NOVEL MODELS FOR CREDIT CARD FRAUD DETECTION

by
Yiğit Kültür
B.S., Computer Engineering, Middle East Technical University, 2006
M.S., Computer Engineering, Boğaziçi University, 2008

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

Graduate Program in Computer Engineering
Boğaziçi University
2017

NOVEL MODELS FOR CREDIT CARD FRAUD DETECTION

APPROVED BY:

Prof. Mehmet Ufuk Çağlayan ………………
(Thesis Supervisor)

Prof. Fatih Alagöz ………………

Assoc. Prof. Taylan Cemgil ………………

Prof. Ekrem Duman ………………

Prof. Albert Levi ………………

DATE OF APPROVAL: 24.05.2017

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Professor Mehmet Ufuk Çağlayan for the continuous support of my Ph.D. research, for his patience, motivation and immense knowledge. His guidance helped me in all the time of research and writing of this thesis.

I would also like to thank Professor Fatih Alagöz, Associate Professor Taylan Cemgil, Professor Ekrem Duman and Professor Albert Levi for kindly accepting to be in my thesis jury.

I am also grateful to the members of Computer Networks Research Laboratory (NetLab) for their friendship.

I would like to express my sincere gratitude to Boğaziçi University Computer Engineering faculty for strengthening my knowledge base with brilliant graduate courses.

My elementary school teacher, Mrs. Fahriye Sümer, deserves special thanks for teaching me the fundamentals of mathematics and logic. If I had not started my education life as her student, it would have been impossible for me to achieve this goal.

My family deserves the biggest thanks for their patience and support in overcoming numerous obstacles I have been facing through my research.

I would like to thank my employers and colleagues in Banksoft for their support and friendship throughout my academic studies.

Finally, I would like to thank Turkish State Planning Organization (DPT) for supporting me under the TAM Project, number 2007K120610.

ABSTRACT

NOVEL MODELS FOR CREDIT CARD FRAUD DETECTION

Financial institutions attach great importance to credit card fraud detection, as a natural consequence of the multi-billion dollar annual losses incurred due to credit card fraud. Rule based systems have been commonly used by financial institutions to detect credit card fraud. The rules applied in such systems are formulated based on the experience of fraud experts and the results of fraud investigations. Rule discovery is a manual process, and this fact is an important disadvantage of rule-based systems. Unlike rule-based systems, artificial intelligence models are expected to learn from past transaction data and consequently no manual process is necessary. Many researchers in the domain of credit card fraud detection have recognized this advantage offered by artificial intelligence models. In this thesis, we propose novel artificial intelligence based models for detecting credit card fraud. First, we propose Cardholder Behavior Model (CBM). CBM is an unsupervised model and uses clustering transaction amounts to represent the spending behavior of cardholders. We propose four focal points to fine-tune CBM, which are single-card versus multi-card focus, holiday season spending focus, time of day focus and inflation focus. The second model we propose is called Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models (OPWEM). OPWEM is an ensemble of six well known artificial intelligence techniques, namely Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine, and K*. We propose optimistic, pessimistic and weighted voting strategies in OPWEM for better detection of credit card fraud. The third model we propose is called Spending Behavior Similarity Model (SBSM). SBSM uses spending behavior similarity measures in order to improve the performance of supervised models. A dataset of real-life credit card transactions from a leading bank in Turkey has been used to evaluate the performance of three proposed models. Finally, we provide a comparative evaluation of three proposed models.

ÖZET

KREDİ KARTI SAHTEKARLIKLARININ TESPİTİ İÇİN YENİ MODEL ÖNERİLERİ

Kredi kartı sahtekarlıkları her yıl milyarlarca dolar zarara neden olmaktadır. Bunun doğal bir sonucu olarak finansal kurumlar kredi kartı sahtekarlık tespitine büyük önem vermektedirler. Finansal kurumlar kredi kartı sahtekarlık tespiti için genelde kural bazlı sistemler kullanmaktadırlar. Bu tip sistemlerde tanımlanan kurallar sahtekarlık uzmanlarının geçmiş tecrübelerinden ve sahtekarlık soruşturma sonuçlarından yararlanarak oluşturulur. Kural tanımlama manuel bir süreçtir ve bu durum önemli bir dezavantaj oluşturur. Yapay zeka modellerinde ise manuel bir süreç ihtiyacı bulunmamaktadır. Kredi kartı sahtekarlık tespiti alanında çalışan çok sayıda araştırmacı yapay zeka modellerinin sunduğu bu avantajın farkına varmıştır. Bu tezde, kredi kartı sahtekarlıklarının tespiti için yapay zeka bazlı yeni modeller öneriyoruz. İlk olarak, Kart Kullanıcısı Davranış Modeli'ni (KDM) öneriyoruz. KDM kart kullanıcısının davranış alışkanlıklarını modellemek için kredi kartı işlem tutarlarını kümeleme metodunu kullanır. KDM eğitilirken sadece gerçek işlemler kullanılır. KDM performansını artırmaya yönelik olarak dört odak noktası öneriyoruz. Bu odak noktalarını tek-kart ve çok-kart odak noktası, tatil dönemi harcamaları odak noktası, günün saatleri odak noktası ve enflasyon odak noktası olarak sıralayabiliriz. Önerdiğimiz ikinci model ise Model Topluluğunda İyimser, Kötümser ve Ağırlıklı Oylama Modeli'dir (TİKA). TİKA iyi bilinen altı yapay zeka tekniğinin bir topluluk olarak bir araya getirilmesiyle oluşturulmuştur. TİKA'yı oluşturan yapay zeka modelleri eğitilirken geçmişteki gerçek ve sahte işlemler birlikte kullanılır. TİKA'da iyimser, kötümser ve ağırlıklı oylama stratejilerini sahtekarlık tespit performansını artırmaya yönelik olarak öneriyoruz. Önerdiğimiz üçüncü model ise Harcama Alışkanlıkları Benzerlik Modeli'dir (HAB). HAB'da harcama alışkanlık benzerlik ölçütlerini sahtekarlık tespit performansını artırmak için kullanıyoruz. Türkiye'nin öncü bankalarından birinden aldığımız gerçek kredi kartı işlemlerini içeren veri kümesini kullanarak önerdiğimiz üç modelin performansını değerlendiriyor ve bu modellerin karşılaştırmalı analizini sunuyoruz.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

AI	Artificial Intelligence
AUC	Area Under Curve
BN	Bayesian Network
CBM	Cardholder Behavior Model
DT	Decision Tree
EM	Expectation Maximization
FN	False Negative
FP	False Positive
HMM	Hidden Markov Model
IFD	Intelligent Fraud Detector
MCC	Merchant Category Code
MJR	Majority Voting
NB	Naïve Bayes
OPT	Optimistic Voting
OPWEM	Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models
PES	Pessimistic Voting
RBF	Radial Basis Function
RF	Random Forest
ROC	Receiver Operating Characteristic
SBSM	Spending Behavior Similarity Model
SOM	Self Organizing Maps
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
WGT	Weighted Voting

1. INTRODUCTION

In 2014, the size of the global card business was $28.84 trillion, whereas fraud losses totalled approximately $16.31 billion, corresponding to a loss rate of 5.65¢ per $100. In 2020, global card fraud amount is expected to exceed $35.54 billion. The total volume figure represents credit, debit, prepaid and private label payment cards worldwide [1]. Naturally, banks are striving to decrease their losses resulting from card fraud.

Banks commonly use rule-based systems for credit card fraud detection. The rules applied in these systems are formulated based on the experience of fraud experts and the results of fraud investigations. Each credit card transaction is evaluated according to the rule set, and an alarm is raised if the transaction matches one or more rules. As new fraud cases occur, new rules are discovered by human fraud experts and added to these rule-based systems. In other words, rule discovery is a manual process, which is an important disadvantage of rule-based systems. By contrast, Artificial Intelligence (AI) models learn from past transaction data. Therefore, unlike in rule-based systems, no manual process is required. Many researchers in the domain of credit card fraud detection have recognized this advantage offered by AI models. As a result, many AI models have been proposed for the detection of credit card fraud.

Approaches to the development of AI models can be generally divided into two types, supervised and unsupervised, both of which have been used in credit card fraud detection. In supervised fraud detection, both fraudulent and legitimate historical transactions are used for training. In unsupervised fraud detection, the spending behavior of each cardholder is modeled using that cardholder's past legitimate transactions. When a new transaction comes, it is checked whether it is similar to the past legitimate transactions of the cardholder. In other words, the new transaction is checked whether it fits the established behavior model. If the new transaction does not fit the behavior model, it is considered to be potentially fraudulent.

There are various fraud types regarding credit cards which can be listed as application fraud, intercept fraud, lost and stolen card fraud, skimming fraud, fake and doctored card fraud, site cloning and false merchant sites and triangulation.

Application fraud occurs when false information is given to acquire a credit card. Application fraud can be grouped into two as assumed identity fraud and financial fraud. In assumed identity fraud, the fraudster pretends to be someone else by using fake identity card. In financial fraud, the fraudster gives false information about his/her financial situation [2].

Intercept fraud occurs when card is stolen from the post service. Lost and stolen cards fraud occurs when the lost or stolen cards are used. Skimming fraud occurs when card data stored on the magnetic strip or the chip is copied from one card to another [2].

Credit card generators are emulation software which can create valid credit card numbers and expiry dates. This information can be used by the fraudsters to commit fake and doctored card fraud [2].

Some merchant web sites can be cloned or false merchant sites can be published to collect card security data. For this purpose triangulation method is commonly used. A fraudster acting as a merchant offers a product with very reasonable price on an internet site. The fraudster tells the customer to pay via e-mail once the item is delivered. The fraudster acting as a merchant uses a fraudulent card information to buy the product from a legal web site and send it to the customer. The customer sends his/her card info to the fraudster acting as the merchant. The fraudster operates in this way before closing the web site to cover his tracks [2].

All fraud types except credit card application fraud are in the scope of our thesis. In this thesis, we propose three novel artificial intelligence based models to detect credit card frauds.

Firstly, we propose Cardholder Behavior Model (CBM). CBM is an unsupervised model and uses clustering transaction amounts to represent the spending behavior of cardholders. We propose four focal points to fine-tune CBM. These focal points are single-card versus multi-card focus, holiday season spending focus, time of day focus and inflation focus.

In previous research, various unsupervised models have been proposed in credit card fraud detection domain. To the best of our knowledge, former unsupervised fraud models have been built on card-specific transaction data. However, a cardholder may hold multiple cards issued by the same bank. Therefore, constructing a behavior model on all cards of a cardholder rather than a single card is expected to improve the fraud detection performance. In CBM, we propose using transactions from multiple cards of the cardholder to form the behavior model.

In most countries, there are holidays such as New Year and religious holidays. It is known that spending is usually higher in holiday seasons. To the best of our knowledge, holiday seasons have not been considered as a focus point in previous research. In CBM, we propose considering holidays as special spending periods and establishing a behavior model specific to holiday seasons.

As can be observed in daily life, the spending behavior may change in relation to the time of day. For example, one can have a reasonable cost lunch but an expensive dinner both in the same restaurant. Based on such observations, it is expected that considering spending behavior separately for different times of the day will improve the fraud detection performance. In CBM, we established specific behavior models for different periods of the day; morning, afternoon, evening and night. To the best of our knowledge, time of day has not been previously taken into account.

As the last focus point of CBM, we propose considering economical inflation to correct transaction amounts before using them to construct the behavior model. To the best of our knowledge, inflation focus has not been previously taken into consideration.

The second model we propose is Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models (OPWEM). OPWEM is an ensemble of six well known AI techniques, namely Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine, and K*. We propose optimistic, pessimistic and weighted voting strategies in OPWEM for a better detection of credit card fraud. In optimistic voting, if at least one AI model judges that the transaction is legitimate, then the decision of the ensemble is that it is legitimate. In pessimistic voting, when at least one AI model judges that the transaction is fraudulent, the decision of the ensemble is that it is fraudulent. In weighted voting, specific weights are assigned to each model in the ensemble depending on the models' fraud detection performance on the validation set.

Combining multiple models has been previously proposed in different enterprise systems [3, 4, 5]. However, voting scheme has not been previously applied in credit card fraud detection domain. OPWEM is novel in this aspect.

In CBM and OPWEM, the models are trained by using transactions which are specific to the cardholder, but similarity of cardholder behavior is not taken into account. Similarity of cardholder behavior can be defined as similar transactions of different cardholders. As the third model, we propose an improved credit card fraud detection model, namely Spending Behavior Similarity Model (SBSM), which takes similarity of cardholder behavior into account. Taking the similarity of cardholder behavior into account implies larger training set size and better quality of training set.

In SBSM, we propose using the mean and median spending amounts as similarity measures. We propose calculating the similarity values offline and storing them in a similarity information table. During training, the model is trained with the cardholder transactions together with the transactions of the cardholders with similar spending behavior. To the best of our knowledge, this point of view is the first in credit card fraud detection domain.

A dataset of real life credit card transactions from a leading bank in Turkey has been used to evaluate the performance of three proposed models. A comparative evaluation of the proposed models is conducted so that the bank management can choose among the proposed models based on their fraud detection and false alarm rate strategy.

The remainder of this thesis is structured as follows. In Chapter 2, we provide background information about credit cards and the credit card fraud detection literature. In Chapter 3, we discuss the proposed models briefly and introduce the credit card transactions dataset, evaluation criteria, experimental setup and software tool. In Chapter 4, Chapter 5 and Chapter 6, we discuss CBM, OPWEM and SBSM in detail and analyze the results. In Chapter 7, we make a comparative evaluation of the proposed models. In Chapter 8, we provide the concluding remarks and state the future work.

2. OVERVIEW OF CREDIT CARD FRAUD DETECTION

Credit cards have been an important part of everyday life for more than half a century since the first universal credit card was introduced by Diners Club in 1950 [6]. By 1959, there were one million Diners Club cardholders [6]. Another leading actor in the credit card industry, American Express, entered the market in 1958, issuing 250,000 cards prior to the official launch date [7]. The first credit cards were paper cards on which the cardholder’s name and account number were typed; however, in 1959, American Express began issuing plastic cards [7]. In 1958, Bank of America became the first bank to issue a credit card, under the BankAmericard brand. Initially, this card could be used only in the state of California, but the BankAmericard Programme went global in 1974. In 1976, BankAmericard became Visa [8]. Master Charge was introduced by several banks in California to compete with BankAmericard in 1966 and was renamed to MasterCard in 1979 [9].

Today, consumers hold approximately 5 billion credit cards throughout the world. In 2014, the total worth of cardholder transactions was approximately $28.84 trillion [1]. Decreasing credit card fraud losses has always been a fundamental goal of banks, which hire fraud detection experts and implement fraud detection tools expressly for this purpose. To date, the fraud detection tools most commonly used by banks have been rule-based systems. Fraud experts define the rules applied in these systems based on historical fraud cases and their investigation results. When a new credit card transaction matches one or more of the previously defined rules, the systems raise an alarm to indicate that the new transaction is potentially fraudulent. The rule-based approach is successful for identifying fraudulent transactions that follow previously observed fraud patterns, but it lacks agility. Before a new rule is added to the existing rule set, a considerable number of fraudulent transactions matching that rule have typically already occurred. A long delay is required before a rule can be added, during which time fraud strategies may change, making the rule obsolete [10]. Besides rule based systems, some data mining based systems are also used in banks. SAS credit card fraud detection tool identify the risk profile of the cardholder across all the banking products that they use [11]. Ethoca credit card fraud detection tool is based on a centralized multi bank data collection system [12]. Banks upload essential data such as card details, IP addresses and E-Mail addresses to the centralized database. Merchants and banks can not see the information of other banks' cardholders but can get a certain risk score from the centralized system.

The size of the credit card industry and the scale of credit card fraud have captured the attention of not only fraudsters but also researchers. Many approaches have previously been proposed for the detection of credit card fraud. The majority of these proposals are based on AI techniques. The previously proposed models for credit card fraud detection are summarized in Table 2.1 and Table 2.2, demonstrating that various types of models have been used in this domain. Tree-based models, unsupervised models, neural-network-based models, Bayesian models and genetic algorithms appear to be the most popular.

Table 2.1. Unsupervised models for credit card fraud detection

Model	References
Distance Based	[2, 10, 13, 14, 15, 16, 17]
Self-Organizing Maps (SOM)	[18]
Hidden Markov Model (HMM)	[19, 20, 21, 22, 23]

In the literature, AI models are grouped into two classes: supervised and unsupervised. Both approaches have been used in credit card fraud detection. In supervised fraud detection, both fraudulent and legitimate transactions are used to train the AI model. The resulting AI model determines whether a new transaction is fraudulent or legitimate. The training process of the AI model is periodically repeated to include more recent transactions. In unsupervised fraud detection, the behavior of a cardholder is modeled based on that cardholder's past legitimate transactions. Thereafter, each new transaction is analyzed to determine whether it lies inside or outside the cardholder's range of typical behavior. When a transaction falls outside this range, an alarm is raised.

Ganji and Mannem proposed an unsupervised fraud detection method called Stream Outlier Detection based on Reverse k Nearest Neighbors (SODRNN) [2]. The past transactions of a credit card form the window, which is entirely stored in the main memory. Each item in the window contains transaction information, k nearest neighbors list and reverse k nearest neighbors list for that transaction. When a new transaction arrives, these lists are updated in one pass. This fact comes out as an important advantage of the proposed methodology. The outliers are easily discovered with a simple scan when needed. Although experimental details have not been given, the proposed method was claimed to be effective on synthetic and real transaction data.

Krivko combined the rule-based system and the unsupervised approach so that one approach's weak points can be made up by the other one [10]. In the proposed model, the transaction is firstly evaluated by the unsupervised model to detect a deviation from the usual spending pattern. If the evaluation result is suspicion the transaction is directed to the rule-based system to get the final decision.

Yu and Wang proposed Euclidean distance as the similarity measure in their proposed unsupervised model [13]. Transaction attributes are normalized so that attributes with large and small values have the same weight. Euclidean distances between any two transactions are computed and summed to get the total distance between a particular transaction and the other transactions. If the total distance is above a pre-determined threshold, the transaction is stated as an outlier. That threshold is determined in the training phase. Personal characteristics such as income, age, profession and marital status are taken into consideration together with spending characteristics such as average amount per transaction, average daily spending and card usage frequency.

Ju and Wang proposed anomaly detection based on coefficient sum [14]. In the proposed model, customer attributes such as income, age, profession, position, marital status and working years are used together with spending behavior such as card usage frequency, average daily spending and average amount per transaction. All categorical attributes are converted to numerical attributes in the preprocessing stage. Thereafter, the distances between every two instances are calculated to form similar coefficient matrix. In this matrix, sum of a row gives the total distance of the corresponding instance to the other instances. A value is calculated using this sum and the maximum sum among all rows. If this value is above a certain threshold, the instance is considered as an outlier. In the validation phase, the threshold value is determined.

Philip and Sherly, proposed an unsupervised fraud detection model which consists of three major modules, namely data engine, rule engine and rule monitor [15]. Data engine transfers recent transaction data from the online database to the replication database and makes pre-processing. Rule engine generates an individual profile for each user by using the pre-processed transaction data on the replication database. A user profile is a rule set stored in a Frequent Pattern Tree. Rule monitor inspects an incoming transaction and raises an alarm when anomaly is detected.

Jha *et al.* started with an initial dataset which consists of primary attributes such as credit card number, transaction type, currency code, transaction date, merchant category code, city and country, e-commerce flag and transaction amount in foreign and local currency units [16]. They derived many attributes such as average amount per transaction over a month and average amount on the same merchant type as the current transaction. The authors used logistic regression as the classification method.

Pun and Lawryshyn suggested combining K Nearest Neighbors, Naïve Bayes and Decision Tree models by using a meta-classifier [17]. In this paper, Naïve Bayes was also used as the meta-algorithm.

Quah and Sriganesh proposed using self-organizing map (SOM) which is a type of artificial neural network that employs unsupervised learning to get the topological properties of input [18]. It maps multi-dimensional input to one or two dimensions. SOM provides clusters that represent spending characteristics of a certain cardholder. A future transaction that falls into a dense cluster can be evaluated as genuine whereas a transaction that does not reside in a cluster may be evaluated as suspicious.

Srivastava *et al.* suggested the use of Hidden Markov Model (HMM) for detecting credit card fraud [19]. The authors modeled the credit card transaction processing sequence by the stochastic process of an HMM. An HMM is trained with the legitimate transactions of the cardholder. If a future credit card transaction is not accepted by the trained HMM, it is considered to be fraudulent.

Bhusari and Patil proposed using a Hidden Markov Model (HMM) whose states are purchase types (bill payment, restaurant, electronic items etc.) and observable values are price ranges (low, medium and high) [20]. The existing spending behavior of the customer is modeled as the initial sequence of observable values. This model has a certain probability. When a new transaction occurs, a new observation value is added to the sequence and the oldest observation value is discarded. The probability is recalculated. If the new probability value is much smaller than the old probability value (determined by a threshold), an alarm is raised.

Rani *et al.* considered transition in the purchase type as state transition in the proposed Hidden Markov Model (HMM) [21]. IP addresses of online card transaction are logged. If this address is different from the known IP address of the customer then the probability of fraud increases. However, static IP addresses are assumed.

Prakash and Chandrasekar proposed using Hidden Markov Model for detecting abnormal spending behavior [22]. A cardholder-specific Hidden Markov Model is trained by using the legitimate past transactions of the cardholder. The authors stated that the information entropy of the fraud state is larger than the information entropy of the legitimate state. Based on this inference, the authors proposed using information entropy criteria as the metric to distinguish between fraudulent transactions and normal transactions. In a following research, Prkasah and Chandrasekar introduced an ensemble of HMM [23]. In this model, they assumed two observation sequences are available as the output of an HMM state sequence. They introduced a random delay between the output sequences so that these two sequences are not synchronized.

Gadi *et al.* proposed using artificial intelligence methods such as Naïve Bayes, Decision Tree, Bayesian Network, neural network and artificial immune system [24]. They focused on exploring the optimum parameter set. There are no or a few parameters for Naive Bayes, Decision Tree and Bayesian Network models. Therefore, exhaustive trial for the sub-optimum parameter set is possible. On the other hand, neural networks and artificial immune system models have a higher number of parameters that makes is impossible to explore exhaustively. The authors performed genetic algorithm based parameter optimization for discovering the optimized parameter set for neural network and artificial immune system. In the following paper, Gadi *et al.* added a cost sensitive approach highlighting that the cost of an undetected fraudulent transaction is much more than the cost of a legitimate transaction reported as fraudulent [25].

Table 2.2. Supervised models for credit card fraud detection

Model	**References**
Decision Trees (DT) Random Forest (RF)	[17, 24, 25, 26, 27, 28, 29, 30, 31, 32]
Neural Networks (NN)	[24, 25, 33, 34, 35, 36, 37, 38, 39]
Bayesian Networks (BN)	[24, 25, 33, 40, 41]
Naïve Bayes (NB)	[17, 24, 25, 30, 40]
Bayesian Decision Theory (BDT)	[42]
Support Vector Machines (SVM)	[28, 29, 36, 43, 44, 45, 46]
Genetic Algorithm (GA)	[47, 48, 49, 50]
Artificial Immune System (AIS)	[24, 25, 50, 51]
Fuzzy Logic	[52, 53]
Sequence Alignment	[54, 55]
Scatter search	[49]
Influence Diagram	[56]
Linear Discriminant Analysis	[57]
Migrating Birds Optimization	[58, 59, 60]

Patil *et al.* proposed a Decision Tree based model which classifies transactions in multiple suspicion levels rather than classifying as either fraudulent or legitimate [26].

Sherly and Nedunchezhian proposed another Decision Tree based approach [27]. Transaction attributes are transaction amount, time, merchant and city of purchase/order placing. These continuous attributes are converted into categorical attributes. Transaction amount is categorized into three as low, medium and high. Transaction time is categorized into eight groups by dividing a month into four weeks and a week into two parts as weekdays and weekend. Merchant is categorized into five groups as groceries, electronics, gold, medical and miscellaneous. Lastly, city of purchase is grouped into three as local, national and international. K-means clustering is used to group the categorized transactions. The categorized attributes are fed into the BOAT algorithm to construct the Decision Tree. BOAT algorithm can incrementally update a Decision Tree as the training set changes. BOAT takes small samples from the training set by using bootstrapping and constructs sample trees for each sample by using any Decision Tree algorithm. For each node of the sample trees, the splitting attribute is checked. If the splitting attribute is not same for all sample trees, the node and its subtrees are deleted. Finally, sample trees are combined to form the final sample tree. The current tree and the final sample tree are compared and differences are detected. These differences are used to achieve the final tree.

Bhattacharyya *et al.* proposed using Random Forests and Support Vector Machines and focused on using deriving complex attributes from primary credit card transaction attributes for a better fraud detection [28].

Sahin and Duman proposed using Decision Tree and Support Vector Machine to detect credit card fraud [29]. Decision Tree models outperform Support Vector Machine models. In the following research, Sahin and Duman proposed using neural network models and stated that neural network models outperform logistic regression models [37]. Sahin *et al.* proposed a cost-based decision tree approach which minimizes the sum of misclassification costs [31].

Alowais and Soon compared the accuracy of models trained using personalized datasets with the accuracy of models trained using aggregated datasets [30]. Three different cardholders' transaction data, which consists of amount, location and time, is collected. Moreover, an online questionnaire on spending behavior is applied to 167 respondents, 47 of which are used since the rest are either incomplete or inconsistent. In this study, three different personalized models are built for each individual. In addition to these personalized models two aggregated models are built, one using the combination of three cardholders' dataset, the other using the answers collected from the questionnaire. Each model uses both Random Forest and Naïve Bayes algorithms. The most surprising part of this study is that personalized models are found out to be less accurate than aggregated models. This result conflicts with studies which claim that personalized models are much more effective because of that fact that each person has a different spending behavior.

Noghani and Moattar proposed dividing the training dataset to several parts and using each part to train a separate decision tree; thus forming a decision forest [32]. In this paper, the authors also introduced wrapper based feature selection to select the most effective credit card transaction attributes. The proposed model is evaluated together with the basic classification algorithms, namely decision tree, Naive Bayes and Bayesian Network. The evaluation results show that the proposed model is superior to the basic classification algorithms.

Maes *et al.* proposed using Bayesian Network and neural network for credit card fraud detection [33]. The evaluation results show that Bayesian Network outperforms neural network in prediction accuracy, training time whereas neural networks give faster responses for incoming transactions.

Ghosh and Reilly presented a neural network based fraud detection system which provides substantial improvements in accuracy and response time [34].

Aleskerov *et al.* proposed detecting irregularities in customer's credit card usage pattern by using neural networks [35]. The system is tested using synthetic credit card transaction data. A neural network is trained for each customer. The accuracy of the system is analyzed by taking merchant categories into account. The proposed idea is provided with a software tool called CARDWATCH, which stands out with its effective graphical user interface.

Mishra and Dash made a comparative evaluation of decision tree and two types of neural networks in terms of classification accuracy and response time [38]. The neural networks which are used in this paper are multi-layer perceptron (MLP) and Chebyshev functional link artificial neural network (CFLANN).

Srivastava *et al.* proposed a merchant-oriented fraud detection approach which uses neural networks [39]. In the proposed approach, neural networks are trained using cardholder information such as profession, earnings and purchase information such as billing and shipping address, location distance and time difference between the current and the previous transaction, currency type and amount.

Filippov *et al.* proposed using Naïve Bayes and Bayesian Networks to detect credit card fraud [40]. Bayesian Networks are found to be more effective. The basic advantage of Bayesian Networks over the other models is found out to be taking the correlation between attributes into account. For instance, currency unit and transaction amount is strongly correlated and this fact deserves to be considered. The authors proposed two fraud detection modules, online and offline. Online fraud detection module provides fraud probability to the provision system during authorization. Therefore, the transaction can be declined if the fraud probability is high. Online fraud detection module has to meet the strict timing requirements of transaction authorization. Offline fraud detection module analyses transactions that have already been authorized without a strict time requirement.

Panigrahi *et al.* proposed a credit card fraud detection system which combines different types of evidences using Dempster-Shafer theory [41]. Rule-based filter (RBF) consists of generic and customer specific rules which classify a transaction as fraudulent with a certain probability. Billing address and shipping address mismatch is an example rule. These probabilities obtained from RBF are combined using Dempster-Shafer theory to compute an initial belief value. If the customer behaves differently than his profile, the transaction can be labeled as suspicious. If the transaction is found to be suspicious, the initial belief value is modified according to the similarity of the transaction with the past fraudulent and legitimate transactions. Bayesian learning is used to calculate posterior probabilities at this point.

Bahsen *et al.* focused on monetary gains and losses and proposed a cost-oriented fraud detection system based on Bayes minimum risk classifier [42].

Chen *et al.* proposed a Support Vector Machine based model and stated that the accuracy is improved by majority voting, weighting and voting and hierarchical SVM’s

[43, 44]. Moreover, it is proposed that an online questionnaire is filled by the cardholder to initialize the fraud model. Therefore, the fraud can be detected even with the initial transaction of the newly issued card. In a related paper, Chen *et al.* compared SVM with ANN again in a questionnaire-based approach [36].

Hejazi and Singh focused on the accuracy and computation time of Support Vector Machine on credit card fraud detection [45]. In this context, different Support Vector Machine algorithms with different kernel methods are evaluated. These Support Vector Machine algorithms are Sequential Minimal Optimization (SMO), C-Support Vector Classification (C-SVC) and v-Support Vector Classification (v-SVC). The kernel methods are linear, polynomial and Radial Basis Function (RBF) kernels. In SMO, polynomial kernel outperforms the others in terms of accuracy. However, RBF kernel is faster than the others. In C-SVC, linear kernel is more accurate than the others. Again, RBF kernel is faster. In v-SVC, RBF shows better classification accuracy. This time, linear kernel is the fastest.

Kamboj and Gupta proposed using Support Vector Machines with Radial Basis Function kernel for credit card fraud detection [46]. A German credit card transaction dataset was used for evaluation.

Ma and Li proposed using genetic algorithm in credit card fraud detection. 78% accuracy is achieved in the experiment on real life transaction data [47].

Ozcelik *et al.* used genetic algorithm in a value based approach [48]. The sum of available limits of the cards whose transactions are labeled as true positive minus the monitoring costs is considered as savings.

Duman and Ozcelik used genetic algorithm [49]. Importance of attributes was analyzed and variables such as MCC and country were not included.

Taklikar and Kulkarni proposed using genetic algorithm and Artificial Immune System to detect credit card fraud [50]. Each transaction is represented as a binary vector. The vector is formed by using card number, product category, amount, day of the week and time. Each attribute is mapped to either 0 or 1. For example, the day of the week is represented as 0 and 1 for weekdays and weekend respectively. Once the vector is formed, crossover and mutation procedures are applied and compared to past fraudulent transactions to determine the fraud score. Crossover and mutation procedures are applied until a certain threshold is reached.

Halvaiee and Akbari addressed using Artificial Immune System for credit card fraud detection [51]. The authors proposed adding some improvements to AIS algorithm to improve precision, decrease the cost and system training time.

Bentley *et al.* used genotype which is a special form of binary tree [52]. In genotype, each node consists of a binary number and a flag defining whether the node has zero, one or two children. Each genotype is mapped onto a phenotype which is a fuzzy rule. Random genotypes are created at the start of the evolution and every evolved genotype is evaluated to provide fitness functions. Crossover, mutation and population overlapping is applied to generate the next generation. Multiple versions of the system are trained by using different setups to form an ensemble which is called a committee. Using multiple systems improved fraud detection accuracy.

Sanchez *et al.* focused on department store credit cards [53]. In South America, the payment with department store credit cards is seven times higher than bank-issued credit cards. Fuzzy logic is applied for detecting frauds. The data is organized as two tables, namely client table and transaction table. Product type, place of purchase, age, gender, years of account held, no-money-down purchase, maximum installments, purchase period and amount are found to be the essential attributes for fraud detection. Continuous attributes, which are age, years of account held, purchase period and amount, are converted to categorical attributes by using K-means algorithm. Thereafter, Fuzzy Query 2+ tool is used to provide the item sets which are likely to indicate a fraudulent transaction.

Kundu *et al.* based their work on the idea that the fraudster spending pattern deviates from the real cardholder's spending pattern [54]. Detecting the deviation means detecting the fraud. For this purpose, Kundu *et al.* used sequence alignment method, which is successfully used firstly in the bioinformatics domain and later in intrusion detection.

BLAST algorithm is selected as the sequence alignment algorithm since it is very fast. Each transaction is categorized as low, medium and high according to amount. Based on past fraudulent transactions, sets of potential fraudulent sequences are retrieved. Likewise, legitimate sequences are retrieved from the transaction set of the card holder. When a new transaction occurs, it is categorized into low, medium or high to become a member of the sequence together with a predetermined number of cardholder's past transactions. This sequence is aligned with the past fraudulent and legitimated sequences to get fraudulent and legitimate scores. If the difference between the legitimate and the fraudulent scores are below a certain threshold, an alarm is raised.

In the following research, Kundu *et al.* considered credit card fraud detection as a sequence matching problem on time-amount dimension [55]. Basic Local Alignment Search Tool (BLAST) and Sequence Search and Alignment by Hashing Algorithm (SSAHA) are combined to propose BLAH. BLAST compiles a list of high scoring words from the given sequence and compares these words with the past sequence database. If the word is matched in a sequence residing in the database a hit is recorded. Thereafter, the matching word is extended in both directions until the similarity value falls below a threshold. SSAHA constructs a hash table from sequences in the database in which the incoming words are searched. Firstly, the incoming transaction is compared with the legitimate transactions and if a deviation is observed, the transaction is compared with the fraudulent transactions to figure out whether the transaction is probably fraudulent or it results from a short-term change in spending behavior. BLAST-SSAHA algorithm is used for producing similarity values.

Cobb proposed detecting the credit card fraud on the merchant side [56]. This paper is different from the others in this perspective. The proposed graphical model, called influence diagram, is based on address characteristics, product characteristics, order value and inspection cost. Address characteristics are listed as whether the shipping and billing addresses are the same, the e-mail is not from a free web-based address (such as gmail, yahoo etc.) and the address is in a foreign country. Product characteristics are listed as whether the customer requests that shipment is left at the door if no one is home, rush shipping and multiple units of the same item. The solution of the influence diagram yields threshold values for each case. For example, if all risky address and product characteristics hold for the current order, the threshold value is $16 whereas it is over $95 if none of them hold.

Mahmoudi and Duman proposed using linear discriminant analysis (LDA) for the first time in credit card fraud detection domain [57]. LDA is a supervised method by which the input is divided into decision regions whose boundaries are called decision boundaries. Decision boundaries are linear function of input vector. Fisher Discriminant Analysis (FDA) is a form of LDA and it minimizes the within-class variance to reduce the overlap and tries to maximize the separation. The authors proposed using FDA with weighted average where the weights are defined as total available limits on each credit card.

Duman and Elikucuk proposed using Migrating Birds Optimization (MBO) which is inspired by birds flying in V formation to spend less energy and improve their range [58, 59, 60]. The algorithm starts with a number of solutions corresponding to birds. The first solution corresponds to the leader bird. This solution is tried to be improved by its neighbor solutions which correspond to the following birds. If the best neighbor results in improvement, the current solution is replaced with that neighbor. After all solutions are improved or tried to be improved, the leader solution is moved to the end and one of the following solutions is moved to the leader position. This process is repeated for a number of iterations.

3. NOVEL MODELS FOR CREDIT CARD FRAUD DETECTION

3.1. Overview of Novel Models

In this thesis, we propose three novel artificial intelligence based models to detect credit card frauds. First, we propose Cardholder Behavior Model (CBM). CBM is an unsupervised model and uses clustering credit card transaction amounts to represent the spending behavior of cardholders. We propose four focal points in CBM to fine-tune its performance. Focal points are single-card versus multi-card focus, holiday season spending focus, time of day of transaction focus and inflation focus.

The second model we propose is Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models (OPWEM). OPWEM is an ensemble of six well known AI techniques, namely Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine, and K*. We propose optimistic, pessimistic and weighted voting strategies in OPWEM for a better detection of credit card fraud.

Lastly, we propose Spending Behavior Similarity Model (SBSM), which takes similarity of cardholder behavior into account. Taking the similarity of cardholder behavior into account implies larger training set size and better quality of training set.

A dataset of real life credit card transactions from a leading bank in Turkey has been used to evaluate the performance of three proposed models.

The overall view of the proposed models is shown in Figure 3.1. Our credit card fraud detection approach involves credit card holders, credit cards, credit card transactions, a credit card transactions database and the proposed models. Credit card holders, called cardholders for short, may have one or more credit cards. They generate credit card transactions for each of their credit cards. These credit card transactions are stored in the Credit Card Transactions Database which is used for training of the proposed models, namely CBM, OPWEM and SBSM.

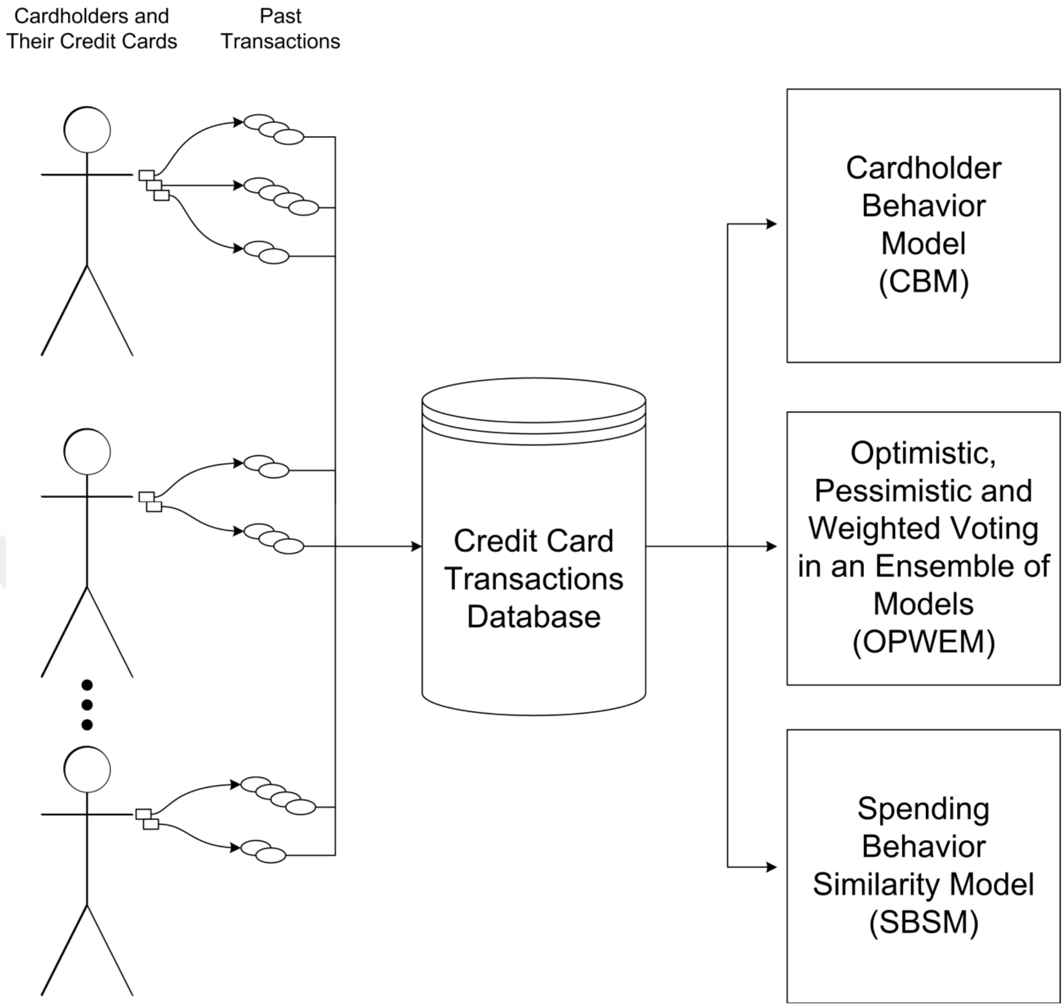

Figure 3.1. Novel models for credit card fraud detection.

A credit card transaction has six attributes: cardholder number, card number, merchant category code, amount, date and time. The cardholder number is a unique number that identifies a specific bankcard holder. A person cannot have multiple cardholder numbers. The card number is a unique 16-digit number that identifies a specific credit card. The Merchant Category Code (MCC) is a four-digit number used by the bankcard industry to classify suppliers into market segments. Approximately 600 different MCCs exist, which denote various types of businesses [61]. Table 3.1 lists some of these MCCs.

Table 3.1. Sample merchant category codes (MCC)

MCC	Merchant Category Description
0742	Veterinary Services
1711	Heating, Plumbing, A/C
4011	Railroads
4119	Ambulance Services
4131	Bus Lines
4511	Airlines, Air Carriers
4814	Telecommunication Services
4899	Cable, Satellite, and Other Pay Television and Radio
5094	Precious Stones and Metals, Watches and Jewelry
5139	Commercial Footwear
5172	Petroleum and Petroleum Products
5411	Grocery Stores, Supermarkets
5462	Bakeries
5651	Family Clothing Stores
5712	Furniture, Home Furnishings, and Equipment Stores, Except Appliances
5722	Household Appliance Stores
5812	Eating Places, Restaurants
5942	Book Stores
7011	Hotels, Motels, and Resorts
7216	Dry Cleaners
8011	Doctors

3.2. Evaluation Dataset

A leading bank in Turkey provided a real-life credit card transaction dataset for the evaluation of the proposed models. This transaction dataset is referred to as the Evaluation Dataset in this thesis. The Evaluation Dataset contains 152,706 credit card transactions from 105 cardholders. The transactions in the Evaluation Dataset occurred between January 2006 and February 2013. Each cardholder included in this dataset has a number of past transactions ranging between 1440 and 1499. More than half of the cardholders in the dataset hold more than one card, as shown in Table 3.2.

Table 3.2. Cardholder, card and transaction counts in Evaluation Dataset

	Cardholder Count	Card Count	Legitimate Transaction Count	Fraudulent Transaction Count	Total Transaction Count
Cardholders with 1 card	52	52	75,193	618	75,811
Cardholders with 2 cards	42	84	60,658	352	61,010
Cardholders with 3 cards	7	21	10,107	17	10,124
Cardholders with 4 cards	4	16	5,729	32	5,761
All Cardholders	105	173	151,687	1,019	152,706

In the Evaluation Dataset, some transactions have been flagged as fraudulent, whereas the rest have been flagged as legitimate. In a bank credit card system, most transactions that are flagged as fraudulent correspond to one of two scenarios. The first scenario arises when the rule-based fraud detection system of the bank raises an alarm on a current transaction. In such a case, the bank's fraud call center contacts the cardholder immediately for notification of the suspicious transaction. If the transaction was not executed by the cardholder, it is flagged as fraudulent. In the second scenario, the cardholder identifies transactions that he or she did not make on his or her credit card statement and calls the bank's fraud call center to inform them of the situation. As a result, the corresponding transactions are flagged as fraudulent.

This study considers only principal card transactions; additional card transactions are ignored because additional cards may be used by other individuals, such as the principal cardholder's family members, who may have different spending behavior. Therefore, additional card transactions cannot be used for model training.

3.3. Evaluation Criteria

Binary classification is the task of classifying elements into two groups on the basis of a classification rule. Credit card fraud detection is a binary classification problem in which a credit card transaction is labeled as either fraudulent or legitimate. As in all binary classification problems, an evaluation of the fraud detection performance of a model is performed by comparing the number of alarms issued with the numbers of transactions that are known to be fraudulent and legitimate. Therefore, criteria that have become standard for the evaluation of binary classification models are used. These criteria are the alarm rate, sensitivity, specificity, false positive rate, precision, negative predictive value and accuracy [62].

The main results of the evaluation can be interpreted in terms of the alarm types and the alarm rate. The number of True Positives (TP) is the number of fraudulent transactions for which alarms are raised correctly—that is, the number of detected fraudulent transactions. The number of False Positives (FP) is the number of legitimate transactions for which false alarms are raised. The number of True Negatives (TN) is the number of legitimate transactions for which no alarm is raised, and the number of False Negatives (FN) is the number of fraudulent transactions for which no alarm is raised—that is, the number of missed fraudulent transactions.

The alarm rate measures the proportion of alarms issued among all decisions made. In this thesis, this is the percentage of alarms with respect to all transactions. The equation for determining the alarm rate is given in Equation 3.1.

$$AlarmRate = \frac{TP + FP}{TP + FN + FP + TN} \tag{3.1}$$

where the sum of TP and FP is the total number of alarms, whereas the sum of TP, FN, FP and TN is the total number of decisions.

The sensitivity measures the proportion of actual positives that are detected correctly. In this thesis, this is the percentage of fraudulent transactions for which the proposed model raises an alarm. The equation for the sensitivity is given in Equation 3.2.

$$Sensitivity = \frac{TP}{TP + FN} \tag{3.2}$$

where the sum of TP and FN is the total number of fraudulent transactions.

The specificity measures the proportion of actual negatives that are correctly identified. In this thesis, this is the percentage of legitimate transactions for which the proposed model does not raise an alarm. The equation for the specificity is given in Equation 3.3.

$$Specificity = \frac{TN}{FP + TN} \tag{3.3}$$

where the sum of FP and TN is the total number of legitimate transactions.

The false positive rate measures the proportion of actual negatives that are falsely identified. In this thesis, this is the percentage of legitimate transactions for which the proposed model raises an alarm. The equation for the false positive rate is given in Equation 3.4.

$$FalsePositiveRate = 1 - Specificity \tag{3.4}$$

The precision measures the proportion of true positives among all positives. In this thesis, this is the percentage of true alarms among all alarms. The equation for the precision is given in Equation 3.5.

$$\Pr ecision = \frac{TP}{TP + FP} \tag{3.5}$$

where the sum of TP and FP is the total number of alarms.

The negative predictive value measures the proportion of true negatives among all negatives. In this thesis, this is the percentage of true “no alarm” decisions among all “no alarm” decisions. The equation for the negative predictive value is given in Equation 3.6.

$$Negative \Pr edictiveValue = \frac{TN}{TN + FN} \tag{3.6}$$

where the sum of TN and FN is the total number of “no alarm” decisions.

The accuracy is the proportion of correct decisions among all decisions. In this thesis, this is the percentage of all transactions for which an alarm is issued for a fraudulent transaction or no alarm is issued for a legitimate transaction. The equation for the accuracy is given in Equation 3.7.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{3.7}$$

where the sum of TP and FN is the total number of correct decisions, whereas the sum of TP, FN, FP and TN is the total number of decisions.

F-measure is defined as the weighted harmonic mean of sensitivity and precision. F-measure can be viewed as a compromise between sensitivity and precision because it is high only when both sensitivity and precision are high. The equation for F-measure is given in Equation 3.8 [63].

$$F - measure = \frac{1}{\alpha \frac{1}{\Pr ecision} + (1-\alpha)\frac{1}{Sensitivity}} \tag{3.8}$$

where the weight $\alpha \in [0, 1]$.

In this thesis, we use balanced F-measure which is commonly denoted as F_1 Score [63]. In F_1 Score, the weights of precision and sensitivity are the same which means α = 0.5. The equation for F_1 Score is given in Equation 3.9.

$$F_1 = \frac{2 * \Pr ecision * Sensitivity}{\Pr ecision + Sensitivity} \tag{3.9}$$

In statistics, a receiver operating characteristic (ROC) curve is a graphical plot that illustrates the performance of a binary classifier as its discrimination threshold is varied [64]. The ROC curve is created by plotting the sensitivity against the false positive rate at various threshold values. We plot ROC curves for each version of the proposed models. Thereafter, we use the ROC curves of the best performing versions in comparative evaluation.

In this thesis, we propose novel models, namely, CBM, OPWEM and SBSM, which possess different ideas. CBM uses clustering of transaction amounts for each MCC in order to provide a cardholder behavior model. OPWEM is an ensemble of artificial intelligence models in which the ensemble decision is provided by aggregating the decision of separate artificial intelligence models that constitute the ensemble. SBSM uses the idea of spending behavior similarity to provide larger and better quality training sets so that fraud detection performance is improved. We standardize the ROC curve plotting methodology and use the standard methodology for each of the proposed models so that the comparative ROC curve evaluation makes sense [65].

We propose a methodology that combines transaction amount and MCC attribute of a transaction to provide a score value for the corresponding transaction. As a natural fact of credit card business, typical transaction amounts are different for each MCC. For instance, jewelry transaction amounts are typically higher than restaurant transaction amounts. Regarding this fact, we propose normalizing each transaction amount by taking the transaction MCC into consideration. The score value range is decided to be between zero and 1000. The scoring equation is given in Equation 3.10.

$$Score_{Trn} = \frac{Amount_{Trn} - MinAmount_{MCC}}{MaxAmount_{MCC} - MinAmount_{MCC}} * 1000 \tag{3.10}$$

where $\mathrm{Score}_{\mathrm{Trn}}$ is the provided score value for a given transaction, $\mathrm{Amount}_{\mathrm{Trn}}$ is the transaction amount for the given transaction, $\mathrm{MinAmount}_{\mathrm{MCC}}$ and $\mathrm{MaxAmount}_{\mathrm{MCC}}$ are the minimum and maximum transaction amounts for the given transaction’s MCC.

As can be seen in Equation 3.10, the score value is zero for a transaction which has the minimum transaction amount among all other transactions with the same MCC. In a similar manner, the score value is 1000 for a transaction which has the maximum transaction amount among all other transaction with the same MCC.

Let us present an example to clarify the score calculation process. Suppose that the minimum and maximum jewelry transaction amounts are 1000 and 5000 respectively. The score value for a given jewelry transaction, Trn$_1$, which is worth 2000, is provided as in Equation 3.11 which is based on Equation 3.10.

$$\begin{aligned} Score_{Trn1} &= \frac{Amount_{Trn1} - MinAmount_{MCC1}}{MaxAmount_{MCC1} - MinAmount_{MCC1}} * 1000 \\ &= \frac{2000 - 1000}{5000 - 1000} * 1000 = 250 \end{aligned} \tag{3.11}$$

Having score values between zero and 1000, we plot the ROC curve by using 50 threshold points in increments of 20.

3.4. Experimental Setup

In experiments, transactions that occurred from January 2006 until the end of 2012 were used for training. The total number of training transactions is 151,939, of which 982 are fraudulent and the remaining 150,957 are legitimate. The transactions that occurred in 2013 were used for testing. The total number of test transactions is 767, of which 37 are fraudulent. The evaluation set-up is summarized in Table 3.3. The statistical significance of the results was assessed by means of t-tests using a 95% confidence level.

Table 3.3. Evaluation setup

	Fraudulent	**Legitimate**	**Total**
Training, January 2006 – December 2012	982	150,957	151,939
Test, January 2013 – February 2013	37	730	767

3.5. Intelligent Fraud Detector

We developed a software tool called Intelligent Fraud Detector (IFD) to implement the proposed models, namely CBM, OPWEM and SBSM, and evaluate their fraud detection performance. IFD was developed in Microsoft Visual Studio 2010 and consists of approximately 3000 lines of C# code based on Microsoft .NET Framework 4.0. It runs on Windows 2012 operating system, and uses Microsoft SQL Server 2014 as its database engine. Expectation Maximization (EM) clustering, Decision Tree (DT), Random Forest (RF), Bayesian Network (BN), Naïve Bayes (NB), Support Vector Machine (SVM) and K* algorithms provided by the WEKA Data Mining Software are used in IFD [66]. IFD has an application programming interface to allow it to be integrated into a bank's credit card system. In the test environment, IFD is running on a HP ProLiant DL380 Gen9 Server with 2 x E5-2630v4 10 Core Intel Xeon Processor, 256 GB DDR4 memory and 6 x 300 GB 12G 15K SAS 2.5'' enterprise disk. Training is made offline. Maximum response time for the test transactions is about 300 milliseconds. After the implementation is moved to the production environment, training will be made during midnight when the number of transactions is less compared to the other hours of the day.

4. CBM: A NOVEL CARDHOLDER BEHAVIOR MODEL FOR DETECTING CREDIT CARD FRAUD

In this section, we focus on analyzing cardholder spending behavior and propose a novel cardholder behavior model for detecting credit card fraud. The model is named Cardholder Behavior Model (CBM). In this section, we discuss details of CBM, four focus points of CBM, experimental results and their analysis. In the first subsection, we discuss CBM in credit card fraud detection process. In the second subsection, we discuss CBM clustering algorithms. In the third subsection, we discuss single-card versus multi-card focus. In the fourth subsection, we discuss holiday spending focus. In the fifth subsection, we discuss time of day focus. In the sixth subsection, we discuss inflation focus. In the last subsection, we provide CBM experimental results and analyze them.

4.1. CBM Credit Card Fraud Detection Process

Our credit card fraud detection approach contains credit card holders, credit cards, credit card transactions, Credit Card Transactions Database and a number of Cardholder Behavior Models. Overall view of this process is given in Figure 4.1. Credit card holders, called cardholders in short, may have one or more credit cards and generate credit card transactions for each of their credit cards. Credit card transactions are stored in Credit Card Transactions Database.

One CBM is trained for each cardholder and MCC. For example, if a cardholder having three credit cards has credit card transactions in a supermarket (MCC: 5411) and in a jewelry store (MCC: 5094), one CBM is trained with supermarket transactions and one CBM is trained with jewelry store transactions for that cardholder. Each CBM may have transactions from three different credit cards.

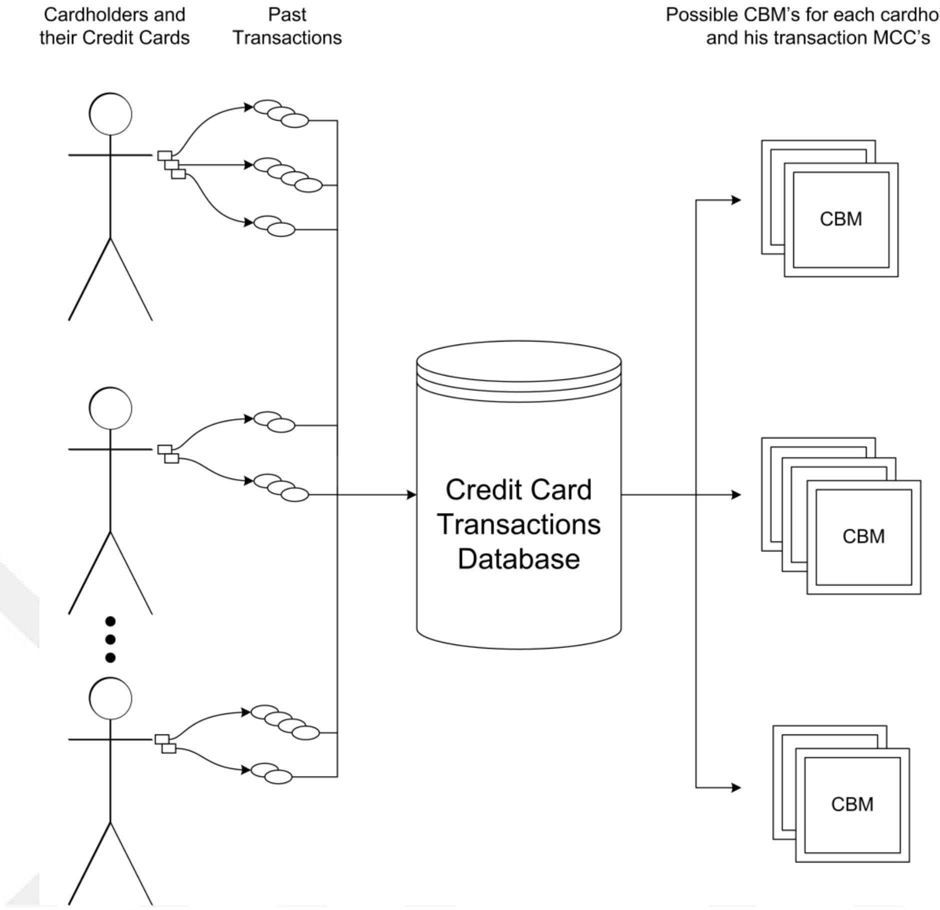
Cardholders and their Credit Cards
Past Transactions
Possible CBM's for each cardholder and his transaction MCC's
Credit Card Transactions Database
CBM
CBM
CBM

Figure 4.1. Overall view of CBM in credit card fraud detection process.

CBM decision process starts with the occurrence of a new transaction. When cardholder makes a new transaction, the corresponding CBM is retrieved by using cardholder number and MCC of the new transaction. The corresponding CBM makes the decision whether the transaction is legitimate or fraudulent. CBM decision process is detailed in Figure 4.2.

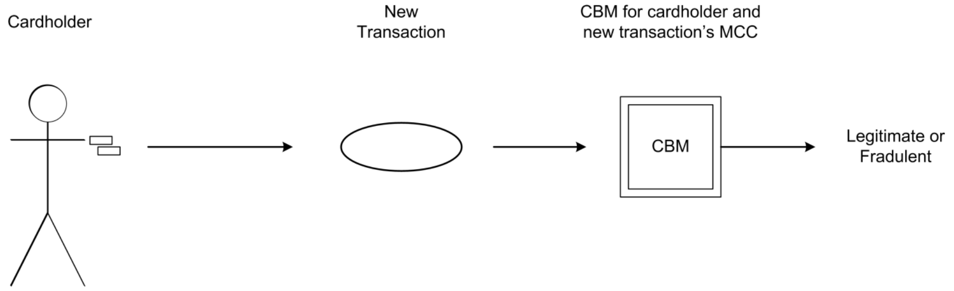

Figure 4.2. CBM decision process.

4.2. CBM Clustering Algorithms

The main approach in CBM for deciding whether the new transaction is legitimate or fraudulent is clustering. If there are no past transactions, clustering is not possible and CBM may decide that new transaction is fraudulent. For example, the cardholder makes a new credit card transaction in a supermarket. Corresponding CBM is the one which is trained by using past supermarket transactions of the cardholder. Amounts of these purchases are fed into the clustering algorithm. In this case, assume that cardholder has past supermarket transactions with amounts 50.00, 65.00, 150.00 and 165.00. Clustering process is shown in Figure 4.3. Past transaction amounts are shown as diamonds. Formed clusters are shown as ovals. The new transaction, which is shown as a square, has amount of 400.00 and it does not fall into any of the clusters. Consequently, CBM decides that the new transaction is fraudulent. If the cardholder makes a new credit card transaction in a jewelry store for the first time, corresponding CBM is not found (i.e. clustering is not possible) and alarm is raised.

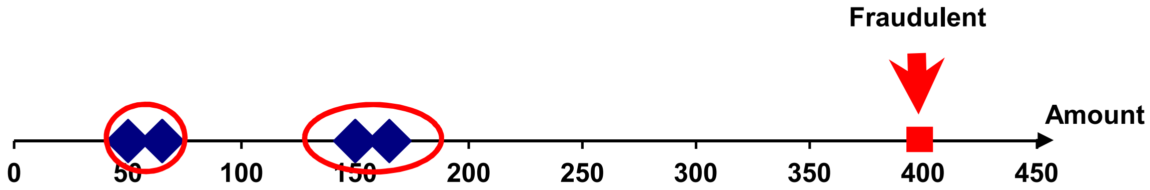

Figure 4.3. Past supermarket transaction clusters and the new supermarket transaction.

In this research, number of amount clusters is previously unknown. Therefore, clustering algorithms, which do not require number of clusters a priori, are selected as candidate algorithms. Candidate algorithms are COBWEB, DBSCAN and Expectation Maximization (EM) [67]. The Experimenter GUI of Weka is used to evaluate candidate algorithms on sample cases [66]. As the result of evaluation, Expectation Maximization (EM) clustering algorithm is selected for clustering amount values [68]. EM clustering decides on the number of clusters by cross-validation. For each cluster formed, we subtracted 10 per cent from the minimum amount in the cluster and added 10 per cent to the maximum amount in the cluster to provide the minimum and maximum borders of the cluster.

4.3. Single-card versus Multi-card Focus of CBM

To the best of our knowledge, former unsupervised fraud models have been built on card-specific transaction data. However, a cardholder may hold multiple cards issued by the same bank. Therefore, constructing a behavior model on all cards of a cardholder rather than a single card is expected to improve the fraud detection performance.

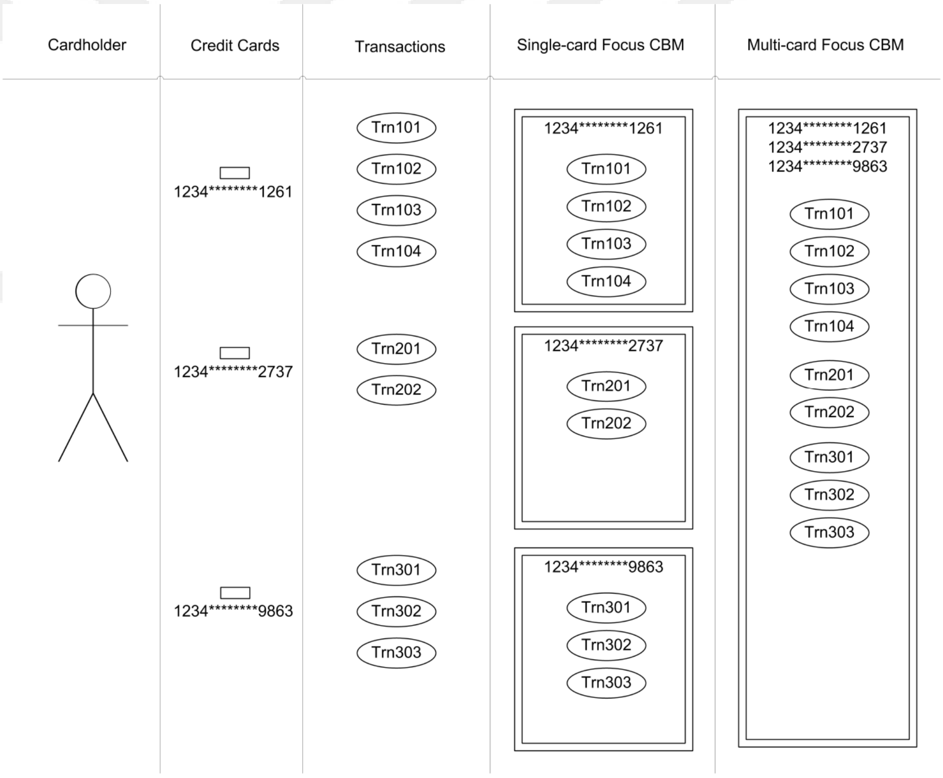

Figure 4.4. Single-card focus versus multi-card focus of CBM.

Single-card versus multi-card focus point is detailed in Figure 4.4 in which the cardholder has three credit cards as 1234********1261, 1234********2737 and 1234********9863. The cardholder made four transactions with 1234********1261, two transactions with 1234********2737 and three transactions with 1234********9863. Three separate CBM's could be built for each of his credit cards. For each of these CBM's, only those transactions made with the corresponding credit card are used. This approach is called single-card focus of CBM. Alternatively, one CBM can be built using transactions made with all cards of that cardholder. This approach is called multi-card focus of CBM.

For further clarification, an example is given using sample transactions. To start with, the transaction records for a cardholder can be seen in Table 4.1. In Table 4.1, transactions have two different MCC's. MCC: 5411 indicates "Grocery Stores, Supermarkets" and MCC: 5094 indicates "Precious Stones and Metals, Watches and Jewelry".

Table 4.1. Transactions of a single credit card

Cardholder No	Card No	MCC	Amount	Date	Time
12345	6789********4321	5411	150.00	03/04/2012	19:25
12345	6789********4321	5411	132.00	10/04/2012	15:00
12345	6789********4321	5411	77.85	12/05/2012	20:05
12345	6789********4321	5094	7500.00	17/05/2012	20:22
12345	6789********4321	5094	12000.00	20/06/2012	19:37
12345	6789********4321	5411	170.09	15/07/2012	19:45

It can be noticed that the cardholder has made four supermarket (MCC: 5411) purchases and two jewelry (MCC: 5094) purchases with the same card. He has spent amounts between 77.85 and 170.09 in a supermarket, while he has spent amounts between 7500.00 and 12000.00 in the jewelry store. Based on these records, it is obvious that, the amount regions for different MCC's may be different. Therefore, transactions with different MCC's are considered separately. Now, suppose that the cardholder makes a new supermarket purchase whose transaction record is as in Table 4.2.

Table 4.2. New supermarket purchase transaction for the credit card

Cardholder No	Card No	MCC	Amount	Date	Time
12345	6789********4321	5411	350.00	19/08/2012	19:54

Up to that time, the cardholder's supermarket purchase amounts were between 77.85 and 170.09. These past transactions are shown as diamonds in Figure 4.5. The clusters formed by these transactions are shown as ovals. The new transaction of amount 350.00 is shown as a square. As can be seen in Figure 4.5, the new transaction falls out of two clusters formed. In this situation, CBM gives an alarm indicating that the cardholder behaves in a different manner.

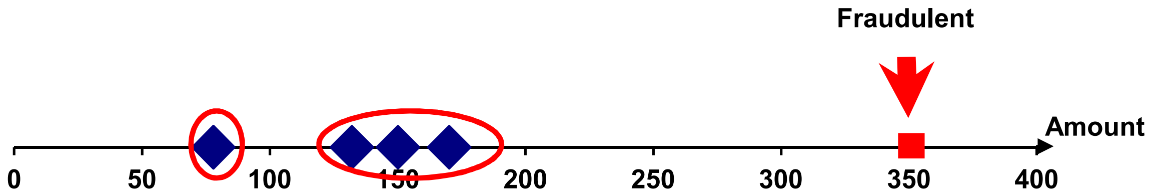

Figure 4.5. Past supermarket transaction clusters and the new supermarket transaction.

Let’s assume that our example cardholder holds a second credit card with the transactions listed in Table 4.3. The cardholder has made four airline purchases (MCC: 4511) and one supermarket (MCC: 5411) purchase with this card.

Table 4.3. Transactions of the cardholder's other credit card

Cardholder No	Card No	MCC	Amount	Date	Time
12345	3456********0912	4511	1800.00	07/03/2012	17:45
12345	3456********0912	4511	750.00	19/04/2012	19:02
12345	3456********0912	4511	150.00	27/05/2012	19:30
12345	3456********0912	5411	128.00	10/06/2012	15:00
12345	3456********0912	4511	1500.00	30/06/2012	18:00

Now, suppose that the cardholder makes an airline ticket purchase (MCC: 4511) whose transaction record is as in Table 4.4.

Table 4.4. New airline purchase transaction for the credit card

Cardholder No	Card No	MCC	Amount	Date	Time
12345	6789********4321	4511	1600.00	19/09/2012	21:54

The airline ticket purchase in Table 4.4 has been made by using the card which has four supermarket (MCC: 5411) purchases and two jewelry (MCC: 5094) purchases. If just the transactions of that card are considered and the other transactions of the cardholder are ignored, an alarm is raised since there are no previous airline ticket purchases for that card. However, if all transactions of that cardholder are considered, it is noticed that the cardholder has made four airline purchases before. These past purchases are plotted as diamonds in Figure 4.6. The clusters are shown as ovals. The new transaction in Table 4.4, which is plotted as a square in Figure 4.6, falls into one of these clusters. Therefore, this new transaction matches with the airline ticket purchase behavior of the cardholder. As a result, CBM does not raise an alarm.

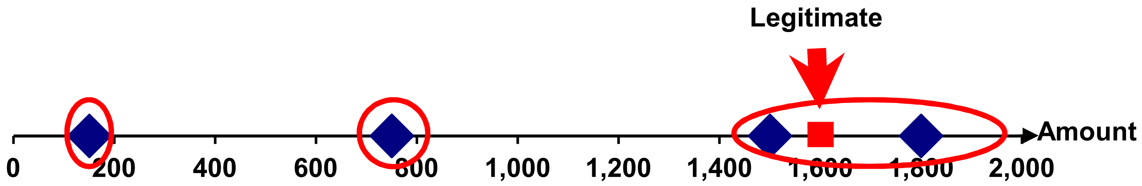

Figure 4.6. Past airline transaction clusters (the other card) and the new transaction.

Focusing on such scenarios, fraud detection performance in single-card and multi-card focus of CBM are evaluated. To the best of our knowledge, this focus point is the first in credit card fraud detection domain.

4.4. Holiday Season Spending Focus of CBM

In most countries, there are holidays such as New Year and religious holidays. It is known that spending is usually higher in holiday seasons. As an example, Christmas is the biggest holiday in the United States. Black Friday is the Friday following Thanksgiving Day in the United States, often regarded as the beginning of the Christmas shopping season. In 2011, each of 152 million Black Friday shoppers spent about $400 on average resulting in $52 billion sales [69]. A similar spending behavior is seen also in Europe. Irish consumers spent about €257 million with an average of €155 for online Christmas shopping [70]. Therefore, it is obvious that holiday seasons may be taken into consideration while building a behavior model for credit card fraud detection.

Holiday season spending focus point is detailed in Figure 4.7. The cardholder having one credit card made two transactions on the holidays and two transactions on the other days. Two separate CBM’s can be trained; one for holidays and one for the other days. For each of these CBM’s, only transactions made within the corresponding days are used. This approach is called holiday season focus of CBM. Alternatively, one CBM can be built for all days. In this approach, all transactions throughout a year are used. This approach is called all time CBM and it treats the whole year as a homogeneous spending period.

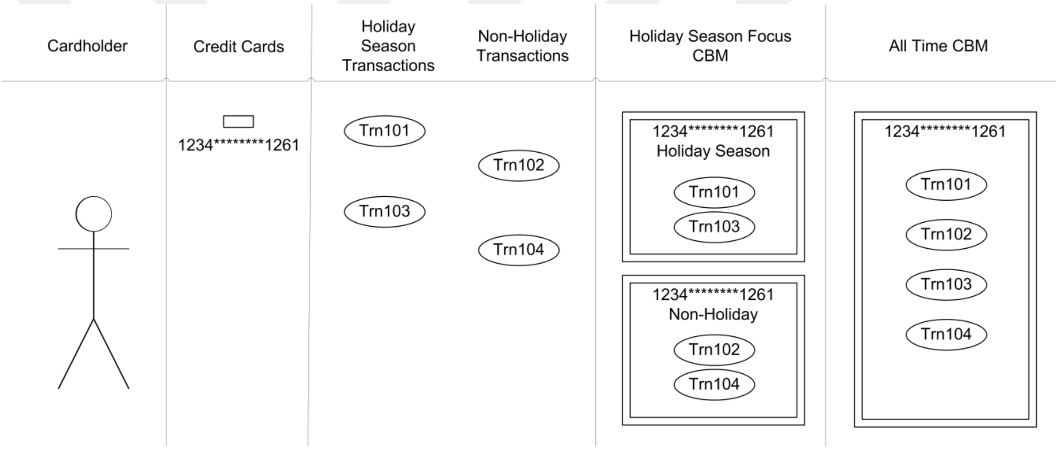

Figure 4.7. Holiday season focus CBM versus all time CBM.

To explain holiday season focus point with an example, the transactions of a cardholder can be seen in Table 4.5. Since this research is evaluated using the transaction dataset from a Turkish bank, Turkish holidays are considered. These are two religious holidays which are depicted as Holiday 1 and Holiday 2 and the New Year. It is expected that if transactions within a number of days before the start of a holiday are taken into account, holiday season spending focus will be meaningful.

The cardholder makes clothing (MCC: 5651) purchases in holiday seasons with amounts between 800.00 and 1250.00. However, the clothing purchase amounts are between 100.00 and 132.00 on non-holidays. In other words, the cardholder spends much more for clothing in holidays.

Table 4.5. Transactions during or out of holiday seasons

Cardholder No	Card No	MCC	Amount	Date	Time	Holiday
35791	4680********5791	5651	125.00	09/07/2012	17:45	No
35791	4680********5791	5651	1150.00	16/08/2012	19:25	Holiday 1
35791	4680********5791	5651	132.00	13/09/2012	15:00	No
35791	4680********5791	5651	1250.00	22/10/2012	19:12	Holiday 2
35791	4680********5791	5651	100.00	11/11/2012	20:05	No
35791	4680********5791	5651	800.00	30/12/2012	20:22	New Year

Now suppose that the card is stolen and a fraudulent clothing purchase is made as given in Table 4.6.

Table 4.6. Fraudulent clothing purchase transaction for the credit card

Cardholder No	Card No	MCC	Amount	Date	Time	Holiday
35791	4680********5791	5651	1200.00	10/01/2013	21:30	No

The transaction date is not a holiday. If holidays are considered as periods of different spending behavior, then this transaction should be evaluated considering the past non-holiday transactions. These past transactions are shown as diamonds in Figure 4.8. One cluster is formed, which is shown as an oval. The new cluster, shown as a square, does not fall into that cluster. Therefore, CBM raises an alarm for the new transaction and the fraudulent transaction is rejected.

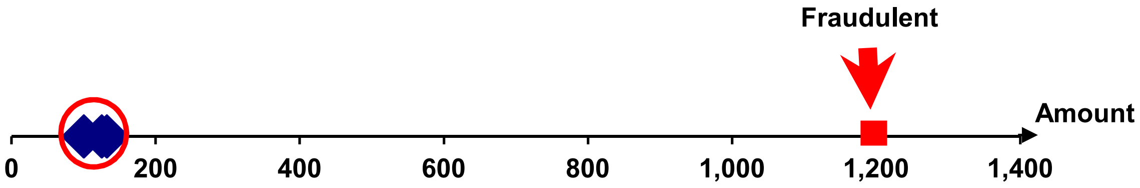

Figure 4.8. Past non-holiday clothing transaction clusters and the fraudulent transaction.

If the holidays are not considered as special spending periods, CBM does not raise an alarm for this transaction since there are past transactions close to this amount. Details can be seen in Figure 4.9, in which all past clothing transactions of the cardholder are shown as diamonds. Three clusters formed are shown as ovals. The new transaction is shown as a square. This new transaction falls in one of the clusters. Therefore, alarm is not raised for the fraudulent transaction worth of 1200.00.

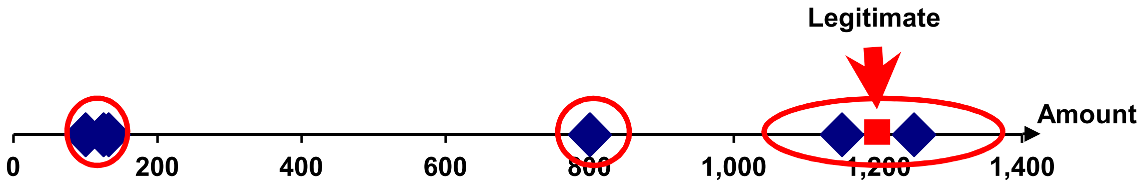

Figure 4.9. Past clothing transaction clusters and the fraudulent clothing transaction.

Focusing on such scenarios, fraud detection performance in holiday season focus CBM and all time CBM are evaluated. To the best of our knowledge, this focus point is also the first in credit card fraud detection domain.

4.5. Time of Day Focus of CBM

As can be observed in daily life, the spending behavior may change in relation to the time of day. For example, a person can have a reasonable cost lunch but an expensive dinner both in the same restaurant. Based on such observations, it is expected that considering spending behavior separately for different times of the day will improve the fraud detection performance. In our approach, the day is divided into four 6-hour intervals as seen in Table 4.7.

Table 4.7. Time of day and related time intervals

	Time of Day Name	Time of Day Interval
1	Night	00:00:00-05:59:59
2	Morning	06:00:00-11:59:59
3	Afternoon	12:00:00-17:59:59
4	Evening	18:00:00-23:59:59

The time of day focus point is detailed in Figure 4.10. The cardholder has one credit card with two transactions at night, one transaction in the afternoon and one transaction in the evening.

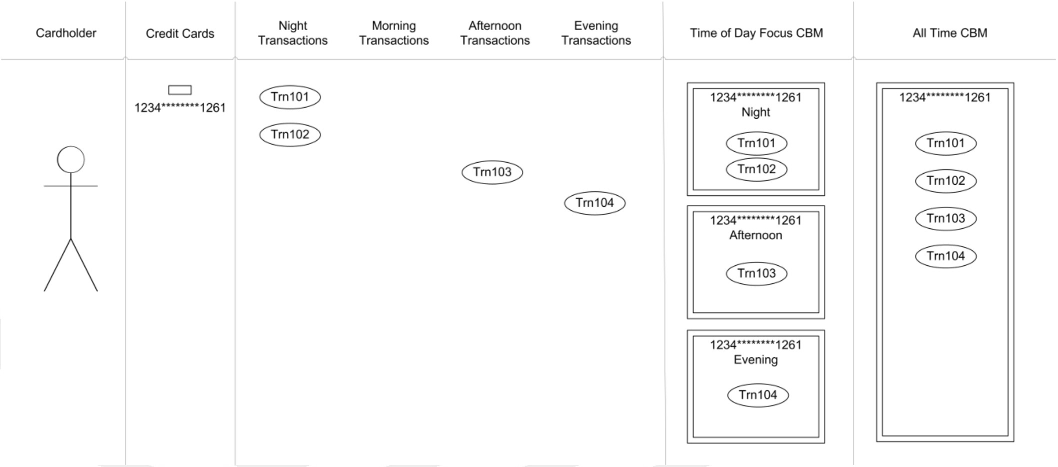

Figure 4.10. Time of day focus CBM versus all time CBM.

Three separate CBM’s may be built for each time of day. For each of these CBM’s, only transactions made with the corresponding time of day interval are used. Alternatively, one CBM can be built for all time in which all transactions are used ignoring time of day.

For further clarification, an example is given using sample transactions stated in Table 4.8. Let’s consider a cardholder who always has dinner in the evening. Suppose that the cardholder has four restaurant purchases (MCC: 5812) as seen in Table 4.8.

Table 4.8. Restaurant purchase transactions of the cardholder

Cardholder No	**Card No**	**MCC**	**Amount**	**Date**	**Time**	**Time of Day**
24680	1122********3344	5812	70.00	07/03/2012	20:45	Evening
24680	1122********3344	5812	50.00	03/04/2012	21:25	Evening
24680	1122********3344	5812	100.00	10/04/2012	22:00	Evening
24680	1122********3344	5812	80.00	19/04/2012	22:32	Evening

Now suppose that the card is stolen and a fraudulent restaurant purchase is made at night as given in Table 4.9.

Table 4.9. Fraudulent restaurant purchase transaction

Cardholder No	Card No	MCC	Amount	Date	Time	Time of Day
24680	1122********3344	5812	75.00	25/04/2013	03:30	Night

If time of day focus is taken into account, then CBM raises an alarm for this purchase since previously no restaurant purchases have been made during the night. In the other case, where time of day focus is ignored, no alarm is raised since there are four former restaurant purchases in the same amount range as shown in Figure 4.11. Former restaurant purchases are shown as diamonds and they form three clusters shown as ovals. The fraudulent purchase is shown as a square in Figure 4.11. The fraudulent purchase falls into one of three clusters. Therefore, an alarm is not raised for the fraudulent purchase.

Focusing on such scenarios, fraud detection performance in time of day focus CBM and all time CBM are evaluated. To the best of our knowledge, this focus point is also the first in credit card fraud detection domain.

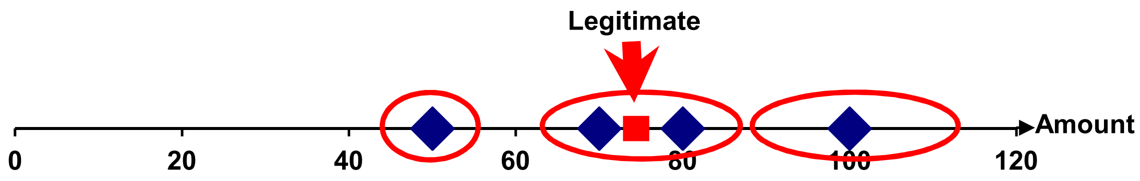

Figure 4.11. Past restaurant transaction clusters and the fraudulent transaction.

4.6. Inflation Focus of CBM

In economics, inflation is a rise in the general level of prices of goods and services in an economy over a period of time. In this research, past transactions from a time span of years is used. Therefore, applying inflation correction to the past transaction records before building the CBM’s is proposed. For this purpose Consumer Price Index (CPI) is used. CPI measures changes in the price level of a market basket of consumer goods and services purchased by households. A partial CPI in Turkey is shown in Table 4.10 [71].

Table 4.10. Partial Consumer Price Index in Turkey, 2003-2013

Year	**Jan.**	**Feb.**	**Mar.**	**Apr.**	**May**	**June**	**July**	**Aug.**	**Sept.**	**Oct.**	**Nov.**	**Dec.**
2013	216.74	217.39	218.83	219.75								
2012	201.98	203.12	203.96	207.05	206.61	204.76	204.29	205.43	207.55	211.62	212.42	213.23
…												
2004	104.81	105.35	106.36	106.89	107.35	107.21	107.72	108.54	109.57	112.03	113.50	113.86
2003	94.77	96.23	98.12	99.09	100.04	100.12	99.93	100.09	101.44	102.38	103.68	104.12

To explain inflation correction method with an example, 100 Turkish Lira’s (TL) in January, 2003 is projected to April, 2013 as in Equation 4.1.

$$Value_{April,2013} = \frac{Amount_{January,2003} * Index_{April,2013}}{Index_{January,2003}} = \frac{100.00 * 219.75}{94.77} = 232.00 \tag{4.1}$$

In this example, a 100 TL transaction record from January, 2003 is projected as 232 TL while building CBM for April, 2013.

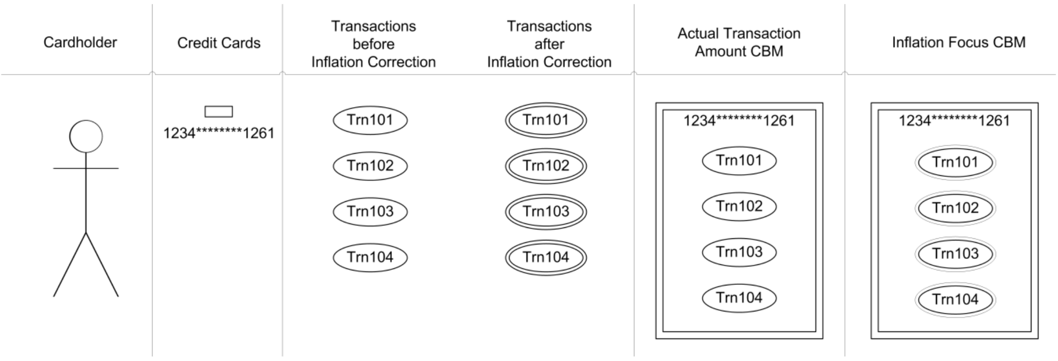

Figure 4.12. Actual transaction amount CBM versus inflation focus CBM.

The inflation focus point is detailed in Figure 4.12. The cardholder having one credit card made four transactions. One CBM may be built using the actual transaction amounts. This approach is called actual transaction amount CBM. Alternatively, inflation correction may be applied on the actual transaction amounts and CBM may be built using the corrected transaction amounts. This approach is called inflation focus CBM.

Considering the inflation fact, fraud detection performance of actual transaction amount CBM and inflation focus CBM are evaluated. To the best of our knowledge, this focus point is also the first in credit card fraud detection domain.

4.7. CBM Experimental Results and Analysis

This section is organized in terms of four focus points of CBM.

4.7.1. Single-card versus Multi-card Focus of CBM

Single-card versus Multi-card Focus of CBM aims to analyze single-card and multi-card CBM's. To the best of our knowledge, previous unsupervised fraud models have been built on card-specific transaction datasets. However, as pointed out in CBM Evaluation Dataset, a cardholder may hold multiple cards issued by the same bank. Therefore, CBM's for all cards of a cardholder rather than a single card have been built. The alarm rates for multi-card focus CBM's and single-card focus CBM's are shown in Table 4.11. Single-card focus CBM's have significantly higher alarm rates than multi-card focus CBM's.

Table 4.11. Alarm rates for multi-card and single-card CBM's

Consider Holidays	Consider Time of Day	Consider Inflation	Alarm Rate	
			Multi-card	Single-card
No	No	No	20.21%	28.42%
Yes	No	No	19.56%	28.03%
No	Yes	No	29.60%	40.68%
Yes	Yes	No	28.94%	40.29%
No	No	Yes	20.34%	29.34%
Yes	No	Yes	19.95%	28.94%
No	Yes	Yes	29.47%	40.29%
Yes	Yes	Yes	29.20%	39.90%

As seen in the sensitivity column of Table 4.12, single-card CBM's have significantly higher sensitivity than multi-card CBM's in the first six cases. The sensitivity values are equal in the last two cases. In other words, single-card CBM’s detected more fraudulent transactions than multi-card CBM’s in the first six cases whereas both detected equal numbers of fraudulent transactions in the last two cases. The maximum sensitivity value is 59.46% and this value means that about 60 per cent of fraudulent transactions are detected by CBM.

Multi-card CBM’s have significantly higher specificity than single-card CBM’s in all cases as seen in specificity column of Table 4.12. In other words, multi-card CBM’s have smaller false alarm rates than single-card CBM’s. The same fact can also be seen in false positive rate column of Table 4.12. In the worst case, single-card CBM’s give false alarm for 39.86% of legitimate transactions whereas multi-card CBM’s give false alarm for 28.49% of legitimate transactions.

Table 4.12. Evaluation results for multi-card and single-card CBM’s

Consider Holidays	Consider Time of Day	Consider Inflation	Sensitivity		Specificity		False Positive Rate	
			Multi-card	Single-card	Multi-card	Single-card	Multi-card	Single-card
No	No	No	43.24%	**54.05%**	**80.96%**	72.88%	**19.04%**	27.12%
Yes	No	No	45.95%	**56.76%**	**81.78%**	73.42%	**18.22%**	26.58%
No	Yes	No	51.35%	**56.76%**	**71.51%**	60.14%	**28.49%**	39.86%
Yes	Yes	No	54.05%	**59.46%**	**72.33%**	60.68%	**27.67%**	39.32%
No	No	Yes	43.24%	**54.05%**	**80.82%**	71.92%	**19.18%**	28.08%
Yes	No	Yes	45.95%	**56.76%**	**81.37%**	72.47%	**18.63%**	27.53%
No	Yes	Yes	56.76%	56.76%	**71.92%**	60.55%	**28.08%**	39.45%
Yes	Yes	Yes	59.46%	59.46%	**72.33%**	61.10%	**27.67%**	38.90%

Consider Holidays	Consider Time of Day	Consider Inflation	Precision		Negative Predictive Value		Accuracy		F_1 Score	
			Multi-card	Single-card	Multi-card	Single-card	Multi-card	Single-card	Multi-card	Single-card
No	No	No	**10.32%**	9.17%	96.57%	96.90%	**79.14%**	71.97%	**16.66%**	15.68%
Yes	No	No	**11.33%**	9.77%	96.76%	97.10%	**80.05%**	72.62%	**18.18%**	16.67%
No	Yes	No	**8.37%**	6.73%	96.67%	96.48%	**70.53%**	59.97%	**14.39%**	12.03%
Yes	Yes	No	**9.01%**	7.12%	96.88%	96.72%	**71.45%**	60.63%	**15.45%**	12.72%
No	No	Yes	**10.26%**	8.89%	96.56%	96.86%	**79.01%**	71.06%	**16.58%**	15.27%
Yes	No	Yes	**11.11%**	9.46%	96.74%	97.06%	**79.66%**	71.71%	**17.89%**	16.22%
No	Yes	Yes	**9.29%**	6.80%	97.04%	96.51%	**71.19%**	60.37%	**15.97%**	12.14%
Yes	Yes	Yes	**9.82%**	7.19%	97.24%	96.75%	**71.71%**	61.02%	**16.86%**	12.83%

Precision column of Table 4.12 shows that multi-card CBM's have significantly higher precision than single-card CBM's. In other words, multi-card CBM's have bigger true alarm rates than single-card CBM's. In the worst case, 8.37% of alarms are true for multi-card CBM's whereas 6.73% of alarms are true for single-card CBM's.

Multi-card CBM's and single-card CBM's have no statistically significant difference in negative predictive values as seen in the corresponding column of Table 4.12. In other words, both multi-card CBM's and single-card CBM's have similar count of true "no alarm" decisions.

As seen in accuracy column of Table 4.12, multi-card CBM's have significantly higher accuracy than single-card CBM's. In other words, multi-card CBM's beat single-card CBM's in correct alarm and "no alarm" decisions.

As seen in F_1 score column of Table 4.12, multi-card CBM's have significantly higher F_1 score than single-card CBM's. Multi-card CBM's should be preferred if the bank fraud detection strategy attaches equal importance to sensitivity and specificity.

Single-card CBM's have significantly higher sensitivity than multi-card CBM's. On the other hand, multi-card CBM's beat single-card CBM's in terms of specificity, false positive rate, precision and accuracy. If the bank strategy is to detect as many frauds as possible at the expense of giving more false alarms, single-card CBM's should be preferred. On the other hand, if the bank strategy is giving fewer false alarms at the expense of detecting fewer frauds, multi-card CBM's should be preferred.

4.7.2. Holiday Season Spending Focus of CBM

Holiday Season Spending Focus of CBM aims to take into account holidays in CBM's. Transactions within 3, 5, 7, 9, 11, 13 and 15 days before the start of holiday are taken into account and the best results are obtained for up to 15 days before the start of a holiday. The alarm rates of holiday season focus CBM's and all time CBM's are shown in Table 4.13. Holiday season focus CBM's and all time CBM's have no statistically significant difference in alarm rates.

Table 4.13. Alarm rates for holiday season focus CBM’s and all time CBM’s

Single-card / Multi-card	**Consider Time of Day**	**Consider Inflation**	**Alarm Rate**	
			Consider Holidays: No	**Consider Holidays: Yes**
Multi-card	No	No	20.21%	19.56%
Multi-card	Yes	No	29.60%	28.94%
Single-card	No	No	28.42%	28.03%
Single-card	Yes	No	40.68%	40.29%
Multi-card	No	Yes	20.34%	19.95%
Multi-card	Yes	Yes	29.47%	29.20%
Single-card	No	Yes	29.34%	28.94%
Single-card	Yes	Yes	40.29%	39.90%

As seen in the sensitivity column of Table 4.14, holiday season focus CBM’s have significantly higher sensitivity than all time CBM’s in all cases. In other words, holiday season focus CBM’s detected more fraudulent transactions than all time CBM’s.

Holiday season focus CBM’s and all time CBM’s have no statistically significant difference in specificity and false positive rate as seen in the corresponding columns of Table 4.14. In other words, holiday season focus CBM’s and all time CBM’s have similar false alarm rates.

Precision column of Table 4.14 shows that holiday season focus CBM’s have significantly higher precision than all time CBM’s in one case whereas there is no statistically significant difference in the other cases.

Holiday season focus CBM’s and all time CBM’s have no statistically significant difference in negative predictive values as seen in the corresponding column of Table 4.14. In other words, holiday season focus CBM’s and all time CBM’s have similar count of true “no alarm” decisions.

As seen in accuracy column of Table 4.14, holiday season focus CBM’s and all time CBM’s have no statistically significant difference in accuracy. In other words, holiday season focus CBM’s and all time CBM’s have similar count of correct alarm and “no alarm” decisions.

As seen in F_1 score column of Table 4.14, holiday season focus CBM’s have significantly higher F_1 score than all time CBM’s. This is not a surprising result since holiday season focus CBM’s beat all time CBM’s in terms of sensitivity whereas they do not have significant difference in terms of precision.

Table 4.14. Evaluation results for holiday season focus CBM’s and all time CBM’s

			Sensitivity		Specificity		False Positive Rate	
Single-card / Multi-card	**Consider Time of Day**	**Consider Inflation**	**Cons. Holiday: No**	**Cons. Holiday: Yes**	**Cons. Holiday: No**	**Cons. Holiday: Yes**	**Cons. Holiday: No**	**Cons. Holiday: Yes**
Multi-card	No	No	43.24%	**45.95%**	80.96%	81.78%	19.04%	18.22%
Multi-card	Yes	No	51.35%	**54.05%**	71.51%	72.33%	28.49%	27.67%
Single-card	No	No	54.05%	**56.76%**	72.88%	73.42%	27.12%	26.58%
Single-card	Yes	No	56.76%	**59.46%**	60.14%	60.68%	39.86%	39.32%
Multi-card	No	Yes	43.24%	**45.95%**	80.82%	81.37%	19.18%	18.63%
Multi-card	Yes	Yes	56.76%	**59.46%**	71.92%	72.33%	28.08%	27.67%
Single-card	No	Yes	54.05%	**56.76%**	71.92%	72.47%	28.08%	27.53%
Single-card	Yes	Yes	56.76%	**59.46%**	60.55%	61.10%	39.45%	38.90%

			Precision		Negative Predictive Value		Accuracy		F_1 Score	
Single-card / Multi-card	**Consider Time of Day**	**Consider Inflation**	**Cons. Holiday : No**	**Cons. Holiday: Yes**	**Cons. Holiday : No**	**Cons. Holiday: Yes**	**Cons. Holiday : No**	**Cons. Holiday: Yes**	**Cons. Holiday : No**	**Cons. Holiday: Yes**
Multi-card	No	No	10.32%	**11.33%**	96.57%	96.76%	79.14%	80.05%	16.66%	**18.18%**
Multi-card	Yes	No	8.37%	9.01%	96.67%	96.88%	70.53%	71.45%	14.39%	**15.45%**
Single-card	No	No	9.17%	9.77%	96.90%	97.10%	71.97%	72.62%	15.68%	**16.67%**
Single-card	Yes	No	6.73%	7.12%	96.48%	96.72%	59.97%	60.63%	12.03%	**12.72%**
Multi-card	No	Yes	10.26%	11.11%	96.56%	96.74%	79.01%	79.66%	16.58%	**17.89%**
Multi-card	Yes	Yes	9.29%	9.82%	97.04%	97.24%	71.19%	71.71%	15.97%	**16.86%**
Single-card	No	Yes	8.89%	9.46%	96.86%	97.06%	71.06%	71.71%	15.27%	**16.22%**
Single-card	Yes	Yes	6.80%	7.19%	96.51%	96.75%	60.37%	61.02%	12.14%	**12.83%**

Holiday season focus CBM’s have significantly higher sensitivity than all time CBM’s in all cases. In terms of precision, holiday season focus CBM’s beat all time CBM’s in one out of eight cases whereas there is no statistically significant difference in the other cases. Additionally, holiday season focus CBM’s and all time CBM’s have no statistically significant difference in specificity, false positive rate, negative predictive value and accuracy. Results show that holiday season focus CBM’s should be preferred.

4.7.3. Time of Day Focus of CBM

Time of Day Focus of CBM takes into account time of day in CBM’s. The alarm rates for time of day focus CBM’s and all time CBM’s are shown in Table 4.15. Time of day focus CBM’s have significantly higher alarm rates than all time CBM’s.

As seen in the sensitivity column of Table 4.16, time of day focus CBM’s have significantly higher sensitivity than all time CBM’s. In other words, time of day focus CBM’s detected more fraudulent transactions than all time CBM’s.

All time CBM’s have significantly higher specificity than time of day focus CBM’s in all cases as seen in specificity column of Table 4.16. In other words, all time CBM’s have smaller false alarm rates than time of day focus CBM’s. The same fact can also be seen in false positive rate column of Table 4.16. In the worst case, time of day focus CBM’s give false alarm for 39.86% of legitimate transactions whereas all time CBM’s give false alarm for 27.12% of legitimate transactions.

Table 4.15. Alarm rates for time of day focus CBM’s and all time CBM’s

Single-card / Multi-card	Consider Holidays	Consider Inflation	Alarm Rate	
			Consider Time of Day: No	Consider Time of Day: Yes
Multi-card	No	No	20.21%	29.60%
Multi-card	Yes	No	19.56%	28.94%
Single-card	No	No	28.42%	40.68%
Single-card	Yes	No	28.03%	40.29%
Multi-card	No	Yes	20.34%	29.47%
Multi-card	Yes	Yes	19.95%	29.20%
Single-card	No	Yes	29.34%	40.29%
Single-card	Yes	Yes	28.94%	39.90%

Precision column of Table 4.16 shows that all time CBM’s have significantly higher precision than time of day focus CBM’s in seven out of eight cases. In other words, all time CBM’s have bigger true alarm rates than time of day focus CBM’s. In the worst case, 8.89% of alarms are true for all time CBM’s whereas 6.80% of alarms are true for time of day focus CBM’s.

All time CBM’s and time of day focus CBM’s have no statistically significant difference in negative predictive values as seen in the corresponding column of Table 4.16. In other words, all time CBM’s and time of day focus CBM’s have similar count of true “no alarm” decisions.

Table 4.16. Evaluation results for time of day focus CBM’s and all time CBM’s

			Sensitivity		**Specificity**		**False Positive Rate**	
Single-card / Multi-card	**Consider Holidays**	**Consider Inflation**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**
Multi-card	No	No	43.24%	**51.35%**	**80.96%**	71.51%	**19.04%**	28.49%
Multi-card	Yes	No	45.95%	**54.05%**	**81.78%**	72.33%	**18.22%**	27.67%
Single-card	No	No	54.05%	**56.76%**	**72.88%**	60.14%	**27.12%**	39.86%
Single-card	Yes	No	56.76%	**59.46%**	**73.42%**	60.68%	**26.58%**	39.32%
Multi-card	No	Yes	43.24%	**56.76%**	**80.82%**	71.92%	**19.18%**	28.08%
Multi-card	Yes	Yes	45.95%	**59.46%**	**81.37%**	72.33%	**18.63%**	27.67%
Single-card	No	Yes	54.05%	**56.76%**	**71.92%**	60.55%	**28.08%**	39.45%
Single-card	Yes	Yes	56.76%	**59.46%**	**72.47%**	61.10%	**27.53%**	38.90%

			Precision		**Negative Predictive Value**		**Accuracy**		**F$_1$ Score**	
Single-card / Multi-card	**Consider Holidays**	**Consider Inflation**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**	**Cons. Time of Day: No**	**Cons. Time of Day: Yes**
Multi-card	No	No	**10.32%**	8.37%	96.57%	96.67%	**79.14%**	70.53%	**16.66%**	14.39%
Multi-card	Yes	No	**11.33%**	9.01%	96.76%	96.88%	**80.05%**	71.45%	**18.18%**	15.45%
Single-card	No	No	**9.17%**	6.73%	96.90%	96.48%	**71.97%**	59.97%	**15.68%**	12.03%
Single-card	Yes	No	**9.77%**	7.12%	97.10%	96.72%	**72.62%**	60.63%	**16.67%**	12.72%
Multi-card	No	Yes	10.26%	9.29%	96.56%	97.04%	**79.01%**	71.19%	**16.58%**	15.97%
Multi-card	Yes	Yes	**11.11%**	9.82%	96.74%	97.24%	**79.66%**	71.71%	**17.89%**	16.86%
Single-card	No	Yes	**8.89%**	6.80%	96.86%	96.51%	**71.06%**	60.37%	**15.27%**	12.14%
Single-card	Yes	Yes	**9.46%**	7.19%	97.06%	96.75%	**71.71%**	61.02%	**16.22%**	12.83%

As seen in accuracy column of Table 4.16, all time CBM’s have significantly higher accuracy than time of day focus CBM’s. In other words, all time CBM’s beat time of day focus CBM’s in correct alarm and “no alarm” decisions.

As seen in F_1 score column of Table 4.16, all time CBM’s have significantly higher F_1 score than time of day focus CBM’s. All time CBM’s should be preferred if the bank fraud detection strategy attaches equal importance to sensitivity and specificity.

Time of day focus CBM’s have significantly higher sensitivity than all time CBM’s. On the other hand, all time CBM’s beat time of day focus CBM’s in terms of specificity, false positive rate, precision and accuracy. If the bank strategy is to detect as many frauds as possible at the expense of giving more false alarms, time of day focus CBM’s should be preferred. On the other hand, if the bank strategy is giving fewer false alarms at the expense of detecting fewer frauds, all time CBM’s should be preferred.

4.7.4. Inflation Focus of CBM

Inflation Focus of CBM aims to take into account inflation in CBM’s. The alarm rates of CBM with ignoring and considering inflation are shown in Table 4.17. In all cases, CBM’s which ignore inflation and CBM’s which consider inflation have no statistically significant difference. There are two reasons for this result. Firstly, inflation rates are low. Secondly, most of the transactions in the case study have occurred in the recent years.

As seen in Table 4.18, CBM’s which ignore inflation and CBM’s which consider inflation have statistically significant difference only in sensitivity and F_1 score columns of the third and the fourth rows. In these rows, evaluation results for multi-card CBM’s which consider time of day are stated. CBM’s which consider inflation have significantly higher sensitivity and F_1 score than CBM’s which ignore inflation. In other words, CBM’s which consider inflation detected more fraudulent transactions than CBM’s which ignore inflation. Additionally, higher F_1 score implies that considering inflation improves sensitivity and precision combined.

Table 4.17. Alarm rates for CBM's ignoring and considering inflation

Single-card / Multi-card	Consider Holidays	Consider Time of Day	Alarm Rate	
			Consider Inflation: No	Consider Inflation: Yes
Multi-card	No	No	20.21%	20.34%
Multi-card	Yes	No	19.56%	19.95%
Multi-card	No	Yes	29.60%	29.47%
Multi-card	Yes	Yes	28.94%	29.20%
Single-card	No	No	28.42%	29.34%
Single-card	Yes	No	28.03%	28.94%
Single-card	No	Yes	40.68%	40.29%
Single-card	Yes	Yes	40.29%	39.90%

Table 4.18. Evaluation results for CBM's ignoring and considering inflation

			Sensitivity		Specificity		False Positive Rate	
Single-card / Multi-card	Consider Holidays	Consider Time of Day	Cons. Infl.: No	Cons. Infl.: Yes	Cons. Infl.: No	Cons. Infl.: Yes	Cons. Infl.: No	Cons. Infl.: Yes
Multi-card	No	No	43.24%	43.24%	80.96%	80.82%	19.04%	19.18%
Multi-card	Yes	No	45.95%	45.95%	81.78%	81.37%	18.22%	18.63%
Multi-card	No	Yes	51.35%	**56.76%**	71.51%	71.92%	28.49%	28.08%
Multi-card	Yes	Yes	54.05%	**59.46%**	72.33%	72.33%	27.67%	27.67%
Single-card	No	No	54.05%	54.05%	72.88%	71.92%	27.12%	28.08%
Single-card	Yes	No	56.76%	56.76%	73.42%	72.47%	26.58%	27.53%
Single-card	No	Yes	56.76%	56.76%	60.14%	60.55%	39.86%	39.45%
Single-card	Yes	Yes	59.46%	59.46%	60.68%	61.10%	39.32%	38.90%

			Precision		Negative Predictive Value		Accuracy		F_1 Score	
Single-card / Multi-card	Consider Holidays	Consider Time of Day	Cons. Infl.: No	Cons. Infl.: Yes	Cons. Infl.: No	Cons. Infl.: Yes	Cons. Infl.: No	Cons. Infl.: Yes	Cons. Infl.: No	Cons. Infl.: Yes
Multi-card	No	No	10.32%	10.26%	96.57%	96.56%	79.14%	79.01%	16.66%	16.58%
Multi-card	Yes	No	11.33%	11.11%	96.76%	96.74%	80.05%	79.66%	18.18%	17.89%
Multi-card	No	Yes	8.37%	9.29%	96.67%	97.04%	70.53%	71.19%	14.39%	**15.97%**
Multi-card	Yes	Yes	9.01%	9.82%	96.88%	97.24%	71.45%	71.71%	15.45%	**16.86%**
Single-card	No	No	9.17%	8.89%	96.90%	96.86%	71.97%	71.06%	15.68%	15.27%
Single-card	Yes	No	9.77%	9.46%	97.10%	97.06%	72.62%	71.71%	16.67%	16.22%
Single-card	No	Yes	6.73%	6.80%	96.48%	96.51%	59.97%	60.37%	12.03%	12.14%
Single-card	Yes	Yes	7.12%	7.19%	96.72%	96.75%	60.63%	61.02%	12.72%	12.83%

The maximum sensitivity value is 59.46% which means that about 60 per cent of fraudulent transactions are detected. In only two cases, CBM's which consider inflation have significantly higher sensitivity than CBM's which ignore inflation.

In all other cases and criteria, CBM's which consider inflation and CBM's which ignore inflation have no statistically significant difference. Still, it is reasonable to prefer CBM's which consider inflation.

To provide the basis of ROC analysis, we define a function to calculate the score values for transactions. Since transactions having different MCC's have different transaction amount ranges, we use MCC as a part of the scoring function. For instance, an average jewelry transaction has naturally a higher amount than an average restaurant transaction. Considering this fact, the transaction amounts in the test set are normalized to a range of 0-1000 to provide score values. The jewelry transaction with the highest amount is scored as 1000 whereas the jewelry transaction with the lowest amount is scored as 0. The same logic is applied to other MCC's separately. In other words, the score value for each transaction is calculated as a function of transaction amount and MCC. Having score values between 0 and 1000, we plotted the ROC curve by using 50 threshold points in increments of 20.

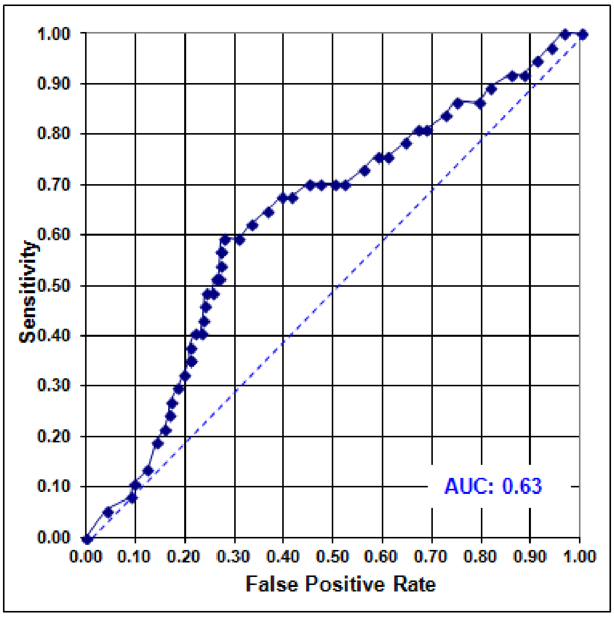

Figure 4.13. CBM ROC curve.

Multi-card CBM’s considering holiday, ignoring time of day and inflation provides the maximum AUC value among all CBM cases. The corresponding ROC curve for CBM is shown in Figure 4.13. CBM achieves Area Under Curve (AUC) value of 0.63.

CBM is fine-tuned by using real credit card transactions dataset from a leading bank in Turkey and credit card fraud detection accuracy is evaluated comparatively from four focus points of view.

5. OPWEM: OPTIMISTIC, PESSIMISTIC AND WEIGHTED VOTING IN AN ENSEMBLE OF MODELS FOR DETECTING CREDIT CARD FRAUD

In OPWEM, we propose combining multiple models in credit card fraud detection domain. We propose the use of six known supervised models: Decision Tree (DT), Random Forest (RF), Bayesian Network (BN), Naïve Bayes (NB), Support Vector Machine (SVM) and K*models. These are combined into an ensemble of AI models for the detection of credit card fraud. Each AI model in the ensemble is trained using the same training set. When a new credit card transaction occurs, each AI model judges whether the transaction is legitimate or fraudulent. The decisions of the AI models are aggregated through voting. In a majority voting mechanism, the decision of the majority is the decision of the ensemble; however, in addition to majority voting, three alternative voting mechanisms are proposed here: optimistic, pessimistic and weighted voting. In optimistic voting, if at least one AI model judges that the transaction is legitimate, then the decision of the ensemble is that it is legitimate. As the word 'optimistic' implies, the ensemble is optimistic about the transaction and judges it as legitimate even when only one AI model indicates that the transaction is legitimate, while the rest determine it to be fraudulent. In pessimistic voting, when at least one AI model judges that the transaction is fraudulent, the decision of the ensemble is that it is fraudulent. As the word 'pessimistic' implies, the ensemble is pessimistic about the transaction; therefore, the transaction is deemed fraudulent even when only one AI model determines it to be fraudulent. On the one hand, the pessimistic voting mechanism results in higher alarm rates than the optimistic voting mechanism. On the other hand, the issuance of fewer alarms results in fewer fraudulent transactions being detected. In the weighted voting mechanism, specific weights are assigned to each AI model in the ensemble depending on the AI models' fraud detection performance on the validation set. A given bank may choose among these voting mechanisms in accordance with its preferred strategy for fraud detection and its desired false alarm rate. The model proposed is called Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models (OPWEM). To the best of our knowledge, this idea is the first to provide a voting scheme in credit card fraud detection domain. Combining multiple models has been previously proposed in different enterprise systems [3, 4, 5].

This section proposes the use of an ensemble of models for credit card fraud detection in combination with the proposed optimistic, pessimistic and weighted voting mechanisms. The first subsection discusses the process applied for credit card fraud detection. The second subsection describes the models that form the ensemble. The third subsection introduces the ensemble of models and the novel optimistic, pessimistic and weighted voting mechanisms used by this ensemble. The fourth subsection provides experimental results and their analysis.

5.1. OPWEM Credit Card Fraud Detection Process

Our credit card fraud detection approach involves credit card holders, credit cards, credit card transactions, a credit card transaction database, individual models and an ensemble of models operating under optimistic, pessimistic, weighted and majority voting strategies. An overall view of the single-model training process is presented in Figure 5.1. Credit card holders, called cardholders for short, may possess one or more credit cards. They generate credit card transactions for each of their credit cards. These credit card transactions are stored in a credit card transaction database.

The training process applied here trains one model of each type (Decision Tree (DT), Random Forest (RF), Bayesian Network (BN), Naïve Bayes (NB), Support Vector Machine (SVM) and K*) for each cardholder.

The decision process begins when a new transaction occurs. At that time, the individual models for the cardholder in question make their decisions regarding whether the transaction is legitimate or fraudulent. The individual models' decision process is detailed in Figure 5.2.

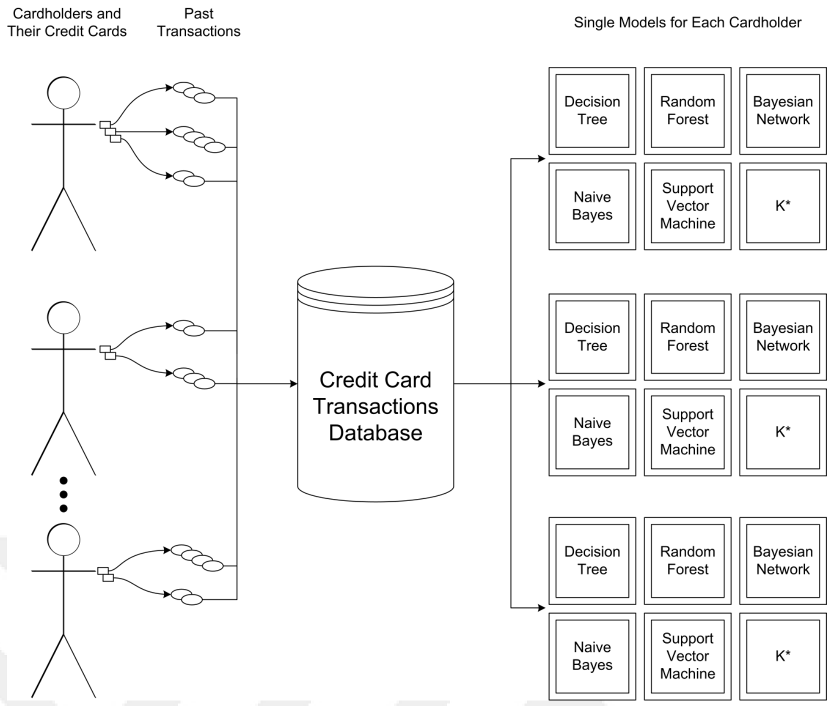

Figure 5.1. Single models training process.

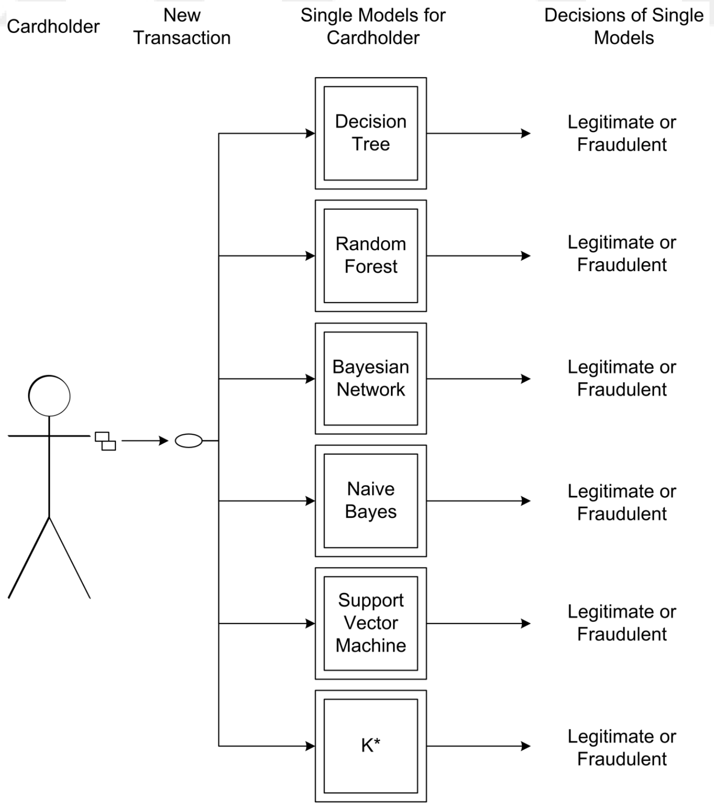

Figure 5.2. Single models decision process.

Taken together, the models form an ensemble. Four types of ensembles may be formed depending on the chosen voting mechanism, namely, majority voting (MJR), optimistic voting (OPT), pessimistic voting (PES) or weighted voting (WGT). The decision process for the ensemble of models is detailed in Figure 5.3. The voting mechanisms are described in Section 5.3.

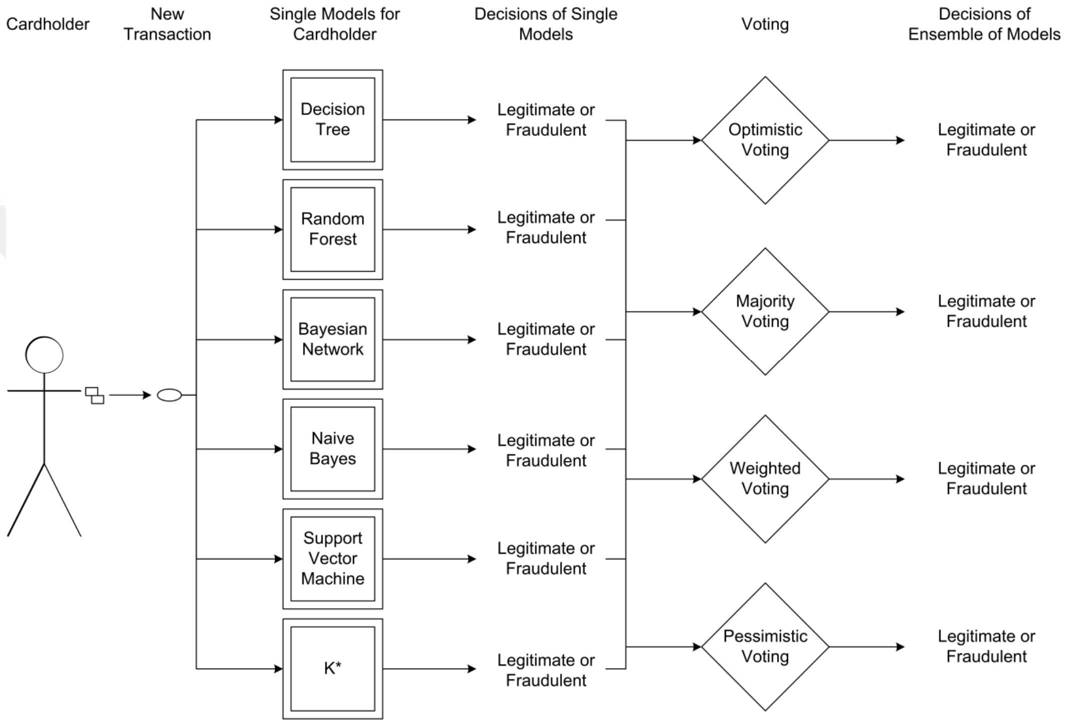

Figure 5.3. Ensemble of models decision process.

5.2. Individual Models for Credit Card Fraud Detection

In this thesis, we use six supervised models that are popular in the credit card fraud detection domain. Initially, the chosen models were the Decision Tree (DT), Random Forest (RF), Neural Network (NN), Bayesian Network (BN), Naïve Bayes (NB) and Support Vector Machine (SVM) models; however, during preliminary work, the NN model was eliminated because of its long computation time and the K* model was added because of its promising accuracy and short computation time [72].

A Decision Tree is a hierarchical model for supervised learning composed of internal decision nodes and terminal leaves. Each decision node implements a test function with discrete outcomes, with which the branches are labeled. Given an input, at each node, a test is applied, and the input is then passed along one of the branches from that node depending on the outcome of the test. This process begins at the root and is repeated recursively until it ends at a leaf node. The class to which the terminal leaf node belongs constitutes the decision of the Decision Tree [73, 74].

Decision Tree models are easy to use and flexible in terms of handling various types of data attributes. However, Decision Tree models can be unstable and overly sensitive to specific training data [28]. In this respect, better prediction results can be obtained by using an ensemble of Decision Trees and aggregating their decisions than by using a single Decision Tree. A Random Forest (RF) is an ensemble of Decision Trees. Each Decision Tree in the ensemble is trained on a separate training set, each of which is created through bootstrapping. Moreover, during training, only a random subset of attributes is selected and considered at each node, rather than all attributes [75].

Bayesian Networks (BNs), also called belief networks, are graphical models that visually represent the interactions between variables. A Bayesian Network is composed of nodes and arcs connecting the nodes. Each node corresponds to a random variable, X, and has a value corresponding to the probability of that random variable, P(X). If a directed arc exists from node X to node Y, this indicates that X has a direct influence on Y. This influence is specified by the conditional probability P(Y|X). A Bayesian Network is a directed acyclic graph, i.e., it contains no cycles [74].

Unlike Bayesian Networks, a Naïve Bayes classifier ignores possible dependencies—correlations—among the inputs. Therefore, Naïve Bayes classifiers are simpler than Bayesian Networks [74].

A Support Vector Machine (SVM) is formulated based on the representation of training instances as points in space and the separation of different classes by distinct gaps that are as wide as possible. A new instance will fall to one side of such a gap, predicting its class label [76].

A K* classifier is an instance-based classifier. The class of a test instance is determined based on the classes of similar training instances. In the K* model, an entropy-based distance function is used as the similarity function [72].

5.3. An Ensemble of Models for Credit Card Fraud Detection: Focus on the Voting Mechanism

We propose the use of an ensemble of models rather than a single model. The proposed ensemble consists of six types of models—Decision Tree (DT), Random Forest (RF), Bayesian Network (BN), Naïve Bayes (NB), Support Vector Machine (SVM) and K* models. Each model in the ensemble is trained using the same training set. The training process for the ensemble of models is detailed in Figure 5.4 for an example in which the cardholder has one credit card (1234********1261) and has completed 15 transactions using that card. These 15 cardholder transactions are used to train the Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine and K* models separately.

Once all models in the ensemble have been trained separately using the training set, the ensemble of models is ready to issue decisions on new transactions. When a new transaction occurs, each model in the ensemble provides a decision regarding whether that transaction is legitimate or fraudulent.

The decisions of the models in the ensemble are aggregated by means of different voting mechanisms. In addition to the classical majority voting mechanism, three novel voting mechanisms, namely, optimistic voting, pessimistic voting and weighted voting, are proposed. In majority voting, the decision of the majority of the models is considered. Optimistic voting corresponds to an optimistic approach to each new transaction; a transaction is deemed legitimate if only one model judges it to be legitimate, even if the remaining models indicate that it is fraudulent. Pessimistic voting corresponds to a pessimistic approach to each new transaction; a transaction is deemed fraudulent even if only one model judges it to be fraudulent while the remaining models determine it to be legitimate. In weighted voting, specific weights are assigned to each model in the ensemble depending on the models' fraud detection performance on the validation set.

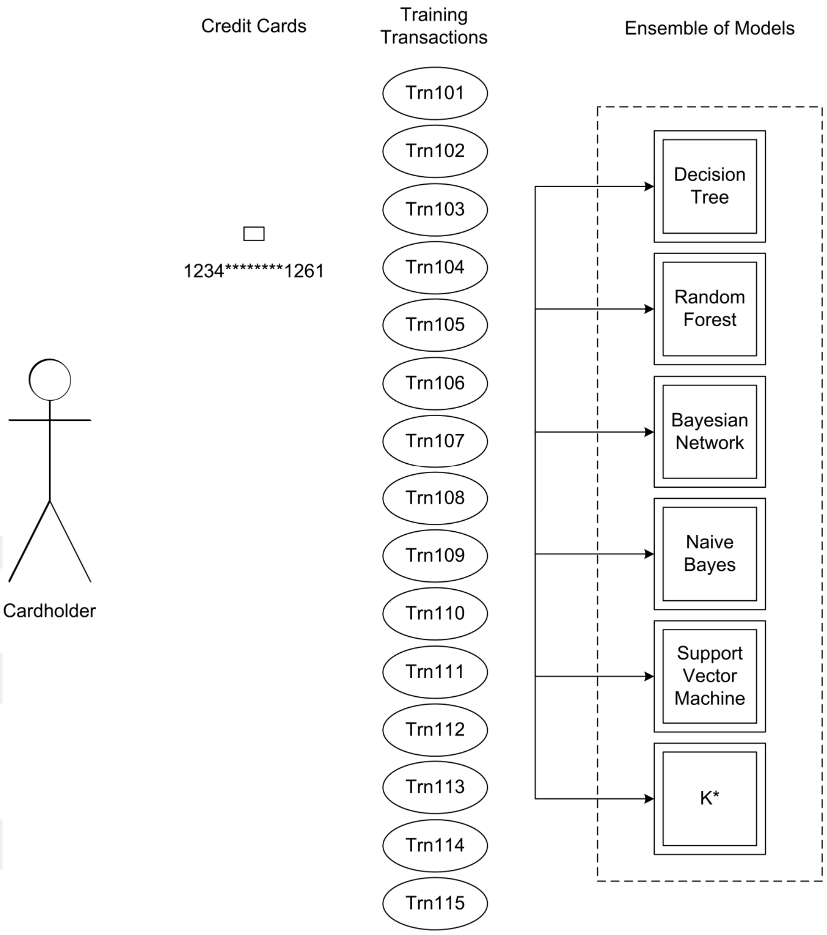

Figure 5.4. Ensemble of models training process.

In majority voting, the ultimate decision is selected based on the judgement of the majority of the models. The value of a decision is either 0 (legitimate) or 1 (fraudulent). Because the ensemble considered in this study consists of six models, a 3-3 draw situation may arise. In a draw situation, the ensemble issues the final decision that the transaction is fraudulent. The majority voting decision formula is given in Equation 5.1.

$$D_{MJR,Trn1} = \begin{cases} 1 & if\ D_{DT,Trn1} + D_{RF,Trn1} + D_{BN,Trn1} + D_{NB,Trn1} + D_{SVM,Trn1} + D_{K*,Trn1} \geq 3 \\ 0 & otherwise \end{cases} \tag{5.1}$$

where $D_{\text{MJR, Trn1}}$ is the Majority Voting (MJR) ensemble decision for transaction 1 and $D_{\text{DT, Trn1}}$, $D_{\text{RF, Trn1}}$, $D_{\text{BN, Trn1}}$, $D_{\text{NB, Trn1}}$, $D_{\text{SVM, Trn1}}$ and $D_{\text{K*, Trn1}}$ are the decisions of the individual models regarding that transaction.

In optimistic voting, if at least one of the models judges a transaction to be legitimate, then the decision of the ensemble is that it is legitimate. Individual decisions take values of either 0 or 1, corresponding to decisions of legitimacy and fraudulence, respectively. The decision values of the individual models are aggregated through logical conjunction to generate the ensemble decision, which also takes a value of either 0 or 1 for legitimacy or fraudulence, respectively. The optimistic voting decision formula is given in Equation 5.2.

$$D_{OPT,Trn1} = D_{DT,Trn1} \wedge D_{RF,Trn1} \wedge D_{BN,Trn1} \wedge D_{NB,Trn1} \wedge D_{SVM,Trn1} \wedge D_{K*,Trn1} \tag{5.2}$$

where $D_{OPT,\ Trn1}$ is the Optimistic Voting (OPT) ensemble decision for transaction 1 and $D_{DT,\ Trn1}$, $D_{RF,\ Trn1}$, $D_{BN,\ Trn1}$, $D_{NB,\ Trn1}$, $D_{SVM,\ Trn1}$ and $D_{K*,\ Trn1}$ are the decisions of the individual models regarding that transaction.

In pessimistic voting, if at least one of the models judges a transaction to be fraudulent, then the decision of the ensemble is that it is fraudulent. The individual decision values can be either 0 (legitimate) or 1 (fraudulent). The individual decision values are aggregated through logical disjunction to generate the ensemble decision, which also takes a value of either 0 (legitimate) or 1 (fraudulent). The pessimistic voting decision formula is given in Equation 5.3.

$$D_{PES,Trn1} = D_{DT,Trn1} \vee D_{RF,Trn1} \vee D_{BN,Trn1} \vee D_{NB,Trn1} \vee D_{SVM,Trn1} \vee D_{K*,Trn1} \tag{5.3}$$

where $D_{PES,\ Trn1}$ is the Pessimistic Voting (PES) ensemble decision for transaction 1 and $D_{DT,\ Trn1}$, $D_{RF,\ Trn1}$, $D_{BN,\ Trn1}$, $D_{NB,\ Trn1}$, $D_{SVM,\ Trn1}$ and $D_{K*,\ Trn1}$ are the decisions of the individual models regarding that transaction.

The optimistic voting process is illustrated in Figure 5.5. The cardholder has made a new transaction using the card with the card number 1234********1261. All models in the ensemble make a decision regarding that new transaction. In this case, the Decision Tree, Bayesian Network, Naïve Bayes, Support Vector Machine and K* models determine the transaction to be fraudulent, but the Random Forest model judges it to be legitimate.

Consequently, optimistic voting results in a decision of legitimacy, whereas majority voting results in a decision of fraudulence. Pessimistic voting also results in a decision of fraudulence because the individual results include at least one such decision.

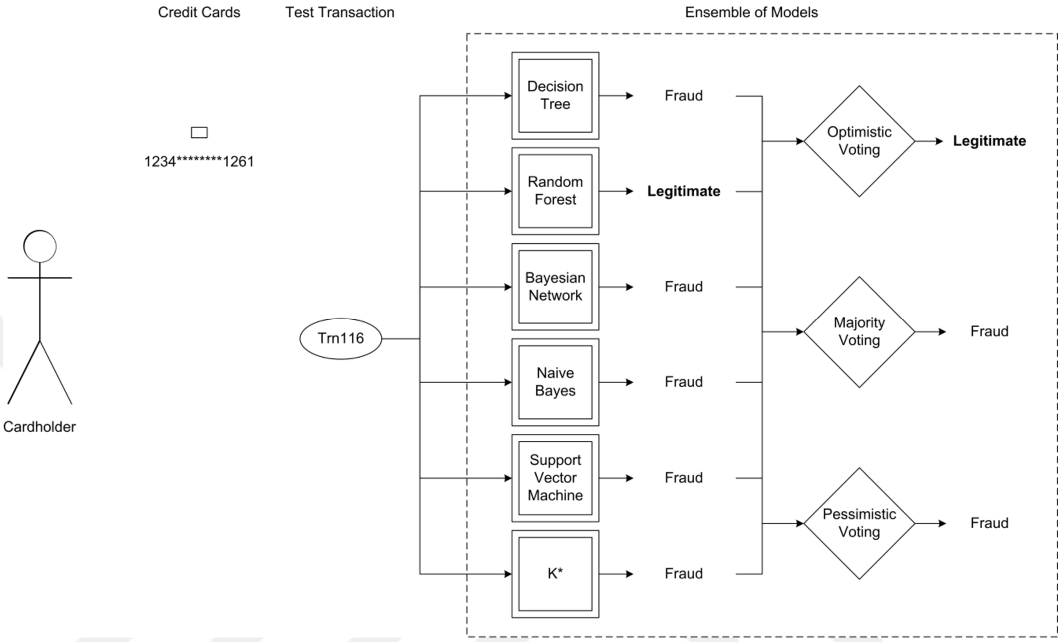

Figure 5.5. Example of optimistic voting.

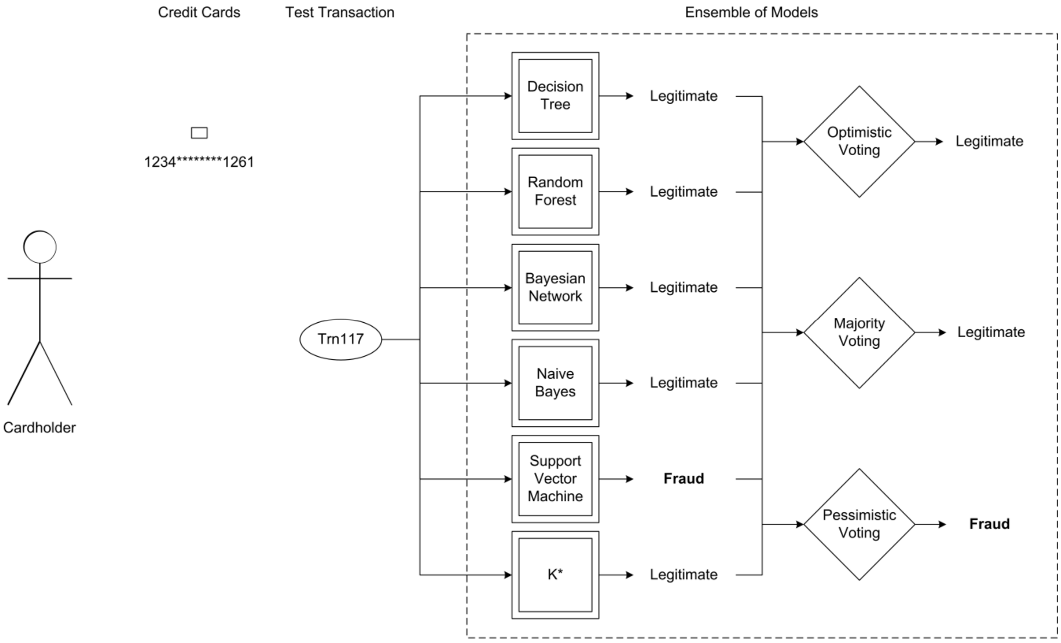

Figure 5.6. Example of pessimistic voting.

The pessimistic voting process is detailed in Figure 5.6. The cardholder has made another new transaction using the card with the card number 1234********1261. All models in the ensemble make a decision regarding this new transaction. The Decision Tree, Random Forest, Bayesian Network, Naïve Bayes and K* models determine the transaction to be legitimate, but the Support Vector Machine model judges it to be fraudulent. Consequently, pessimistic voting results in a decision of fraudulence, whereas majority voting results in a decision of legitimacy. Optimistic voting also results in a decision of legitimacy because there is at least one such decision among the individual results.

In weighted voting, two weights are calculated for each cardholder/model pair. One weight is calculated for legitimate transactions, and the other is calculated for fraudulent transactions. The weight of a model for legitimate transactions is calculated as shown in Equation 5.4.

$$W_{Model,Legitimate} = \frac{P_{Model,Legitimate}}{P_{DT,Legitimate} + P_{RF,Legitimate} + P_{BN,Legitimate} + P_{NB,Legitimate} + P_{SVM,Legitimate} + P_{K*,Legitimate}} \qquad (5.4)$$

where $P_{Model,\ Legitimate}$ is the performance of the model for legitimate transactions, $W_{Model,\ Legitimate}$ is the calculated weight of the model for legitimate transactions, and $P_{DT,\ Legitimate}$, $P_{RF,\ Legitimate}$, $P_{BN,\ Legitimate}$, $P_{NB,\ Legitimate}$, $P_{SVM,\ Legitimate}$ and $P_{K*,\ Legitimate}$ are the performances of all individual models for legitimate transactions.

The weight of a model for fraudulent transactions is calculated as shown in Equation 5.5.

$$W_{Model,Fraudulent} = \frac{P_{Model,Fraudulent}}{P_{DT,Fraudulent} + P_{RF,Fraudulent} + P_{BN,Fraudulent} + P_{NB,Fraudulent} + P_{SVM,Fraudulent} + P_{K*,F}} \qquad (5.5)$$

where $P_{Model,\ Fraudulent}$ is the performance of the model for fraudulent transactions, $W_{Model,\ Fraudulent}$ is the calculated weight of the model for fraudulent transactions, and $P_{DT,\ Fraudulent}$, $P_{RF,\ Fraudulent}$, $P_{BN,\ Fraudulent}$, $P_{NB,\ Fraudulent}$, $P_{SVM,\ Fraudulent}$ and $P_{K*,\ Fraudulent}$ are the performances of all individual models for fraudulent transactions. Equation 5.5 considers the performance of the models only for fraudulent transactions, whereas Equation 5.4 considers their performance only for legitimate transactions.

Let us present an example to clarify the weight calculation process. Suppose that the validation set for a cardholder contains 8 legitimate and 2 fraudulent transactions. The detection performance for each model is shown in Table 5.1.

Table 5.1. Example of weight calculation

Model	**Legitimate Performance**	**Fraudulent Performance**
Decision Tree (DT)	7/8	1/2
Random Forest (RF)	5/8	2/2
Bayesian Network (BN)	8/8	2/2
Naïve Bayes (NB)	7/8	0/2
Support Vector Machine (SVM)	4/8	1/2
K*	7/8	2/2

The weights of the Decision Tree (DT) model for legitimate and fraudulent transactions are calculated as shown in Equation 5.6 and Equation 5.7. Equation 5.6 is based on Equation 5.4, whereas Equation 5.7 is based on Equation 5.5.

$$W_{DT,Legitimate} = \frac{P_{DT,Legitimate}}{P_{DT,Legitimate} + P_{RF,Legitimate} + P_{BN,Legitimate} + P_{NB,Legitimate} + P_{SVM,Legitimate} + P_{K*,Legitimate}}$$
$$= \frac{7/8}{7/8+5/8+8/8+7/8+4/8+7/8} = 7/38 \tag{5.6}$$

where $P_{DT,\ Legitimate}$ is the performance of the Decision Tree (DT) model for legitimate transactions, $W_{DT,\ Legitimate}$ is the calculated weight of the Decision Tree (DT) model for legitimate transactions, and $P_{RF,\ Legitimate}$, $P_{BN,\ Legitimate}$, $P_{NB,\ Legitimate}$, $P_{SVM,\ Legitimate}$ and $P_{K*,\ Legitimate}$ are the performances of the other models for legitimate transactions. The weighting calculation for fraudulent transactions is shown in Equation 5.7.

$$W_{DT,Fraudulent} = \frac{P_{DT,Fraudulent}}{P_{DT,Fraudulent} + P_{RF,Fraudulent} + P_{BN,Fraudulent} + P_{NB,Fraudulent} + P_{SVM,Fraudulent} + P_{K*,Fraudulent}}$$
$$= \frac{1/2}{1/2 + 2/2 + 2/2 + 0/2 + 1/2 + 2/2} = 1/8 \qquad (5.7)$$

where $P_{DT,\ Fraudulent}$ is the performance of the Decision Tree (DT) model for fraudulent transactions, $W_{DT,\ Fraudulent}$ is the calculated weight of the Decision Tree (DT) model for fraudulent transactions, and $P_{RF,\ Fraudulent}$, $P_{BN,\ Fraudulent}$, $P_{NB,\ Fraudulent}$, $P_{SVM,\ Fraudulent}$ and $P_{K*,\ Fraudulent}$ are the performances of the other models for fraudulent transactions.

The weights for the other models are calculated as in Equation 5.6 and Equation 5.7, except that the performance of the model of interest is the performance that appears in the numerator. The denominator remains the same.

The test set for each cardholder is then re-evaluated using the calculated weights as shown in Equation 5.8 and Equation 5.9 for legitimate and fraudulent transactions, respectively. The decisions of the individual models take values of 0 (legitimate) or 1 (fraudulent). However, the decision of the ensemble takes a continuous value between 0 and 1. When the decision value is less than or equal to 0.5, the transaction is considered to be legitimate. When the decision value is greater than 0.5, the decision of the ensemble is that the transaction is fraudulent.

$$\begin{aligned} D_{WGT,Trn1} &= D_{DT,Trn1} * W_{DT,Legitimate} + D_{RF,Trn1} * W_{RF,Legitimate} + D_{BN,Trn1} * W_{BN,Legitimate} \\ &+ D_{NB,Trn1} * W_{NB,Legitimate} + D_{SVM,Trn1} * W_{SVM,Legitimate} + D_{K*,Trn1} * W_{K*,Legitimate} \end{aligned} \qquad (5.8)$$

where $D_{WGT,\ Trn1}$ is the Weighted Voting (WGT) ensemble decision for transaction 1, for which majority voting decision is legitimate; $D_{DT,\ Trn1}$, $D_{RF,\ Trn1}$, $D_{BN,\ Trn1}$, $D_{NB,\ Trn1}$, $D_{SVM,\ Trn1}$ and $D_{K*,\ Trn1}$ are the decisions of the individual models regarding that transaction; and $W_{DT,\ Legitimate}$, $W_{RF,\ Legitimate}$, $W_{BN,\ Legitimate}$, $W_{NB,\ Legitimate}$, $W_{SVM,\ Legitimate}$ and $W_{K*,\ Legitimate}$ are the weights of the individual models for legitimate transactions. The weighted decision for a fraudulent transaction is obtained as shown in Equation 5.9.

$$D_{WGT,Trn2} = D_{DT,Trn2} * W_{DT,Fraudulent} + D_{RF,Trn2} * W_{RF,Fraudulent} + D_{BN,Trn2} * W_{BN,Fraudulent} \\ + D_{NB,Trn2} * W_{NB,Fraudulent} + D_{SVM,Trn2} * W_{SVM,Fraudulent} + D_{K*,Trn2} * W_{K*,Fraudulent} \tag{5.9}$$

where $D_{WGT,\ Trn2}$ is the Weighted Voting (WGT) ensemble decision for transaction 2, for which majority voting decision is fraudulent; $D_{DT,\ Trn2}$, $D_{RF,\ Trn2}$, $D_{BN,\ Trn2}$, $D_{NB,\ Trn2}$, $D_{SVM,\ Trn2}$ and $D_{K*,\ Trn2}$ are the decisions of the models regarding transaction 2; and $W_{DT,\ Fraudulent}$, $W_{RF,\ Fraudulent}$, $W_{BN,\ Fraudulent}$, $W_{NB,\ Fraudulent}$, $W_{SVM,\ Fraudulent}$ and $W_{K*,\ Fraudulent}$ are the weights of the individual models for fraudulent transactions.

The weighted voting process is detailed in Figure 5.7. All models in the ensemble make their decisions regarding the test transaction for which majority voting decision is legitimate. The Decision Tree, Random Forest, Naïve Bayes and Support Vector Machine models judge the transaction to be legitimate, whereas the Bayesian Network and K* models determine it to be fraudulent. The decision value (0=legitimate or 1=fraudulent) for each model is multiplied by the weight for legitimate transactions for that model because the MJR decision is legitimate. The values sum to obtain a value of less than 0.5; therefore, the decision of the ensemble with weighted voting is that the transaction is legitimate.

5.4. OPWEM Experimental Results and Analysis

We proposed the use of an ensemble of models rather than a single model for credit card fraud detection. The ensemble consists of six models: Decision Tree (DT), Random Forest (RF), Bayesian Network (BN), Naïve Bayes (NB), Support Vector Machine (SVM) and K* models. The decisions of the models in the ensemble are aggregated through voting. In addition to the majority voting (MJR) approach, we have proposed three alternative voting mechanisms: optimistic (OPT), pessimistic (PES) and weighted (WGT) voting. The fraud detection performances of the individual models and the proposed voting mechanisms are summarized in Table 5.2.

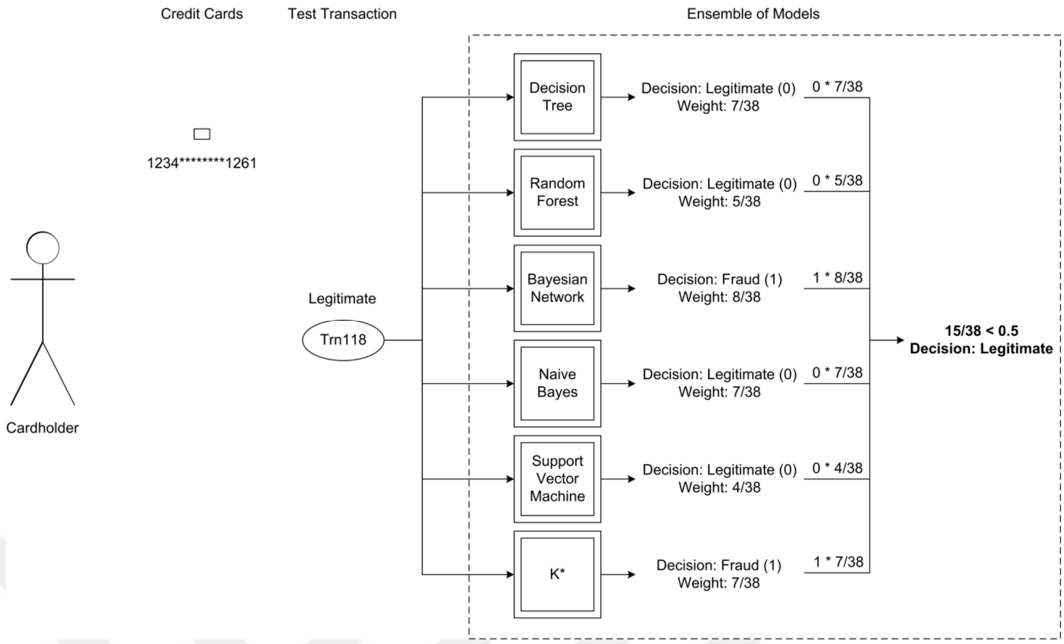

Figure 5.7. Example of weighted voting.

As shown in the Sensitivity column in Table 5.2, pessimistic voting exhibited significantly higher sensitivity than any of the individual models or the other voting mechanisms. In other words, pessimistic voting detected more fraudulent transactions than any individual model or any of the other voting mechanisms.

Table 5.2. Evaluation results for single models and voting mechanisms

Model / Voting Approach	**Sensitivity**	**Specificity**	**False Positive Rate**	**Precision**	**Negative Predictive Value**	**Accuracy**	**Alarm Rate**	**F_1 Score**
DT	52.53%	97.35%	2.65%	51.09%	97.59%	95.19%	5.05%	51.80%
RF	50.84%	98.09%	1.91%	58.87%	97.53%	95.81%	4.27%	54.56%
BN	50.00%	**99.30%**	**0.70%**	78.67%	97.51%	96.92%	3.08%	61.14%
NB	92.57%	94.18%	5.82%	44.97%	**99.60%**	94.10%	10.01%	60.53%
SVM	66.89%	95.55%	4.45%	43.98%	98.28%	94.17%	7.46%	53.07%
K*	73.14%	92.67%	7.33%	33.90%	98.56%	91.73%	10.50%	46.33%
MJR	67.40%	97.98%	2.02%	62.92%	98.34%	96.50%	5.17%	65.08%
OPT	31.59%	**99.90%**	**0.10%**	**94.65%**	96.65%	96.60%	**1.62%**	47.37%
PES	**93.92%**	86.28%	13.72%	26.05%	**99.65%**	86.65%	17.58%	40.79%
WGT	64.02%	**99.25%**	**0.75%**	82.12%	98.20%	**97.55%**	3.81%	**71.95%**

Optimistic voting, weighted voting and the Bayesian Network model offered significantly higher specificity than the other approaches, as shown in the Specificity column in Table 5.2. In other words, optimistic voting, weighted voting and the Bayesian Network model produced lower false alarm rates. This same result can also be observed in the False Positive Rate column in Table 5.2. Optimistic voting results in a decision of fraudulence only if all models in the ensemble issue decisions of fraudulence. It is the most deliberate approach among all of the voting mechanisms. Consequently, optimistic voting resulted in a false alarm for only 0.10% of legitimate transactions, whereas majority voting resulted in a false alarm for 2.02% of legitimate transactions—a rate that is 20 times greater.

The Precision column in Table 5.2 shows that optimistic voting yielded significantly higher precision than either the individual models or the other voting mechanisms. In other words, optimistic voting resulted in a higher true alarm rate than did any individual model or any of the other voting mechanisms. When optimistic voting was used, 94.65% of the issued alarms were raised for actually fraudulent transactions.

Pessimistic voting and the Naïve Bayes model exhibited significantly higher negative predictive values, as shown in the corresponding column in Table 5.2. In other words, pessimistic voting and the Naïve Bayes model were superior to the other models and voting mechanisms in terms of the number of true “no alarm” decisions.

As shown in the Accuracy column in Table 5.2, weighted voting offered significantly higher accuracy than any of the individual models or the other voting mechanisms. In other words, weighted voting was superior to the other approaches in terms of correct decisions overall (both alarm and “no alarm” decisions).

The Alarm Rate column in Table 5.2 showed that optimistic voting results in a significantly lower alarm rate than did the individual models and the other voting mechanisms. This result is not surprising because optimistic voting is the most deliberate approach among all voting mechanisms.

As shown in the F_1 score column in Table 5.2, weighted voting offered a significantly higher F_1 score than any of the individual models or the other voting mechanisms. In other words, weighted voting achieves the best performance value in terms of sensitivity and precision combined.

The choice of voting approach depends on bank strategy about fraud detection and false alarm rate. False alarms have a negative impact on cardholder satisfaction and consequently bank reputation. If bank management decides to detect frauds with minimal false alarm rate, optimistic voting should be used. Ensemble with optimistic voting detects 31.59% of fraudulent transactions with only 0.10% false alarm rate. On the other hand, if bank management decides to detect as many frauds as possible, pessimistic voting should be used. Ensemble with pessimistic voting detects 93.92% of fraudulent transactions with 13.72% false alarm rate. Weighted voting appeared to be a good alternative for optimistic and pessimistic voting. Ensemble with weighted voting detects 64.02% of fraudulent transactions with only 0.75% false alarm rate. This figure means that, weighted voting detects twice as many fraudulent transactions of optimistic voting together with a similar false alarm rate.

OPWEM consists of six artificial intelligence models and the decisions of individual models are combined by the proposed voting strategies namely, optimistic voting, pessimistic voting and weighted voting. In optimistic voting, the ensemble decision is legitimate if at least one individual model decides that the transaction is legitimate. In pessimistic voting, the ensemble decision is fraudulent if at least one individual model decides that the transaction is fraudulent. In both of these voting strategies, the decision of one model may affect the ensemble decision and the decisions of the rest of the models in the ensemble are ignored. In weighted voting, specific weights are assigned to each individual model and each individual model affects the ensemble decision in proportion to its assigned weight. As a natural consequence of such specific voting strategies, ROC curve analysis based on the individual model is not usable in OPWEM. Within the context of ROC curve analysis, OPWEM is considered as a whole model without focusing on individual models that constitutes the ensemble [65, 77].

Some past research papers have been published regarding ROC curves and ensembles but they do not possess a generic methodology to be applied to an ensemble. Barreno *et al.* proposed a methodology to find the optimal combination of individual artificial intelligence models rather than defining generic methodology to plot an ROC curves for ensembles [78]. Jurkowski *et al.* provided ROC curve analysis of an ensemble of individual models but no detailed definition about ROC curve plotting methodology was given [79]. Evangelista *et al.* focused on ensemble methods for unsupervised classification and provided ROC curve analysis for the proposed aggregation techniques but they did not provide details about the ROC curve plotting methodology [80].

Score calculation for OPWEM ROC curve analysis is applied by using the same methodology as in CBM ROC analysis. The score value for each transaction is calculated as a function of transaction amount and MCC. Considering the fact that, transactions having different MCC's have different transaction amount ranges, we use MCC as a part of the scoring function. The transaction amounts in the test set are normalized to a range of 0-1000 to provide score values. Normalization is applied separately for each MCC since each MCC has a different amount range. Having score values between 0 and 1000, we plot the ROC curve by using 50 threshold points in increments of 20.

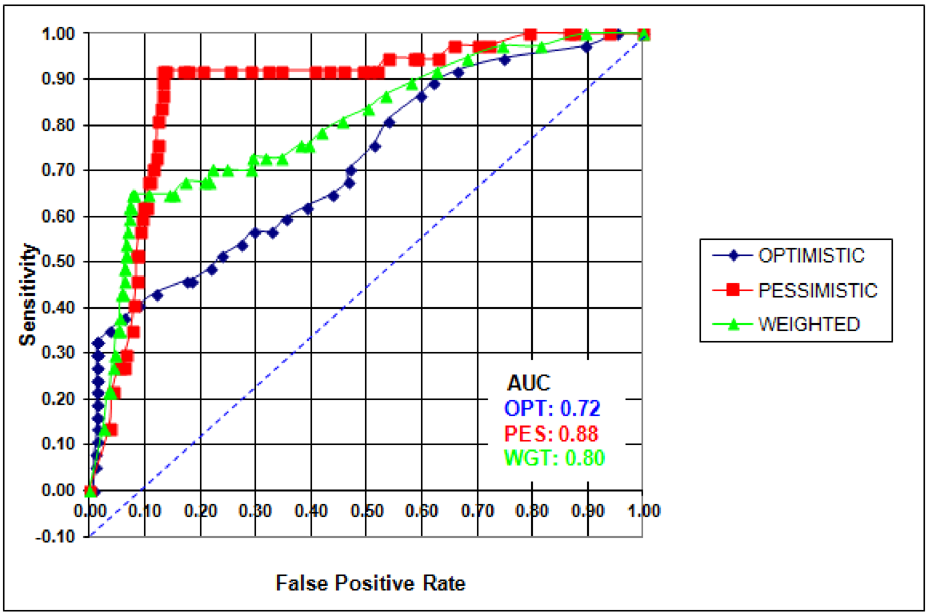

Figure 5.8. ROC curves for optimistic, pessimistic and weighted voting in OPWEM.

OPWEM ROC curves for the proposed voting strategies are shown in Figure 5.8. Pessimistic voting outperforms optimistic and weighted voting providing an AUC value of 0.88. Weighted voting provides an AUC value of 0.80 whereas optimistic voting provides an AUC value of 0.72.

6. SBSM: SPENDING BEHAVIOR SIMILARITY MODEL TO IMPROVE CREDIT CARD FRAUD DETECTION

6.1. SBSM Credit Card Fraud Detection Process

In CBM and OPWEM, the models are trained by using transactions which are specific to the cardholder, but similarity of cardholder behavior is not taken into account. Similarity of cardholder behavior can be defined as similar transactions of different cardholders. In this section, we propose an improved credit card fraud detection model, namely Spending Behavior Similarity Model (SBSM), which takes similarity of cardholder behavior into account. Taking the similarity of cardholder behavior into account implies larger training set size and better quality of training set.

Similar transactions of different cardholders mean that transactions have the same MCC but spending amount may or may not be the same. If the mean and median values of spending amounts are within a certain range, we assume transactions are similar. As an example, the mean spending amount of a cardholder in supermarket transactions (MCC: 5411) is 105.73. Other cardholders with a mean amount of supermarket transactions around 105.73 are selected and their supermarket transactions are added to the training set. The same idea is also applied to the median amounts.

An overall view of the SBSM training process is given in Figure 6.1. Cardholders may have one or more credit cards. They generate credit card transactions for each of their credit cards. These credit card transactions are stored in a credit card transactions and similarity database. In addition to credit card transactions, the database contains similarity information table. Similarity information table contains mean and median amounts for each cardholder and merchant category code. Similarity information table is populated using the original credit card transactions in credit card transactions and similarity database. Training set for each cardholder contains transactions of that cardholder and transactions of cardholders which have similar spending behavior. The training process used here trains one of each AI model for each cardholder.

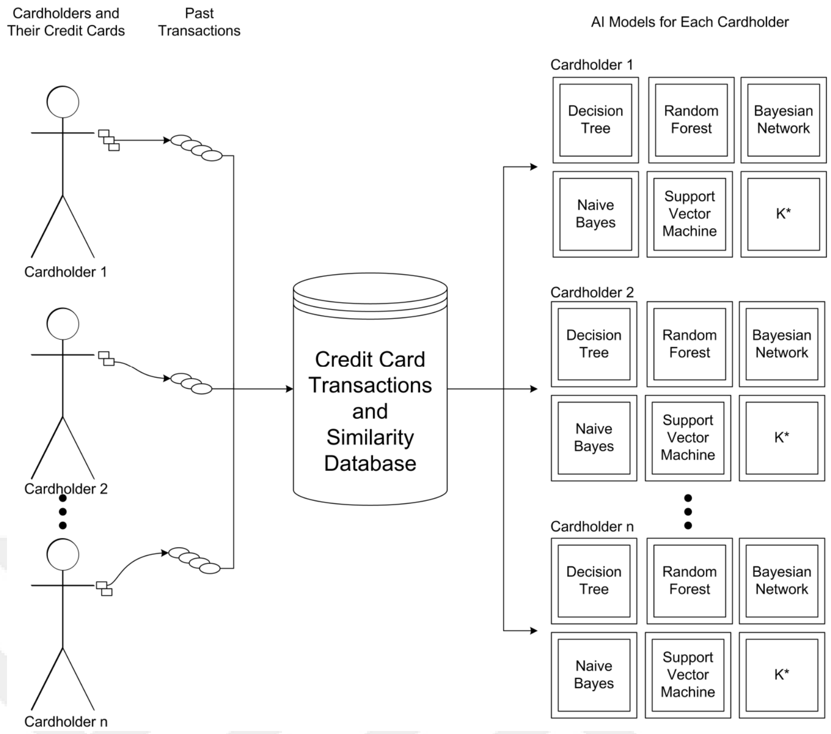

Figure 6.1. SBSM training process.

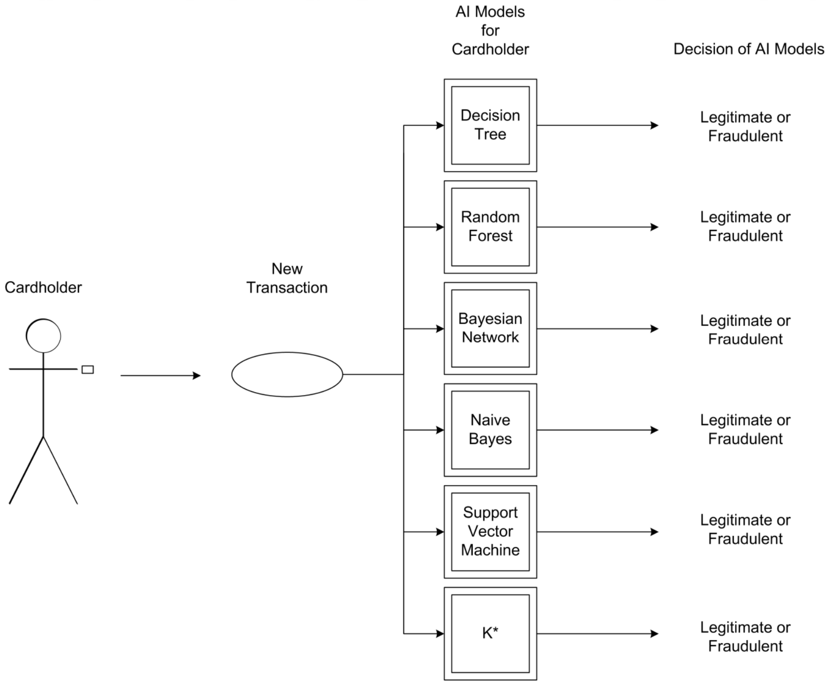

Figure 6.2. SBSM decision process.

The decision process begins when a new transaction occurs. At that point, the corresponding AI models make their decisions as to whether the transaction is legitimate or fraudulent. The decision process is detailed in Figure 6.2.

6.2. Calculating and Using Spending Behavior Similarity Statistics

In this study, mean and median are used as similarity measures. The mean and median formulae are given in Equation 6.1 and Equation 6.2 respectively:

$$Mean = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (6.1)$$

$$Median = \begin{cases} x_{\frac{n-1}{2}} & if\ \ n\ is\ odd \\ \frac{1}{2}\left(x_{\frac{n}{2}} + x_{\frac{n}{2}+1} \right) & if\ \ n\ is\ even \end{cases} \qquad (6.2)$$

where n is the number of transactions for each cardholder – MCC pair and x is transaction amount.

Mean and median amounts for each cardholder – MCC pair are calculated and stored in similarity information table. A similarity information table record has four attributes: cardholder number, MCC, statistics type and amount. Statistics type is mean or median. Amount attribute contains mean or median amount depending on the statistics type.

For further clarification, an example is given using sample transactions. To start with, the transaction records for two cardholders can be seen in Table 6.1. In Table 6.1, transactions have MCC: 5411 which indicates “Grocery Stores, Supermarkets”.

It can be noticed that each cardholder has made four supermarket (MCC: 5411) purchases. Mean and median values for each cardholder – MCC pair is calculated and added to the similarity information table. The corresponding statistics can be seen in Table 6.2.

Table 6.1. Transactions of two cardholders

Cardholder No	Card No	MCC	Amount	Date	Time
12345	6789********4321	5411	123.54	03/04/2012	19:25
12345	6789********4321	5411	165.22	10/04/2012	15:00
12345	6789********4321	5411	172.23	12/05/2012	20:05
12345	6789********4321	5411	135.52	15/07/2012	19:45
67890	6789********1234	5411	174.22	05/04/2012	19:47
67890	6789********1234	5411	115.23	18/04/2012	20:13
67890	6789********1234	5411	197.54	06/05/2012	15:15
67890	6789********1234	5411	95.43	23/05/2012	19:25

Table 6.2. Similarity information table records of two cardholders

Cardholder No	MCC	Statistics Type	Amount
12345	5411	MEAN	149.13
12345	5411	MEDIAN	150.37
67890	5411	MEAN	145.60
67890	5411	MEDIAN	144.73

During the training process for a cardholder, firstly his records in the similarity information table are selected. Secondly, these similarity records are used to select similarity records of other cardholders having similar spending behavior. In this study, we used 5% similarity range. In other words, if another cardholder’s mean or median amount is within 5% range of the original cardholder’s mean or median amount, that cardholder’s transactions with the same MCC are selected and added to the training set. We evaluated using mean and median measures individually and together.

To continue with the example above, during the training process for cardholder 12345, his records in the similarity information table are selected, which are mean: 149.13 and median: 150.37 as seen in Table 6.2. Similarity records of cardholder 67890 are mean: 145.60 and median: 144.73 as seen in Table 6.2. Since similarity records of cardholder 67890 is within 5% range of similarity records of cardholder 12345, cardholder 67890 can be said to have a similar spending behavior with cardholder 12345 in supermarket. Therefore, supermarket transactions of cardholder 67890 are selected from credit card transactions database and added to the training set, which already contains transactions of cardholder 12345.

6.3. SBSM Experimental Results and Analysis

In this study, we proposed improving the size and the quality of the training set by finding the cardholders with similar spending behavior and using their corresponding transactions. Six AI models are used in evaluation: Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine and K*. The fraud detection performance of AI models without considering similarity and with considering mean and median similarity are summarized in Table 6.3.

As shown in the Sensitivity column of Table 6.3, considering mean or median similarity individually or together improves sensitivity, which means more fraudulent transactions are detected.

Table 6.3. Evaluation results for no similarity and similarity approaches

Approach	**Sensitivity**	**Specificity**	**False Positive Rate**	**Precision**	**Negative Predictive Value**	**Accuracy**	**Alarm Rate**	**F_1 Score**
No Similarity	65.77%	95.41%	4.59%	42.07%	98.21%	93.98%	7.54%	51.32%
Mean Similarity	72.97%	93.88%	6.12%	37.67%	98.56%	92.87%	9.34%	49.69%
Median Similarity	75.68%	92.31%	7.69%	33.27%	98.68%	91.50%	10.97%	46.22%
Mean + Median Similarity	**89.64%**	**96.58%**	**3.42%**	**57.02%**	**99.46%**	**96.24%**	7.58%	**69.70%**

The Specificity, the False Positive Rate, the Precision, the Accuracy and the Alarm Rate and F_1 Score columns in Table 6.3 show that using mean and median similarity individually has a slightly negative impact on fraud detection performance. However, using mean and median similarity together provides an improvement. This result shows that as we extend training set, we may damage its quality. So using multiple similarity measures is a good solution. It both increases the number of detected fraudulent transactions and decreases the number of false alarms.

As shown in the Negative Predictive Value column in Table 6.3, there is no statistically significant difference between no similarity and individual mean and median similarity approaches. Only mean and median similarity approach provides an improvement.

Overall, considering mean and median similarity together provides significant improvement in credit card fraud detection.

Score calculation for SBSM ROC curve analysis is applied by using the same methodology as in CBM and OPWEM ROC analysis. The score value for each transaction is calculated as a function of transaction amount and MCC. Considering the fact that, transactions having different MCC's have different transaction amount ranges, we use MCC as a part of the scoring function. The transaction amounts in the test set are normalized to a range of 0-1000 to provide score values. Normalization is applied separately for each MCC since each MCC has a different amount range. Having score values between 0 and 1000, we plot the ROC curve by using 50 threshold points in increments of 20.

ROC curves for the proposed similarity approaches in SBSM are shown in Figure 6.3. Mean + median similarity approach outperforms the individual mean similarity and median similarity approaches providing an AUC value of 0.92.

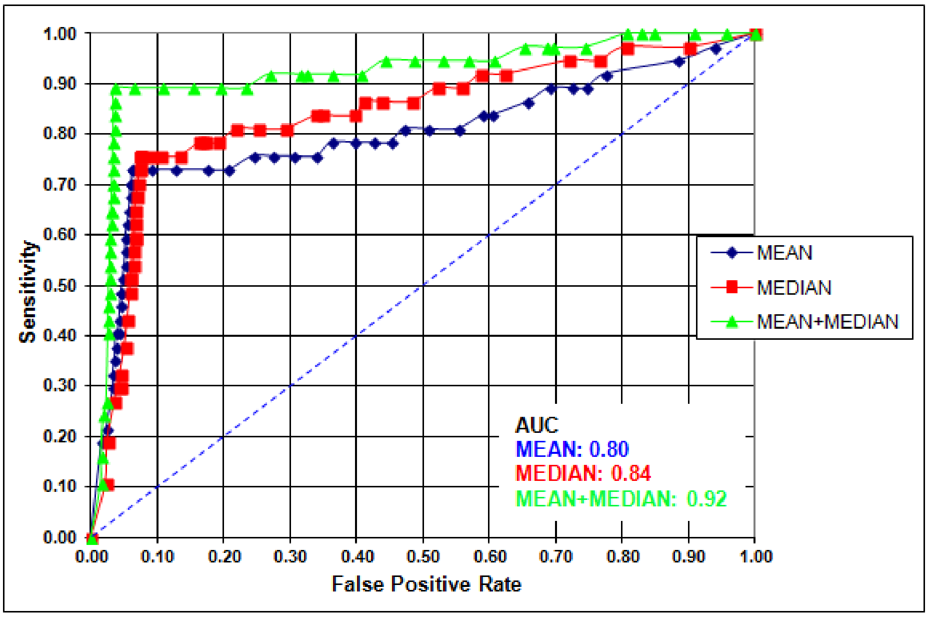

Figure 6.3. ROC curves for mean, median and mean+median similarity in SBSM.

7. COMPARATIVE EVALUATION OF CBM, OPWEM AND SBSM

In this thesis, we proposed three novel artificial-intelligence-based models for detecting credit card frauds. First, we proposed Cardholder Behavior Model (CBM). CBM is an unsupervised model and uses clustering to represent the spending behavior of cardholders. We proposed four focal points to fine-tune CBM, which are single-card versus multi-card focus, holiday season spending focus, time of day focus and inflation focus. The second model we proposed is called Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models (OPWEM). OPWEM is an ensemble of six well known artificial intelligence techniques, namely Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine, and K*. We proposed optimistic, pessimistic and weighted voting strategies in OPWEM for better detection of credit card fraud. Lastly, we proposed Spending Behavior Similarity Model (SBSM) to improve the performance of supervised models. SBSM uses the same artificial intelligence techniques used in OPWEM.

From the point of view of a practitioner, a fraud detection system should detect as many fraud cases as possible. Also false alarm rate should be as low as possible because false alarms have a negative impact on cardholder satisfaction. Additionally, accuracy and alarm rate provides a general picture of the proposed models' fraud detection performance. Considering these facts, we used sensitivity, false alarm rate and accuracy criteria in comparative evaluation. The sensitivity measures the percentage of fraudulent transactions for which the proposed model raises an alarm. The false positive rate is the percentage of legitimate transactions for which the proposed model raises an alarm.

For CBM, the maximum sensitivity and the corresponding false positive rate have been taken from Table 4.12 which shows the evaluation results for multi-card and single-card CBM's. The maximum sensitivity value, 59.46% has been achieved by multi-card CBM's considering holiday, time of day and inflation. The corresponding false positive rate is 27.67%.

For OPWEM, we used the values in Table 5.2 which shows the evaluation results for single models and the voting mechanisms. The maximum sensitivity value, 93.92% has been achieved by pessimistic voting. The corresponding false positive rate is 13.72%.

For SBSM, we used the values in Table 6.3 which shows evaluation results for no similarity and similarity approaches. The maximum sensitivity value, 89.64% has been achieved by mean + median similarity approach. The corresponding false positive rate is 3.42%.

Sensitivity-based comparative evaluation results of the proposed models are summarized in Table 7.1. OPWEM achieves the maximum sensitivity value of 93.92% together with a false positive rate of 13.72%. On the other hand, SBSM achieves a sensitivity value of 89.64% with a false positive rate of 3.42%. These values show that, SBSM provides a similar fraud detection performance to OPWEM with a much smaller false alarm rate. CBM sensitivity value is worse than OPWEM and SBSM.

Table 7.1. Sensitivity-based comparison of the proposed models

Proposed Model	**Case**	**Maximum Sensitivity (Higher is Better)**	**Corresponding False Positive Rate (Lower is Better)**
CBM	Multi-card + Consider Holidays + Consider Time of Day + Consider Inflation	59.46%	27.67%
OPWEM	Pessimistic Voting	93.92%	13.72%
SBSM	Mean + Median Similarity	89.64%	3.42%

For CBM, the minimum false positive rate and the corresponding sensitivity have been taken from Table 4.12. The minimum false positive rate value for CBM, 18.22% has been achieved by multi-card CBM’s considering holiday, ignoring time of day and inflation. The corresponding sensitivity value is 45.95%.

As seen in Table 5.2, the minimum false positive rate value for OPWEM, 0.10% has been achieved by optimistic voting. The corresponding sensitivity value is 31.59%.

Table 6.3 shows that the minimum false positive rate value for SBSM, 3.42% has been achieved by mean + median similarity. The corresponding sensitivity value is 89.64%.

False positive rate based comparative evaluation results of the proposed models are summarized in Table 7.2. OPWEM achieves the minimum false positive rate value of 0.10% together with a sensitivity value of 31.59%. On the other hand, SBSM achieves a sensitivity value of 89.64% with a false positive rate of 3.42%. OPWEM with optimistic voting provides a very small false positive rate. Therefore, OPWEM should be preferred especially when the negative impact of false alarm rate on cardholder satisfaction is high. On the other hand, SBSM can be preferred for better fraud detection with a reasonable false alarm rate. CBM has worse performance than OPWEM and SBSM in terms of sensitivity and false positive rate.

Table 7.2. False positive rate based comparison of the proposed models

Proposed Model	**Case**	**Minimum False Positive Rate (Lower is Better)**	**Corresponding Sensitivity (Higher is Better)**
CBM	Multi-card + Consider Holidays + Ignore Time of Day + Ignore Inflation	18.22%	45.95%
OPWEM	Optimistic Voting	0.10%	31.59%
SBSM	Mean + Median Similarity	3.42%	89.64%

Accuracy and alarm rate based comparative evaluation results of the proposed models are summarized in Table 7.3. OPWEM with weighed voting achieves the maximum accuracy value of 97.55% together with an alarm rate of 3.81%. On the other hand, SBSM with mean and median similarity provides a smaller accuracy value of 96.24% with a higher alarm rate of 7.58%. CBM provides an accuracy value of 80.05% with an alarm rate of 19.56%. CBM values are far worse than OPWEM and SBSM.

Therefore, OPWEM with weighted voting should be preferred for achieving maximum accuracy together with the minimum possible alarm rate.

Table 7.3. Accuracy and alarm rate based comparison of the proposed models

Proposed Model	**Case**	**Maximum Accuracy (Higher is Better)**	**Corresponding Alarm Rate (Lower is Better)**
CBM	Multi-card + Consider Holidays + Ignore Time of Day + Ignore Inflation	80.05%	19.56%
OPWEM	Weighted Voting	97.55%	3.81%
SBSM	Mean + Median Similarity	96.24%	7.58%

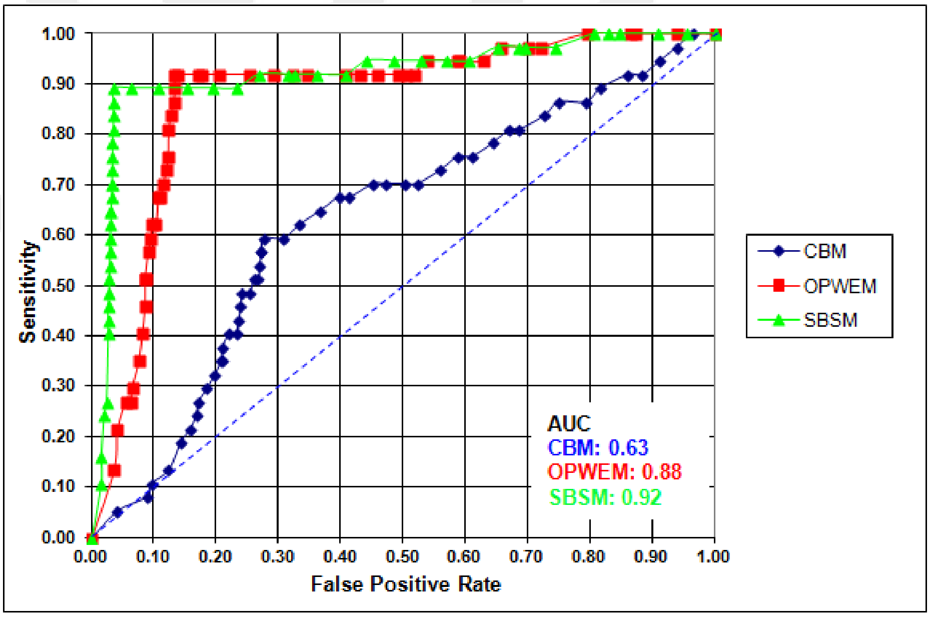

Figure 7.1. CBM, OPWEM and SBSM comparative ROC curve analysis.

The best performing cases of the proposed models are used so as to provide ROC based comparative evaluation of them. Multi-card CBM’s considering holiday, ignoring time of day and inflation provides the maximum AUC value among all CBM cases. Pessimistic voting in OPWEM provides a better AUC value than other voting strategies. Using mean and median similarity together in SBSM provides a better AUC value than using mean and median similarity individually. The ROC curves for best performing cases of CBM, OPWEM and SBSM are shown in Figure 7.1. SBSM provides a slightly better overall classification performance than OPWEM. SBSM using mean+median similarity approach provides an AUC value of 0.92 whereas OPWEM with pessimistic voting provides an AUC value of 0.88. CBM performance is worse than the OPWEM and SBSM. CBM provides an AUC value of only 0.63.

8. CONCLUSION AND FUTURE WORK

Losses due to credit card fraud are billions of dollars each year and a major goal of financial institutions is to minimize credit card fraud losses. Rule-based systems have been widely used to detect and help fight credit card fraud. The rules applied in such systems are formulated based on the experience of fraud experts and the results of fraud investigations. Rule discovery is a manual process and this fact is an important disadvantage of rule-based systems. Unlike rule-based systems, artificial intelligence models are expected to learn from past transaction data and consequently no manual process is required. Many researchers in credit card fraud detection have recognized the advantage offered by artificial intelligence techniques.

In this thesis, we proposed three novel artificial-intelligence-based models for detecting credit card frauds. A dataset of real-life credit card transactions from a leading bank in Turkey has been used to evaluate the performance of three proposed models. Also, we provided a comparative evaluation of three proposed models.

First, we proposed Cardholder Behavior Model (CBM). CBM is an unsupervised model and uses clustering to represent the spending behavior of cardholders. We proposed four focal points to fine-tune CBM, which are single-card versus multi-card focus, holiday season spending focus, time of day focus and inflation focus.

The first focus point takes into account the single-card and multi-card ownership of cardholders. The evaluation results show that single-card CBM detects more frauds than multi-card CBM but single-card CBM gives more false alarms. It is shown that single-card CBM is preferable for detecting more frauds whereas multi-card CBM is preferable for giving fewer false alarms.

The second focus point is to take into account holidays of cardholders. The evaluation results show that holiday season focus in CBM detects more frauds than all time CBM while giving similar number of false alarms. Consequently, we proposed that holiday seasons should always be considered in building credit card fraud detection.

Our third focus point is to take into account time of day of credit card transactions. Our evaluation results show that time of day focus CBM detected more frauds than all time CBM while giving more false alarms. In short, it is discovered that time of day should be considered for detecting more frauds whereas time of day should be ignored for giving fewer false alarms.

Our last focus point is to take into account economical inflation in CBM. Evaluation results show that, considering inflation results in better fraud detection performance without a negative effect on false alarms rates. Therefore, inflation should be considered in building CBM.

We have focused on the practical problem of credit card fraud detection by proposing CBM together with a number of focus points. We have empirically shown the effects of the proposed focus points on the alarm rates and fraud detection performance. The practical impact of CBM is to provide a supportive fraud detection tool which will work together with the existing rule-based tools. CBM detects about 60 per cent of the frauds by modeling the behavior of the cardholders in proposed focus points.

The second model we proposed is called Optimistic, Pessimistic and Weighted Voting in an Ensemble of Models (OPWEM). OPWEM contains six well known artificial intelligence techniques, namely Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine, and K*. We proposed optimistic, pessimistic and weighted voting strategies in OPWEM for better detection of credit card fraud. In optimistic voting, if at least one AI model judges that the transaction is legitimate, then the decision of the ensemble is that it is legitimate. In pessimistic voting, when at least one AI model judges that the transaction is fraudulent, the decision of the ensemble is that it is fraudulent. In weighted voting, specific weights are assigned to each model in the ensemble depending on the models' fraud detection performance on the validation set. The results are promising. Optimistic voting in ensemble of models leads to the detection of 31.59% of fraudulent transactions with a false alarm rate of only 0.10%, the pessimistic voting strategy detects 93.92% of fraudulent transactions with a false alarm rate of 13.72%, and the weighted voting strategy detects 64.02% of fraudulent transactions with a false alarm rate of 0.75%. Banks can choose from among these voting mechanisms in accordance with their preferred strategies regarding fraud detection and desired false alarm rates.

The practical contribution of OPWEM is to provide a supportive fraud detection system that can operate in collaboration with existing rule-based systems. From the point of view of a practitioner;

- All past credit card transactions are stored in Credit Card Transactions Database.
- Past transactions are used for training. Training is made every day during midnight since number of transactions is minimum during this period.
- When a new transaction occurs, OPWEM makes a decision whether to flag the new transaction as fraudulent or legitimate.
- Bank management selects one of the voting approaches depending on the bank strategy about false alarm rates.
- Considering the fact that false alarms have a negative impact on cardholder satisfaction, bank strategy may aim minimal false alarm rate. For this purpose, optimistic voting should be selected.
- If bank management aims to detect as many fraud cases as possible, pessimistic voting should be selected.
- Weighted voting detected more frauds than optimistic voting with a slightly more false alarm rate. Therefore, it may be chosen as a good alternative to optimistic and pessimistic voting.

Lastly, we proposed Spending Behavior Similarity Model (SBSM) to improve the performance of supervised models. The basic aim of SBSM is using spending behavior similarity to improve the size and the quality of the training set, hence the resulting artificial intelligence model accuracy. Mean and median spending amounts for each merchant category are used as the similarity measures. We used six widely known artificial intelligence models, namely, Decision Tree, Random Forest, Bayesian Network, Naïve Bayes, Support Vector Machine and K*. These models were also used in OPWEM. The evaluation results show that the proposed approach improve fraud detection performance significantly. Artificial intelligence models which are trained considering mean and median similarity together detect 89.64% of fraudulent transactions with a false alarm rate of only 3.42%. On the other hand, AI models which are trained with only corresponding cardholder transactions detect only 65.77% of fraudulent transactions with a false alarm rate of 4.59%.

In future research regarding CBM, we will focus on internet transactions to improve CBM. In the first case, we will build two separate models; one with just internet transactions and the other with just card-present transactions. In the second case, we will build a single model with all internet and card-present transactions together. Thereafter, we will evaluate the alarm rates and fraud detection performance in each case.

In future research regarding OPWEM, we will focus on adding associative memory to OPWEM and evaluate the effect of associative memory on fraud detection performance. Additionally, we plan to modify OPWEM to be able to process credit card transactions over the Internet.

In future research regarding SBSM, we will apply other similarity measures to SBSM. These similarity measures are card types (i.e. gold, platinum) and personal information such as age and gender.

Moreover, we aim to repeat the experiments for CBM, OPWEM and SBSM using datasets from other leading banks in Turkey to minimize the threat to external validity.

REFERENCES

1. HSN Consultants Inc., *The Nilson report*. Oxnard, California, 2015.

2. Ganji, V.R. and S.N.P. Mannem, “Credit card fraud detection using anti-k nearest neighbor algorithm”, *International Journal on Computer Science and Engineering (IJCSE)*, Vol. 4, No. 6, pp. 1035-1039, 2012.

3. Duan, L. and L.D. Xu, “Business Intelligence for Enterprise Systems: A Survey”, *IEEE Transactions on Industrial Informatics*, Vol. 8, No. 3, pp. 679-687, 2012.

4. Kultur, Y., B. Turhan and A.B. Bener, “ENNA: software effort estimation using ensemble of neural networks with associative memory”, *Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering (SIGSOFT ’08/FSE-16)*, pp. 330-338, 2008.

5. Kultur, Y., B. Turhan and A.B. Bener, “Ensemble of neural networks with associative memory (ENNA) for estimating software development costs”, *Knowledge-Based Systems,* Vol. 22, No. 6, pp. 395-402, 2009.

6. Diners Club, “About Diners Club International”, 2017, https://www.dinersclub.com/about-us/history, accessed at May 2017.

7. Grossman, P.Z., *American Express: The unofficial history of the people who built the great financial empire*. New York: Crown Publishers, 1987.

8. Visa, “History of Visa”, 2017, https://usa.visa.com/about-visa/our_business/history-of-visa.html, accessed at May 2017.

9. Mastercard, “About Mastercard”, 2017, https://www.mastercard.us/en-us/about-mastercard/who-we-are.html, accessed at May 2017.

10. Krivko, M., “A hybrid model for plastic card fraud detection systems”, *Expert Systems with Applications*, Vol. 37, No. 8, pp. 6070-6076, 2010.

11. SAS, “SAS Fraud Management”, 2017, http://www.sas.com/en_my/industry/banking/fraud-management.html, accessed at May 2017.

12. Ethoca, “Solutions”, 2017, https://www.ethoca.com/solutions, accessed at May 2017.

13. Yu, W.F. and N. Wang, “Research on credit card fraud detection model based on distance sum”, *Proceedings of the international joint conference on artificial intelligence (JCAI '09)*, pp. 353-356, Hainan Island, China, 2009.

14. Ju, C.H. and N. Wang, “Research on credit card fraud detection model based on similar coefficient sum”, *Proceedings of the 1st international workshop on database technology and applications*, pp. 295-298, Wuhan, China, 2009.

15. Philip, N. and K.K. Sherly, “Credit card fraud detection based on behavior mining”, *TIST International Journal for Science, Technology & Research*, Vol. 1, pp. 7-12, 2012.

16. Jha, S., M. Guillen and J.C. Westland, “Employing transaction aggregation strategy to detect credit card fraud”, *Expert Systems with Applications*, Vol. 39, No. 16, pp. 12650-12657, 2012.

17. Pun, J. and Y. Lawryshyn, “Improving Credit Card Fraud Detection using a Meta-Classification Strategy”, *International Journal of Computer Applications*, Vol. 56, No. 10, pp. 41-46, 2012.

18. Quah, J.T.S. and M. Sriganesh, “Real-time credit card fraud detection using computational intelligence”, *Expert Systems with Applications*, Vol. 35, No. 4, pp. 1721-1732, 2008.

19. Srivastava, A., A. Kundu, S. Sural and A.K. Majumdar, “Credit Card Fraud Detection Using Hidden Markov Model”, *IEEE Transactions on Dependable and Secure Computing*, Vol. 5, No. 1, pp. 37-48, 2008.

20. Bhusari, V. and S. Patil, “Application of Hidden Markov Model in credit card fraud detection”, *International Journal of Distributed and Parallel Systems (IJDPS)*, Vol. 2, No. 6, pp. 203-211, 2011.

21. Rani, J.K., S.P. Kumar, U.R. Mohan and C.U. Shankar, “Credit card fraud detection analysis”, *International Journal of Computer Trends and Technology*, Vol. 2, No. 1, pp. 24-27, 2011.

22. Prakash, A. and C. Chandrasekar, “A Novel Hidden Markov Model for Credit Card Fraud Detection”, *International Journal of Computer Applications*, Vol. 59, No. 3, pp. 35-40, 2012a.

23. Prakash, A. and C. Chandrasekar, “An Ensemble Approach for Credit Card Fraud Detection”, *International Journal of Computer Applications*, Vol. 59, No. 19, pp. 1-6, 2012b.

24. Gadi, M.F.A., X. Wang and A.P. Lago, “Credit card fraud detection with artificial immune system”, *Lecture Notes in Computer Science: Artificial Immune Systems*, Vol. 5132, pp. 119-131, 2008a.

25. Gadi, M.F.A., X. Wang and A.P. Lago, “Comparison with parametric optimization in credit card fraud detection”, *Proceedings of the 7th international conference on machine learning and applications (ICMLA '08)*, pp. 279-285, San Diego, California, USA, 2008.

26. Patil, D.D., S.M. Karad, V.M. Wadhai, J.A. Gokhale and P.S. Halgaonkar, “Efficient scalable multi-level classification scheme for credit card fraud detection”, *International Journal of Computer Science and Network Security (IJCSNS)*, Vol. 10, No. 8, pp. 123-130, 2010.

27. Sherly, K.K. and R. Nedunchezhian, "BOAT adaptive credit card fraud detection system", *Proceedings of IEEE international conference on computational intelligence and computing research (ICCIC)*, pp. 1-7, Coimbatore, India, 2010.

28. Bhattacharyya, S., S. Jha, K. Tharakunnel and J.C. Westland, "Data mining for credit card fraud: A comparative study", *Decision Support Systems*, Vol. 5, No. 3, pp. 602-613, 2011.

29. Sahin, Y. and E. Duman, "Detecting credit card fraud by decision trees and support vector machines", *Proceedings of the international multiconference of engineers and computer scientists (IMECS)*, pp. 442-447, Hong Kong, 2011a.

30. Alowais, M.I. and L.K. Soon, "Credit card fraud detection: Personalized or aggregated model", *Proceedings of the 3rd FTRA international conference on mobile ubiquitous and intelligent computing (MUSIC)*, pp. 114-119, Vancouver, Canada, 2012.

31. Sahin, Y., S. Bulkan and E. Duman, "A cost-sensitive decision tree approach for fraud detection", *Expert Systems with Applications*, Vol. 40, No. 15, pp. 5916-5923, 2013.

32. Noghani, F.F and M.H. Moattar, "Ensemble classification and extended feature selection for credit card fraud detection", *Journal of AI and Data Mining*, 2016.

33. Maes, S., K. Tuyls, B. Vanschoenwinkel and B. Manderick, "Credit card fraud detection using Bayesian and neural networks", *Interactive image-guided neurosurgery*, American Association Neurological Surgeons, pp. 261-270, 1993.

34. Ghosh, S. and D.L. Reilly, "Credit Card Fraud Detection with a Neural-Network", *Proceedings of the 27th Hawaii International Conference on System Sciences*, pp. 621-630, 1994.

35. Aleskerov, E., B. Freisleben and B. Rao, "CARDWATCH: A neural network based database mining system for credit card fraud detection", *Proceedings of the IEEE/IAFE*

1997 computational intelligence for financial engineering (CIFEr), pp. 220-226, New York City, New York, USA, 1997.

36. Chen, R.C., S.T. Luo, X. Liang and V.C.S. Lee, “Personalized Approach Based on SVM and ANN for Detecting Credit Card Fraud”, *Proceedings of the International Conference on Neural Networks and Brain (ICNN&B '05)*, pp. 810-815, 2005b.

37. Sahin, Y. and E. Duman, “Detecting credit card fraud by ANN and logistic regression”, *Proceedings of international symposium on innovations in intelligent systems and applications (INISTA)*, pp. 315-319, Istanbul, Turkey, 2011b.

38. Mishra, M.K. and R. Dash, “A Comparative Study of Chebyshev Functional Link Artificial Neural Network, Multi-Layer Perceptron and Decision Tree for Credit Card Fraud Detection”, *Proceedings of the 2014 International Conference on Information Technology*, pp. 228-233, 2014.

39. Srivastava, A., M. Yadav and S. Basu, “Credit Card Fraud Detection at Merchant Side using Neural Networks”, *Proceedings of the 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 667-670, March 2016.

40. Filippov, V., L. Mukhanov and B. Shchukin, “Credit card fraud detection system”, *Proceedings of the 7th IEEE international conference on cybernetic intelligent systems (CIS 2008)*, pp. 1-6, London, UK, 2008.

41. Panigrahi, S., A. Kundu, S. Sural and A.K. Majumdar, “Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning”, *Information Fusion*, Vol. 10, No. 4, pp. 354-363, 2009.

42. Bahsen, A.C., A. Stojanovic, D. Aouada and B. Ottersen, “Cost Sensitive Credit Card Fraud Detection using Bayes Minimum Risk”, *Proceedings of the 12th International Conference on Machine Learning and Applications*, pp. 333-338, 2013.

43. Chen, R.C., M.L. Chiu, Y.L. Huang and L.T. Chen, "Detecting Credit Card Fraud by Using Questionnaire-Responded Transaction Model Based on Support Vector Machines", *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2004)*, pp. 800-806, 2004.

44. Chen, R., T. Chen, Y. Chien and Y. Yang, "Novel questionnaire-responded transaction approach with SVM for credit card fraud detection", *Lecture Notes in Computer Science: Advances in Neural Networks*, Vol. 3497, pp. 916-921, 2005a.

45. Hejazi, M. and Y.P. Singh, "Credit data fraud detection using kernel methods with support vector machine", *Journal of Advanced Computer Science and Technology Research*, Vol. 2, No. 1, pp. 35-49, 2012.

46. Kamboj, M. and S. Gupta, "Credit card fraud detection and false alarms reduction using support vector machines", *International Journal of Advance Research, Ideas and Innovations in Technology*, Vol. 2, No. 4, 2016.

47. Ma, H. and X. Li, "Application of data mining in preventing credit card fraud", *Proceedings of international conference on management and service science (MASS)*, pp. 1-6, Wuhan, China, 2009.

48. Ozcelik, M.H., M. Isik, E. Duman and T. Cevik, "Improving a credit card fraud detection system using genetic algorithm", *Proceedings of international conference on networking and information technology (ICNIT)*, pp. 436-440, Manila, Philippines, 2010.

49. Duman, E. and M.H. Ozcelik, "Detecting credit card fraud by genetic algorithm and scatter search", *Expert Systems with Applications*, Vol. 38, No: 10, pp. 13057-13063, 2011.

50. Taklikar, S.H. and R.P. Kulkarni, "Credit Card Fraud Detection System Based on User Based Model with GA and Artificial Immune System", *Proceedings of the 14th*

International Conference on Applications of Computer Engineering (ACE '15), Recent Advances on Computer Engineering, Seoul, South Korea, pp. 163-168, 2015.

51. Halvaiee, N.S. and M.K. Akbari, “A novel model for credit card fraud detection using Artificial Immune Systems”, *Applied Soft Computing*, Vol. 24, pp. 40-49, 2014.

52. Bentley, P.J., J. Kim, G.H. Jung and J.U. Choi, “Fuzzy Darwinian detection of credit card fraud”, *Proceedings of the 14th annual fall symposium of the Korean information processing society*, pp. 1-4, 2000.

53. Sanchez, D., M.A. Vila, L. Cerda and J.M. Serrano, “Association rules applied to credit card fraud detection”, *Expert Systems with Applications*, Vol. 36, No. 2, pp. 3630-3640, 2009.

54. Kundu, A., S. Sural and A.K. Majumdar, “Two-stage credit card fraud detection using sequence alignment”, *Information Systems Security*, Vol. 4332/2006, pp. 260-275, 2006.

55. Kundu, A., S. Panigrahi, S. Sural and A.K. Majumdar, “BLAST-SSAHA hybridization for credit card fraud detection”, *IEEE Transactions on Dependable and Secure Computing*, Vol. 6, No. 4, pp. 309-315, 2009.

56. Cobb, B.R., “An influence diagram model for detecting credit card fraud”, *Proceedings of the 5th European conference on probabilistic graphical models*, pp. 89-96, Helsinki, Finland, 2010.

57. Mahmoudi, N. and E. Duman, “Detecting credit card fraud by modified fisher discriminant analysis”, *Expert Systems with Applications*, Vol. 45, No. 5, pp. 2510-2516, 2015.

58. Duman, E. and I. Elikucuk, “Solving Credit Card Fraud Detection Problem by the New Metaheuristics Migrating Birds Optimization”, *Proceedings of the International Work-*

Conference on Artificial Neural Networks (IWANN 2013) Advances in Computational Intelligence, pp. 62-71, 2013a.

59. Duman, E. and I. Elikucuk, “Applying Migrating Birds Optimization to Credit Card Fraud Detection”, *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2013) Trends and Applications in Knowledge Discovery and Data Mining*, pp. 416-427, 2013b.

60. Duman, E., A. Buyukkaya and I. Elikucuk, “A Novel and Successful Credit Card Fraud Detection System Implemented in a Turkish Bank”, *Proceedings of the IEEE 13th International Conference on Data Mining Workshops*, pp. 162-171, 2013.

61. Visa USA, *Merchant category codes for IRS form 1099-MISC reporting*, 2004.

62. Fawcett, T., “ROC Analysis in Pattern Recognition”. *Pattern Recognition Letters*, Vol. 27, No. 8, pp. 861-874, 2006.

63. Rijsbergen, V., *Information Retrieval*, Butterworth, 1979.

64. Spackman, K.A., “Signal detection theory: Valuable tools for evaluating inductive learning”, *Proceedings of the 6th International Workshop on Machine Learning*, pp. 160–163, San Mateo, California, USA, 1989.

65. Tape, T.G., “Interpreting Diagnostic Tests”, 2017, http://gim.unmc.edu/dxtests/Default.htm, accessed at June 2017.

66. Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I.H. Witten, “The WEKA data mining software: An update”, *SIGKDD Explorations*, Vol. 11, No. 1, 2009.

67. Sharma, N., A. Bajpai and L. Ritoriya, “Comparison the various clustering algorithms of weka tools”, *International Journal of Emerging Technology and Advanced Engineering*, Vol. 2, No. 5, pp. 73-80, 2012.

68. Dempster, A.P., N.M. Laird and D.B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm”, *Journal of the Royal Statistical Society*, Vol. 39, No. 1, pp. 1-38, 1977.

69. Shen, A., “INFOGRAPHIC: Americans are spending a whopping $704.18 on gifts this year”, 2011, http://www.businessinsider.com/what-americans-spend-on-christmas-2011-12, accessed at May 2017.

70. Mccarra, D., “Irish consumers will spend €257 million online this Christmas”, 2011, http://sociable.co/web/irish-consumers-will-spend-e257-million-online-this-christmas/, accessed at May 2017.

71. Chamber of Certified Public Accountants of Ankara, *Consumer Price Indexes and Change Ratios*, 2013.

72. Cleary, J.G. and L.E. Trigg, “K*: An instance-based learner using an entropic distance measure”, *Proceedings of the 12th international conference on machine learning*, pp. 108-114, Tahoe City, California, USA, 1995.

73. Quinlan, J.R., “Induction of decision trees”, *Machine Learning*, Vol. 1, No. 1, pp. 81-106, 1986.

74. Alpaydın, E., *Introduction to machine learning*, MIT Press, 2004.

75. Breiman, L., “Random forests”, *Machine Learning*, Vol. 45, No. 1, pp. 5-32, 2001.

76. Cortes, C. and V. Vapnik, “Support-vector networks”, *Machine Learning*, Vol. 20, No. 3, pp. 273-297, 1995.

77. Stack Exchange, “Computing the ROC curve for ensemble classifier”, 2017, https://stats.stackexchange.com/questions/266123/computing-the-roc-curve-for-ensemble-classifier, accessed at June 2017.

78. Barreno, M., A.A. Cardenas and J.D. Tygar, “Optimal ROC curve for a combination of classifiers”, *Advances in Neural Information Processing Systems (NIPS)*, pp. 57-64, 2004.

79. Jurkowski, P., M. Cwiklinska-Jurkowska, Z. Doniec and A. Szaflarska-Poplawska, “Receiver operating characteristic (ROC) and other curves measuring discriminability of classifiers’ ensemble for asthma diagnosis”, *Annales Academiae Medicae Bialostocensis*, Vol. 50, No. 2, pp. 65-67, 2005.

80. Evangelista, P.F., M.J. Embrechts, P. Bonissone and B.K. Szymanski, “Fuzzy ROC curves for unsupervised nonparametric ensemble techniques”, *Proceedings of the international joint conference on neural networks (IJCNN ‘05)*, Montreal, Quebec, Canada, 2005.