

Research of Classification techniques in machine learning for Predicting Credit Risk modelling

*A project report submitted in partial fulfillment of the requirements for
B.Tech. Project*

B.Tech.

By

ANKIT (2017IMG-066)



विश्वजीवनामृतं ज्ञानम्

**ABV INDIAN INSTITUTE OF INFORMATION
TECHNOLOGY AND MANAGEMENT
GWALIOR-474 010**

2008

CANDIDATES DECLARATION

We hereby certify that the work, which is being presented in the report, entitled **Research of Classification techniques in machine learning for Predicting Credit Risk modeling**, in partial fulfillment of the requirement for the award of the Degree of Bachelor of Technology and submitted to the institution is an authentic record of our own work carried out during the period May 2020 to September 2020 under the supervision of **Dr. Rajesh Rajagopal and Dr. Jeevaraj S.** We also cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.

Date: 26/10/2020

Signatures of the Candidates

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Date: 10/11/2020

Signatures of the Research Supervisors

ABSTRACT

With the enhancement in the banking sector, many of people are applying for bank loans but the bank has its limited assets which it has to allow to limited people only, therefore finding out to whom the loan can be given which will be a safer option for the bank is a typical process. So in this project, we try to decrease this risk factor behind selecting the safe person so as to save lots of bank efforts and assets. This is done by mining the Big Data of the previous records of the people to whom the loan was allowed before and on the basis of these records/experiences the machine was trained using the machine learning model which gives the most correct result. The main objective of this project is to predict whether assigning the loan to a particular person will be safe or not. This paper is separated into four sections (1)Dataset Collection (2) Comparison of machine learning models on collected data set (3) Training of system on the most promising model (4) training and Testing . In this paper we are predict the loan data by using some machine learning algorithms they are classification, logic regression, Decision Tree etc.

Keywords - Loan Prediction, Machine Learning, Classification techniques, Model deploy in Django.

ACKNOWLEDGEMENTS

We are highly indebted to **Dr. Rajesh Rajagopal** and **Dr. Jeevaraj S** , and are obliged for giving us the autonomy of functioning and experimenting with ideas. We would like to take this opportunity to express our profound gratitude to them not only for their academic guidance but also for their personal interest in our project and constant support coupled with confidence boosting and motivating sessions which proved very fruitful and were instrumental in infusing self-assurance and trust within us. The nurturing and blossoming of the present work is mainly due to their valuable guidance, suggestions, astute judgment, constructive criticism and an eye for perfection. Our mentor always answered myriad of our doubts with smiling graciousness and prodigious patience, never letting us feel that we are novices by always lending an ear to our views, appreciating and improving them and by giving us a free hand in our project. It's only because of their overwhelming interest and helpful attitude, the present work has attained the stage it has. Finally, we are grateful to our Institution and colleagues whose constant encouragement served to renew our spirit, refocus our attention and energy and helped us in carrying out this work.

TABLE OF CONTENTS

ABSTRACT	3
CHAPTER 1 : INTRODUCTION.....	7
1.1 INTRODUCTION.....	7
1.2 NEED OF THE STUDY	8
1.3 OBJECTIVES.....	8
1.4 DATA SOURCES	8
1.5 TOOLS & TECHNIQUES	9
CHAPTER 2 : SYSTEM ARCHITECTURE AND DATA PREPARATION OR UNDERSTANDING ...	10
2.1 ARCHITECTURE.....	10
2.1.1 DATA COLLECTION.....	10
2.2 PRE PROCESSING	11
2.2.1 Data Extraction and Cleaning:	11
2.3 Feature Engineering.....	13
2.3.1 Data Transformation and Visualization	13
2.3.2 Correlation:	16
CHAPTER 3 : FITTING MODELS TO DATA	17
3.1 RANDOM FOREST	17
3.2 LOGISTIC REGRESSION.....	19
3.3 SUPPORT VECTOR MACHINE.....	21
3.4 K-Nearest-Neighbors Model	23
3.5 Feature Importance Analysis	24
3.6 Model Comparison.....	25
3.7 Model deploy in graphical User interface	26
CHAPTER 4 : Enhancement and Conclusion	27
4.1 Future enhancement.....	27
4.2 Conclusion.....	27
CHAPTER 5 : REFERENCES	28
5.1 REFERENCE	28

List of Figures

Figure 2.1 model	10
Figure 2.2 code of filling the missing value	12
Figure 2. 3 Code of Label encoding.....	13
Figure 2.4 Plot of Application Income beform trasformation and after transformation.....	14
Figure 2.5 plot of Co-Application Income beform trasformation and after transformation	14
Figure 2.6 Plot of Total income after log-transformation.....	14
Figure 2.7 plot of loan amount term beform trasformation and after transformation.....	15
Figure 2.8 Bar Chart categorical attributes	15
Figure 2. 9 Correlation Matrix of the given attribute.....	16
Figure 3.1 confusion matrix for the Random forest classifier model	18
Figure 3.2 confusion matrix for the optimized model of random forest classifier	19
Figure 3.3 confusion matrix for logistic regression model	20
Figure 3.4 confusion matrix for the optimized SVM model.....	23
Figure 3.5: confusion matrix for the K-nearest-neighbors model when K = 10.....	24
Figure 3.6 graphical user interface of deploy model	26

List of table

Table 2. 1 Data Type Table.....	11
Table 2.2 Data percent of missing value.....	12
Table 3.1 Best Parameter for Optimized SVM.....	22
Table 3.2 Model Comparison table.....	26

CHAPTER 1 : INTRODUCTION

This chapter includes the details of credit risk, loan prediction method and model, our objective tools and techniques used in deploying the project.

1.1 INTRODUCTION

The prediction of defaulting the borrower in future is a challenging task for Financial institutions . this project intends to develop prediction models using binary classification techniques for defaulting the borrower in future. The project aims to solve the complex problem of identifying the loan defaulters based on different factors using Risk analytics .this will aid the financial firms to take more informed credit scoring decisions and reduce the defaulters rate in the future. This helps the banks to reduce the possible losses and can boost the volume of credits. The results of this credit risk assessment will be the prediction of the Probability of Default (PD) of an applicant. Therefore, it becomes necessary to build a model that will consider the various features of the applicants and produces an assessment of the Probability of Default of the applicant. This parameter PD, helps the bank to decide if they can offer the loan to the applicant or not. In such a scenario the data being analyzed is huge and complicated and using data mining techniques to reach the result is the most suitable option provided its efficient analytical methodology that attains useful knowledge. There are many such works has been done previously, but they have not explored the use of the features available in machine learning. machine learning is an excellent statistical and data mining tools that can handled any volume of structured as well as unstructured data and provide the result in a super fast manner and presents the result in both text and graphical manners. This allows the decision-maker to make more reliable predictions and analysis of the findings. The goal of this work is to propose a data mining framework using machine learning techniques for predicting PD for the new loan applicant of a Bank. The data used for analysis holds many inconsistencies like missing values, outliers and inconsistencies and they have to be handled before being used to develop the model. Only a few of the customer parameters really contribute to the prediction of the defaulter. Therefore, those parameters or features need to be identified before a model is used. To classify if the applicants is a defaulter or not, the good data mining approach is the classification modelling techniques. The above-said steps are integrated into a unique model and prediction is done based on this

INTRODUCTION

model. Related works have been discussed in the need of study Section and the gap in exploring using machine learning techniques has been highlighted. The “Methodology” Section examines the approach that has been followed using text as well as block diagram. In the “Results and Discussions” Section explores the coding and the resultant model applied in this work. It is also essential to note that the metrics derived out of this model determines the high accuracy and efficiency of the built model.

1.2 NEED OF THE STUDY

In the past year, the bank sector faced a lot of losses due to bad loans. and some banks have come to Bankruptcy stage such as yes bank, Punjab national bank, etc. Every year the bank sector faces a lot of losses . So that Prediction of loan defaults is critical to financial institutions in order to minimize losses from loan non-payments. We are motivated to solve that problem using machine learning technology. In this paper we use different techniques to automate tasks and provide better accurate taking decisions and reduce the risk of the bank to provide a loan.

1.3 OBJECTIVES

The main objectives of our project can be summarized in the following points:

- ☐ Create a predictive model to classify each borrower a defaulter or not using the data collected when loans has been given to applicant.
- ☐ Minimize the risk of borrowers defaulting the loan using created model.
- ☐ Study about different types of classification algorithms, mainly supervised learning algorithms such as Logistic Regression, Decision Tree and Random Forest, etc on a given data set which containing information of customer. and compare their accuracies and build a powerful model.

1.4 DATA SOURCES

INTRODUCTION

The provided dataset corresponds to all loans issued to individuals in the past from (<https://datahack.analyticsvidhya.com/contest/practice-problem-loan-prediction-iii>). The dataset has 720 observations and 13 features. The data contains the indicator of default, payment information, credit history, etc. Customers under 'current' status have been considered as non-defaulters in the dataset. We have also been provided with a Data dictionary that best describes the features.

The dataset has quite a lot of missing values and the figures can be considered as ground truth, but lots of columns are either irrelevant, very sparse or non informative. Moreover, the dataset is unbalanced, with approximately 4% of loans considered as defaulted.

1.5 TOOLS & TECHNIQUES

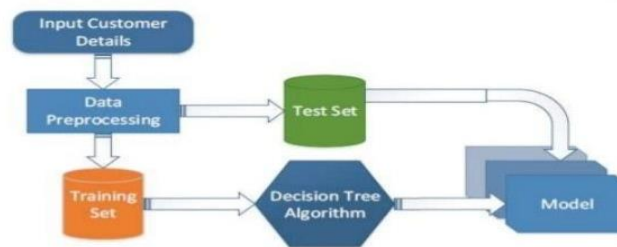
Tools: *Python 3.7.2, Django, html, CSS, Jupyter Notebook, Numpy, Pandas, Matplotlib, Seaborn, Scikit-learn, Scipy,*

Techniques: *Logistic regression, Random Forest Classifier Etc.*

CHAPTER 2 : SYSTEM ARCHITECTURE AND DATA PREPARATION OR UNDERSTANDING

This chapter tell about the implementation of loan predication model and system architecture and data preparation or understanding about dataset

2.1 ARCHITECTURE



Architecture of Proposed Model

FIGURE 2.1 ARCHITECTURE OF PROPOSED MODEL

2.1.1 DATA COLLECTION

The dataset taken for predicting loan default customers is predicted into the Training set and testing set. Usually, a 70:30 ratio is used to break the training set and testing set . model which was developed by using machine learning classification techniques is used to the training set and based on the test result accuracy, Test set prediction is done. The following are the attributes.

Attribute Name	Category
Loan_ID	Qualitative
Gender	Categorical
Married	Categorical
Dependents	Qualitative
Education	Categorical
Self_Employed	Categorical
ApplicantIncome	Qualitative
CoapplicantIncome	Qualitative
LoanAmount	Qualitative
Loan_Amount_Term	Qualitative
Credit_History	Qualitative
Property_Area	Categorical

TABLE 2. 1 DATA TYPE TABLE

2.2 PRE PROCESSING

The collected information which bear missing values that may lead to inconsistency. In order to achieve more useful results, data must be pre-processed in order to boost the algorithm 's effectiveness. The outliers need to be eliminated and variable conversion needs to be done as well. We use a map function in order to solve these problems.

One of the first steps we engaged in was to outline the sequence of steps that we will be following for our project. Each of these steps are elaborated below

2.2.1 DATA EXTRACTION AND CLEANING:

- **Missing Value Analysis and Treatment**

In our dataset our target shows that 94% have not defaulted and 6% are defaulters or charged off. So this is clearly an unbalanced dataset.

The first problem was knowing whether the columns were filled or mostly empty with useful details. Many empty or almost empty columns that were removed from the dataset were discovered by data exploration because it would prove a difficult task to go back and attempt to respond to each data point that did not seem relevant at the time of the loan application.

Our dataset has 614 rows \times 13 features including the target out of which 7 feature have missing values or NAN. Below we will look at below.



	<pre># find the null values na=df.isnull().sum()/len(df) na*100</pre>	
	Loan_ID	0.000000
	Gender	2.117264
	Married	0.488599
	Dependents	2.442997
	Education	0.000000
	Self_Employed	5.211726
	ApplicantIncome	0.000000
	CoapplicantIncome	0.000000
	LoanAmount	3.583062
	Loan_Amount_Term	2.280130
	Credit_History	8.143322
	Property_Area	0.000000
	Loan_Status	0.000000
	dtype: float64	

TABLE 2.2 DATA PERCENT OF MISSING VALUE

It would be very difficult to look at each column one by one and find the NA or missing values, as we can see there are 614 observations & 13 columns in the dataset. So let's figure out all columns where the missing values are more than a certain number, and use the average of specific columns to fill the missing values for nursery words. And the missing mean ing is filled with categorical terms using mode.

```
[185] # fill the missing values for numerical terms - mean
df['LoanAmount'] = df['LoanAmount'].fillna(df['LoanAmount'].mean())
df['Loan_Amount_Term'] = df['Loan_Amount_Term'].fillna(df['Loan_Amount_Term'].mean())
df['Credit_History'] = df['Credit_History'].fillna(df['Credit_History'].mean())

[186] # fill the missing values for categorical terms - mode
df['Gender'] = df["Gender"].fillna(df['Gender'].mode()[0])
df['Married'] = df["Married"].fillna(df['Married'].mode()[0])
df['Dependents'] = df["Dependents"].fillna(df['Dependents'].mode()[0])
df['Self_Employed'] = df["Self_Employed"].fillna(df['Self_Employed'].mode()[0])
```

FIGURE 2.2 CODE OF FILLING THE MISSING VALUE

2.3 Feature Engineering

Casting continuos variables to numeric:

We have Cast all continuos variables that are necessary for our analysis to numeric so that we can find a correlation between them.

Label Encoding

```
[ ] from sklearn.preprocessing import LabelEncoder
    cols = ['Gender','Married','Education','Self_Employed','Property_Area','Loan_Status','Dependents']
    le = LabelEncoder()
    for col in cols:
        df[col] = le.fit_transform(df[col])
```

FIGURE 2. 3 CODE OF LABEL ENCODING

2.3.1 DATA TRANSFORMATION AND VISUALIZATION

We must first transform the data before training the data to account for any skewness in the distribution of the variable. Various techniques for transformation are available, ranging from log transformation to power transformation. We'll be using Log-transformation for our study. It is used to adjust the distributional form of a data set to be distributed more normally so that it is possible to correctly use tests and confidence limits that require normality.

We found out most of data was not normalized and did have skewness but .so we have found right skewed distributing so we used log transformation to transform these variable so it did normalize the data. it looked beform transformation and how the data was normalized after transformation .

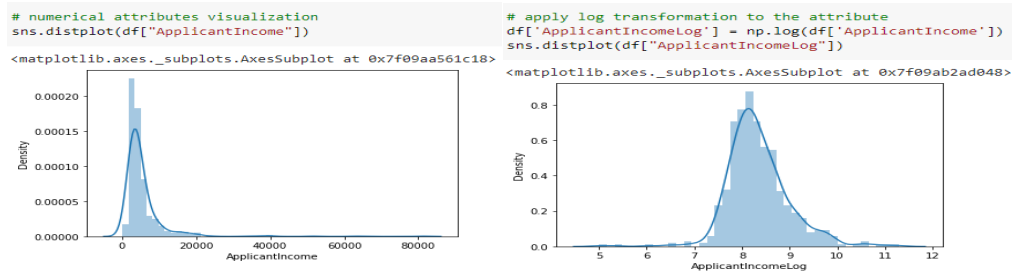


FIGURE 2.4 PLOT OF APPLICATION INCOME BEFORE TRANSFORMATION AND AFTER TRANSFORMATION

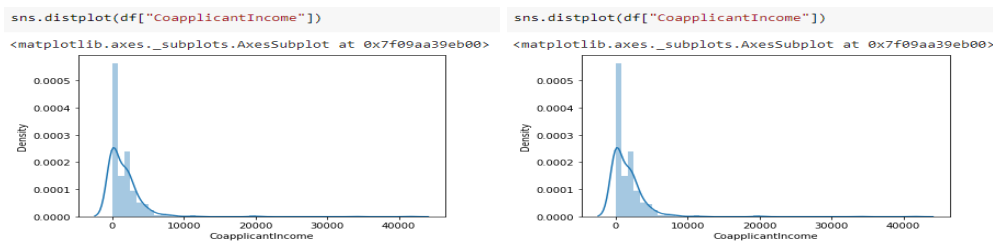


FIGURE 2.5 PLOT OF CO-APPLICATION INCOME BEFORE TRANSFORMATION AND AFTER TRANSFORMATION

Creation of new attributes

```
[ ] # total income
df['Total_Income'] = df['ApplicantIncome'] + df['CoapplicantIncome']
df.head()
```

```
df['Total_Income_Log'] = np.log(df['Total_Income'])
sns.distplot(df["Total_Income_Log"])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f09a9e6d2b0>
```

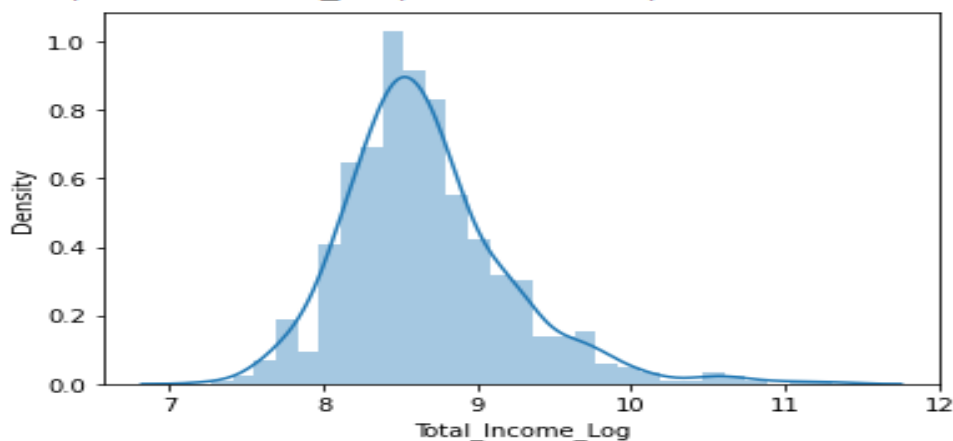


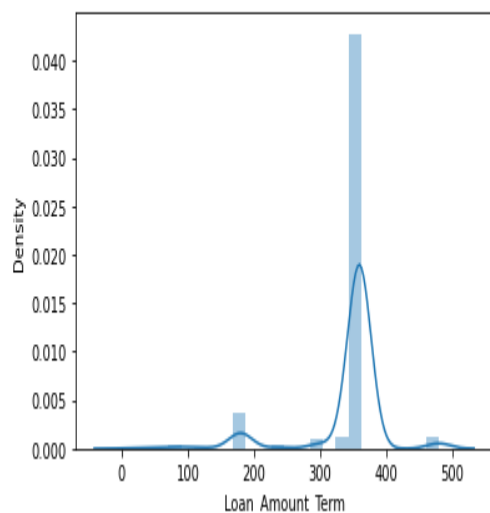
FIGURE 2.6 PLOT OF TOTAL INCOME AFTER LOG-TRANSFORMATION

Loan Amount Term

Change the loan amount term data in normalized form by taking used log transformation

```
sns.distplot(df['Loan_Amount_Term'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f09aa277128>
```



```
df['Loan_Amount_Term_Log'] = np.log(df['Loan_Amount_Term'])  
sns.distplot(df["Loan_Amount_Term_Log"])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f09a9f3ab70>
```

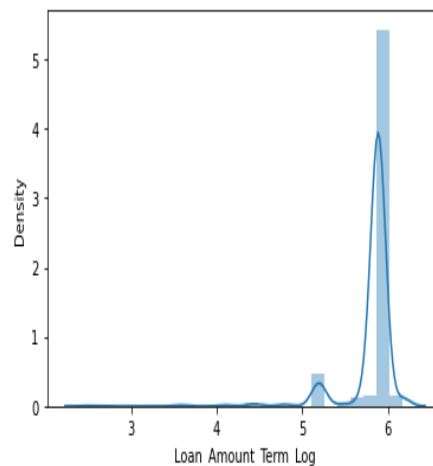


FIGURE 2.7 PLOT OF LOAN AMOUNT TERM BEFORE TRANSFORMATION AND AFTER TRANSFORMATION

Visualization of categorical attributes

Now We are plot bar chart of Gender ,married ,education and self employed for which helps of analysis dataset . after visualization of attributes we will find that maximum which apply loan are man also they are married ,self-employed and also graduate.

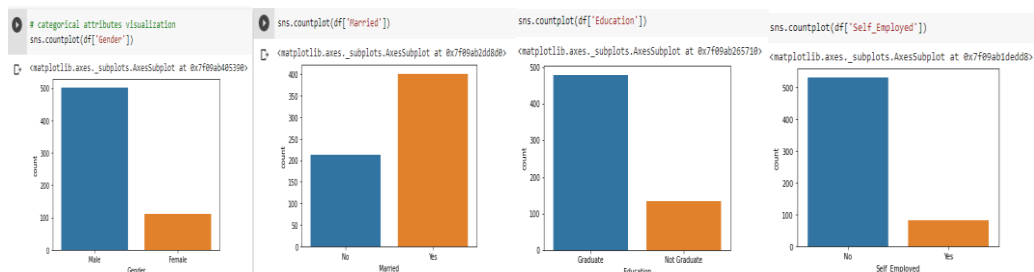


FIGURE 2.8 BAR CHART CATEGORICAL ATTRIBUTES

2.3.2 CORRELATION:

Finding the correlation between variables

We will now look at the structure of the association between our variables selected above. The variables tested for correlation are: This will tell us about any dependencies between different variables and help us reduce the dimensionality a little more.

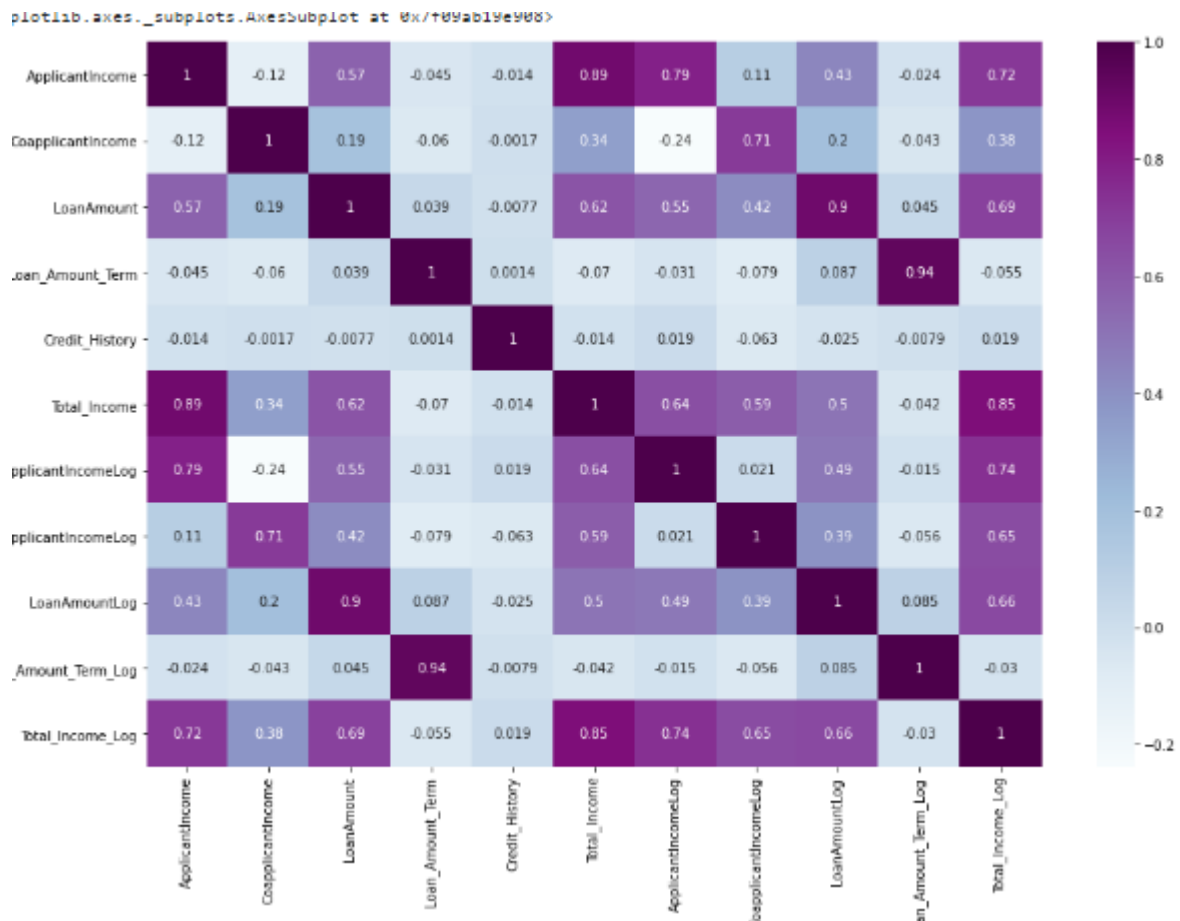


FIGURE 2.9 CORRELATION MATRIX OF THE GIVEN ATTRIBUTE

CHAPTER 3 : FITTING MODELS TO DATA

The classification task mentioned above in machine learning is generally referred to as supervised learning. There is a defined set of classes in supervised learning and example objects are identified with the appropriate class. In this case, the aim is to determine if the customer will be able to repay the loan. Several supervised models will be implemented on the loan repayment dataset in this chapter.

The models include: Logistic Regression, Random Forest, SVM (supporting vector machine), KNN (K nearest neighbours). We will use 10-fold cross validation instead of making validation set to tune hyper parameters and compare various models, since it is a more accurate generalisation error calculation.

Note that not only will the accuracy score be used as the metric to evaluate the model 's success as we solve the classification problem with imbalanced data, but we will also use Precision and Recall as the performance index.

Precision = True Positives/ (True Positives + False Positives)

Recall = True Positives/ (True Positives + False Negatives)

3.1 RANDOM FOREST

Random Forest is an algorithm for supervised learning. It's like an ensemble of bagging process decision trees. The basic principle of the method of bagging is that the final outcome is strengthened by a mixture of learning models. To create multiple decision trees, the Random Forest algorithm randomly selects observations and features and then averages the results. Unlike decision trees, however, for most of the time, random forest avoids overfitting problem, as it generates many random subsets of features and only constructs smaller subtrees. We will use loan repayment data in this part to train a random forest model.

FITTING MODELS TO DATA

Step 1: construct a random forest model by default using Scikit-learn package and conduct a confusion matrix to see how the model performs on the loan repayment dataset.

```
from sklearn.ensemble import RandomForestClassifier, ExtraTreesClassifier
model = RandomForestClassifier()
classify(model, X, y)
```

Accuracy is 80.51948051948052
Cross validation is 78.17806210848993

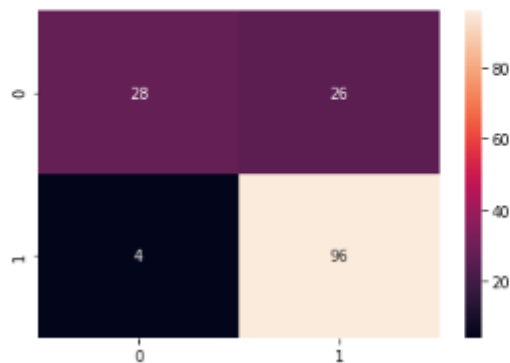


FIGURE 3.1 CONFUSION MATRIX FOR THE RANDOM FOREST CLASSIFIER MODEL

Precision: 0.78688

Recall: 0.9600

Step 2: Use the Randomized Search Cross validation technique to evaluate the optimal parameter combinations.

Firstly, in Scikit-Learn, we describe a grid of hyperparameter ranges based on random forest documentation. Second, for any combination of values, we randomly sample from the grid, and perform K-Fold CV. Parameters include the following:

1. n_estimators = number of trees in the forest
2. max_features = max number of features considered for splitting a node
3. max_depth = max number of levels in each decision tree
4. min_samples_split = min number of data points placed in a node before the node is split
5. min_samples_leaf = min number of data points allowed in a leaf node

FITTING MODELS TO DATA

6. bootstrap = method for sampling data points (with or without replacement)

Step 3: Train another random forest model with the optimal parameters combination we found in step2 and evaluate the model.

Figure 3.2 is the confusion matrix for the optimized model

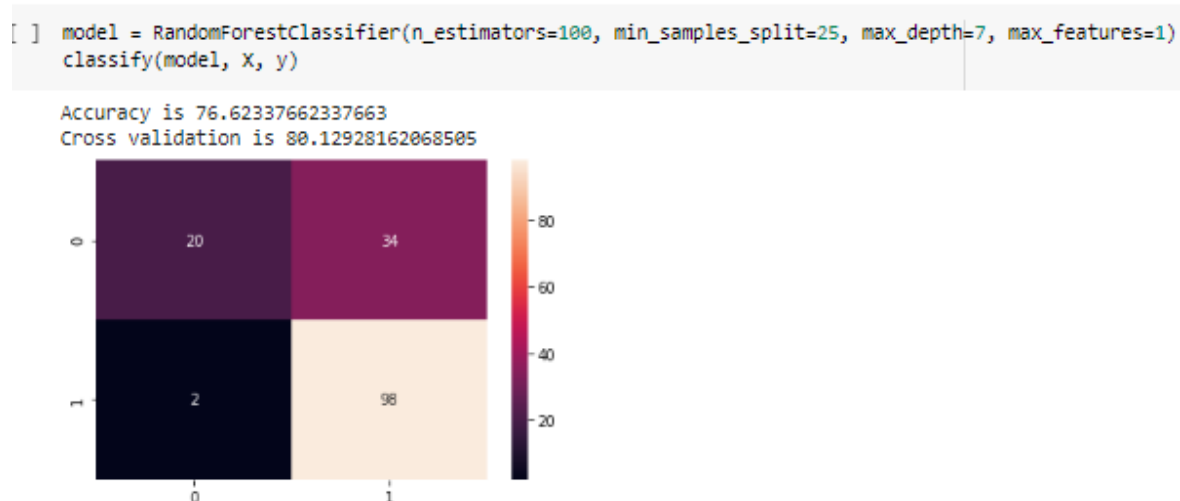


FIGURE 3.2 CONFUSION MATRIX FOR THE OPTIMIZED MODEL OF RANDOM FOREST CLASSIFIER

Precision: 0.74242

Recall: 0.9800

Comparing the confusion matrix, we can find that the optimized random forest model would have higher precision, recall .

3.2 LOGISTIC REGRESSION

Logistic regression is another algorithm that is perfect for performing supervised learning when the binary variable is dependent. It is generally used in the presence of more than one explanatory variable to achieve an odds ratio. Linear regression is somewhat similar to the method, but its response variable is binomial. The effect of each variable on the odds ratio of the observed interest event is the result. The general evidence of logistic regression is below:

FITTING MODELS TO DATA

We will fit the logistic regression into the loan data in this section. Also, we will try to decide the optimal logistic regression parameter. In the last part, the approach is identical to the approach.

Step 1: construct logistic regression model with regularization to avoid overfitting and conduct a confusion matrix to see how the model performs on the loan repayment dataset.

In this case, since it causes the coefficients to be lower, we can use ridge regression here, but it doesn't force them to be zero. That is, it will not get rid of irrelevant characteristics, but rather mitigate their effect on the model being educated. The model will therefore appear to have more power to predict. Below is the Logistic Regression Cost Function with Ridge Penalty:

$$\sum_{i=1}^n \left(Y_i - \sum_{j=1}^p X_{ij} \beta_j \right)^2 + \lambda \sum_{j=1}^p \beta_j^2$$

Figure 3.3 is the default confusion matrix in the logistic regression model. We could calculate on the basis of the matrix to determine the model for the index.



FIGURE 3.3 CONFUSION MATRIX FOR LOGISTIC REGRESSION MODEL

Precision: 0.7205

Recall: 0.9800

Step 2: Now we would test different parameters in order to see how accuracy changes and find the optimized parameter for the ridge regression.

Lambda (λ) monitors the trade-off between bias and variance in ridge regression. In other words, if λ is 0 or close to 0, the model would have ample power to increase its complexity by assigning weights to large values for each parameter, leading to the problem of overfitting. The model will appear to underfit if we increase the value of λ , as the model will become too basic. We use parameter C as our regularisation parameter in this case. (Where $1 / \lambda = C$)

Figure 3.4 is the validation curve which indicates how each C value affects the accuracy of the training set and the testing set. In this case, the parameter C is not as important as we expected. As we observe from the validation curve, the change of C value won't affect the training and testing accuracy so much. Besides, the accuracy of training data and the accuracy of the testing data are very close no matter what value C chooses. The difference is only within 0.1 percent. Therefore, we will just consider the default model as the optimised Logistic regression model for loan repayment prediction. Table 3.1 is the coefficient for the Logistic Regression with L2 penalty.

3.3 SUPPORT VECTOR MACHINE

A Support Vector Machine (SVM) is also a supervised learning algorithm which is used to separating hyperplane. In other words, given labelled training data, the algorithm outputs an optimal hyperplane which classifies new examples. In two-dimensional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side. In multidimensional space, the separation of the class is a hyperplane.

In this section, we will fit SVM model into the loan data. Also, we will try to determine the optimal parameter of the model and evaluate the SVM model. The method is similar with the method that we use for the random forest model.

FITTING MODELS TO DATA

One thing we need to notice is that we are using RBF kernel for the SVM model in this case because, unlike linear kernel, it can handle the situation when the non-linear relationship between the class labels and attributes. In addition, contrast to poly kernel, the number of hyperparameters in RBF kernel is easier to control which reduces the model complexity.

To optimize the SVC model, we are using Grid-Search Cross validation. There are parameters for RBF kernel: C and γ

C: it is like a regularization parameter, which tells the SVM optimization how much you want to avoid misclassifying each training example. For example, large values of C will result the optimization choosing a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly.

Gamma (γ): tells how far the influence of a single training example would reach. High 17 Gamma only consider the nearby points of the separation line; and low Gamma would also consider the far away points of the separation line.

Table 3.2 is the result of Grid-Search Cross Validation, the table shows the optimal combination of the parameters for SVM model.

Parameter	Value
C	10
γ	0.01

TABLE 3.1 BEST PARAMETER FOR OPTIMIZED SVM

This is the confusion matrix parameter for SVM model by default.

Precision = 0.7609

Recall = 0.9535

FITTING MODELS TO DATA

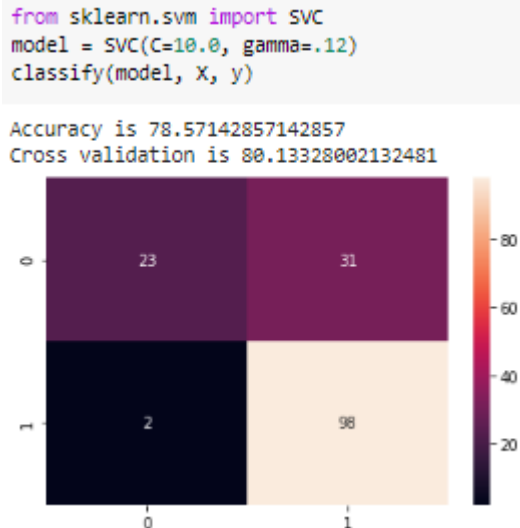


FIGURE 3.4 CONFUSION MATRIX FOR THE OPTIMIZED SVM MODEL.

Precision = 0.7597

Recall = 0.9800

As we observed from the result of two models, we noticed that the precision rate drops in the exchange of the increase of recall. In this case, since the cost that we miss classify ineligible loaner is much higher than the cost we miss classify the eligible loaner. Therefore, we would prefer the model with higher recall score.

3.4 K-Nearest-Neighbors Model

The k-nearest neighbours algorithm (KNN) is a non-parametric method that can be used for classification and regression problems. In classification problems, an object is classified by a vote of its neighbors, with the object being assigned to the class most common among its k 18 nearest neighbours.

The prediction accuracy based on the k-NN model is highly contingent on the value of K. The best choice of K depends upon the data. Usually, larger values of K would reduce the effect of the noise on the classification but make boundaries between each category less distinct. Smaller value of K would reduce the error rate for the training sample, but it would cause the overfitting problem. In this section, we attempt to identify the optimal K value (from 1 to 30) and evaluate the model.

FITTING MODELS TO DATA

this is the Error rate vs K value for the loan repayment data set. as we observe from the plot, we may notice that when $K = 1$, the error rate tends to be zero. As K value increases, the error rate would experience a significant increase first, and then it tends to have a slight drop and becomes relatively stable when $K > 10$.

Figure 3.5 is validation curve vs K value for the data set. It indicates that as K increases, the train accuracy will decrease, and the testing accuracy will increase. When $K > 10$, the training and testing accuracy tends to be very close, and they are converging as K increases. Therefore, in this case we would choose $K = 10$ to train the KNN model.



FIGURE 3.5: CONFUSION MATRIX FOR THE K-NEAREST-NEIGHBORS MODEL WHEN $K = 10$.

Precision: 0.717

Recall: 0.940

3.5 Feature Importance Analysis

Methods of feature selection are a way of reducing the dimensions efficiently without much loss of total knowledge. It also helps to make sense of the features and its importance. In this section, by constructing a gradient boosting model, we will check the importance of the function.

FITTING MODELS TO DATA

The partial dependency plots are shown in Figure 3.5 to see what the most significant characteristics are and their associations with whether the borrower is most likely to pay the loan in full before mature data. Notice that, to make it easier to read, only the top 8 features were plotted. We found that fico score greatly affects the status of whether the loaner will be able to repay the loan, according to Figure 3.5. Borrowers with higher fico scores are far less likely to fail to repay the loan in full.

"The partial dependence plot shows that" log.annual.inc," "instalment," "credit.policy," "inq.last.6mths," "revol.bal "are essential features consistent with the results of the logistic regression model (Table 3.1). The partial dependence plot shows that" log.annual.inc," "instalment," "credit.policy," "inq.last.6mths," "revol.bal The "int.rate" is the element that tends to be more significant than that in the logistic regression in the gradient boosting model. This happened because the borrowers with higher interest rates are considered riskier, which are less likely to repay the loan.

3.6 Model Comparison

In this segment, you will evaluate and compare the results of the selected models. As we observed from the ROC curve map, we can note that the Logistic Regression, Random Forest, and SVM models' ROC curves are relatively similar to the plot's left-top than the KNN model's roc curve. And we can remember that the SVM model has the highest precision of 78.93 and the highest recall rate of 0.98 on the basis of the results of previous research.

Furthermore, the SVM Model has the highest accuracy score of 78.407 and the highest accuracy rate of 0.7411. In this scenario, the cost of misclassifying the loaner who is unable to repay is much higher, so we would prefer to use the model which would result in a higher recall rate. Thus, the svm model is favoured among the selected models in this study.

	MLA Name	MLA Test Accuracy	MLA Cross Val Score
1	Random Forest Classifier	76.623	79.480
2	Logistic Regression	77.272	80.646
3	SVC	78.935	80.994
4	K-Neighbours Classifier	72.077	75.079

TABLE 3.2 MODEL COMPARISON TABLE

3.7 Model deploy in graphical User interface

After complete build a model now we moved to building a **graphical user interface builder** (or **GUI builder**), also known as **GUI designer**, is a software development tool that simplifies the creation of GUIs by allowing the designer to arrange graphical control elements. Our GUI which contains an interface to take input data from the user and then make predictions on that data using machine learning algorithm and display the relevant outputs from the model.

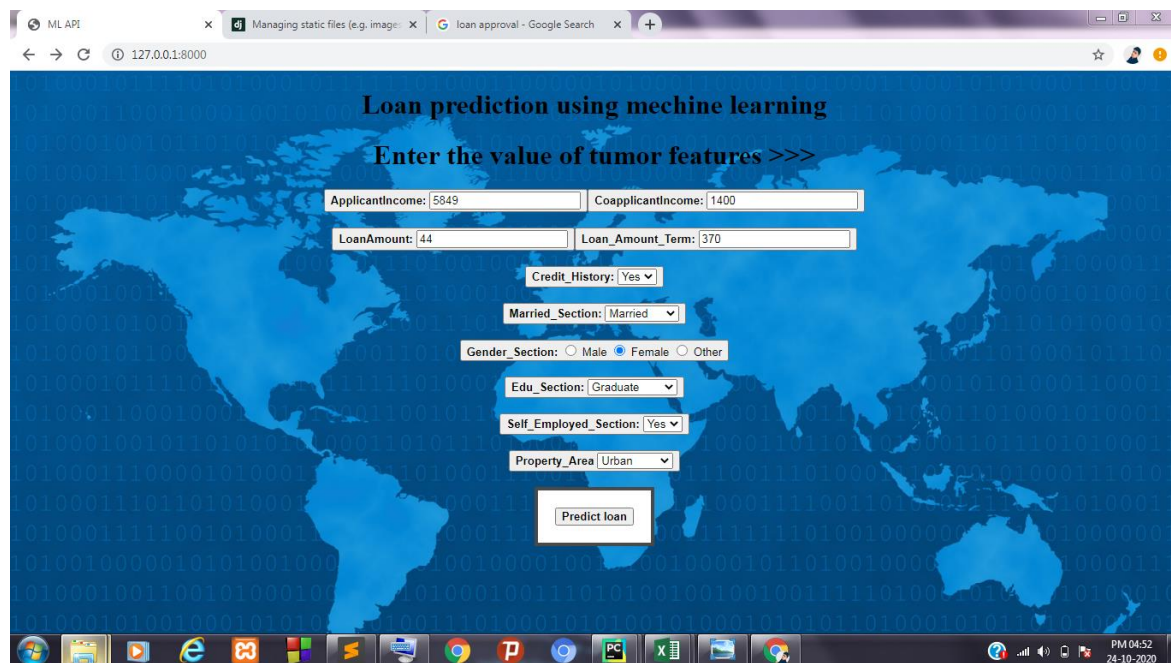


FIGURE 3.6 GRAPHICAL USER INTERFACE OF DEPLOY MODEL

CHAPTER 4 : ENHANCEMENT AND CONCLUSION

4.1 Future enhancement

There are some developments in this research that we might make in the future. Outlier problems, for example, are not included in the study of exploratory results. If there are outliers in the sample, it will not be as accurate as the effects of the predictive model.

In addition, the deep learning algorithm approach should also be applied when the repayment status is expected for the loan. In addition, we would have more training samples if we were to have a larger dataset. Therefore, it could help fix the issue of high variance and make our research more relevant.

4.2 Conclusion

The loan business is becoming more and more common nowadays, and many individuals apply for loans for different reasons. There are, however, occasions where individuals do not repay the bulk of the loan sum to the bank, resulting in a massive financial loss. Therefore, if there is a way that can accurately identify the loaners in advance, the financial loss will be greatly avoided.

In this study, the dataset was cleaned first, and the exploratory data analysis and feature engineering were performed and model deploy in GUI. It covered the techniques for coping with both missing values and imbalanced data sets. We then suggest four machine learning models, which are Random Forest, Logistic Regression, Support Vector Machine, and K-Nearest Neighbours, to predict whether the borrower will repay the loan. The two Randomized Search Cross Validation methods are implemented in various circumstances when tuning parameters. By way of trials, it is noticed that the model was found which best fits the dataset with highest accuracy is the SVM model,

CHAPTER 5 : REFERENCES

5.1 REFERENCE

1. Chen, N., Ribeiro, B., & Chen, A. (2016). Financial credit risk assessment: a recent review. *Artificial Intelligence Review*, 45(1), 1-23.
2. Trustorff, J., Konrad, P., & Leker, J. (2011). Credit risk prediction using support vector machines. *Review of Quantitative Finance and Accounting*, 36(4), 565-581.
3. Zięba, M., Tomczak, S., & Tomczak, J. (2016). Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction. *Expert Systems with Applications*, 58, 93-101.
4. Galinndo, J., & Pablo T. (2000). Credit Risk Assessment using Statistical and Machine Learning: Basic Methodology and Risk and Risk Modelling Applications. *Computational Economics* 15: 107-43.
5. Weidong Cai,Zhihua Zhou,Xing Li,” Research on Support Vector Machine”,*Computer Engineering and Applications*,pp.58-61,Jan.2001.
6. Datahack analyticsvidhya contest practice “problem-loan-prediction” .
7. Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. “A practical guide to support vector classification.” 2003.
8. Alvira Swalin. “How to handle missing value.” *Towards Data Science*, 2018.
9. Hongri. “Jia Bank Loan Default Prediction with Machine Learning.” pp. 137–163, Apr 10, 2018.
10. Sebastian Raschka. “About feature scaling and normalization.” Sebastian Racha. *Disques*, nd Web. Dec, 2014.

REFERENCES

11. Chao-Ying Joanne Peng, Kuk Lida Lee, and Gary M Ingersoll. "An introduction to logistic regression analysis and reporting." *The journal of educational research*, 96(1):3–14, 2002.
12. Hongri. "Jia Bank Loan Default Prediction with Machine Learning." pp. 137–163, Apr 10, 2018.
13. Niklas Donges. "The Random Forest Algorithm." *Statistical Methods*, Feb 2018.
14. Ofir Chakon. "PRACTICAL MACHINE LEARNING: RIDGE REGRESSION VS.LASSO." August 2017.
15. Abhishek Bhagat et al. Predicting Loan Defaults using Machine Learning Techniques. PhD thesis, California State University, Northridge, 2018.
16. Sudharsan Asaithambi. "Why, How and When to apply Feature Selection." Jan 2018.
17. Naomi S Altman. "An introduction to kernel and nearest-neighbor nonparametric regression." *The American Statistician*, 46(3):175–185, 1992.
18. Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. "A practical guide to support vector classification." 2003.
19. Rattle data mining tool: available from <http://rattle.togaware.com/rattle-download.html>
20. Aafer Y, Du W &Yin H 2013, DroidAPIMiner: 'Mining API-Level Features for Robust Malware Detection in Android', in *Security and privacy in Communication Networks* Springer, pp 86-103 .
21. Ekta Gandotra, Divya Bansal, Sanjeev Sofat 2014, 'Malware Analysis and Classification: A Survey'available from [http:// www.scirp.org/journal/ji](http://www.scirp.org/journal/ji)

REFERENCES

22. Andy Liaw and Matthew Wiener. Classification and Regression by randomForest. R News(<http://CRAN.R-project.org/doc/Rnews>. 2(3):9–22, 2002.
23. S.S. Keerthi and E.G. Gilbert. Convergence of a generalizeSMO algorithm for SVM classifier design. Machine Learning, Springer, 46(1):351–360, 2002.
24. J.M. Chambers. Computational methods for data analysis. Applied Statistics, Wiley, 1(2):1–10, 1977.