CREDIT APPROVAL ANALYSIS AND RISK PREDICTION BASED ON MACHINE LEARNING (SUMMARY)

Nowadays, banks play an important role in Turkey's informatics sector. They must provide a good service for their customers. Also they want to get maximum efficiency from their projects and works. Therefore, they must keep pace with the speed of technology. In this perspective, machine learning is highly important for the financial area. Since it is useful to determine what factors most greatly affect people or businesses. So banks consistently and quickly realize their project with the aid of machine learning.

Machine learning is a subfield of artificial intelligence which has great significance in computer sciences. Machine learning is a name of computer algorithms which deduces existing data using mathematical and statistical methods and predicts about unknown things with these deduction.

One of the most important system of the banks is credit unit. In this study, credit approval interface is developed based on machine learning techniques. The aim of this interface and this project is making suitable algorithm analysis and providing that banks can be able to respond their customer about credit approval in fastly. In this way, suitability of customer's credit appeals are analysed in terms of risk prediction.

In this system, some machine learning algorithms are used and compared each other with the aid of suitable test and training dataset. In this way, algorithms are choose in order to ensure high accuracy in prediction. Machine learning tool which is called WEKA is used for this graduation project. There are also datasets such that training and test using cross-validation method so that the computer teaches these datasets and makes a decision about the related problems. Dataset is analysed with the statistical results using R script language. Moreover, Yapı Kredi Bank decision mechanism for the evaluating credit approvals are explained. Project are compared existing related systems of Yapı Kredi Bank and Consumer Reporting Agency (Kredi Kayıt Bürosu). Comparison results are explained in detail. In the light of these information, it is discoursed that how the more reliable and correctly system is improved.

Finally, according to the comparison results, a web service is implemented for the bank to decide credit approvals in terms of customer features. In order to realize web service, Azure Machine Learning Tool and MVC technologies are used. In this way, a machine learning model is designed in Azure Machine Learning Tool. This design are modelled in terms of WEKA and R script language results. When the customer's information are entered in form application, system requests to machine learning and then response a message from machine learning tool whether customer's credit approval are risky or not.

MAKİNE ÖĞRENMESİNE DAYALI KREDİ ONAY ANALİZİ VE RİSK TAHMİNİ (ÖZET)

Günümüzde bankalar Türkiye'deki bilişim sektörü için önemli bir yere sahiptir. Bankalar müşterilerine iyi hizmet sunmalıdırlar. Ayrıca bankalar yaptıkları projeler ve işlerden maksimum düzeyde verim almak isterler. Bu yüzden teknolojinin hızına ayak uydurmak zorundadırlar. Bu perspektifte makine öğrenmesi finansal dünyada kendine yer bulan ve her geçen gün gelişen bir teknoloji olarak karşımıza çıkmaktadır. Zira insanları ve iş dünyasını etkileyen faktörleri belirlemede, makine öğrenmesi tekniklerinin geliştirilmesi önemli bir avantajdır. Bu yüzden bankalar makine öğrenmesi yardımıyla projelerini tutarlı ve hızlı bir şekilde gerçeklerler.

Makine öğrenmesi bilgisayar bilimleri içinde önemli bir yer tutan Yapay Zeka'nın bir alt dalıdır. Makine öğrenmesi matematiksel ve istatistiksel yöntemler kullanılarak mevcut verilerden çıkarımlar yapan ve bu çıkarımlarla bilinmeyene dair tahminlerde bulunan bilgisayar algoritmalarının genel adıdır.

Bankaların önemli sistemlerinin başında kredi birimleri gelir. Bu çalışmada makine öğrenmesi tekniğine dayalı kredi başvuru arayüzü geliştirilmiştir. Bu uygulamadaki ve projenin genelindeki amaç uygun algoritma analizleri yapıp bankaların kredi başvurularında müşterilere hızlı ve tutarlı bir şekilde cevap vermelerini sağlamaktır. Bu yolla müşterilerin kredi başvuru uygunluğu risk tahmini açısından analiz edilmiştir.

Bu sistemde bazı makine öğrenmesi algoritmaları uygun test ve eğitim verileri kullanılarak analiz edilip, birbirleriyle karşılaştırılmıştır. Algoritmalar en yüksek performansı ve doğruluğu sağlayacak şekilde seçilmiştir. Algoritmaların analizi için makine öğrenmesi aracı olan WEKA kullanılmıştır. Eğitim ve test verileri cross-validation tekniğine göre oluşturulmuştur. Böylelikle bilgisayarın ilgili problemle alakalı veri setini öğrenmesi ve bir karar verme mekanizması oluşturması sağlanmıştır. İlgili analizde bazı istatiksel verileri çıkarmak adına WEKA' ya yardımcı olarak R dili de kullanılmıştır.

Buna ek olarak yapılan proje dahilinde Yapı Kredi bankasının kredi başvurularında kullandığı karar mekanizması değerlendirilmiş, yapılan proje ile halihazırda Yapı Kredi Bankası ile Kredi Kayıt Bürosunun mevcut sistemleri karşılaştırılmıştır. Karşılaştırma sonuçları detaylıca anlatılıp bu bilgiler ışığında nasıl daha güvenilir ve tutarlı sistemler geliştirilebilir konusu üzerinde durulmuştur.

Son olarak karşılaştırma sonuçlarına göre yukarıda da belirtildiği üzere müşteri bilgilerine göre kredi başvuru uygunluğunu kontrol etmek için bir web servis yazılmıştır. Web servis için Azure Machine Learning aracı ile beraber bir MVC projesi geliştirilmiştir. Makine öğrenmesi modeli Azure Machine Learning aracı içinde dizayn edilmiştir. Modelleme WEKA'dan ve R dilinde yapılan analizler sonrasında en uygun algoritma seçilerek yaratılmış ve local'deki form ekranından istek gönderip ilgili müşteriye kredi verilip verilmemesine yönelik bir yanıt alınmıştır.

TABLE OF CONTENTS

1. INTRODUCTION

In our countries, banks' personal credit approval units are very critical in terms banks' profits and loss. Personal credit appeal is frequently encountered a situation in Turkey. At the present time, many people apply for getting credit according to their necessities. These appeals are done via SMS, Internet, bank's offices or ATMs. Because of the surplus of the number of appeals, banks must provide correctly and rapidly evaluation and responds to their customers. Within the process, the most critical point for the banks is to give credit to the right person and right amount in quickly. In other words, if the banks give credit a customer who cannot discharge, likely they cannot withdraw, and this situation will be backfire. According to the Turkey's Banks Union Risk Center, there are about 1 million people who cannot pay credit dept. [1]

The motive behind of this project is to ease evaluation of banks' personal credit approval units. The primary aim of this project is to improve fertile system about banks' credit approval decision mechanism. There are two ways in order to acceleration credit approval process. Banks employs more staff to respond more quickly to credit application or a computer makes this process to conclude a lot more quickly. Here is in the second way machine learning.

Machine learning is one of the most well improved branch of the computer science. The goal is to teach research topic to computer with the aid of specially designed algorithms. By this means, machine learning provides teaching through the existing the data to the computers in order to decide itself. ML is generally named a system which fulfils a duty about Artificial Intelligence methods. [2]

In this study, an analysis of credit approvals and risk prediction of banks using some machine learning methods. In this way, a dataset which is about bank customers' features are taught to the computer and the computer analyse this data. Then it shows the risk prediction and determine whether this customer is suitable or not. Dataset which is called german-credit is provided as a public. It is analysed with the algorithms of Decision Tree, Support Vector Machine, Artificial Neural Network, Random Forest and Logistic Regression. Results at WEKA are explained in details as some statistical techniques with the aid of R script language. Then results are evaluated with Yapı Kredi Bank credit approval system. Previous literature researches are generally about determine the risk prediction according to the some statistical results. [3][4] In this Project, comparison of the new system and existing system of machine learning in Yapı Kredi Bank is useful so as to determine a risk criteria and accuracy algorithms.

Finally, a web service is created with the aid of Azure Machine Learning and about risk of customer's credit approval according to this risk criteria.

This study consists of six main parts:

- **Project Description and Plan:** This section includes general information about plan of the project and risk management.. In this part, work steps that is situated in plan is described and sequence and time of this work steps is given.

- **Theoretical Information:** In this part, platforms which used in project are explained in details.
- **Analysis and Modelling:** In this section, algorithms are analysed and web service model which is created in Azure Machine Learning is explained.

- **Design, Implementation:** In this stage, interface which is developed in Visual Studio as MVC project is explained
- **Experimental Result:** In this section, project is compared Yapı Kredi Bank credit approval system.
- **Conclusion and Future Work:** In this stage, project is explained according to analysis. Also in future, it is mentioned about which studies may be improved about related project.

2. PROJECT DESCRIPTION AND PLAN

2.1 Project Description

Credit approval analysis and risk prediction project is about machine learning which contains analysis of algorithms and a web service as an interface. As it is mentioned above, the fundamental aim of the project is to speed up to banks' credit approval process and increase approval's accuracy in order to gain maximum fertile of their system. In this way, the most performance and correct algorithm and system are designed according to the dataset's attributes which are about customers' information. This design is modelled for using the banks' as a tool.

2.2 Project Scope

This project was design by one person and it was done to enlighten the deficiency of the banks' credit approval system. The major aim of the project is to develop more fertile system and to provide quickly respond to the customer's need. In this Project, Dataset which is called german-credit is obtained as a public. Firstly, dataset is analysed with the algorithms of Decision Tree, Artificial Neural Network, Support Vector Machine, Random Forest and Logistic Regression. Also some statistical information are created in R script language. Then Project is compared Yapı Kredi Bank credit approval system. According to the comparison, a model is created in Azure for a web service and MVC technology is used.

2.3 Project Schedule

Project planning and development will be done according to the gantt chart. (Figure 2.1). Project is planned to be concluded successfully until January-2016. Tasks which will be done till this time.

Görev Modu	Görev Adı	Süre	Başlangıç	Bitiş
	Project	**201 gün**	**Pzt 13.4.15**	**Pzt 18.1.16**
	Initialization Stage	**53 gün**	**Pzt 13.4.15**	**Çar 24.6.15**
	Determining the subject line of the project	23 gün	Pzt 13.4.15	Çar 13.5.15
	Researching methods which will be used	23 gün	Çar 13.5.15	Cum 12.6.15
	Requirement Analysis	9 gün	Cum 12.6.15	Çar 24.6.15
	Development Stage	**118 gün**	**Çar 24.6.15**	**Cum 4.12.15**
	Anaysis of related algorithms	45 gün	Çar 24.6.15	Sal 25.8.15
	Design Project	40 gün	Sal 25.8.15	Pzt 19.10.15
	Software Development	35 gün	Pzt 19.10.15	Cum 4.12.15
	Conclusion Stage	**32 gün**	**Cum 4.12.15**	**Pzt 18.1.16**
	Comparison Results	22 gün	Cum 4.12.15	Pzt 4.1.16
	Checking future marketing of the project	6 gün	Pzt 4.1.16	Cmt 9.1.16
	Preparation od Documents	7 gün	Cmt 9.1.16	Pzt 18.1.16

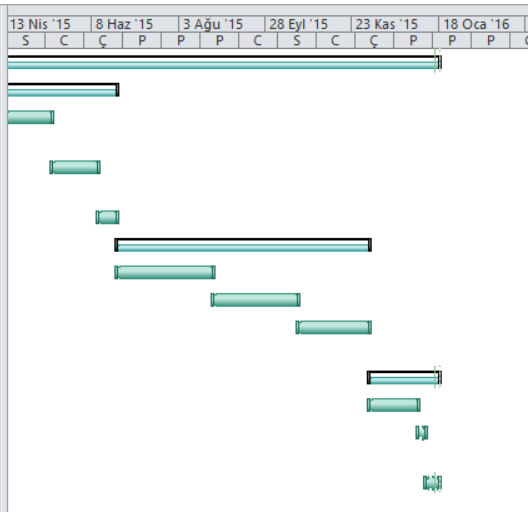


Figure 2.1: Gantt Chart

2.4 Risk Management

2.4.1 Time Managing

It is an important problem of time scheduling of huge project. Thus, project may not finish before deadline. In order to keep safe from this situation, project is organized in terms of time. It is listed as the following:

Mitigation: Timeline of the project should be controlled regularly to prevent delay.

Monitoring: Another preventing issue which is about checking the project schedule so as to predict possible latencies.

Management: Time scheduling should be reorganized against the all possible scheduling violations.

2.4.2 Lack of Experience

This project is important for making good prediction in terms of the scoring of customers' credit approval. So, it is required that having a good experience to complete project for analysis of algorithms.

Mitigation: As it is mentioned above, researching about machine learning tools such that WEKA, Azure Machine Learning plays an important role in order to decrease lack of experience.

Monitoring: In addition to research, every stages of projects must be examine in details.

Management: One of the most important method for solving the lack of experience issue is to understand related examples and tutorials on the web very carefully.

Another possibilities about risk issues are listed below:

- Changes that may occur later in the Project.
- Programming language which is used for web service may be changed with respect to performance or any other effect.
- Procure of dataset for machine learning may be a problem.
- Risk criteria of bank may be change.
- Machine learning algorithms that I used may be change according to this risk criteria.

3. THEORETICAL INFORMATION

3.1 Machine Learning

Machine learning programs the computer so as to make suitable a success criterion. In other saying, implementation of machine learning methods is called data mining.[7] Here is the important thing is "learning" and "model". Machine learning uses training dataset or past experiences, and provides to optimize this model's parameters. This is about "learning". And as for "model", it is an estimator to make a prediction about in the future or it is a depictor to obtain information from the data. From this perspective, it can be mentioned about supervised and unsupervised learning. In supervised learning; it is known that the number of class and distribution of the object. In contrast, in unsupervised learning; it is not known the number of class and which object is in which class. [7]

In machine learning, analysis of different area accompanies the different expectations. So, it is classified into four parts: [6]

Classification: In this method, there are predefined classes and they will be known which class is a new object belongs to. Supervised learning is in the category of classification.

Clustering: It tries to group a set of objects and find whether there is some relationship between the objects. Unsupervised learning is in the clustering.

Regression: The output variable takes continuous values.

Associative Rule: It is another method which discovers interesting relations between in large databases.

3.2 Development Tools

This project is developed with the several platforms with respect to analysis of algorithms, statistical information and web service. The part of analysis algorithms are made in WEKA. Also some statistical information such that T-test, heatmap, variance, standard deviation and feature importance technique for random forest are made in R script language. Furthermore, Azure Machine Learning and MVC technology are used for web service. In Azure, a machine learning model is created according to the displaying the best performance of algorithm in WEKA. Then MVC project is created in Visual Studio 2013. MVC project is typically created as a Model, View and Controller. A form application is prepared in the part of View which makes a request. Then in the part of Model, an object is created in order to receive user input to controller. Then controller receives user input and makes call to model objects and the view to perform appropriate actions.

3.2.1 Weka

Waikato Environment for knowledge Analysis also known as WEKA is a tool which consists of several machine learning algorithms for data mining tasks. It has been improved at Waikato University in New Zealand. The word of WEKA means that Weka is a flightless bird with an inquisitive bird which found only in New Zealand. WEKA is an open source which is developed with Java programming language. Also WEKA uses "arff" file format. WEKA's interface is shown as the following: [2][6]

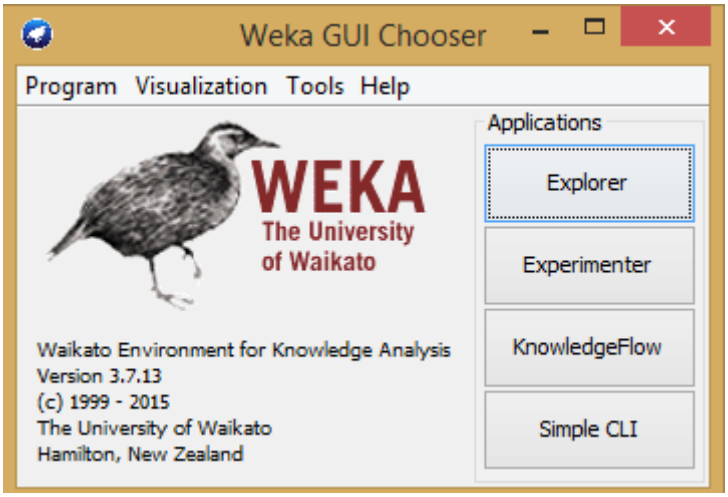


Figure 3.1 – WEKA Interface

Explorer: It is an environment which is used for exploring data with WEKA. For this project this menu is used for analysis algorithms. In Explorer menu are shown and listed as below:

- Data pre-processing
- Classification
- Regression
- Clustering
- Association Rule

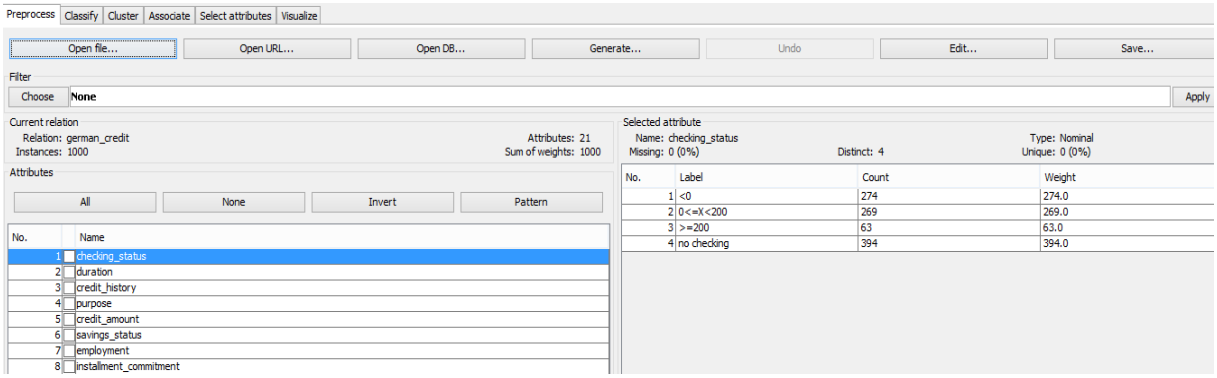


Figure 3.2: Explorer Interface

Experimenter: It is used for working on a more than one datasets. So it is necessary for adjustment kinds of parameters and kinds of datasets:

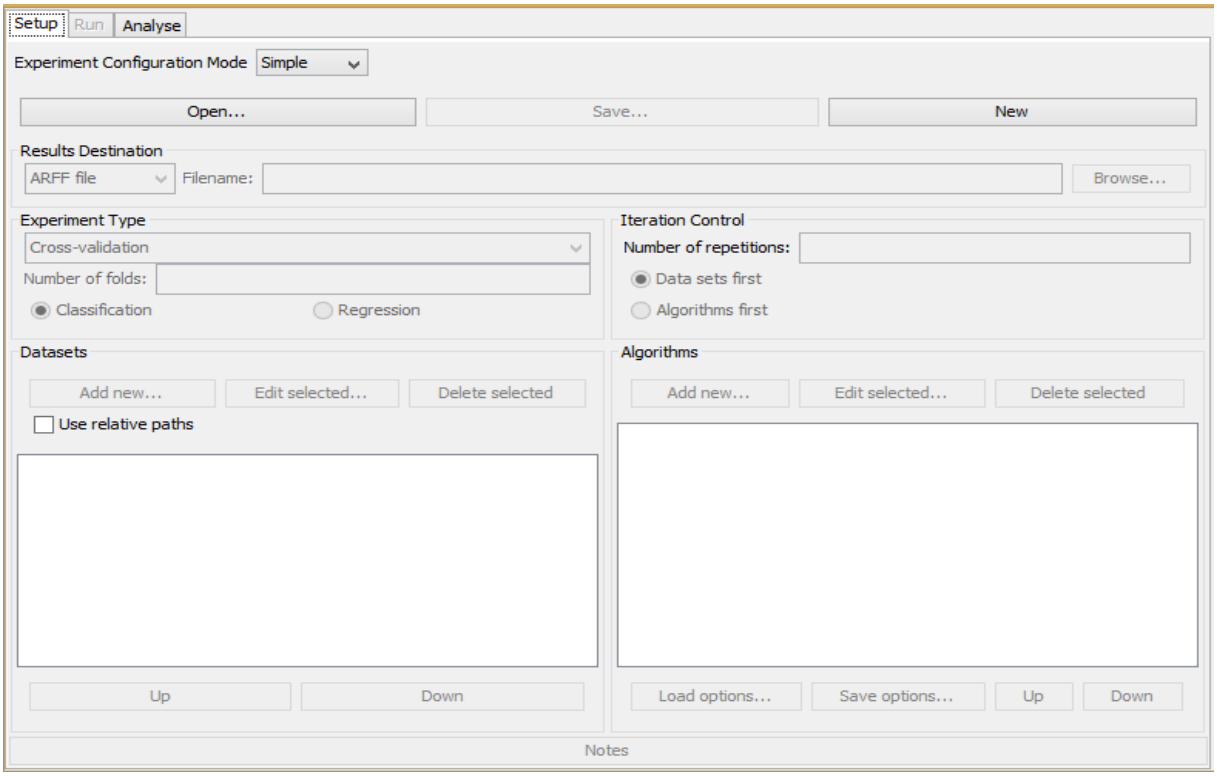


Figure 3.3: Experimenter

KnowledgeFlow: It is an alternate interface to explorer which is used for graphical operations.

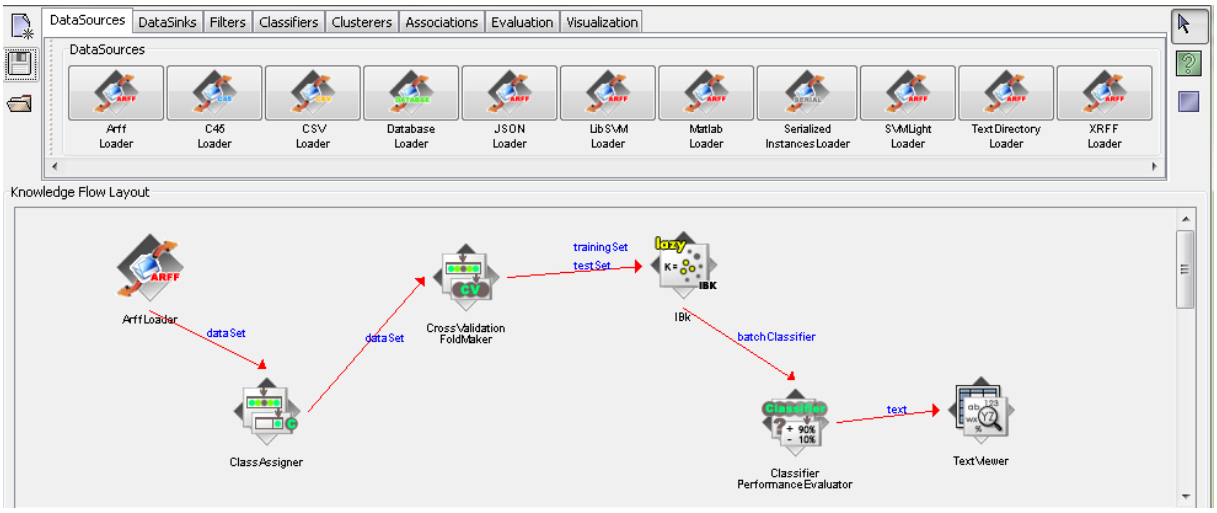


Figure 3.4: Knowledge Flow

Simple CLI: This part is an advantage for experiment form command line. Loading dataset, choosing algorithms can be done with command files.

In WEKA, dataset is like arff file format. ARFF file format is devided into three parts. These are relation, attribute and data. In this project, german-credit dataset which is called credit-g.arff is used. The struct of arff file is shown as the following:

```
@relation german_credit
@attribute checking_status { '<0', '0<=X<200', '>=200', 'no checking'}
@attribute duration numeric
@attribute credit_history { 'no credits/all paid', 'all paid', 'existing paid', 'delayed previously
@attribute purpose { 'new car', 'used car', furniture/equipment, radio/tv, 'domestic appliance', re
@attribute credit_amount numeric
@attribute savings_status { '<100', '100<=X<500', '500<=X<1000', '>=1000', 'no known savings'}
@attribute employment { unemployed, '<1', '1<=X<4', '4<=X<7', '>=7'}
@attribute installment_commitment numeric
@attribute personal_status { 'male div/sep', 'female div/dep/mar', 'male single', 'male mar/wid', '
@attribute other_parties { none, 'co applicant', guarantor}
@attribute residence_since numeric
@attribute property_magnitude { 'real estate', 'life insurance', car, 'no known property'}
@attribute age numeric
@attribute other_payment_plans { bank, stores, none}
@attribute housing { rent, own, 'for free'}
@attribute existing_credits numeric
@attribute job { 'unemp/unskilled non res', 'unskilled resident', skilled, 'high qualif/self emp/mg
@attribute num_dependents numeric
@attribute own_telephone { none, yes}
@attribute foreign_worker { yes, no}
@attribute class { good, bad}
@data
'<0',6,'critical/other existing credit',radio/tv,1169,'no known savings','>=7',4,'male single',none
'0<=X<200',48,'existing paid',radio/tv,5951,'<100','1<=X<4',2,'female div/dep/mar',none,2,'real est
'no checking',12,'critical/other existing credit',education,2096,'<100','4<=X<7',2,'male single',no
```

Figure 3.5: Dataset arff file format

Algorithms analysis are done explorer menu. After the loading dataset, algorithm choose in classify. Sample analyse and output are indicated as below:

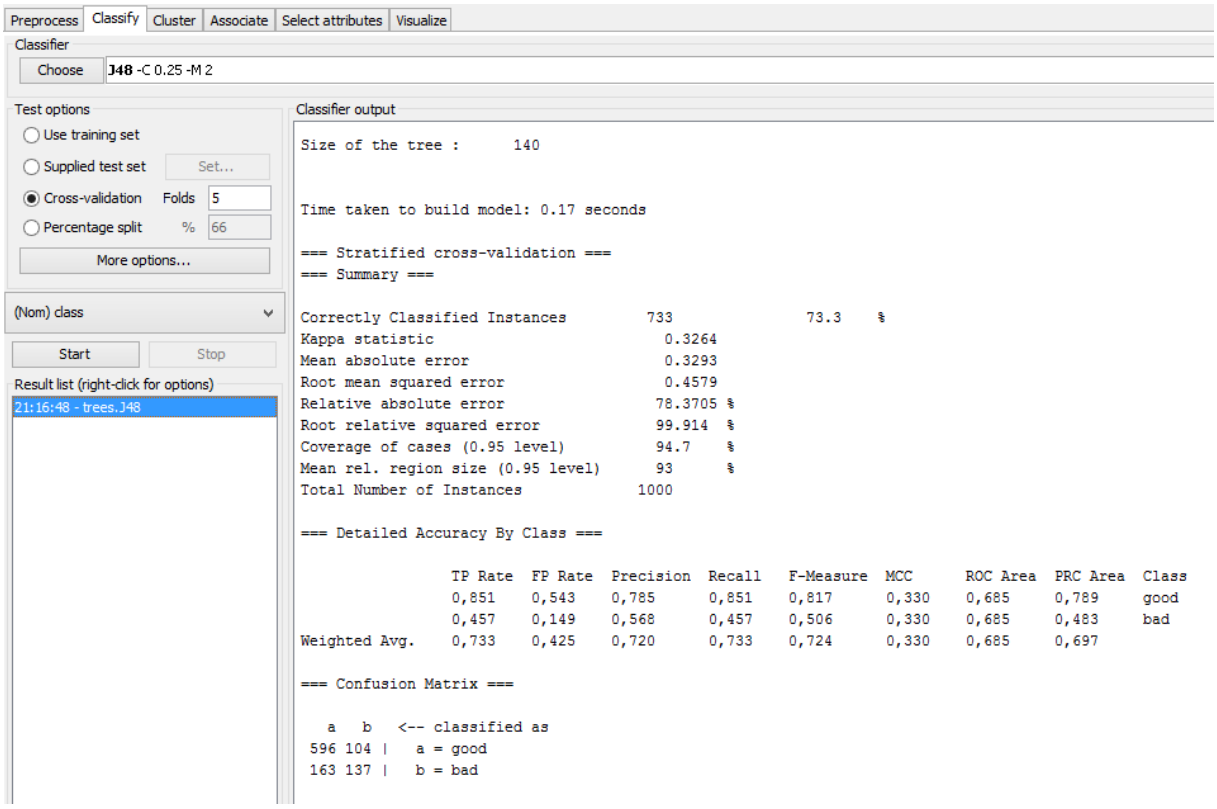


Figure 3.6: Sample Output

As it is seen above, output shows some explanation about dataset and related algorithms. In stratified cross-validation part gives estimates of algorithms' predictive performance. This part outputs the lists some statistical calculations. Firstly, correctly classified instances shows accuracy that how many data is classified in good categories. Kappa statistics measures the agreement of prediction with the true class. The other ones are abaut probability of predicted and actual value. In Detailed Accuracy By Class part indicate detail of classifier's accuracy. Finally confusion matric shows the number of true/false – positive/negative rate.

3.2.2 R Language

R is a script language which provides statistical computing and graphics. Within the scope of the project, some statistical data is obtained with the aid of R. T-test is applied on the dataset in order to find difference of samples. Also, a heatmap is indicated in terms of dataset's correlation between class and attributes. Moreover, feature importance technique for random forest is implemented in R. Finally, for a decision mechanism in terms of evaluation of credit approval criteria, a script is written in order to determine banks' credit scoring. [10]

3.2.3 Azure Machine Learning

It is another machine learning tool which is used for web service. According to the WEKA results, a model is created in this tool. Score model and evaluate model determines the predicted value in the future. Score model creates a scoring probabilities in every field in dataset with respect to the threshold values. Dataset is loaded in Azure Machine Learning Tool with some codes. These code equals to real attribute names. Example of Attribute information are listed as below: [8] [9]

Attribute 1: (qualitative)
Status of existing checking account
A11 : ... < 0 DM A12 : 0 <= ... < 200 DM A13 : ... >= 200 DM / salary assignments for at least 1 year A14 : no checking account

Attribute 2: (numerical) Duration in month

Attribute 3: (qualitative)
Credit history
A30 : no credits taken/ all credits paid back duly A31 : all credits at this bank paid back duly
A32 : existing credits paid back duly till now A33 : delay in paying off in the past A34 : critical account/ other credits existing (not at this bank)

Attribute 4: (qualitative)
Purpose
A40 : car (new) A41 : car (used) A42 : furniture/equipment A43 : radio/television A44 : domestic appliances A45 : repairs A46 : education A47 : (vacation - does not exist?) A48 : retraining
A49 : business A410 : others

Attribute 5: (numerical) Credit amount

Attibute 6: (qualitative)
Savings account/bonds
A61 : ... < 100 DM A62 : 100 <= ... < 500 DM A63 : 500 <= ... < 1000 DM
A64 : .. >= 1000 DM A65 : unknown/ no savings account

Attribute 7: (qualitative)
Present employment since
A71 : unemployed A72 : ... < 1 year A73 : 1 <= ... < 4 years
A74 : 4 <= ... < 7 years A75 : .. >= 7 years

Attribute 8: (numerical)
Installment rate in percentage of disposable income

Attribute 9: (qualitative)
Personal status and sex
A91 : male : divorced/separated A92 : female : divorced/separated/married
A93 : male : single A94 : male : married/widowed
A95 : female : single

Attribute 10: (qualitative)
Other debtors / guarantors
A101 : none A102 : co-applicant A103 : guarantor

4. ANALYSIS AND MODELLING

The part of project's analysis is done in WEKA. As it is mentioned above, because of the aim of project, analysis is done according to the "predicted the future". Algorithms scheme that are used in WEKA --also in data mining is indicated in Figure 4.1.

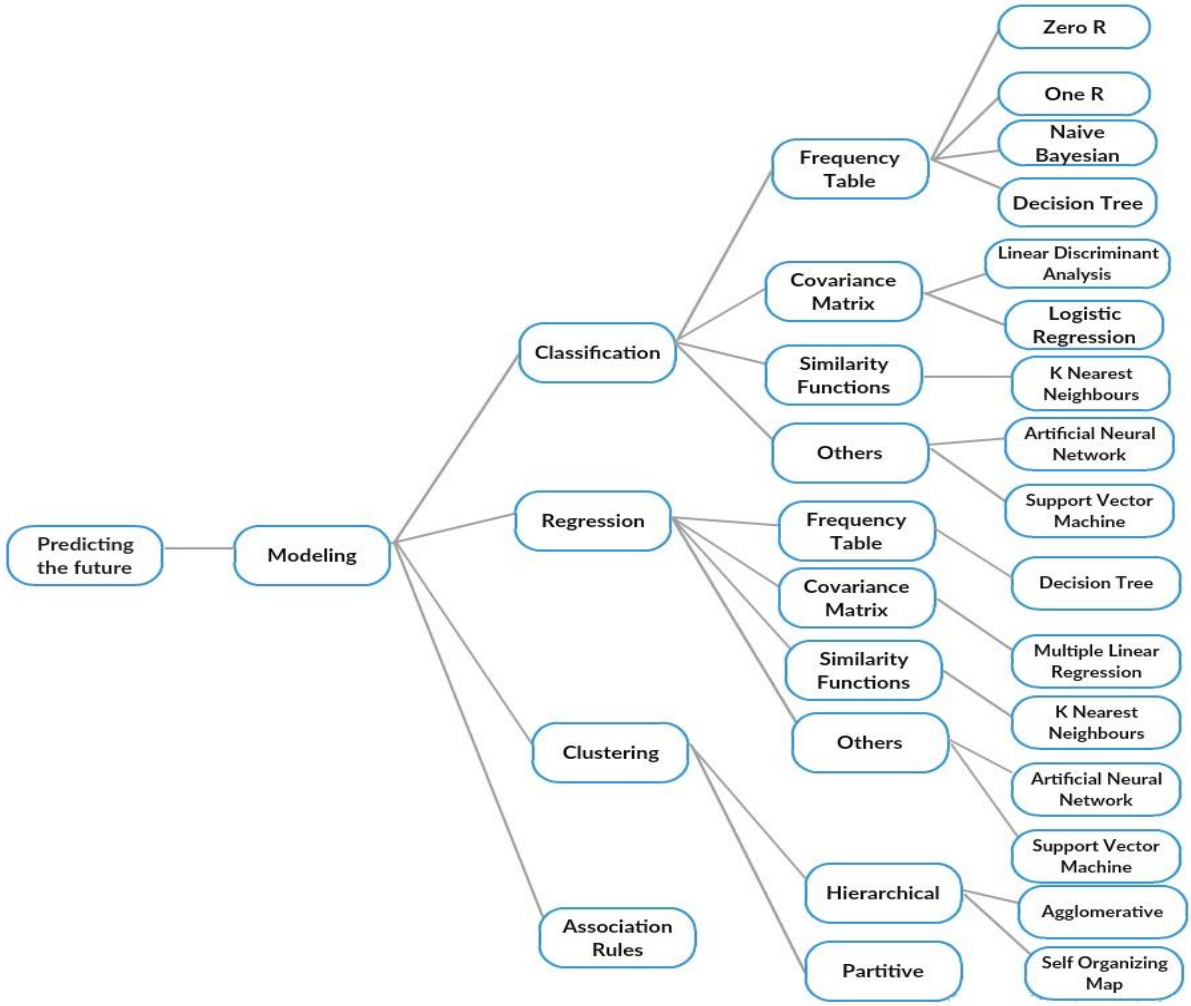


Figure 4.1: Algorithm Map in WEKA

Since the project is analysed on an existing dataset and the main issue is the "predicted the future", analysis is made according to the classification and regression model.

4.1 Analysis

In this section, project is analysed with the algorithms of Decision Tree, Artificial Neural Network, Support Vector Machine, Random Forest and Logistic Regression. Before analyse with these algorithms, explanation of dataset and related attributes are indicated.

4.1.1 Dataset- (german credit)

As it is mentioned above, dataset is procured as a public in WEKA data library. It is about the union of personal, social and financial information past bank customers. Attributes and their types are listed as the following: (Table 4.1)

Table 4.1: List of Attributes

Number	Attribute	Type
1	Checking Status	Nominal
2	Duration	Numeric
3	Credit History	Nominal
4	Purpose	Nominal
5	Credit Amount	Numeric
6	Savings	Nominal
7	Employment Duration	Nominal
8	Installment Rate	Numeric
9	Personal Status	Nominal
10	Debtors	Nominal
11	Residence	Numeric
12	Property	Nominal
13	Age	Numeric
14	Installment plans	Nominal
15	Housing	Nominal
16	Existing Credits	Numeric
17	Job	Nominal
18	Liable People	Numeric
19	Telephone	Nominal
20	Foreign Worker	Nominal
21	**CLASS**	Nominal

As seen above, there are 20 attributes and one class attribute. Class attribute contains two value which its type is nominal as a binary—**good** or **bad.** According to the data, class shows that suitable of customers' credit approval is whether risky or not. Dataset consists of 1000 instances. Overview of the dataset in WEKA is indicated as below: (Figure 4.2 and 4.3)

No.	1: checking_status Nominal	2: duration Numeric	3: credit_history Nominal	4: purpose Nominal	5: credit_amount Numeric	6: savings_status Nominal	7: employment Nominal	8: installment_commitment Numeric	9: personal_status Nominal	10: other_parties Nominal
1	(0	6.0	critical/other ex...	radio/tv	1169.0	no known savings	)=7	4.0	male single	none
2	0(=X(200	48.0	existing paid	radio/tv	5951.0	(100	1(=X(4	2.0	female div/dep/mar	none
3	no checking	12.0	critical/other ex...	education	2096.0	(100	4(=X(7	2.0	male single	none
4	(0	42.0	existing paid	furniture/...	7882.0	(100	4(=X(7	2.0	male single	guarantor
5	(0	24.0	delayed previo...	new car	4870.0	(100	1(=X(4	3.0	male single	none
6	no checking	36.0	existing paid	education	9055.0	no known savings	1(=X(4	2.0	male single	none
7	no checking	24.0	existing paid	furniture/...	2835.0	500(=X(1000	)=7	3.0	male single	none
8	0(=X(200	36.0	existing paid	used car	6948.0	(100	1(=X(4	2.0	male single	none
9	no checking	12.0	existing paid	radio/tv	3059.0	)=1000	4(=X(7	2.0	male div/sep	none
10	0(=X(200	30.0	critical/other ex...	new car	5234.0	(100	unemployed	4.0	male mar/wid	none

Figure 4.2: Overview of Dataset Part 1

11: residence_since Numeric	12: property_magnitude Nominal	13: age Numeric	14: other_payment_plans Nominal	15: housing Nominal	16: existing_credits Numeric	17: job Nominal	18: num_dependents Numeric	19: own_telephone Nominal	20: foreign_worker Nominal	21: **class** Nominal
4.0	real estate	67.0	none	own	2.0	skilled	1.0	yes	yes	good
2.0	real estate	22.0	none	own	1.0	skilled	1.0	none	yes	bad
3.0	real estate	49.0	none	own	1.0	unskill...	2.0	none	yes	good
4.0	life insurance	45.0	none	for free	1.0	skilled	2.0	none	yes	good
4.0	no known property	53.0	none	for free	2.0	skilled	2.0	none	yes	bad
4.0	no known property	35.0	none	for free	1.0	unskill...	2.0	yes	yes	good
4.0	life insurance	53.0	none	own	1.0	skilled	1.0	none	yes	good
2.0	car	35.0	none	rent	1.0	high q...	1.0	yes	yes	good
4.0	real estate	61.0	none	own	1.0	unskill...	1.0	none	yes	good

Figure 4.3: Overview of Dataset Part 2

In the light of these information, algorithms are performed on the dataset. Results and comparison are explained in detail. Before analyse, to do lists are itemized as the following:

- Firstly, normal evaluation is done according to the dataset attributes in algorithms of Decision Tree, Artificial Neural Network, Support Vector Machine and Random Forest.
- Then this time, reduction of attribute in WEKA on the feature selection menu is performed on dataset and Algorithms are analysed again with the feature selection. Feature selection is done using wrapper method.
- **Five fold cross validation** technique is used and in order to decrease chance factor, it is tried 10 times. Results are shown with respect to accuracy. This means that correctly classified instances represents good class for credit approval.
- T- test and Homogeneity of Variances (Fisher's F-test) are implemented in R on the samples. The aim of the T-test is performed to obtain the normal evaluation and feature selections are whether significantly different or not. Also homogeneity of variances means that dependent and independent variables are distributed similar. Results are evaluated in terms of p-value. p-value is used for testing statistical hypothesis Assuming that the p-value is 0.05 as a threshold value.
- In addition to feature selection, for random forest algorithm, random forest feature importance is shown and tried on the random forest and SVM algorithm.
- Precision/Recall and ROC curve are drawn for every algorithms with respect to the confusion matrix.
- Heatmap is plotted in order to show correlation of the class and attributes.

Five Fold Cross Validation: Holdout method is used. In this method, dataset is divided into two sets called the training and testing. In this way dataset is separated into 5 parts (folds). Holdout method is repeated 5 times. Each part is used once for testing and 4 times training.

Wrapper and Search Method: Wrapper method is performed in WEKA in WrapperSubsetEvaluator. It works five fold cross validation and stops when standard deviation is less than the threshold values. Wrapper uses BestFirst search method as searching. BestFirst works like breath first search.

Algorithms and above written are explained in detail for every algorithm in separate titles:

4.1.2 Decision Tree

J48 algorithm is implemented. Decision Tree is like a greedy algorithm that attributes are divided into top to down. Divide and Conquer strategies are used. The algorithm starts with the whole dataset in a single node. If the sample of data is same attribute class, node becomes a leaf in the decision tree. Otherwise, algorithm search better divides data into individual classes. Five fold cross validation performed and random seed is worked ten times. Divide Operations and decision nodes are realized according to the Shannon Entropy: [2]

$p1x\log_2(p1)+p2x\log_2(p2)+\ldots\ldots$ where p is the probability of the related attribute

Results are shown as the following: (Table 4.2)

Table 4.2 : Accuracy with normal evaluation for decision tree

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	733	73.3 %
2	728	72.8 %
3	711	71.1 %
4	702	70.2 %
5	730	73 %
6	706	70.6 %
7	721	72.1 %
8	707	70.7 %
9	736	73.6 %
10	716	71.6 %

Here is the average, standard deviation and variance of table:

Sample Mean: 71.9
Standard Deviation: 1.231981
Variance: 1.517778

4.1.2.1 Decision Tree with Feature Selection

After the Wrapper method, most effective attributes for decision tree are shown as the following:

- checking_status
- duration
- credit_history
- credit_amount
- savings_status
- other_parties
- age
- existing_credits
- foreign_worker

Then J48 algorithm is again implemented in terms of only this attributes. After this, we can see that correctly classified instances increases: (Table 4.3)

Table 4.3: Accuracy with feature selection for decision tree

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	747	74.7 %
2	751	75.1 %
3	734	73.4 %
4	719	71.9 %
5	727	72.7 %
6	735	73.5 %
7	743	74.3 %
8	735	73.5 %
9	746	74.6 %
10	747	74.7 %

Sample Mean: 73.84
Standard Deviation: 1.018932
Variance: 1.038222

Decision Tree Homogeneity Variance and T-test

Related R codes and comparison results are indicated as below:

dt_normal = c(73.3,72.8,71.1,70.2,73,70.6,72.1,70.7,73.6,71.6)
dt_feature = c(74.7,75.1,73.4,71.9,72.7,73.5,74.3,73.5,74.6,74.7)

var.test(a,b) // p-value =0.5807

t.test(dt_normal,dt_feature, var.equal=TRUE, paired=FALSE) // p-value = 0.001207

Result:

When Fisher's F-test is applied, we obtained p-value greater than 0.05, then we can assume that the two variances are homogeneous. Then T-Test is applied, we obtained p-value smaller than 0.05, then we can conclude that the averages of two groups are significantly diffe rent.

4.1.3 Artificial Neural Network (ANN)

In ANN, there are two algorithms which are called voted perceptron and multilayer perceptron. Multilayer perceptron are more success on the dataset than the other. MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. In MLP, each neuron connects the other neuron with the synapsis and each neuron has a weight. Transmission can be change neurons' weight. Overview of the MLP is indicated as below:[2]

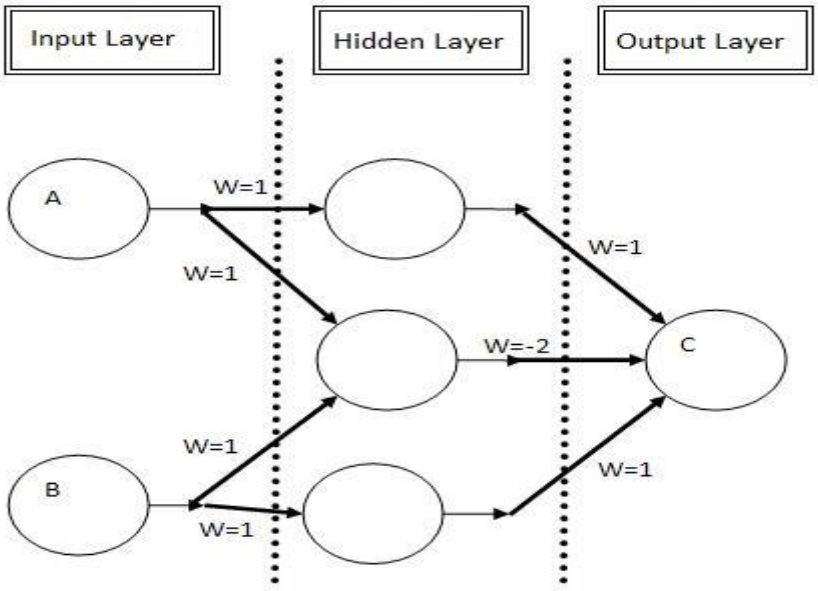


Figure 4.4: Multilayer Perceptron [11]

In MLP neurons have activation function which is called Sigmoid. Activation function carries a variable to different dimension. Here is the sigmoid function: $s(a) = 1 / 1 + e^{-a}$

In this network, a variable number of hidden layers 3-5-10 can be used with a different number of neurons:

Hidden Layers: 3

Table 4.4: Accuracy with ANN for 3 hidden layer

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	728	72.8 %
2	718	71.8 %
3	698	69.8 %
4	733	73.3 %
5	725	72.5 %
6	736	73.6 %
7	717	71.7 %
8	730	73 %
9	743	74.3 %
10	713	71.3 %

Sample Mean: 72.41
Standard Deviation: 1.299957
Variance: 1.689889
Here is the figure of artificial neural network with three hidden layer: (Figure 4.4)

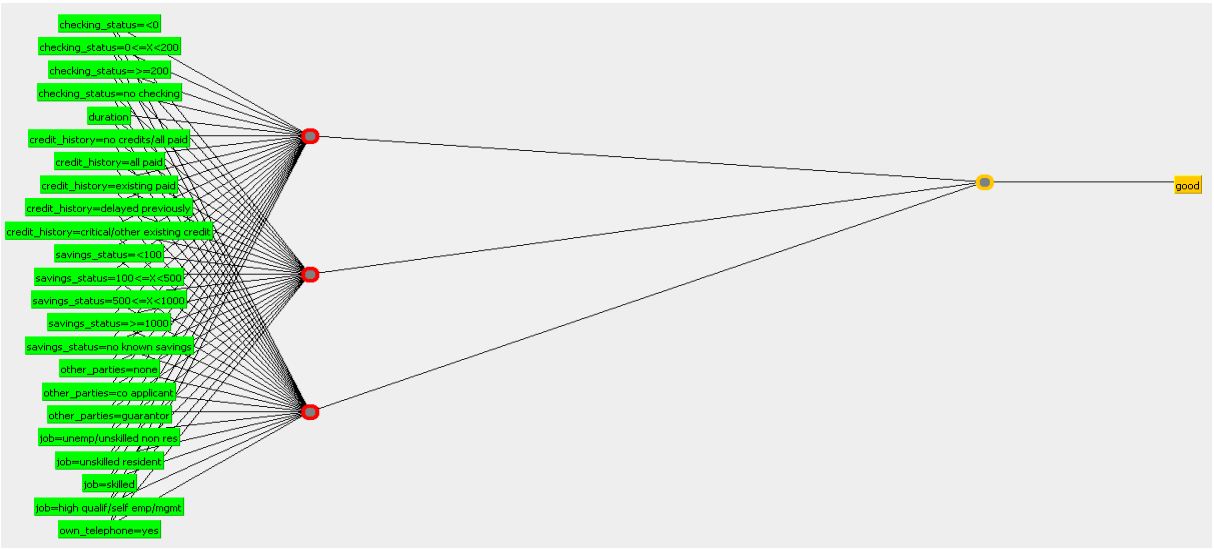


Figure 4.5: Artificial Neural Network-3 Hidden Layer

Hidden Layers: 5

Table 4.5: Accuracy with ANN for 5 hidden layer

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	736	73.6 %
2	729	72.9 %
3	718	71.8 %
4	722	72.2 %
5	725	72.5 %
6	703	70.3 %
7	718	71.8 %
8	716	71.6 %
9	721	72.1 %
10	723	72.3 %

Sample Mean: 72.11
Standard Deviation: 0.8672434
Variance: 0.7521111

Hidden Layers: 10

Table 4.6: Accuracy with ANN for 10 hidden layer

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	721	72.1 %
2	707	70.7 %
3	705	70.5 %
4	718	71.8 %
5	712	71.2 %
6	698	69.8 %
7	698	69.8 %
8	732	73.2 %
9	704	70.4 %
10	714	71.4 %

Sample Mean: 71.09
Standard Deviation: 1.074399
Variance: 1.154333

As seen above, three hidden layers displays better performance the others. Feature selection is done with respect to 3 hidden layers.

4.1.3.1 ANN with Feature Selection

After the Wrapper method, most effective attributes for ANN are shown as the following:

- checking_status
- duration
- credit_history
- credit_amount
- savings_status
- other_parties
- foreign_worker

Table 4.7: Accuracy with ANN for feature selection

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances
1	740 74 %
2	744 74.4 %
3	763 76.3 %
4	743 74.3 %
5	748 74.8 %
6	726 72.6 %
7	740 74 %
8	757 75.7 %
9	742 74.2 %
10	735 73.5 %

Sample Mean: 74.38
Standard Deviation: 1.047537
Variance: 1.097333

ANN Homogeneity Variance and T-test

Related R codes and comparison results are indicated as below:

```
ann_normal=c(72.8,71.8,69.8,73.3,72.5,73.6,71.7,73,74.3,71.3)
ann_feature=c(74,74.4,76.3,74.3,74.8,72.6,74, 75.7,74.2,73.5)

var.test(a,b)       // p-value =0.5303

t.test(ann_normal,ann_feature, var.equal=TRUE, paired=FALSE) //   p-value = 0.00158
```

Result

When Fisher's F-test is applied, we obtained p-value greater than 0.05, then we can assume th at the two variances are homogeneous. Then T-Test is applied, We obtained p-value smaller t han 0.05, then we can conclude that the averages of two groups are significantly different.

4.1.4 Support Vector Machine (SVM)

In WEKA, SMO algorithm is used and rbf, polynomial kernel and normalized polynomial kernel (linear kernel) are tested on the algorithm. SVM algorithm uses kernel structure. Kernels are an advantage to easily computing. So data is transformed from any space easily. [5]. Details are given in every typed of kernel as below:

Linear Kernel:

$$k(x, y) = x^T y + c$$

The aim of the Linear Kernel is to separate to class each other with a special line also known hyperplane. This line is the furthest line two class. After the finding hyperplane using training dataset, determine which side of the testing data with respect to hyperplane. Then attribute is included this new side. Here is the linear kernel example.

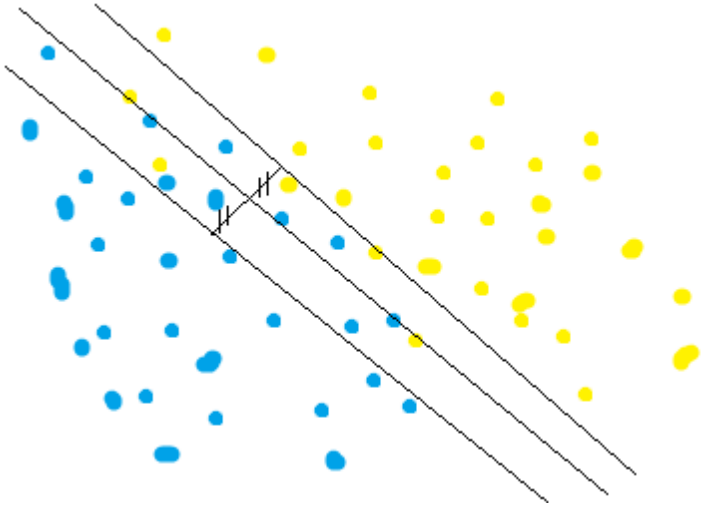


Figure 4.6: SVM Linear Kernel

Table 4.8: Accuracy with SVM-Linear Kernel

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	760	76 %
2	761	76.1 %
3	739	73.9 %
4	751	75.1 %
5	763	76.3 %
6	763	76.3 %
7	747	74.7 %
8	742	74.2 %
9	754	75.4 %
10	743	74.3 %

Sample Mean: 75.23
Standard Deviation: 0.9226171
Variance: 0.8512222

Normalized Poly Kernel:

The polynomial kernel is non-stationary kernel. It is suitable if training dataset is all normalized.

$k(x, y) = (\alpha x^T y + c)^d$ where the polynomial degree **d,** constant term **c** and the slope parameters **alpha**

Table 4.9: Accuracy with SVM-Normalized Poly Kernel

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	771	77.1 %
2	763	76.3 %
3	758	75.8 %
4	750	75 %
5	763	76.3 %
6	754	75.4 %
7	754	75.4 %
8	761	76.1 %
9	765	76.5 %
10	744	74.4 %

Sample Mean: 75.83
Standard Deviation: 0.7944949
Variance: 0.6312222

RBF Kernel:

It is also known Gaussian Kernel.

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right)$$

In RBF kernel, data is not linearly separable in original kernel. Also, low degree gives soft margin. Example of RBF is as below:

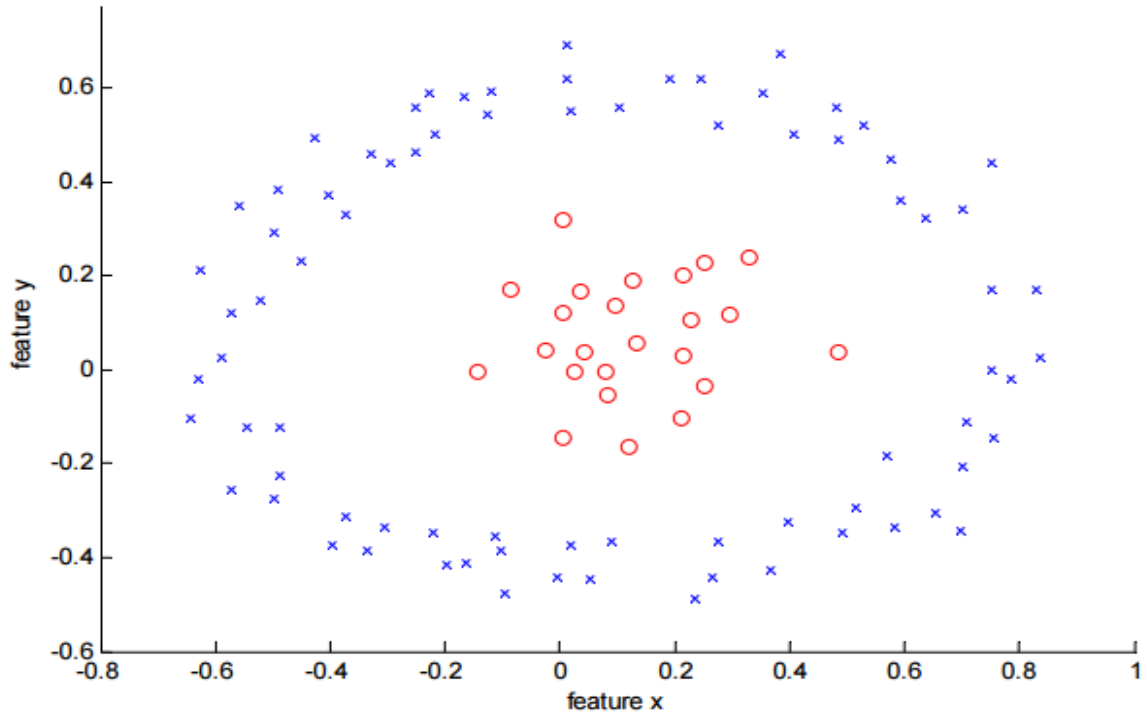


Figure 4.7: SVM RBF Kernel[12]

Table 4.10: Accuracy with SVM-RBF Kernel

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	700	70 %
2	700	70 %
3	702	70.2 %
4	701	70.1 %
5	700	70 %
6	700	70 %%
7	700	70 %
8	700	70 %
9	700	70 %
10	700	70 %

Sample Mean: 70.03
Standard Deviation: 0.06749486
Variance: 0.004555556

As seen above, poly-kernel displays better performance others. Since feature selection is done according to the normalized poly kernel

4.1.4.1 Svm with Feature Selection

After the Wrapper method, most effective attributes for SVM are shown as the following:

- checking_status
- credit_history
- credit_amount
- savings_status
- employment
- installment_commitment
- personal_status
- other_parties
- property_magnitude
- age
- other_payment_plans
- existing_credits
- job
- own_telephone
- foreign_worker

Table 4.11: Accuracy with SVM-Feature Selection

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	767	76.7 %
2	757	75.7 %
3	766	76.6 %
4	748	74.8 %
5	758	75.8 %
6	762	76.2 %
7	753	75.3 %
8	761	76.1 %
9	758	75.8 %
10	752	75.2 %

Sample Mean: 75.82
Standard Deviation: 0.6069962
Variance: 0.3684444

SVM Homogeneity Variance and T-test

Related R codes and comparison results are indicated as below:

```
svm_normal=c(77.1,76.3,75.8,75,  76.3,75.4,75.4,76.1,76.5,74.4)
svm_feature=c(76,75.6,74.6,75.1,75,74.8,76.2,76,76,74.3)

var.test(a,b)          // p-value = 0.661
t.test(svm_normal,svm_feature, var.equal=TRUE, paired=FALSE) //   p-value = 0.1732
```

Result:

When Fisher's F-test is applied, We obtained p-value greater than 0.05, then we can assume that the two variances are homogeneous. Then T-Test is applied, We obtained p-value is bigger than 0.05, then we can conclude that the averages of two groups are significantly similar.

4.1.5 Random Forest

Random Forest is same as decision tree. However, the difference is that random forest produces lots of trees. Algorithm is realized 100 trees.[2]

Table 4.12: Accuracy with Random Forest

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	759	75.9 %
2	763	76.3 %
3	766	76.6 %
4	751	75.1 %
5	758	75
6	768	76.8 %
7	755	75.5 %
8	758	75.8 %
9	758	75.8 %
10	736	73.6 %

Sample Mean: 75.64
Standard Deviation: 0.9252027
Variance: 0.856

4.1.5.1 Random Forest with Feature Selection

In random forest, after the Wrapper method, only three attributes remains:

- checking_status
- credit_history
- other_parties

Table 4.13: Accuracy with Random Forest Feature Selection

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	742	74.2 %
2	738	73.8 %
3	736	73.6 %
4	719	71.9 %
5	743	74.3 %
6	741	74.1 %
7	741	74.1 %
8	737	73.7 %
9	740	74 %
10	731	73.1 %

Sample Mean: 73.68
Standard Deviation: 0.7177124
Variance: 0.5151111

Random Forest Homogeneity Variance and T-test

Related R codes and comparison results are indicated as below:

```
a=c(75.9,76.3,76.6,75.1,75,76.8,75.5,75.8,75.8,73.6)
b=c(74.2,73.8,73.6,71.9,74.3,74.1,74.1,73.7,74,73.1)

var.test(a,b)                                    // p-value = 0.461
t.test(a,b, var.equal=TRUE, paired=FALSE)    //   p-value = 4.944e-05
```

Result

When Fisher's F-test is applied, We obtained p-value greater than 0.05, then we can assume t hat the two variances are homogeneous. Then T-Test is applied, We obtained p-value bigger t han 0.05, then we can conclude that the averages of two groups are significantly similar.

As seen above, random forest feature selection has bad performance in terms of reduction attribute. Thus, feature selection importance is implemented in R that is only special to random forest.

Random Forest Feature Importance Measure

For random forest, feature importance can be measured. Measure operation can be done in R script language. MeanDecreaseAccuracy and MeanDecreaseGini techniques are used. MeanDecreaseAccuracy is calculated using out of bag (OOB) data. For each tree MeanDecreaseAccuracy is calculated on observations not used to form that particular tree. In contrast, MeanDecreaseGini is a summary of how impure the leaf nodes of a tree are. It is calculated using the same data used to fit trees. Related R codes are shown a below:

```
install.packages('randomForest')
library('randomForest')

rmod = randomForest(deneme_numerical[,1:20], y=deneme_numerical[,21],ntree=500)

rmod$importance
```

Then results are:

	MeanDecreaseAccuracy	MeanDecreaseGini
checking_status	0.0326129162	44.772923
duration	0.0147884507	36.060343
credit_history	0.0115957909	35.356963
purpose	0.0079748038	37.817870
credit_amount	0.0112510338	46.918941
savings_status	0.0043298451	20.921139
employment	0.0030691522	21.032386
installment_commitment	0.0032510536	15.046631
personal_status	0.0006395535	21.408518
other_parties	0.0020744718	7.787615
residence_since	0.0013163651	13.989478
property_magnitude	0.0042324776	23.056050
age	0.0054132237	36.094765
other_payment_plans	0.0027330966	9.910620
housing	0.0015034026	10.024363
existing_credits	0.0016787993	7.949957
job	0.0032050047	16.367684
num_dependents	0.0001087696	5.088920
own_telephone	0.0016994088	5.542816
foreign_worker	0.0002107329	1.457451

And the confusion matrix:

```
rmod$confusion
      bad good class.error
bad  116  184  0.61333333
good  52  648  0.07428571
```

SVM and Random Forest with Random Forest Feature Importance

In this section, according to the above results, SVM and Random Forest are analysed after the measure of random forest feature importance. Top 15 attributes are selected according to the Gini values so as to perform analyse:

- credit_amount
- checking_status
- purpose
- age
- duration
- credit_history
- property_magnitude
- personal_status
- employment
- saving_status
- job
- installment_commitment
- residence_since
- housing
- other_payment_plans
- existing_credit

Random Forest

Table 4.14: Accuracy with Random Forest Feature Selection

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	761	76.1 %
2	753	75.3 %
3	759	75.9 %
4	770	77 %
5	756	75.6 %
6	764	76.4 %
7	773	77.3 %
8	748	74.8 %
9	747	74.7 %
10	754	75.4 %

Sample Mean: 75.85
Standard Deviation: 0.8682678
Variance: 0.7538889

Support Vector Machine

Table 4.15: Accuracy with Random Forest Feature Selection

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	768	76.8 %
2	756	75.6 %
3	748	74.8 %
4	757	75.7 %
5	764	76.4 %
6	764	76.4 %
7	757	75.7 %
8	760	76 %
9	770	77 %
10	753	75.3 %

Sample Mean: 75.97
Standard Deviation: 0.6848357
Variance: 0.469

4.1.6 Logistic Regression (LR)

Logistic regression test is used to estimate class probabilities directly. Difference between linear regression, dependent variable is categorical. So in linear regression class attribute of dataset must be changed in the filter as nominal to binary.
In Logistic Regression analyse, logit transform uses to predict probabilities directly. Logit function also means S-curve. The goal of logistic regression is to explain the relationship between the explanatory variables and the outcome, so that an outcome can be predicted for a new set of explanatory variables.[2]

Table 4.16: Accuracy with Logistic Regression

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances	
1	757	75.7 %
2	759	75.9 %
3	734	73.4 %
4	756	75.6 %
5	757	75.7 %
6	750	75 %
7	744	74.4 %
8	751	75.1 %
9	755	75.5 %
10	752	75.2 %

Sample Mean: 75.15
Standard Deviation: 0.7560864
Variance: 0.5716667

4.1.6.1 LR with Feature Selection

After the Wrapper method, most effective attributes for logistic regression are shown as the following:

- checking_status
- duration
- credit_history
- credit_amount
- savings_status
- employment
- installment_commitment
- personal_status
- other_parties
- property_magnitude
- age
- existing_credits
- num_dependents
- own_telephone
- foreign_worker

Table 4.17: Accuracy with Logistic Regression

5 Fold Cross Validation-Random Seed:	Correctly Classified Instances
1	773 77.3 %
2	762 76.2 %
3	756 75.6 %
4	764 76.4 %
5	750 75 %
6	760 76 %
7	756 75.6 %
8	766 76.6 %
9	762 76.2 %
10	753 75.3 %

Sample Mean: 76.02
Standard Deviation: 0.6746192
Variance: 0.4551111

LR Homogeneity Variance and T-test

Related R codes and comparison results are indicated as below:

```
a=c(75.7,75.9,73.4,75.6,75.7,75,74.4,75.1,75.5,75.2)
b=c(77.3,76.2,75.6,76.4,75,76,75.6,76.6,76.2,75.3)

var.test(a,b)          // p-value = 0.7396
t.test(a,b, var.equal=TRUE, paired=FALSE) //   p-value = 0.01419
```

Result:

When Fisher's F-test is applied, We obtained p-value greater than 0.05, then we can assume t hat the two variances are homogeneous. Then T-Test is applied, We obtained p-value smaller than 0.05, then we can conclude that the averages of two groups are significantly different.

4.1.7 Precision/Recall Curve And ROC Curve

In this section, precision/recall curve and roc curve are sketched according to the confusion matrix. Confusion Matrix contains information about actual and predicted classifications done by a classification system. It is shown as the following:

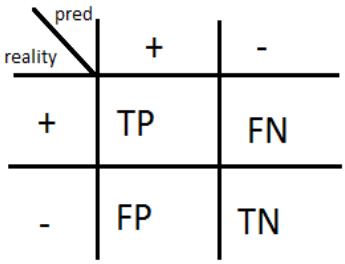


Figure 4.8: Content of Confusion Matrix

TP: TP is the number of **correct** predictions that an instance is **positive**.

TN: TN is the number of **correct** predictions that an instance is **negative**,

FP: FP is the number of **incorrect** predictions that an instance is **positive**,

FN: FN is the number of **incorrect** of predictions that an instance **negative**

- Precision is defined as the number of true positives (TP) over the number of true positives plus the number of false positives (FP).

$$P = \frac{T_p}{T_p+F_p}$$

- Recall (R) is defined as the number of true positives (TP over the number of true positives plus the number of false negatives (FN).

$$R = \frac{T_p}{T_p+F_n}$$

- ROC curve is the ratio of true positive and false positive.

According to the confusion matrix, precision/recall and roc curve are plotted for every algorithms:

4.1.7.1 Artificial Neural Network

```
=== Confusion Matrix ===
   a   b   <-- classified as
 635 65 |    a = good
 195 105 |   b = bad
```

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,907	0,650	0,765	0,907	0,830	0,314	0,762	0,861	good
	0,350	0,093	0,618	0,350	0,447	0,314	0,762	0,587	bad
Weighted Avg.	0,740	0,483	0,721	0,740	0,715	0,314	0,762	0,779	

Precision/Recall Curve – Class: Bad

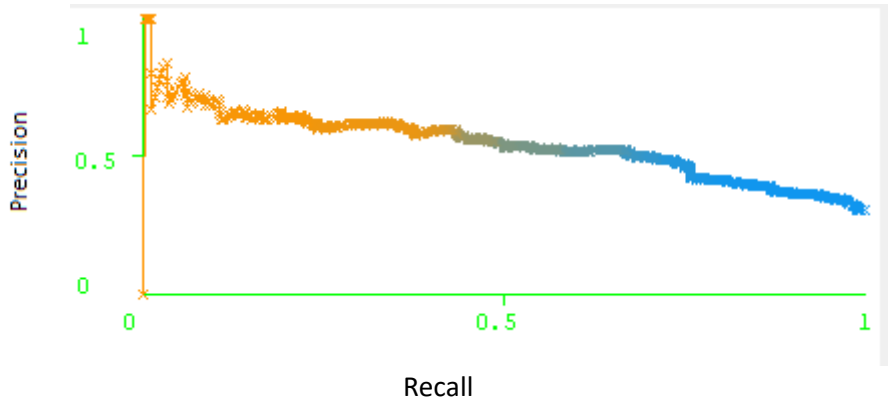


Figure 4.9: Precision Recall Curve for ANN

At first, precision starts at the point of 1. Because there is no error at start so threshold is 1. Then precision value decreases because of the negative sample (class: bad). On the other handrecall is 0 at start because there is no recognition sample. Then recall increases after the recognition of good sample. Finally recall is 1 the whole good sample.

ROC Curve

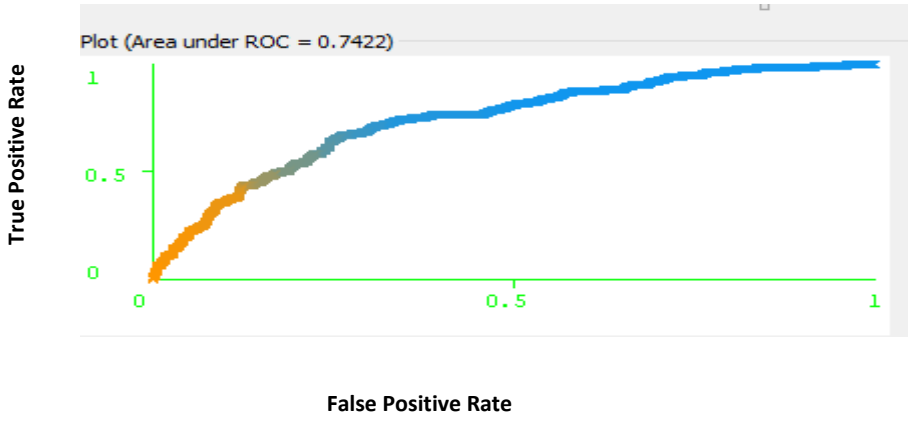


Figure 4.10: ROC Curve for ANN

ROC curve indicates that true positive rate versus false positive rate.

4.1.7.2 Decision Tree

=== Confusion Matrix ===

a b <-- classified as

590 110 | a = good

143 157 | b = bad

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,843	0,477	0,805	0,843	0,823	0,379	0,711	0,805	good
	0,523	0,157	0,588	0,523	0,554	0,379	0,711	0,509	bad
Weighted Avg.	0,747	0,381	0,740	0,747	0,743	0,379	0,711	0,716	

Precision/Recall Curve – Class: Bad

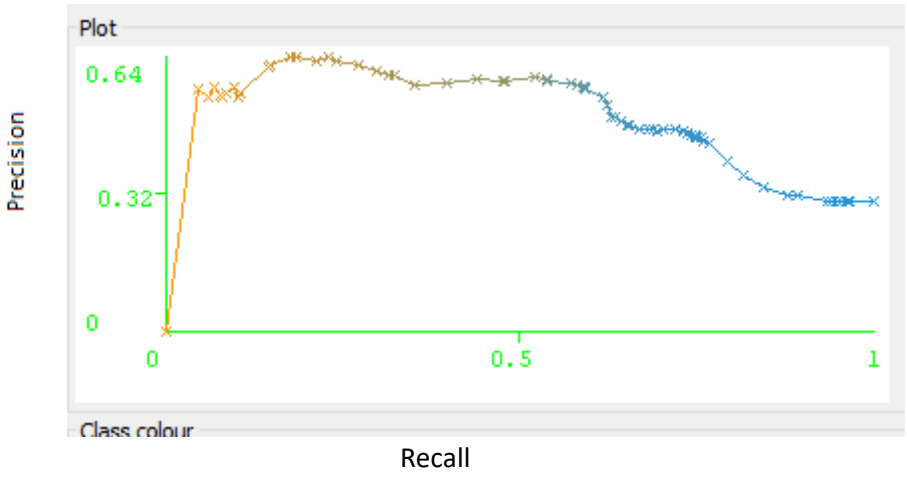


Figure 4.11: Precision/Recall Curve for DT

ROC Curve

Figure 4.12: ROC Curve for DT

4.1.7.3 Support Vector Machine

a b <-- classified as

635 65 | a = good

168 132 | b = bad

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,907	0,560	0,791	0,907	0,845	0,400	0,674	0,782	good
	0,440	0,093	0,670	0,440	0,531	0,400	0,674	0,463	bad
Weighted Avg.	0,767	0,420	0,755	0,767	0,751	0,400	0,674	0,686	

Precision/Recall Curve – Class: Bad

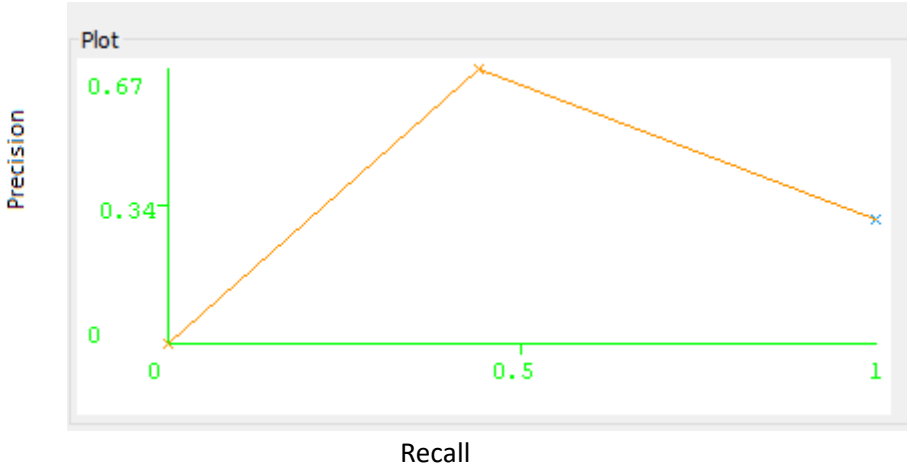


Figure 4.13: Precision/Recall Curve for SVM

ROC Curve

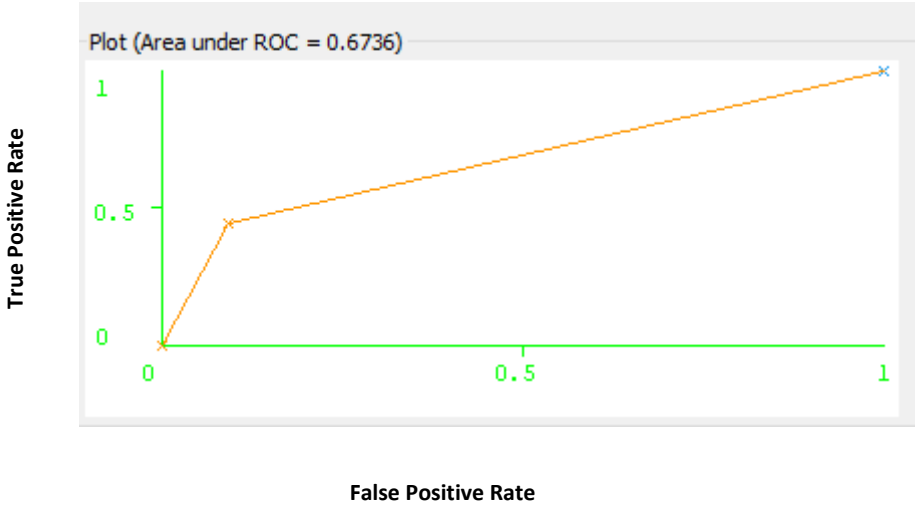


Figure 4.14: ROC Curve for SVM

4.1.7.4 Random Forest

```
     === Confusion Matrix ===

   a   b   <-- classified as
 636  64 |   a = good
 175 125 |   b = bad
```

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,909	0,583	0,784	0,909	0,842	0,381	0,786	0,887	good
	0,417	0,091	0,661	0,417	0,511	0,381	0,786	0,615	bad
Weighted Avg.	0,761	0,436	0,747	0,761	0,743	0,381	0,786	0,805	

Precision/Recall Curve – Class: Bad

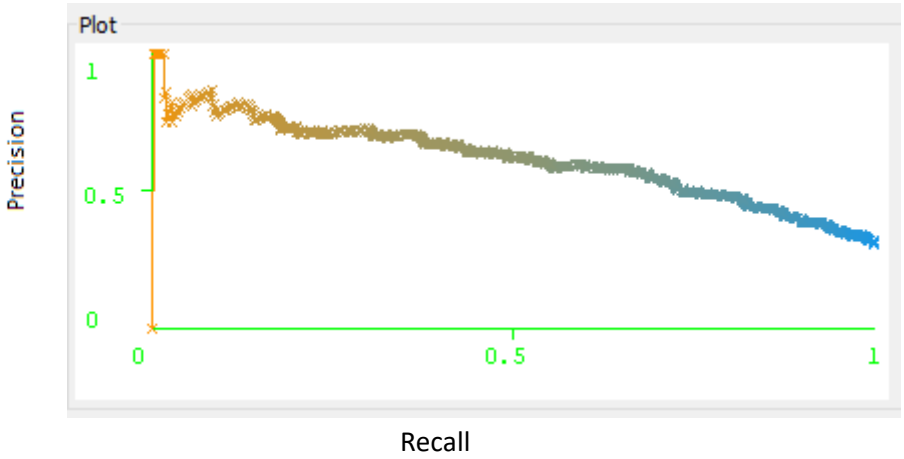


Figure 4.15: Precision Recall Curve for Random Forest

ROC Curve

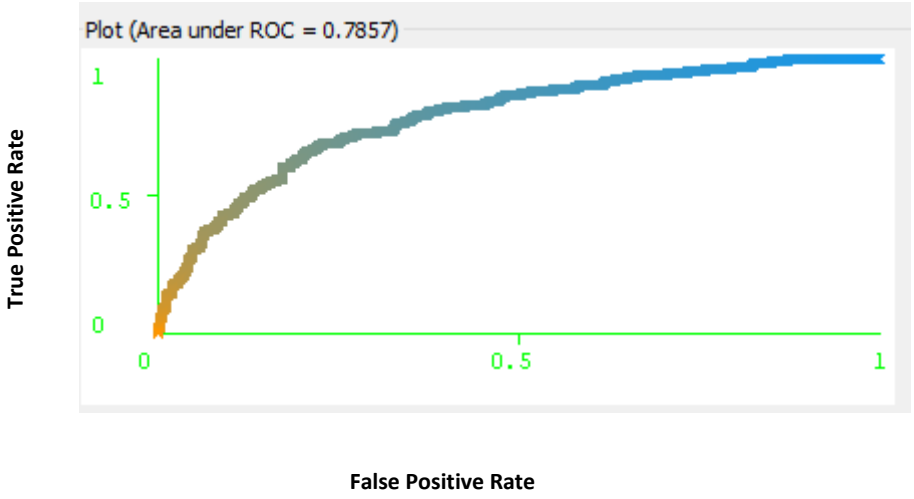


Figure 4.16: ROC Curve for Random Forest

4.1.7.5 Logistic Regression

=== Confusion Matrix ===

a	**b**	**<-- classified as**
623	**77**	**a = good**
150	**150**	**b = bad**

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,890	0,500	0,806	0,890	0,846	0,427	0,783	0,883	good
	0,500	0,110	0,661	0,500	0,569	0,427	0,783	0,605	bad
Weighted Avg.	0,773	0,383	0,762	0,773	0,763	0,427	0,783	0,799	

Precision/Recall Curve – Class: Bad

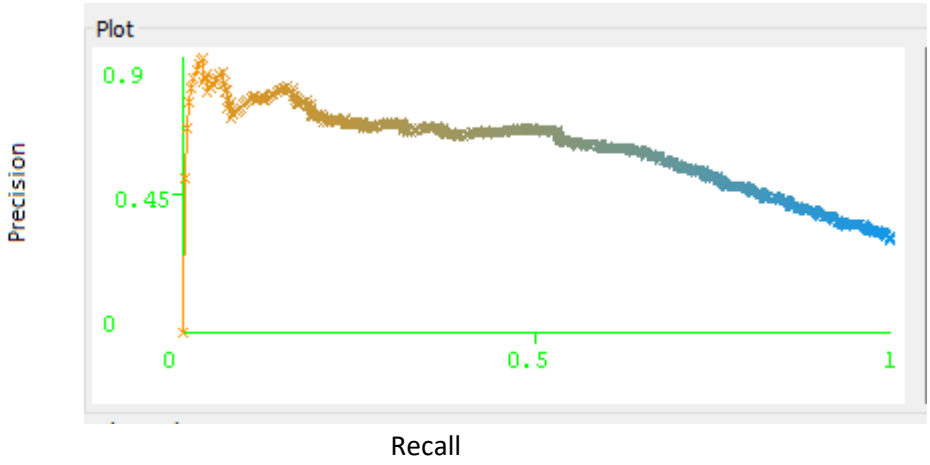


Figure 4.17: Precision/Recall Curve for Logistic Regression

ROC Curve

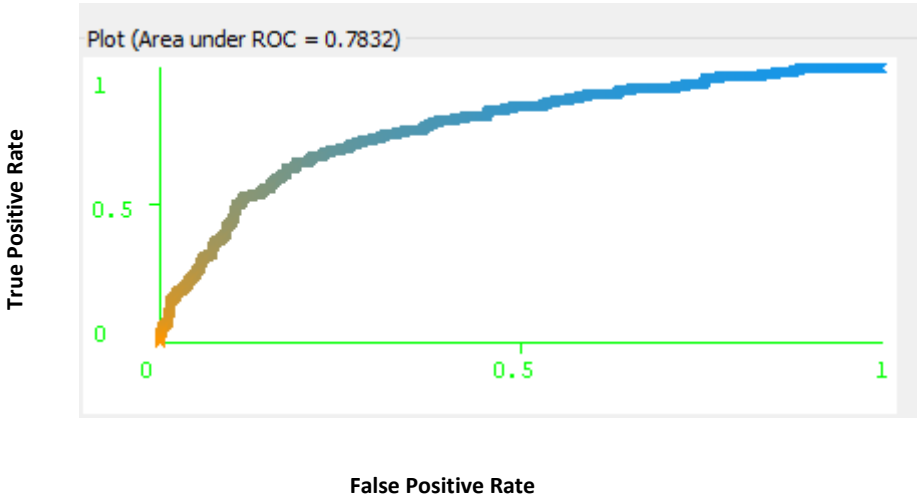
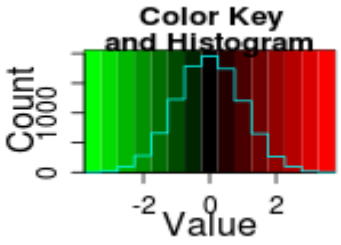


Figure 4.18: ROC Curve for Logistic Regression

4.1.8 Heatmap

In this section, correlation matrix are scratched with the aid of heatmap.2 function in R. Heatmap indicates the relationship between attributes. In this heatmap, read-green heatmap are used. The color's value according to the correlation and heatmap are shown as the following. Red areas means that there is a high relationship between attributes.



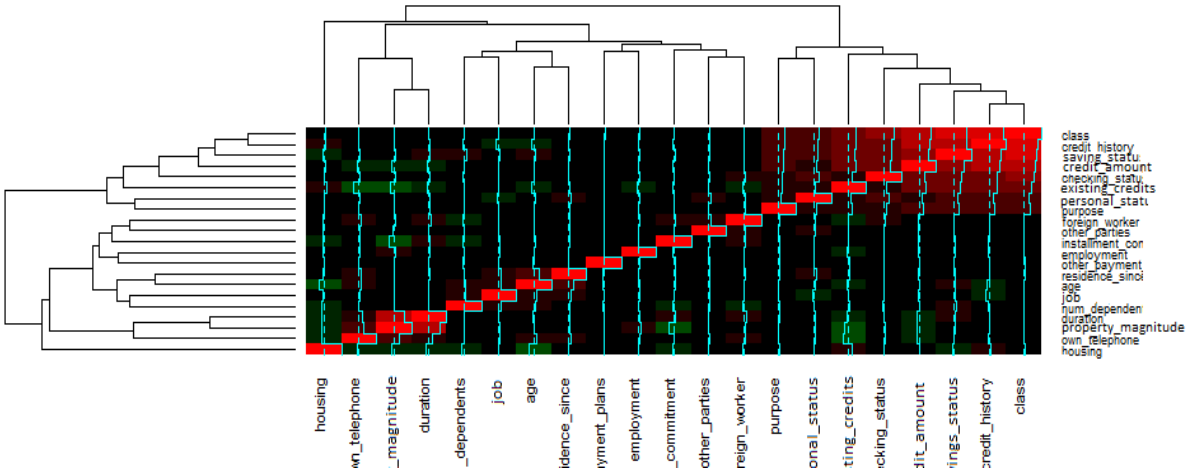


Figure 4.19: HeatMap

4.2 Modelling

This section is prepared in Azure Machine Learning. According the algorithms performance in WEKA, the best performance for accuracy is obtained in logistic regression. After the loading dataset in Azure Machine Learning, firstly attributes' real name are added with the aid MetaEditor model. Since, as it is mentioned dataset is loading in tool as some coded number. (A11, A12, A13 and so on). Then Logistic Regression model is added in the workspace. It is connected trained model. In train model, class attribute is eliminated because of determine predict. Then it is connected to score model because of the scoring probabilities. Finally Score model is connected to the evaluate model. Related flow diagram is shown as the following:

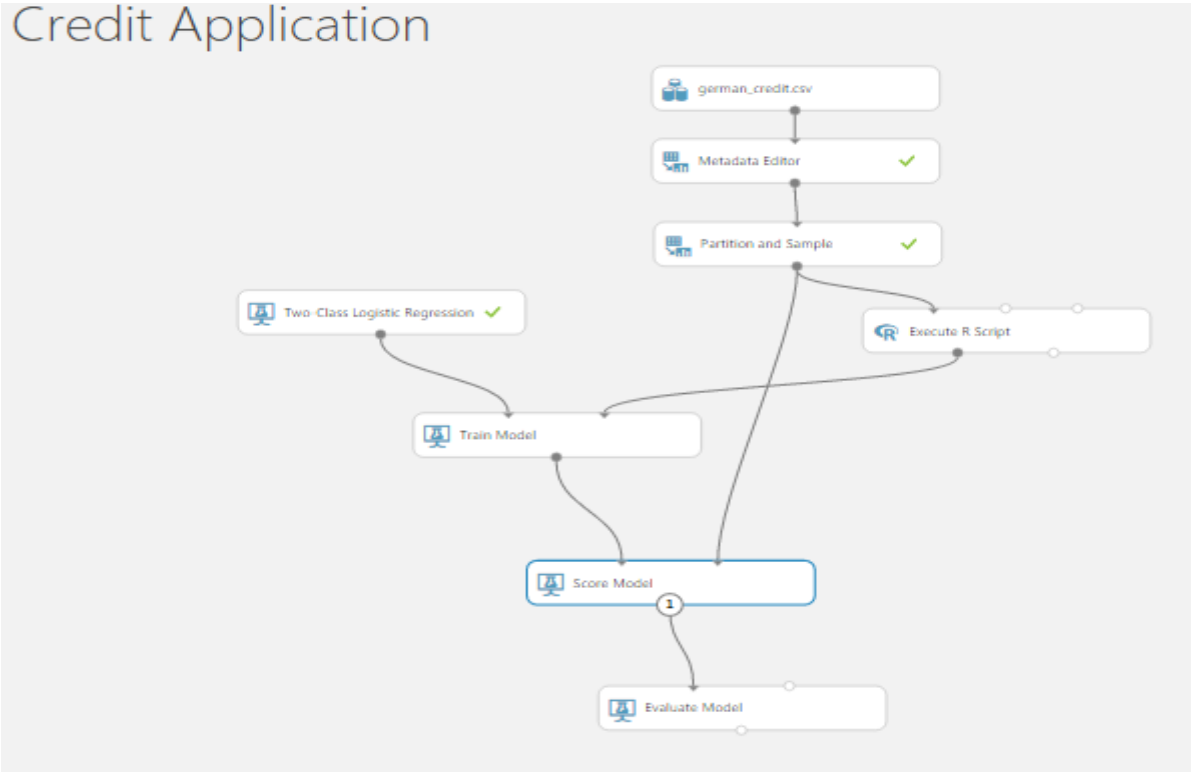


Figure 4.20: Azure Machine Learning Model

Screen of sample form application is indicated as the following:

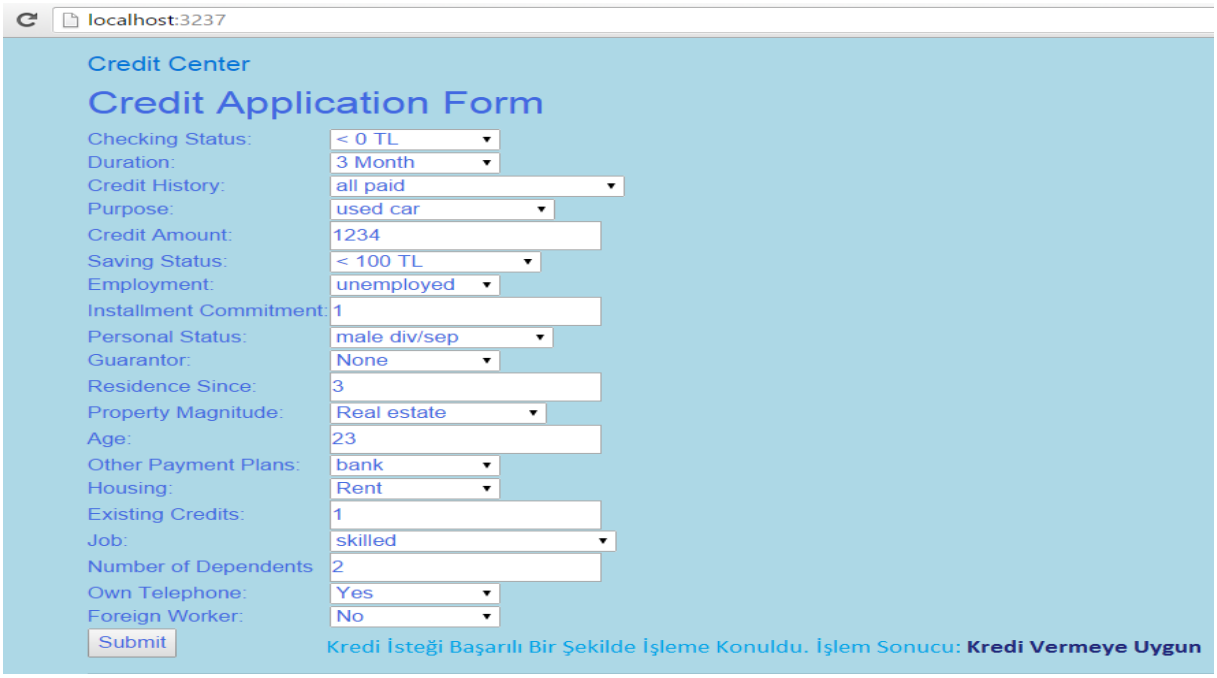


Figure 4.21: Form Application

As it is seen, when Submit button is pressed, it is requested from web service, then web service responses message in screen as a result.

5. DESIGN AND IMPLEMENTATION

In this section, project which is related to MVC is explained. Visual Studio 2013 platform is used for creating this model. MVC project is typically created as a Model, View and Controller. A form application is prepared in the part of View which makes a request to the web service (Azure Machine Learning). Then in the part of Model, an object is created in order to receive user input to controller. Then controller receives user input and makes call to model objects and the view to perform appropriate actions. Concept Diagram and Class Diagram of MVC project are shown as the following:

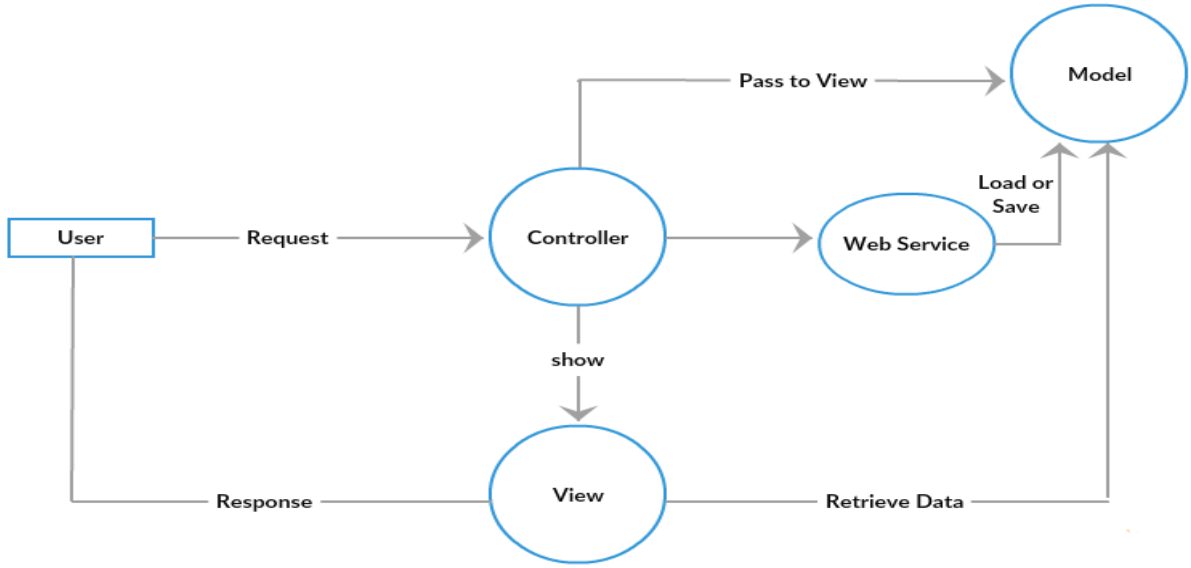


Figure 5.1: Concept Diagram of MVC

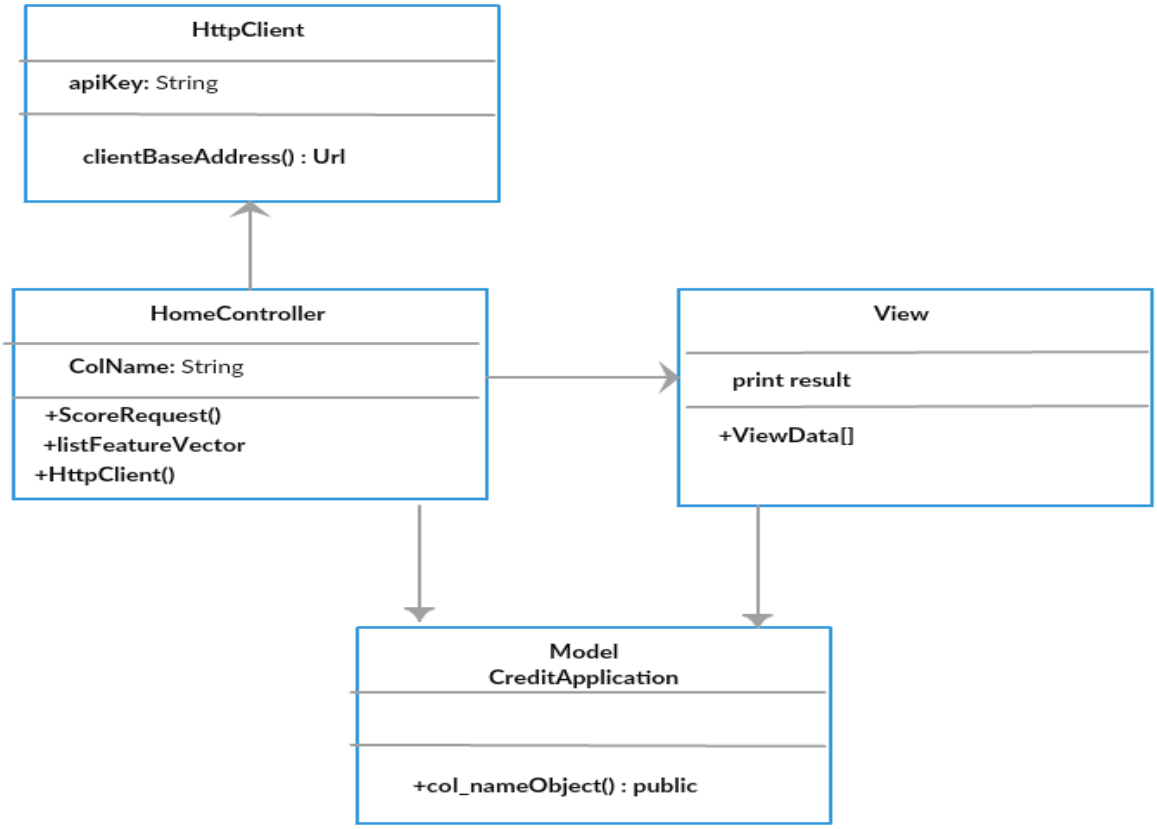


Figure 5.2: Class Diagram of MVC

6. EXPERIMENTAL RESULT

According to the analysis, summary of the algorithm performance, distribution of the attribute importance and heatmap correlation are shown as below:

Algorithm Accuracy Table

Table 6.1: Algorithm Accuracy Table

Algorithms	Normal Evaluation Accuracy – S.deviation	Evaluation with Feature Selection Accuracy – S. deviation	Random Forest Feature Importance Accuracy – S.deviation
Decision Tree	71.9 - 1.231981	73.84 – 1.047537	
Artificial Neural Network	72.41 – 1.299957	74.38 – 1.047537	
Support Vector Machine	75.83 – 0.7944949	75.82 -0.6069962	75.97 – 0.86826
Random Forest	75.64 – 0.9252027	73.68 – 0.7177124	75.85 – 0.6848357
Logistic Regression	75.15 – 0.7560864	76.02 – 0.6746192	

Attribute Distribution Table

Table 6.2: Attribute Distribution Table

Number	Attribute	Decision Tree	Artificial Neural Network	Support Vector Machine	Random Forest	Logistic Regression
1	Status	√	√	√	√	√
2	Duration	√	√			√
3	Credit History	√	√	√	√	√
4	Purpose					
5	Credit Amount	√	√	√		√
6	Savings	√	√	√		√
7	Employment Duration			√		√
8	Installment Rate			√		√
9	Personal Status			√		√
10	Debtors	√	√	√	√	√
11	Residence					
12	Property			√		√
13	Age	√		√		√
14	Installment plans			√		
15	Housing					
16	Existing Credits	√		√		√
17	Job			√		
18	Liable People					√
19	Telephone			√		√
20	Foreign Worker	√	√	√		√

HeatMap Correlation Table

Table 6.3: HeatMap Correlation Table

Attribute	Correlation with Class (Good /Bad)
Status	Very High
Duration	Normal
Credit History	Very High
Purpose	High
Credit Amount	Very High
Savings	Very High
Employment Duration	Normal
Installment Rate	Normal
Personal Status	High
Debtors	Normal
Residence	Normal
Property	Normal
Age	Normal
Installment plans	Normal
Housing	Normal
Existing Credits	High
Job	Normal
Liable People	Normal
Telephone	Normal
Foreign Worker	Normal

Evaluations of Tables:

According to the tables, it can be said that which features are more important than the others. In the Algorithm Accuracy Tables; it can be said that the best performance is showed by the algorithm of logistic regression feature selection. In Attributes distribution table and Heatmap table, it can be seen that there is almost consistently distribution. In this way, one of most important attributes are listed as below:

- Checking Status
- Credit History
- Savings Status
- Credit Amount
- Existing Credits
- Personal Status
- Age

7. CONCLUSION AND FUTURE WORK

At the end of the project, after the analysis of algorithms; logistic regression have the best performance to determine credit scoring with respect to related attributes. On the other hand SVM-Polynomial Kernel another displays good performance. As it is mentioned above, existing systems which are related to credit appeals based on machine learning in banks are still a problem to determine evaluation of credit appeals. Although there is an existing decision mechanism systems of the bank, manual evaluation is still using in banks in terms of credit appeal. Therefore, in order to eliminate manual evaluation entirely, existing systems must be further improved with the aid of analyse different algorithms.

On the other hand, one of the future aim of the project is to used banks' real data and scoring evaluations. Thus, comparison between existing system and future project is more reliable.

8. REFERENCES

[1] Türkiye Bankalar Birliği Risk Merkezi. 2015.

http://www.riskmerkezi.org/tr-TR/istatistikiBilgiler.aspx

[2] Witten I.H., Frank E., 2005, Data mining: practical machine learning tools and techniques – 2nd ed. p. cm. – Morgan Kaufmann series in data management systems. ISBN: 0-12-088407-0.

[3] Ye, Y., Liu, S. & Li, J. 2008. A multiclass learning approach to credit rating prediction. *International Sysposiums on Information Processing*

[4] Reshmy, A.K., Paulraj, D. 2015. An efficient unstructured big data analysis method for enhancing performance using machine learning algorithm. *International Conference on Circuit, Power and Computing Technologies. IEEE,* 2015, 1-7.

[5] Zhou, H., Lan, Y., Soh, Y. C., Huang, G. B., & Zhang, R. 2012. Credit risk evaluation with extreme learning machine. *International Conference on Systems, Man, and Cybermeics. IEEE.* October 14-17, Korea, 1064-1069.

[6] Machine Learning Group at the University of Waikato

http://www.cs.waikato.ac.nz/ml/weka/

[7] Alpaydın, E., Introduction to Machine Learning, The MIT Press, 2004, Printed and bound in the United States of America. ISBN 0-262-01211-1.

[8] Microsoft Azure Corporation. Azure Machine Learning.
https://studio.azureml.net/

[9] UC Irvine Machine Learning Repository, “UCI Machine Learning Repository”,
https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data)

[10] R script Language
https://www.r-project.org/

[11] http://bilgisayarkavramlari.sadievrenseker.com/2008/11/02/ileri-beslemeli-aglar-feedforward- neural-networks/

[12] University of Oxford, 2015. Information Engineering.
http://www.robots.ox.ac.uk/