Table of Contents

Abstract:

Credit card fraud is causing serious problems in the finance industry. Every year, a large sum of amount is lost by banks because of undetected credit card fraud. Fraud detection has become an effective tool in reducing this issue, and it is likely the most effective way to prevent these frauds. In this study, new credit card applicants are categorized into “good” and “bad” customers from their previous loan repayment status (if available). Since, typically in such scenarios, the data is highly imbalanced, a number of data cleansing and pre-processing techniques have been employed along with various supervised machine learning models to obtain the best results possible. After testing out the available data on Logistic Regression, KNN, Decision Tree and Random Forest algorithms, the Random Forest model showed the best results. Final test-accuracy of 85.8% is achieved by the predictive model.

Introduction and Literature review:

An organization typically loses 4% of its revenue to frauds each year. According to an RTI query, 2480 Frauds involving a total amount of 7 million USD rattled 18 Public Sector banks in a single quarter. Thus, fraud possess a serious concern for all organizations. Fraud can be considered an act of deception that is used to illegally deprive another individual or an organization of money or property.

There are four fundamental stages to construct an efficient and useful Machine Learning model. First step is to select and set up a preparation training dataset. Supervised learning training data with labels which is then used to train the machine so that when the machine encounters a new set of data, it can efficiently produce a correct outcome (based on its learning from training data).

This stage is followed by choosing an algorithm to run on the collected training dataset. The kind of algorithm relies upon the sort (labelled or unlabelled) and measure of data in the preparation of training dataset and on the kind of issue to be addressed.

Some common machine learning algorithms widely used in the past include:

•**Regression algorithms**: A linear regression is one of the most fundamental algorithms in the Machine learning. It is an algorithm that tries to find a straight line that will fits through all the observation points. The algorithm can use a standard error, for example, least square error method to find the best fitting line.

Applications of regression include forecasting trends and sales estimates, analysing the impact on price changes, and assessment of risk in financial services and insurance domain.

•**Decision trees**: Decision tree is a type of classification algorithm which comes under the supervised learning technique. It is a graphical representation of all the possible solution to a decision.

Decision tree is really very easy to read and understand. It belongs to one of the few models that are interpretable where we can understand exactly why the classifier has made a particular decision.

A decision tree is a classical representation of all the possible solutions to a decision based on certain conditions. It is called a decision tree because it starts with a root and then branches off to a number of solutions just like a tree. It has a root which keeps on growing with increasing number of decision and the conditions.

The third stage comprises of training the model on the selected algorithm. Most machine learning algorithms are iterative in nature. The model takes input as one data point at a time and keeps on updating its parameters. The model is ready after it has processed each data point in the training dataset.

The final stage consists of using and improving the model using hyperparameter tuning. Once the model is trained using the training data, few parameters in the model (for example, value of k in KNN algorithm, number of estimators in Random Forest algorithm, etc.) can be tweaked in a way to improve the model's performance. This is generally a hit and trial method, usually carried out using in-built functions like GridSearchCV and RandomSearchCV.

A.Shen (2007)[1] The authors evaluated the influence of different classifiers in identifying credit card fraud and proposed three classification models: decision tree, neural network, and logistic regression.

Y. Sahin and E. Duman(2011)[2] has employed several classification techniques and listed the study for credit card fraud detection. To mitigate the likelihood of the banks, they used decision trees and SVMs in this report. They stated how Artificial Neural Networks and Logistic Regression classification models are more effective for detecting fraud.

Y. Sahin, E. Duman(2011)[3] has cited the research , Artificial Neural Network(ANN) classifiers perform better than Logistic Regression (LR) classifiers in solving the particular investigation, according to authors who used ANN and LR. Here, the allocation of the training data sets became more skewed, and the performance of all models in identifying fraud transactions declined.

Kokkinaki [4] suggested to build a user profile for each credit card account and to test incoming transactions against the corresponding user's profile. Credit card information, transaction dates, type of organization, place, expenses incurred, margin requirement, and expiration time were used to construct these profiles. To capture a user's behaviours, researchers proposed a Similarity Tree algorithm, which is a variation of Decision Trees. According to the research, the method has a very low likelihood of achieving false negative errors. However, since the user profiles used in this methodology are often not reactive, they must be updated on a constant schedule as the needs of users and fraudulent trends shift.

Chan and Stolfo [5] On the credit card fraud system, authors evaluated the class distribution of a test data and its influence on the performance of multi-classifiers. It was discovered that increasing the number of minority samples in the training process resulted in fewer losses to suspicious purchases. Moreover, the fraudulent percentage used in training was ranged from 10% to 90%, and it was revealed that the maximum benefits were discovered whenever the fraudulent rate used in training was 50%.

Brause and others [6] Integrating a rule-based classification approach with a neural network algorithm, researchers looked at credit card payment fraud and identified fraudulent activities. In this method, a rule-based classifier first tests to see whether a payment is illegitimate, and then a neural network validates the transaction classification. This method exploits the likelihood of a successful "fraud" detection, reducing the estimation error whilst increasing trust.

Problem Statement:

The main objective of this study is to use personal information along with data provided by credit card applicants to predict the likelihood of future defaults and credit card borrowings. Thus, banks can use the findings of this project as an important factor in deciding whether to issue a credit card to the applicant or not. The data available to solve this task includes features such as client number/ID of the applicant along with their gender, number of children, annual income, education level, marital status, age, employment status, occupation, family size etc. Another dataset contains the loan repayment status for various applicant IDs over the past several months. Applicants have been categorised into numerous groups which are explained in the methodology section of the report. This data has been downloaded from https://www.kaggle.com/rikdifos/credit-card-approval-prediction.

Methodology:

I. Conversion of categorical data into numerical data:

Categorical features may only have a certain number of possible values, which is normally set. In this study, for example, categorical features like gender, occupation, etc. are present. Typically, these features are stored as text values that reflect various characteristics of the observations. All of the machine learning algorithms are algebraic. This necessitates the use of numerical data in their input. To use these models, all such categories are converted into numerical data before applying the learning algorithm to them. In this project, Sci-Kit Learn module's built-in "LabelEncoder" function is used for this task.

II. Outlier detection and removal

A datapoint can be considered as an outlier if it significantly deviates from the rest of the datapoints. Outliers are generally introduced in the dataset because of incorrect data entry or misreporting of information during data collection. Outliers are extremely sensitive to most parametric statistics, such as means, standard deviations, and correlations, as well as any statistic dependent on them. Outliers can really mess up an analysis because the assumptions of standard statistical techniques like linear regression and Decision Tree are often based on these statistics. A simple way to detect outliers is by visualizing the given data. The following figure (Figure 1) represents a small group of few points that deviate from the rest of the data

under different columns namely, number of children, income of the applicant and the number of family members.

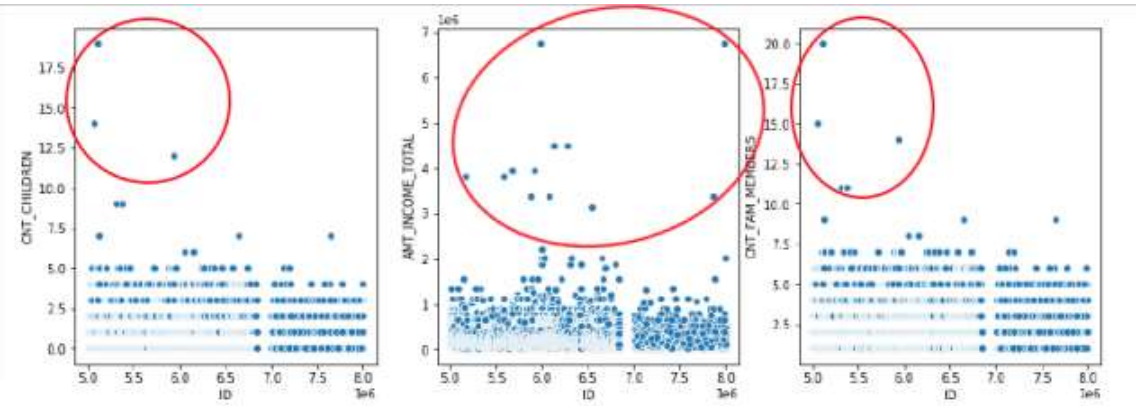

Figure 1

Since the outliers in this case are present at one extreme end, we only consider the values from 0 to 99.9 percentile i.e., in other words, we disregard the extreme 0.1% values and categorise them as outliers.

Following figure (Figure 2) represents the distribution in the above-mentioned columns after outliers have been removed.

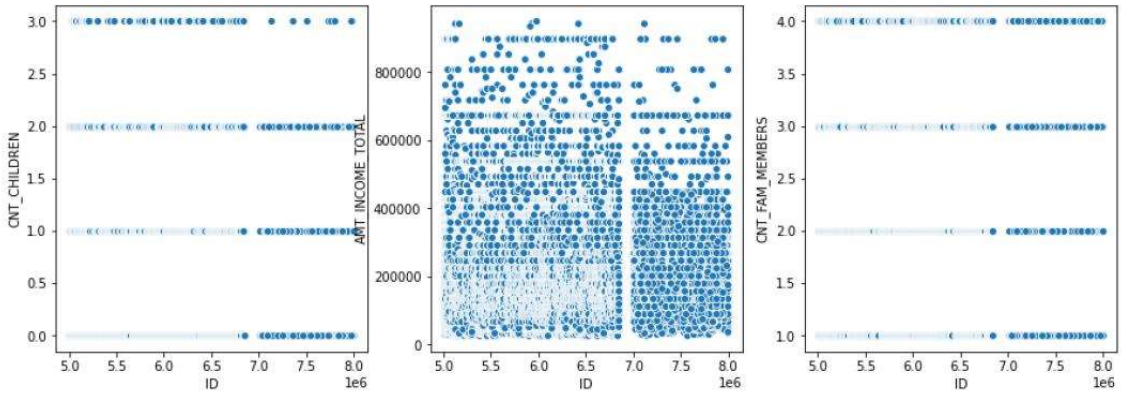

Figure 2

III. Normalization:

Rescaling real-valued numeric attributes into a 0 to 1 range is referred to as Normalisation. In machine learning, data normalisation is used to make model training less responsive to feature size. As a result, our model will converge to better weights, resulting in a more accurate model.

In this study, normalization has been applied on continuous variables like total income, age and work experience of the applicant. The "MinMaxScaler" scaler operation, which is built-in the SciKit Learn module is used to normalize the data according to the following formula:

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

IV. Oversampling using SMOTE:

After the data is pre-processed, the final dataset contains 31730 datapoints for "good" classified customers whereas only 4226 datapoints for "bad" classified customers. The problem with working on such an imbalance dataset is that the classification is likely to perform poorly on minority class although we need it to perform better especially on the minority class ("bad" applicant class). A simple way to overcome this problem is oversampling of the minority class. This can be done by adding duplicate examples to the smaller class in the training data before feeding it to the machine learning model.

A better and a more effective way to oversample data is to create new examples from the available data of the minority class which are highly similar in nature to the original data. This new synthesized data can now be added to the training dataset to balance it.

SMOTE functions based on K- Nearest Neighbour, the number should be specified by the programmer. SMOTE identified the K number of neighbour data points for all the datapoints in the minority class. It creates a line with the K-nearest neighbour forming a complex polygon with all the datapoints. Based on the value of the additional datapoint required to match with the majority class, SMOTE creates additional data points on the lines of the polygon artificially. Once this is done, the variation between both classes is removed. Thus, SMOTE oversampling has been carried out on the training dataset to equalize the number of "good" applicant data points and "bad" applicant data points.

Logistic Regression:

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable. Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications.

Logistic Regression produces results in a binary format which is used to predict the outcome of a categorical dependent variable. So, the outcome should be discrete/categorical. Such as: 0 or 1; Yes or no; True or false.

$$P = \frac{e^{a+bX}}{1 + e^{a+bX}}$$

Where, P is the probability of a 1, and a and b are parameters of the model.

Following figure (Figure 3) represents the difference between Linear Regression and Logistic Regression.

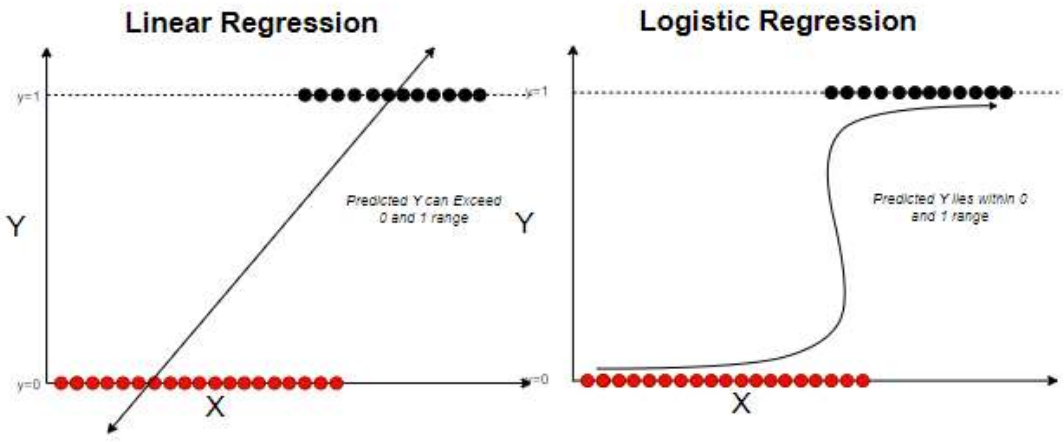

Figure 3

KNN Algorithm:

Unlabelled findings are classified using the kNN classifier by assigning them to the class of the most related labelled instances. For both the training and evaluation datasets, observational characteristics are collected. KNN is based on the assumption that any data point that is next to another belongs to the same class. In other words, it uses similarity to classify a new data point. Figure 4 represents the basic working principle of KNN algorithm.

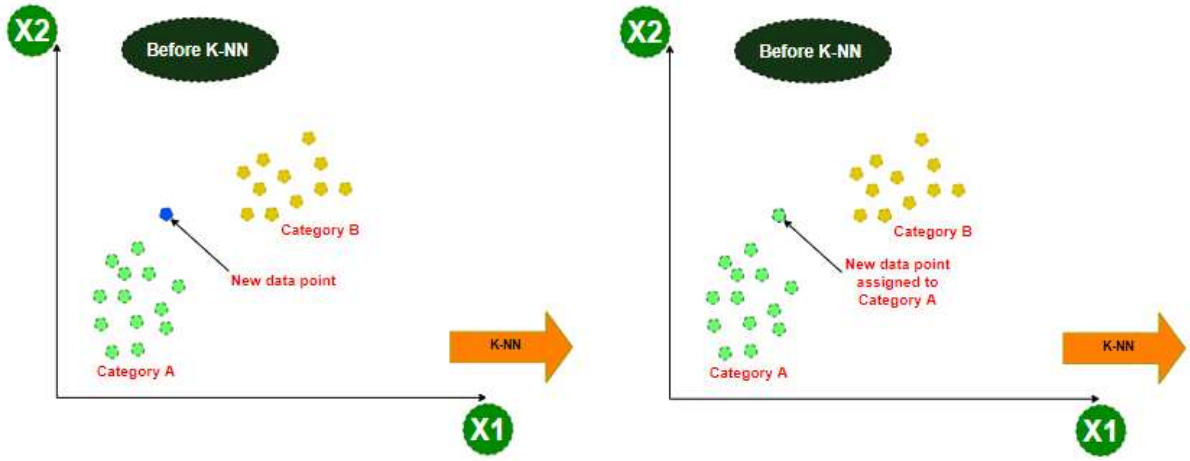

Figure 4

Decision Tree:

A basic Machine Learning model used in classification problems is the Decision Tree Classifier. It is one of the most basic Machine Learning models used in classifications, but when done correctly and with sufficient training data, it can be extremely successful in solving certain tasks.

Decision Trees Classifiers are a form of Supervised Machine Learning in which we create a model, feed it with training data that is matched to correct outputs, and then let the model learn from these patterns. Then we feed new data to our model that it hasn't seen before to see how it responds. Often, we need to see exactly what needs to be trained for a Decision Tree.

Decision Trees classifiers are used in classification tasks where the dataset is small, and a simplified model can be used. This classifier can also be used when only a few features are available or when a model needs to be visualised and clarified in simple terms. Figure 5 demonstrates the basic working principle of decision tree algorithm.

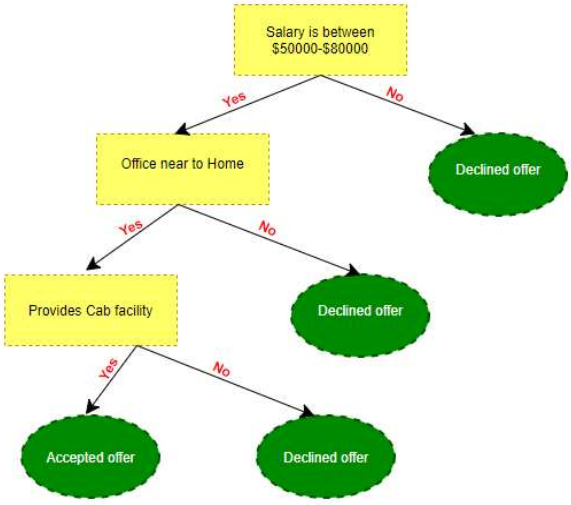

Figure 5

Selection of root node in decision tree algorithm:

The following expressions are used to calculate the entropy and information gain for a feature variable. After computing the information gain for all the features at all possible values, the feature with the highest value of information gain is used to split the dataset into two branches. This is an iterative process which continues till the data is divided into minimum number data points (which is generally specified by the user depending upon the desired results).

Entropy

Entropy measures the impurity or uncertainty present in the data.

$$H(S) = -\sum_{i=1}^{N} p_i \log_2 p_i$$

Where:

S - set of all instances in the dataset

N – number of distinct class values

Pi – event probability

Information Gain (IG)

IG indicates how much "information" a particular feature/variable gives us about the outcome.

$\text{Gain}(A, S) = H(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} . H(S_j) = H(S) - H(A, S)$ Where:

H (S) - entropy of the whole dataset S

|Sj| - number of instances with j value of an attribute A

|S| - total number of instances in dataset S

v - set of distinct values of an attribute A

H(Sj) - entropy of subset of instances for attribute A

H(A,S) - entropy of an attribute A

Random Forest Classifier:

Random Forest Classifier is the best suited type of machine learning used in regression and classification. The classification functioning is similar to that of a decision tree method.

When we provide a train dataset to the random forest classifier, the datasets are separated into N- number of sample datasets (models), where-in the random forest classifier trains creates a decision trees in which there are root logic and sub dataset classified based on various attributes that we specify.

If we provide 100 train set datapoint and 10 estimators (No. of decision trees), the 100 points will be randomly separated to 10 samples with each containing 10 datapoints each. The RF classifier analyses this samples separately by creating a decision trees using the sample dataset and attributes of the groups and identifies which results in true logic and which results in false logic.

Finally, voting is done based on the overall comparison of all the algorithms that ran under the single Forest Classifier to finalise the classification of the test dataset. The accuracy of the test dataset output can be varied by varying the number of estimators; however, the user identifies the threshold accuracy percentage.

The main difference between RF classifier and Decision tree methods is that decision trees are less accurate and it usually overfits.

Numerical results

Numerous performance parameters such as accuracy, precision, recall, F1-score can be used to evaluate the performance of our model. Only 11.8% (1193 records of the total 10,068) records were data points for "Bad" customers in the test dataset. This clearly demonstrates that the available data for testing of the model is highly imbalanced. Accuracy for classification problem is given by the following formula:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

Classification accuracy is a widely used metric for evaluating performance of classification models as it is easy to interpret and calculate and is a single number to quantify the model's performance.

However, for unbalanced datasets, accuracy can become an unreliable performance metric. High values of accuracy (e.g 90%,95% or even 99%) can become trivial for such problems depending upon the degree of imbalance. To understand why this is the case, we consider a simple example dataset with 1:99 class imbalance, (i.e for every 100 data points, 99 belong to class 0 and 1 belongs to class 1). In such a case, if we construct a model that predicts the output to be 0 for all test data, we can end up with an accuracy close to 99% (because of the high degree of imbalance, 99/100 predictions will be true). However, such a model will not be of any use since we cannot predict any instance of class 1 examples. Thus, for highly imbalanced datasets, accuracy is not always the most important performance metric.

To overcome this problem, confusion matrix can be used to depict a more conclusive and interpretable representation of a machine learning classification model. A **confusion matrix** is a correlation between the predictions of a model and the actual class labels of the data points and is represented as follows.

Confusion Matrix:

	Actual Positive (1)	**Actual Negative (0)**
Predicted Positive (1)	**True Positives (TPs)**	**False Positives (FPs)**
Predicted Negative (0)	**False Negatives (FNs)**	**True Negatives (TNs)**

The primary focus of this study is to accurately predict the applicants classified as "bad" according to their loan repayment history. Hence, the number of true positives in the confusion matrix obtained after running the model on the test data should be of prime significance. Following metrics can be derived from the confusion matrix to quantify the observations.

$$\text{Precision} = \frac{\text{tp}}{\text{tp} + \text{fp}}$$

$$\text{Recall} = \frac{\text{tp}}{\text{tp} + \text{fn}}$$

F- Measure: Harmonic mean of precision and recall.

$$\text{F} = 2 . \frac{\text{precision} . \text{recall}}{\text{precision} + \text{recall}}$$

Thus, precision, recall and f-score can be considered as excellent performance parameters as they are directly related to the true positive value in the confusion matrix. This helps in evaluating model performances based on their ability to detect "bad" classified applicants.

Following table represents the comparison of models working on different algorithms judged by the above-mentioned metrics.

	Model Algorithm	**Recall**	**Precision**	**Accuracy**	**F1**
1	**Logistic Regression**	0.517	0.13	54.41	0.207
2	**KNN**	0.368	0.405	86.11	0.385
3	**Decision Tree**	0.41	0.387	85.31	0.398
4	**Random Forest**	0.432	0.4	85.54	0.42

The model working on the Random Forest algorithm displayed the best results on the test dataset.

Next, a correlation matrix is plotted which demonstrates the relationship between the input variables. Features with high positive or negative correlation to each other demonstrate that they are highly dependent on each other and adversely affect the performance of the model (especially models working on Logistic Regression algorithm).

From the correlation matrix we can see that the number of children of an applicant is highly correlated to the number of family members of the applicant. Thus, we can safely drop one of those features and re-run the model with the same parameters to check any improvement in the result.

From the new confusion matrix, it is evident that there is a slight improvement in the model. The number of correct positive (true positive) predictions have increased from 516 to 522. Also, the false negative values have dropped from 677 to 673.

Finally, we check the feature importance of different input features in predicting the output. This is done by using the in-built SciKit Learn module command (model.feature_importance_)

It is observed that the applicant's annual income, age and work experience play a major role in categorising them as "good" or "bad" applicants.

Conclusion

Credit card fraud identification is a major issue in the financial services industry. With the rise of e-commerce, the amount of money lost due to credit card fraud is rising. This project focuses on methods for detecting credit card fraud. In this report, Machine learning techniques like Logistic regression, Decision Tree and Random forest were used to detect fraud in credit card system. Accuracy, Precision of various Model algorithms are used to evaluate the performance for the proposed system. The accuracy for logistic regression, KNN, Decision tree and Random forest classifier are 54.41, 86.11, 85.31, 85.54 respectively. By comparing all the methods based on performance metrics such as precision, recall and f-score, we found that Random Forest algorithm is the best suitable algorithm for this problem

References

[1] A. Shen, R. Tong, Y. Deng, "Application of classification models on credit card fraud detection", Service Systems and Service Management 2007 International Conference, pp. 1-4, 2007.

[2] Y. Sahin, E. Duman, "Detecting Credit Card Fraud by Decision Trees and Support Vector Machines", Proceedings of International Multi-Conference of Engineers and Computer Scientists (IMECS 2011), vol. 1, pp. 1-6, Mar. 16-18 2011, ISSN 2078-0966, ISBN 978-988-18210-3-4.

[3] Y. Sahin, E. Duman, "Detecting credit card fraud by ANN and logistic regression", Innovations in Intelligent Systems and Applications (INISTA) 2011 International Symposium, pp. 315-319, 2011.

[4] Kokkinaki, A. 1997. On Atypical Database Transactions: Identification of Probable Frauds using Machine Learning for User Profiling, Proc. of IEEE Knowledge and Data Engineering Exchange Workshop; 107-113.

[5] P. K. Chan, W. Fan, A. L. Prodromidis and S. J. Stolfo, "Distributed data mining in credit card fraud detection," in IEEE Intelligent Systems and their Applications, vol. 14, no. 6, pp. 67-74, Nov.-Dec. 1999.

[6] R. Brause, T. Langsdorf, M. Hepp,Neural Data Mining for Credit Card Fraud Detection, November 1999 ICTAI '99: Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence.