn	**ROC Area Naïve Bayes**	**ROC Area Id3**	**ROC Area Conjunctive Rule**
1330	0.859	0.597	0.827
6730	0.871	0.642	0.572
19000	0.98	0.643	0.582
22400	0.972	0.667	0.542
33230	0.952	0.5	0.507

Secondly, we can sketch graphs. At these figures, x axis stand for predicted value, and y axis is stands for drawn money. We can easily see, all these algorithms predictions are on the intersecting line except Conjunctive Rule. Naïve Bayes, ID3 and Anfis predictions are close to drawn money, but Conjunctive Rule predictions are less than drawn money.

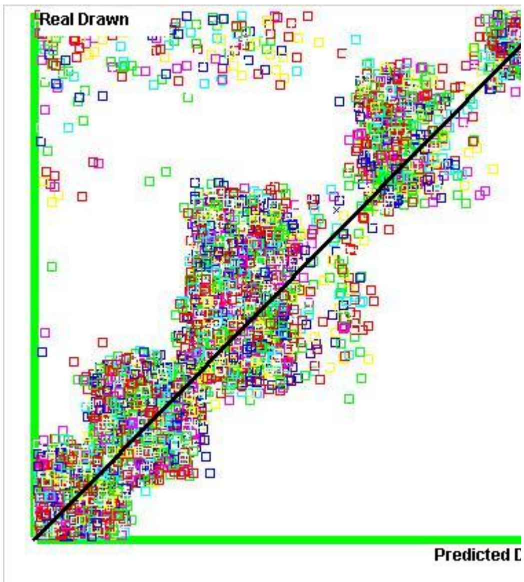

Figure 7.1: Naïve Bayes Prediction Graph

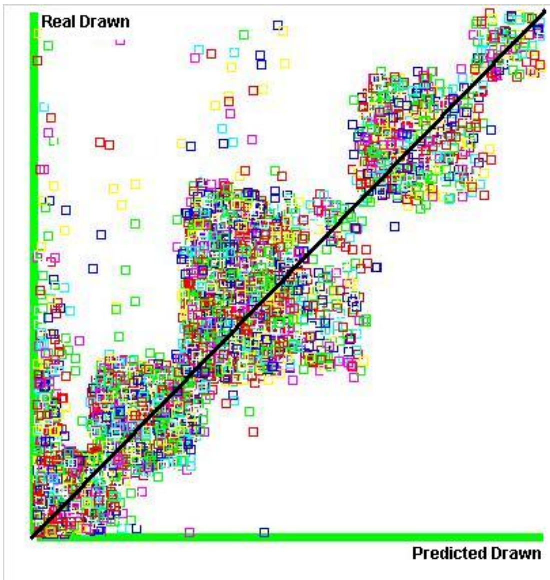

Figure 7.2: ID3 Prediction Graph

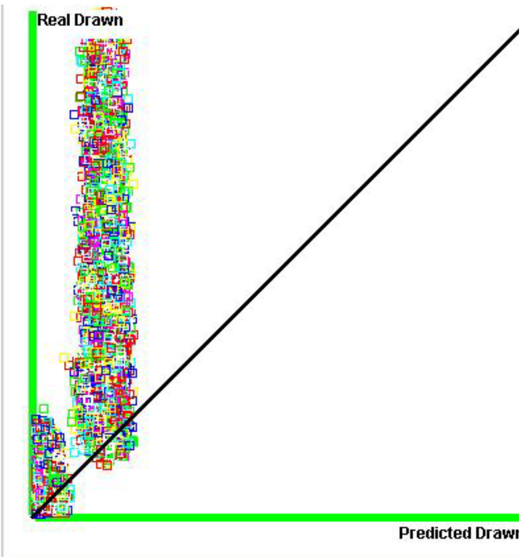

Figure7.3: Conjunctive Rule Prediction Graph

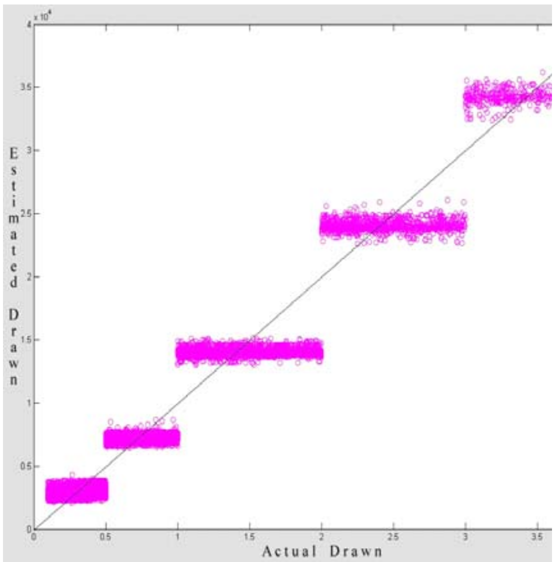

Figure 7.4: Anfis Prediction Graph

If we click on the squares, we can see instance information. Some of these instance information is showed below.

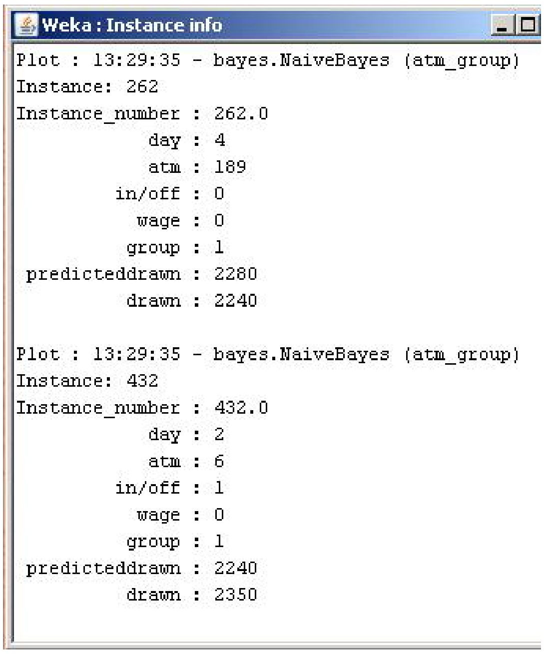

Figure 7.5: Group 1 Instance Information

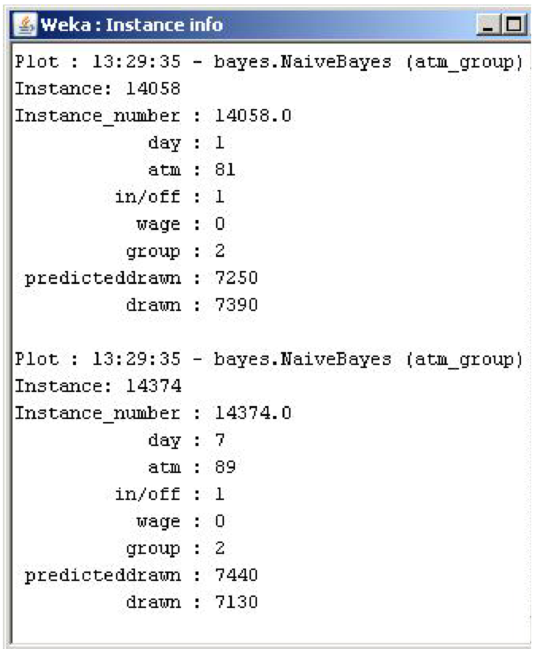

Figure 7.6: Group 2 Instance Information

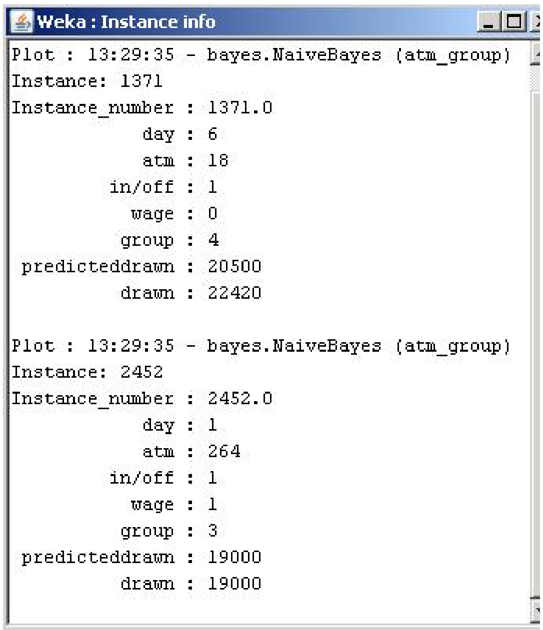

Figure 7.7: Group 3-4 Instance Information

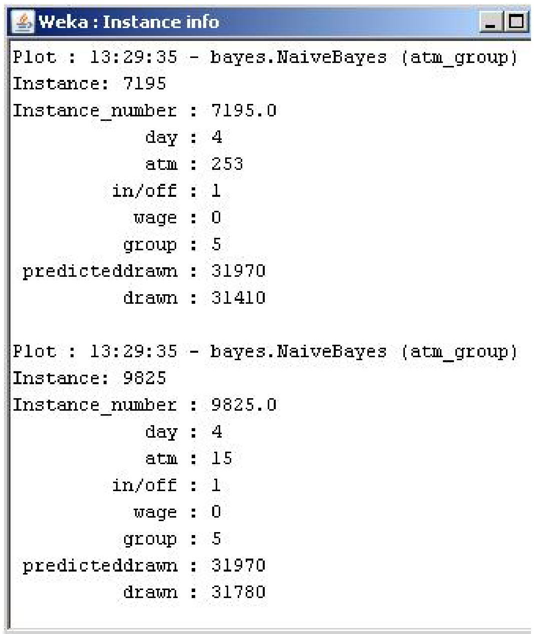

Figure 7.8: Group 5 Instance Information

Table 7.2 Naïve Bayes Summary

```
Time taken to build model: 0.16 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          55                0.3809 %
Incorrectly Classified Instances     14386               99.6191 %
Kappa statistic                          0.0029
Mean absolute error                      0.0007
Root mean squared error                  0.0186
Relative absolute error                 99.89    %
Root relative squared error             99.9577 %
Total Number of Instances            14441
```

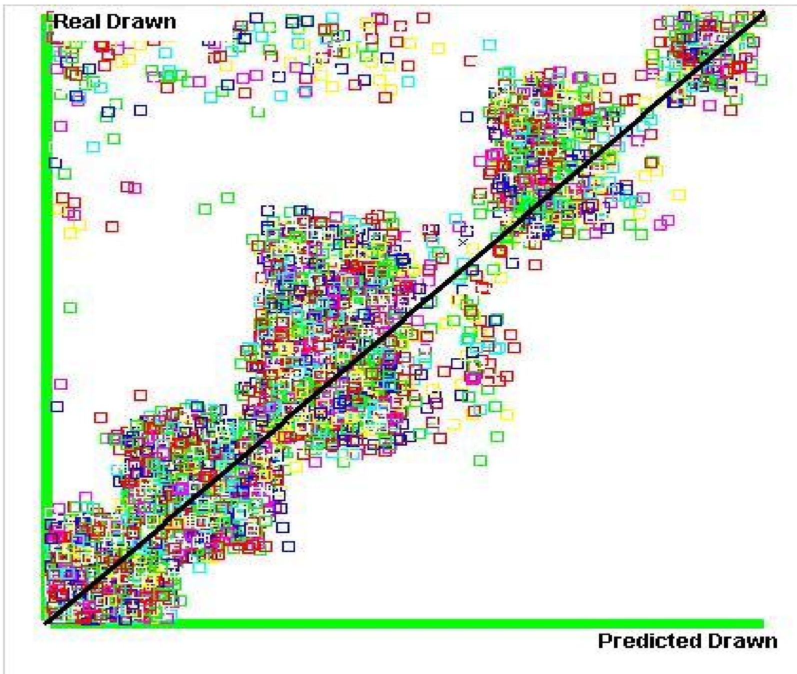

Figure 7.9: Estimated Drawn versus Actual Drawn (Naive Bayes)

Table 7.3 ID3 Summary

```
Time taken to build model: 5.94 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          39               0.2701 %
Incorrectly Classified Instances     12853              89.0035 %
Kappa statistic                          0.0023
Mean absolute error                      0.0007
Root mean squared error                  0.0222
Relative absolute error                111.7747 %
Root relative squared error            126.2021 %
UnClassified Instances                1549              10.7264 %
Total Number of Instances            14441
```

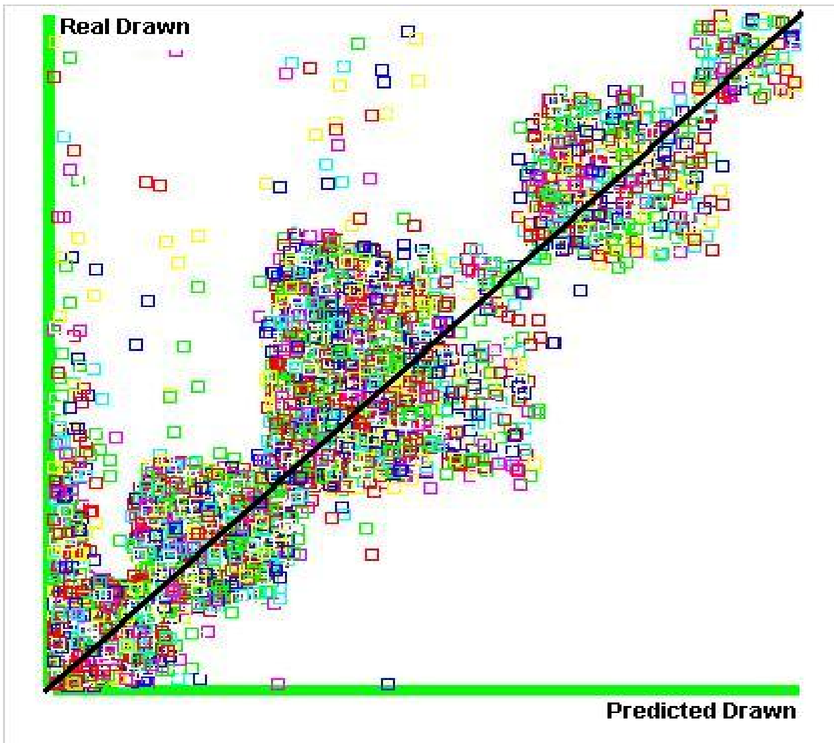

Figure 7.10: Estimated Drawn versus Actual Drawn (Iterative Dichotomiser 3)

Table 7.4 Conjunctive Rule Summary

```
Time taken to build model: 1.03 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          44               0.3047 %
Incorrectly Classified Instances     14397              99.6953 %
Kappa statistic                          0.0016
Mean absolute error                      0.0007
Root mean squared error                  0.0186
Relative absolute error                 99.9196 %
Root relative squared error             99.9636 %
Total Number of Instances            14441
```

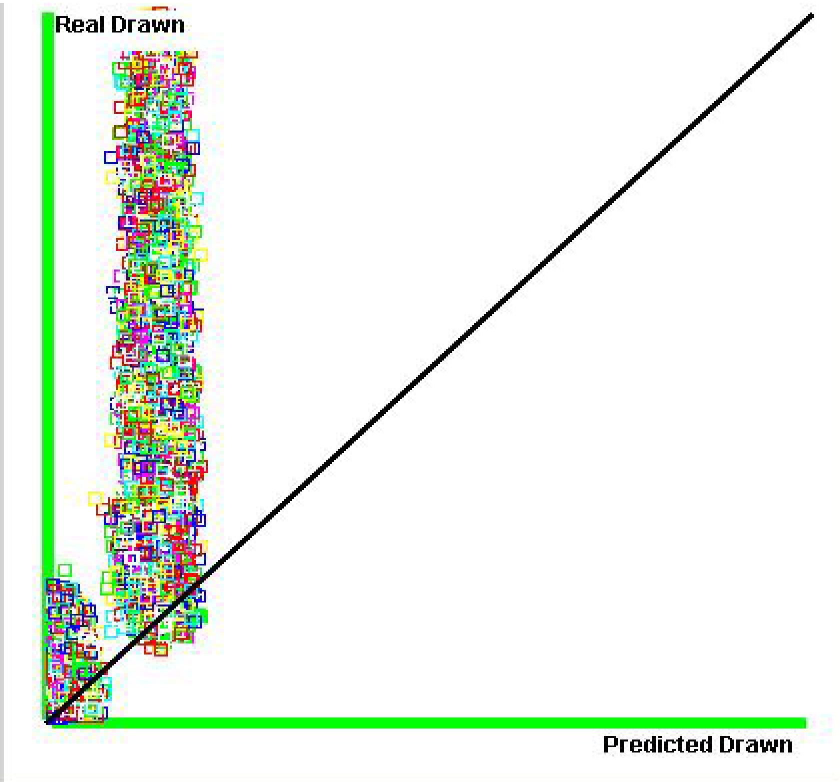

Figure 7.11: Estimated Drawn versus Actual Drawn (Conjunctive Rule)

8. CONCLUSION

After applying all methods on the same dataset, different results are obtained in every method. To find out which method is best for predicting drawn money from ATMs, different error metrics like Root Mean Square Error (RMSE) and some graphics of prediction results are used.

For ANFIS, RMSE value for prediction is 1929,9499.

For Knowledge Extraction Engines (KXEN), which is a commercial data mining automation software, RMSE value is 7388,3920.
For Naive Bayes, normalized RMSE value for prediction is 0,0186.
For Iterative Dichotomiser 3, normalized RMSE value is 0,0222.

After the comparison, best results are obtained from ANFIS. The following best results are given by Naive Bayes and Iterative Dichotomiser 3.

REFERENCES

Degroot, Morris (1980). Probability and Statistics, 2, Addison-Wesley.

Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27, 861-874.

G. Holmes; A. Donkin and I.H. Witten (1994). Weka: A machine learning workbench. Proc Second Australia and New Zealand Conference on Intelligent Information Systems, Brisbane, Australia. Retrieved on 2007-06-25.

Hand, D.J., & Till, R.J. (2001). A simple generalization of the area under the ROC curve to multiple class classification problems. Machine Learning, 45, 171-186.

Ian H. Witten; Eibe Frank (2005). Data Mining: Practical machine learning tools and techniques, 2nd Edition. Morgan Kaufmann, San Francisco. Retrieved on 2007-06-25.

Ian H. Witten; Eibe Frank, Len Trigg, Mark Hall, Geoffrey Holmes, and Sally Jo Cunningham (1999). Weka: Practical Machine Learning Tools and Techniques with Java Implementations. Proceedings of the ICONIP/ANZIIS/ANNES'99 Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems 192-196. Retrieved on 2007-06-26.

J.-S. Roger Jang, "Fuzzy modeling using generalized nureal networks and Kalman filter algorithm," in Proc. Ninth Nat. Conj Arrifical Intell, 2004

Mamdami, E.H.. Application of fuzzy algorithms for the control of a simple dynamic plant,2000

Mitchell, Tom M. Machine Learning. McGraw-Hill, 1997

M.H. Zweig and G. Campbell (1993). "Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine". Clinical chemistry 39 (8): 561–577. PMID 8472349.http://faculty.washington.edu/marzban/roc.pdf

Mozina M, Demsar J, Kattan M, & Zupan B. (2004). "Nomograms for Visualization of Naive Bayesian Classifier". In Proc. of PKDD-2004,

S.R. Garner; S.J. Cunningham, G. Holmes, C.G. Nevill-Manning, and I.H. Witten (1995). Applying a machine learning workbench: Experience with agricultural databases. Proc Machine Learning in Practice Workshop, Machine Learning Conference, Tahoe City, CA, USA 14-21. Retrieved on 2007-06-25.

S.Kotsiantis, P. Pintelas, Increasing the Classification Accuracy of Simple Bayesian Classifier, Lecture Notes in Artificial Intelligence, AIMSA 2004, Springer-Verlag Vol 3192, pp. 198-207, 2004 http://www.math.upatras.gr/~esdlab/en/members/kotsiantis/improvingNB.pdf)

Von Altrock, Consantin(1995).Fuzzy logic and NeuraFuzzy applications explained. Upper Saddle River, NJ: Prentice Hall PTR.

CURRICULUM VITAE

Name Surname: Mustafa KARA

Address : Selamiali Mah. Sözebesi Sok. Dilek Apt No:20/9 Üsküdar/İstanbul

Birth Place / Year : Mersin- 1984

Languages : Turkish (native) - English

High School : Erdemli Anatolian High School

BSc: Bahçeşehir University - 2008

MSc: Bahçeşehir University- 2011

Name of Institute: Institute of Science

Name of Program: Computer Engineering

Work Experience:

Software Engineer, Albaraka Turk Bank İstanbul December 2008 - Present