# A Hybrid Approach to Detect Credit Card Fraud

**Nadisha Abdulla[*], Rakendu R, Surekha Mariam Varghese[**]**

[*]Department of Computer Science and Engineering
[**]Mar Athanasius College of Engineering, Kothamangalam, Kerala, India

***Abstract-*** With the tremendous growth of e-commerce use of credit cards for online purchases has increased to a great extent and it caused an explosion in the credit card fraud. Fraud has become one of the major ethical issues in the credit card industry. Fraud associated with credit card are also rising today as it has the most popular mode of payment for both online as well as regular purchase. In order to detect frauds from the mix of genuine as well as fraudulent transactions, efficient fraud detection techniques to detect them accurately are vital rather than simple pattern matching techniques. Here an approach is done to detect the credit card fraud using a hybrid approach which involve stages of pre-processing in which anonymous transactions are removed, genetic algorithm modelled for feature selection and support vector machine for classification. The proposed model is done on UCSD-FICO data mining contest 2009 dataset (anonymous and imbalanced). It is the dataset used in competition which was organized by FICO, the leading provider of analytics and decision management technology and the University of California, San Diego UCSD. This paper describes a simple fraud detection mechanism which can effectively detect fraud with great accuracy.

***Index Terms-*** Data Pre-processing, Clustering, Feature selection, genetic algorithm based K Nearest Neighbour method, Classification, support vector machine

## I. Introduction

With the emerging rise of technology today, the ecommerce and the online payments has grown to such a large extent and people rely on it for most of their needs. It has become a great boon to the modern world to carry out an easy way of life. As the credit card gives a lot of convenience to the users, Frauds caused due to these are potentially dangerous and are even more. As our lives become increasingly digital, a growing number of financial transactions are conducted online. Fraudsters have been quick to adapt to this trend, and to devise clever ways to defraud online payment systems. While this type of activity involves criminal rings, a well-educated fraudster can create a very large number of synthetic identities on his own, and use these to carry on sizeable schemes. New types of frauds are getting devised and hence the detection of frauds are becoming difficult. While taking ecommerce transactions the main problem that has been faced is that, the fraudulent transactions appears in a most cunning way as it looks similar as the legal one's. This puts many financial institutions and enterprises in trouble. Hence the efficient way of fraud detection mechanism is very much necessary rather than using simple classification techniques and the pattern matching techniques [1]. The challenging part is to detect frauds in the highly imbalanced datasets were the legal transactions are on the maximum and the fraudulent transactions are about very less amount.

The research papers about credit card fraud detection are very few and that is mainly because of the lack of publicly available datasets. This makes the researchers to have great difficulty in performing experiments. Since the credit card information are confidential, the bank owners and the other financial enterprise owners are not ready to disclose the credit card information's about their customers because of the privacy concerns. Due to this, only fewer papers seems to be implemented and still there are some of the successful applications have been developed and evolved from numerous research communities, which are driven by artificial immune systems, artificial intelligence, database, distributed and parallel computing, expert systems, fuzzy logic, genetic algorithms, machine learning, neural networks[2], pattern recognition and others. Since the emergence of many advanced computing and classification systems including the support vector machine [3] and the optimization technique like genetic algorithm [4] shows a greater fluctuations in the implementation of many different technologies due to the accuracy and efficiency it produces. This project is to perform credit card fraud using a hybrid approach of Genetic Algorithm and Support vector machines.

## II. RELATED WORKS

A number of data mining techniques are there like classification, clustering, advanced neural networks, prediction and regression models used for different data mining approaches are used for fraud detection. A Hidden Markov Model [5] is a double embedded stochastic process with used to model much more complicated stochastic processes as compared to a traditional Markov model. If an incoming credit card transaction is not accepted by the trained Hidden Markov Model with sufficiently high probability, it is considered to be fraudulent transactions. HMM, Baum Welch algorithm is used for training purpose and K-means algorithm for clustering. HMM sores data in the form of clusters depending on three price value ranges low, medium and high [5]. The probabilities of initial set of transaction have chosen and FDS checks whether transaction is genuine or fraudulent. Since HMM maintains a log for transactions it reduces tedious work of employee but produces high false alarm as well as high false positive. Sometimes it may take long time to process requests. Random forests is a widely used method in detecting fraud. Random forest [6] is an ensemble of decision trees. The basic principle behind ensemble methods is that a group of "weak learners" can come together to form a "strong learner." Random forests grow many decision trees. Here each individual decision tree is a "weak learner," while all the decision trees taken together are a "strong learner."

When a new object is to be classified, it is run down in each of the trees in the forest. Each tree gives a classification output or “vote” for a class. The forest classifies the new object into the class having maximum votes. Random forests are fast and they can efficiently handle unbalanced and large databases with thousands of features. Due to the construction of long decision trees it shows slower performances. The K-nearest neighbor (KNN) technique[7]is a simple algorithm which stores all vailable instances; then it classifies any new instances based on a similarity measure. The KNN algorithm is example of an instance based learner. In the nearest neighbor classification method, each new instance is compared with existing ones by using a distance metric, and the closest existing instance is used to assign the class to the new one. Sometimes more than one nearest neighbor is used, and the majority class of the closest K neighbors is assigned to the new instance. Among the various credit card fraud detection methods, the KNN achieves consistently high performance, without a priori assumptions about the distributions from which the training examples are drawn. In the process of KNN, we classify any incoming transaction by calculating nearest point to new incoming transaction. If the nearest neighbor is fraudulent, then the transaction is classified as fraudulent and if the nearest neighbor is legal, then it is classified as legal. Yan-Qing et al [9] propose neural network based security solution they described their solution as a parallel granular neural network (GNN) is developed to speed up data mining and knowledge discovery process for credit card fraud detection. The data are classified into three categories: first for training, second for prediction, and third for fraud detection. After learning from training data, the GNN is used to predict on second set of data and later the third set of data is applied for fraud detection. Around eight scenarios are employed for detecting purpose. GNN gives fewer average training errors with larger amount of past training data. We also found that the number of training error is inversely proportional to the number of training cycles. The higher the fraud detection error is, the greater the possibility of that transaction being actually fraudulent.

Support vector machines are based on the conception of decision planes which define decision boundaries. A decision plane is one that separates between a set of different classes. Basically SVM classification algorithms tend to construct a hyper plane as the decision plane which does separate the samples into the two classes— positive and negative. The strength of SVMs comes from two main properties: kernel representation and margin optimization. There lies difficulty in achieving good level of accuracy using SVM. The proposed model is designed in such a way that the maximum accuracy of support vector is achieved by adopting feature selection using genetic algorithm approach. FraudMiner[10] is the work done on the UCSD dataset using the process of frequent itemset mining in which apriori is used. By using apriori algorithm, the frequently occurring pattern of transaction items are mined. The largest frequent itemsets would be the legal pattern corresponding to the customer and the rest is taken as the fraud patterns and this way the two pattern databases for legal and fraud are constructed. Through the mining the legal and fraud transactions of the customers are separated. A matching algorithm is then performed for each incoming transaction with the already created pattern databases to check for the classification in which the transaction falls. This method needs a very long time of frequent itemset mining.

## III. PROPOSED WORK

The proposed fraud detection model is based on building up of a classifier through processes including preprocessing, clustering, feature selection and SVM training. The UCSD dataset used in the proposed work consists of two sets of datasets including training set and testing test. It contains 1 lakh transactions of train data and 50000 transactions of test data where there are numerous vague transactions exists. Hence in order to remove such vague transactions the dataset should be subjected to certain processes like preprocessing. Through the preprocessing technique the single as well as anonymous transactions are filtered and it outputs the reduced datasets with relevant transactions only (train set with 21,850 transactions and test set with 9.425 transactions). Thus the dataset is now ready for the next phase of the work which is the clustering phase. The dataset should necessarily contain a class indicating whether each transaction is ether legal or fraud. Since the UCSD dataset do not have such a label, it should be labeled to the respective classes by subjecting it to the required process like clustering. The clustering phase actually works on the preprocessed data which is the output of preprocessing phase. The train and test datasets are clustered by using KMeans clustering approach. KMeans iteratively clusters the data and finally outputs the two clusters. From the two clusters obtained, take minimum population of the cluster containing the transactions which are to be labelled as fraud and the other cluster containing the population of transactions would be labelled as legal. Thus the labeled dataset is now ready. It is then subjected to the feature selection process in which the potential attributes are given more preference.

In order to make the final SVM classifier achieve good performance, selection of best combination of features from the total features is vital for training the SVM classifier. Genetic algorithm with KNearest Neighbor method is used for selecting the best subset of features. Initial population is randomly generated of genome length 8 which is the total number of attributes as that of the dataset and that is referred to as chromosomes. KNN method is used for fitness function evaluation in which the nearest neighbors of different combinations of attributes are calculated. The next generations are then created by selecting the chromosomes of high fitness values. By applying genetic operators like crossover and mutation next generations are created. The best individuals are selected by using tournament selection method. This is repeated till the best individuals are obtained. This way the informative features are selected for classification and are given as input to the next phase. The last phase comes the construction of SVM classifier which should be trained with the selected features that are obtained. With these specified features SVM classifier is constructed by training it with the train data. Since the data are to be separated only to two classes i.e.; legal or fraud, the default linear kernel function is used which can give good classification performance. As the dataset do not contain larger attributes it is best to use linear kernel for faster training. The input parameters are set to default values. After constructing the SVM classifier the test data can be applied to it and each transactions can be classified as legal or fraud. The detailed explanation of each of the above phases would be discussed in coming sections. The block diagram is given as below:

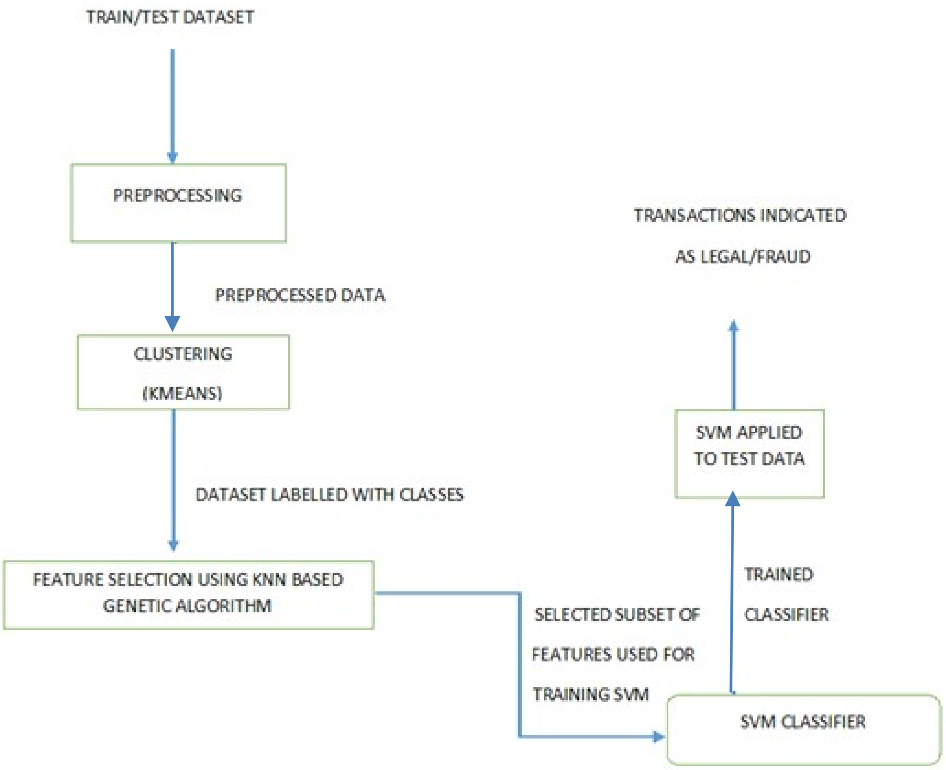

TRAIN/TEST DATASET
PREPROCESSING
PREPROCESSED DATA
CLUSTERING
(KMEANS)
DATASET LABELLED WITH CLASSES
FEATURE SELECTION USING KNN BASED GENETIC ALGORITHM
SELECTED SUBSET OF FEATURES USED FOR TRAINING SVM
SVM CLASSIFIER
TRAINED CLASSIFIER
SVM APPLIED TO TEST DATA
TRANSACTIONS INDICATED AS LEGAL/FRAUD

**Figure 3.1 Block Diagram**

3.1 DATA PREPROCESSING

The main aim of the preprocessing module is to preprocess the datasets. It aims at obtaining the datasets consisting of only relevant transactions with no unique transactions. The training set and the test set of the UCSD dataset are the inputs of the data preprocessing. This processing is done separately to both the datasets so as to make it into a reduced form removing the unwanted transactions which are mainly the single ones and thereby keeping only necessary number of transactions. The preprocessing is done to the train/test data by listing out the unique customer ID’s. Out of the total number of 19 attributes in both the datasets search for the attribute “custattr1” which refers to the customer ID. For each customer ID, scan the entire dataset for the respective transactions of the same ID. Only if there are more than two transactions exists for a particular customer ID, it is kept in the preprocessed lists of transactions else it will be removed. This procedure is done for train set and test set and thus obtain the preprocessed datasets as outputs. After preprocessing the train set has been reduced to 21,850 number of transactions and test set has been reduced to 9,425 number of transactions.

***Algorithm 1 Data Preprocessing***

```
Input: Load the train / Test data
Output: Preprocessed list of data
Initialize: Attribute matrix
CustId = find ( fieldTitles , custattr1)
For I =1 to n
If (CustId>2)
          Add     ( CustId ⟶ Accepted Set of Users)
End for
End if
```

3.2 CLUSTERING

Clustering is a process of organizing data into clusters such that there is high intra class similarity and low inter class similarity. The former refers to that how closely the units of same group resembles each other and latter refers to how the units of different groups resembles. Hence it divides the data into groups of similar objects. It is an unsupervised machine learning technique in which the data has been analyzed to perform the grouping without having any explicit training. It means that for a given unlabeled dataset it performs the natural grouping of instances without having any sort of learning approach with pre labelled instances as in supervised learning. To form clusters it is necessary to check whether the data points are close to each other and it is done by measuring distances among data points. In the proposed work, the clustering comes to role as to label the UCSD transactions. Through clustering the transactions are clustered into two sets. Each transaction is referred to as a data point and the distance among them are found so as to group it. This way the transactions are labeled into its corresponding classes i.e.; legal or fraud. In the proposed work the dataset contains total of eight potential attributes which are of Boolean values. By taking random weight value (here value is taken as 2) the dataset is partitioned. h = c/w is calculated in which C=8, where 'C' refers to the total attributes, 'W'= 2 where W is the random weight value. And 'h' is the partition value, the value of h is obtained as 4. The attributes would then be divided into two parts containing each of four attributes out of total eight. The Boolean values of each partitioned block are then converted to its respective decimal values. These are then subjected to perform K Means clustering process. The data points are plotted and the initial centroid are randomly assigned. The distance measure of data points to the centroid are calculated. The distance measure used is the city block distance.

City block distance measure also known as Manhattan distance or absolute value distance. It examines the absolute differences of coordinates of a pair of objects. The equation is as follows:

$$\sum_{j=1}^{k} |aj - bj|$$

$a_j$ and $b_j$ are the data points and k refers to the cluster.

Performing city block distance measure between data points and centroids. It clusters the data point to the centroids where the distance found to be minimum. New centroids are then calculated for each cluster and the process is repeated until there is no convergence. Finally the two clusters are obtained with different population. The minimum population cluster would be taken as fraud class and the other population of cluster would be taken as legal class.

***Algorithm 2 Clustering***

```
Input: Set of Preprocessed datasets
Output: population of cluster1 and cluster2
Initialize w= [2] , c=8
Compute h= c/ w[i]
$X_1$ = bin to dec ( 1 to h) $X_2$ = bin to dec (h+1 to c)
Input for K Means: $X_1$ , $X_2$
Initialize centroids: $c_1$ $c_2$ ....... $c_k$ , number of clusters = 2
While (no change in mean)
  For j =1 to k
    Compute $a_j – b_j$
      $C_j$ = new mean( $c_1$ $c_2$ ........... $c_{k)}$
       Print  C1,C2
       End for
     End while
 Fraud( min( class1, class2))
```

### 3.3 FEATURE SLECTION USING GENETIC ALGORITHM BASED KNN

The feature selection refers the task of identifying and selection a useful subset of features to be used to represent patterns from a larger set of features. Feature selection plays a vital role in optimizing the performance of the classifier. Training a classifier with many attributes is tedious and that increases the computational costs. Hence choosing the attributes that potentially contributes to the class is relevant and so does the feature selection. This work propose a method using genetic algorithm to identify subset of features combinations from the set of attributes so as to improve the classification accuracy. These combination of features are used to train SVM and finally classifier is prepared for classification. GA based KNN approach i.e.; genetic algorithm with K nearest neighbor approach have been used to select out the feature combination from the list of random combinations. Figure 3.2 shows simple structure of feature selection in which the total number of features refers to the total eight attributes of the dataset which are subjected to feature selection process using GA-KNN and which finally outputs the reduced number of features.

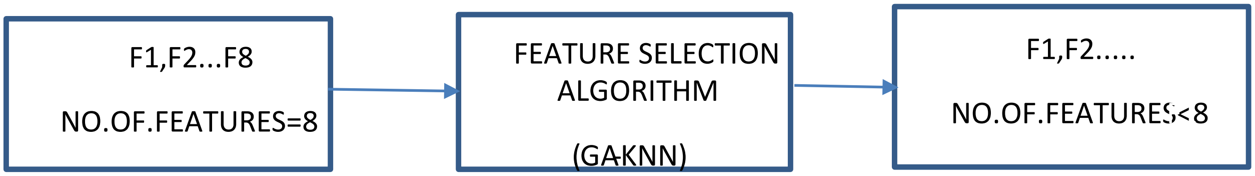


**Fig 3.2 Feature selection format**

The overall feature selection using genetic algorithm based KNN approach is modelled as given below:

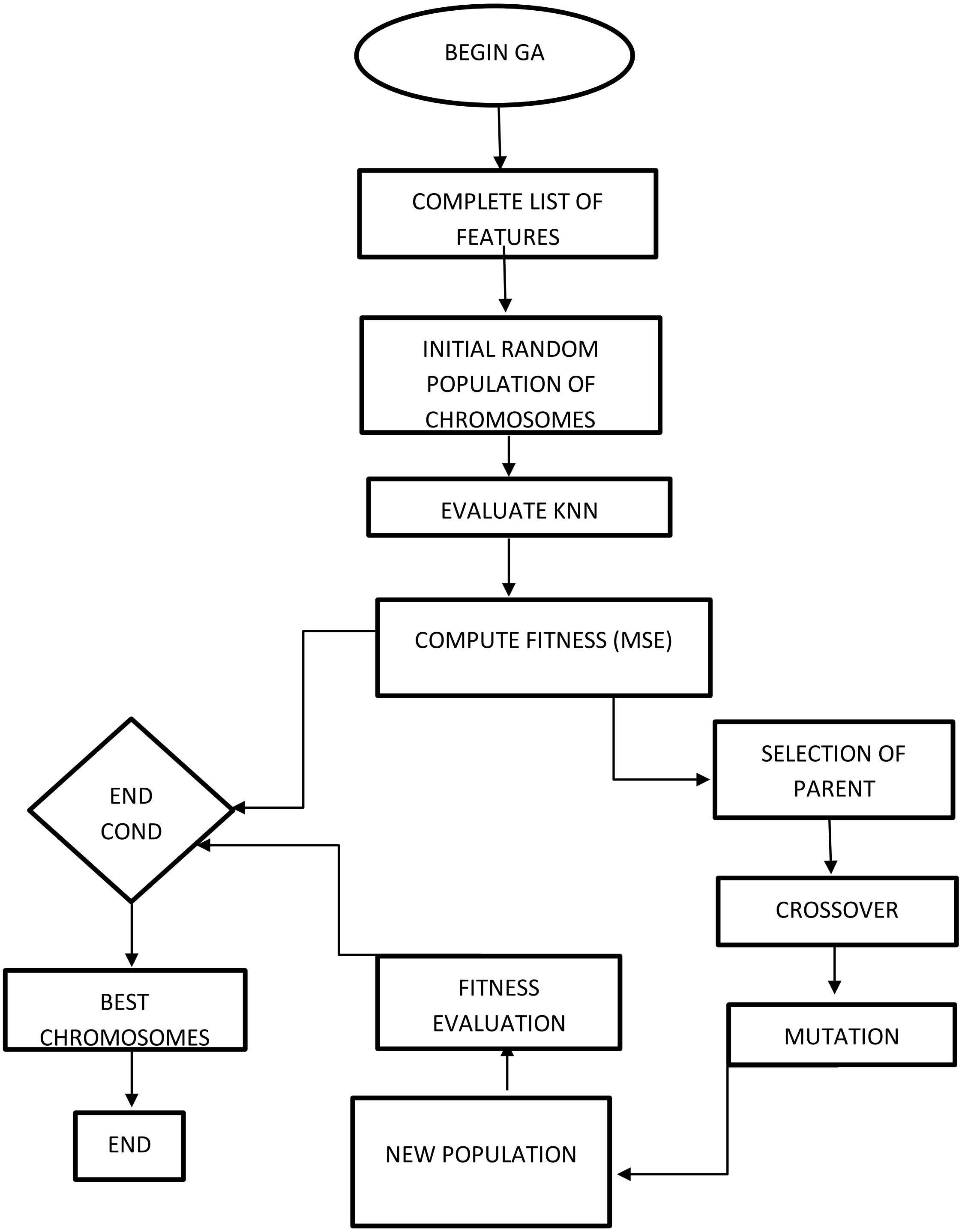
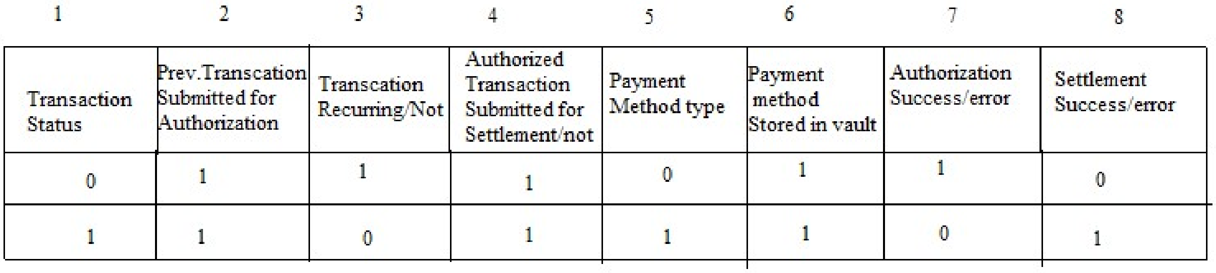


**Fig 3.3 Flowchart for Feature selection**

Genetic algorithms are evolutionary algorithms which aim at obtaining better solutions as time progresses. Since their first introduction by Holland, they have been successfully applied to many problem domains from astronomy to sports, from optimization to computer science, etc. They have also been used in data mining mainly for variable selection and are mostly coupled with other data mining algorithms. The algorithm begins with multi-population of randomly generated chromosomes. These chromosomes undergo the operations of selection, crossover and mutation. Crossover combines the information from two parent chromosomes to produce new individuals, exploiting the best of the current generation, while mutation or randomly changing some of the parameters allows exploration into other regions of the solution space. Natural selection via the fitness function assures that only the best fit chromosomes remain in the population to mate and produce the next generation. Upon iteration, the genetic algorithm converges to a global solution. In the proposed work, The chromosomes are designed as shown in figure 3.3 below. The attributes of the dataset TS, PT, TR, PU, PM, PV, AS AND SS are the features. These 8 attributes are the total features that are taken for GA. The random population is created of genome length 8 which refers to the set of combinations. The gene value '1' depicts that the particular feature indexed by the position of 1 is selected. Otherwise (if it is 0) the feature is not selected for chromosomal evaluation. For example, from the figure the first row have the values 0 1 1 1 0 1 1 0 which implies the features 2 3 4 6 7 are selected.



| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| Transaction Status | Prev.Transcation Submitted for Authorization | Transcation Recurring/Not | Authorized Transaction Submitted for Settlement/not | Payment Method type | Payment method Stored in vault | Authorization Success/error | Settlement Success/error |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |

**Figure 3.4 Chromosome representation**

The chromosomes refers to the random population of combinations created of genome length 8 indicating the population containing eight attributes. These are one set of input. Training data of eight attributes and the class vector constructed using K Means are the other set of inputs. Aim of KNN is to select out the combination having great significance contributing to the class. For each combination selected from the random population a new class vector is created by finding its nearest neighbors with respect to the train data. For each combination, the columns in the train data according to the value of given combination are taken. Hence a set of transactions for a given combination is obtained. Euclidean distance measure is computed on each row of transaction in the combination set of data to the original training data. K value taken as smallest indicates the probability of having effective results. Here k value is taken as 3. Hence 3 nearest neighbors are computed for each row of data. The class that is found to have the majority vote among the neighbors is assigned to the selected transaction in the combination set. This way KNN is applied to all rows of transactions and a new class vector is thus obtained for all rows of transaction in the selected combination.

Fitness Evaluation:

Fitness function evaluation is done by finding MSE (Mean Squared Error). This is done by comparing the newly obtained class vector to the already available class vector.

MSE (Mean Squared Error) = $1 \div n \sum (y' - y)^2$

Where $y'$ refers to the new obtained class vector and y refers to the already available class vector. This process is done for all other combinations in the random population. Fitness values for each combination is found out. The combination with lower fitness values are passed to the next generation. The new set of chromosomes are found by applying crossover and mutation.

For instance: 1 1 0 1 0 0 1 0 is the chromosome

It denotes the attributes 1 2 4 and 7 are selected. Then the columns in the training dataset according to these values are taken. KNN is applied to it. Euclidean distance between the each row of transaction in the combination set of data and the training set is calculated as:

Distance= $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + \dots\dots\dots}$.

Nearest three neighbors are taken and the class corresponding to the majority is assigned to the data. This is done for all rows in the combination set and obtains a class vector of $y'$. MSE is calculated for $y'$ and the already available class vector y. Fitness score is hence obtained. Fitness scores for set of combinations are obtained in a similar way and the lower fitness value combinations are passed to the next generation to perform mutation and cross over. Tournament selection method is used here to select the chromosomes due to its simplicity, speed and efficiency. This selection method make sure that no worst individuals passed to the next generation.

Crossover:

Uniform crossover is applied in which the random bits of parents are selected. This way child is created. For example:

Suppose parents selected are p1 and p2
P1: 1 2 5 6 8 → 11001101
P2: 1 2 4 5 6 → 11011100
Applying crossover
~~1 1~~0 0 1 1 ~~0 1~~
1 1 0 1 1 1 0 0
Result: ~~1~~ 1 0 1 1 1 0 1 ( 1 2 4 5 6 8)

P1 and P2 of combinations 1 2 5 6 8 and 1 2 4 5 6 are selected which are then subjected to uniform crossover where the random bits of p1 and p2 are exchanged to form the resultant output as 1 1 0 1 1 1 0 1 which is of the combination 1 2 4 5 6 8.

Mutation:

The random bits of the crossover output is changed in some position to form the mutated output.

For example: taking the chromosome 1 1 0 **1** 1 1 0 1 in which the fourth position is mutated and the resultant is obtained as:

1 1 0 0 1 1 0 1 which is of the combination 1 2 5 6 8

For instance: If the chromosome of best featured index of combination obtained as

1 1 0 0 1 1 0 1, indicates that the attributes 1 2 5 6 and 8 columns of the data affects the classification most and these would be taken for training SVM.

The new population are generated in this manner. The process is repeated till the maximum number of generations are reached and the best chromosome features are obtained which can be made to train SVM.

## 3.4 CLASSIFICATION USING SUPPORT VECTOR MACHINE

An SVM classifier classifies data by finding the best hyperplane that separates all data points of one class from those of the other class. The *best* hyperplane for an SVM means the one with the largest margin between the two classes. Margin means the maximal width of the slab parallel to the hyperplane that has no interior data points. The support vectors are the data points that are closest to the separating hyperplane; these points are on the boundary of the slab. As with any supervised learning model, first train a support vector machine, and then cross validate the classifier. Use the trained machine to classify (predict) new data. In addition, to obtain satisfactory predictive accuracy, various SVM kernel functions are used, and must tune the parameters of the kernel functions. SVM is trained with the featured set of indices and the class vector. Train data refers to the data to be given for training which the transactions of the dataset are and the class information refers to the class type i.e. legal or fraud. Training is the process of taking content that is known to belong to specified classes and creating a classifier on the basis of that known content. The training is performed by giving the input arguments. The input is given as training data with each row corresponds to the transaction and each column represents the attributes. The next argument is the grouping variable which refers to the class vector indicating legal or fraud. To map the data to feature space for finding hyper planes the kernel functions are must. There are many kernel functions are available including linear, Gaussian and radial basis function. The default one is linear and in the proposed work linear kernel has been used as the dataset attributes are of small and it can faster do the training compared to nonlinear kernels. As the proposed work consists of only 2 classes (legal/fraud) it is best to use linear kernels for linearly separable data. Linear kernel finds the dot product so as to find the maximal hyperplane for separating the data. Training is the process of taking content that is known to belong to specified classes and creating a classifier on the basis of that known content. The training is performed here by giving the input arguments. The input is given as training data with each row corresponds to the transaction and each column represents the attributes. The next argument is the grouping variable which refers to the class vector indicating legal or fraud. To map the data to feature space for finding hyper planes the kernel functions are must. There are many kernel functions are available including linear, Gaussian and radial basis function. The default one is linear and in the proposed work linear kernel has been used as the dataset attributes are of small and it can faster do the training compared to nonlinear kernels. As the proposed work consists of only 2 classes (legal/fraud) it is best to use linear kernels for linearly separable data. Linear kernel finds the dot product so as to find the maximal hyperplane for separating the data.

Svmtrain() function is used for training. The function takes the inputs as train data and the class. An SVM classifier is built and this classifier is then used for classifying the new incoming data. Svmclassify() function is used for classifying the test data. The function takes the input as the classifier and the test data. Flowchart for the SVM is given as below:

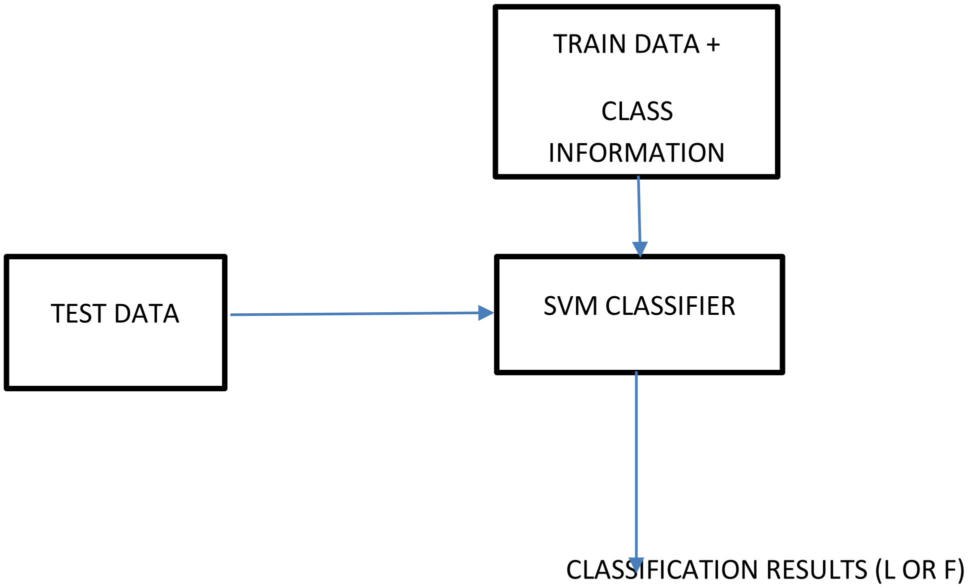


**Fig 3.5 Flowchart for SVM**

## IV. EXPERIMENTAL RESULTS

The experiment was carried out in Intel Core i5 processor with 2GB RAM implemented in MATLAB R2013a. In order to evaluate the proposed model, UCSD-FICO Data mining contest 2009 data set is used. The competition was organized by FICO, the leading provider of analytics and decision management technology, and the University of California, San Diego (UCSD). The dataset is a real dataset of e-commerce transactions and the objective was to detect anomalous e- commerce transactions. The train dataset contains 100000 transactions and the test data consists of 50,000 transactions. The dataset contains 19 fields including class labels—amount, hour1, state1, zip1, custAttr1, field1, custAttr2, field2, hour2, flag1, total, field3, field4, indicator1, indicator2, flag2, flag3, flag4, flag5, and Class. It is found that custAttr1 is the account/card number and custAttr2 is e-mail id of the customer. Both these fields are unique to a particular customer and thus decided to keep only custAttr1. The fields total and amount as well as hour1 and hour2 are found to be the same for each customer and thus removed total and hour2. Similarlystate1 and zip1 are also found to be representing the same information and thus removed state1. All other fields are anonymized and therefore decided to keep them as they are. Thus the final dataset contains 15 fields—amount, hour1, zip1, custAttr1, field1, field2, flag1, field3, field4, indicator1, indicator2, flag2, flag3, flag4, flag5. Data preprocessing is done on all these fifteen attributes of both training s well test set. The preprocessed datasets are then clustered to get the label class and this results clusters as 19,605 of class1 and 11,670 of class 2 in which class2 population is taken as fraud. After clustering the feature selection using genetic algorithm based K Nearest Neighbour approach ahs been done with the following parameters:

| GA PARAMETER | VALUE |
|---|---|
| Population size | 50 |
| Genome Length | 8 |
| Population Type | Bit string |
| Fitness Function | MSE(Mean Squared Error) |
| Number Of Generations | 10 |
| Crossover | Scattered |
| Cross over probability | 0.8 |
| Mutation | Uniform mutation |
| Mutation probability | 0.1 |
| selection | Tournament of size 2 |
| Elite count | 2 |

**Table 4.1 Genetic algorithm parameters**

Feature selection using genetic algorithm is done in order to select the significant combination of features that contributes most to the class. 8 features of the dataset are selected and the random combinations of it are made. KNN is applied to the initial population, the original train data and the already available class vector. The fitness values are also obtained corresponding to each combination by finding it mean squared error. At each generation the best chromosomes are passed to the next generation by applying genetic operators. The genetic algorithm terminates as it reaches the maximum number of generations. The best and mean values should be close to each other for accurate genetic algorithm result. The best fitness shows the best fitness value that the chromosome should have and the mean fitness refers to the average of mean fitness values of all generations. The table 4.2 below shows the best chromosomes at each generation.

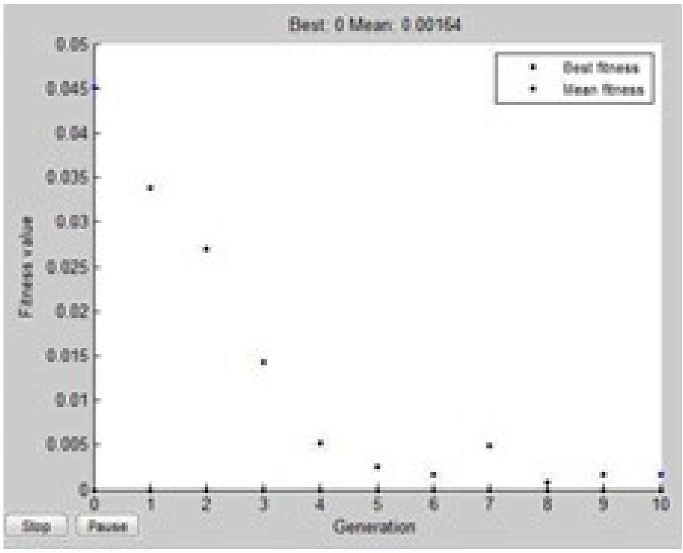


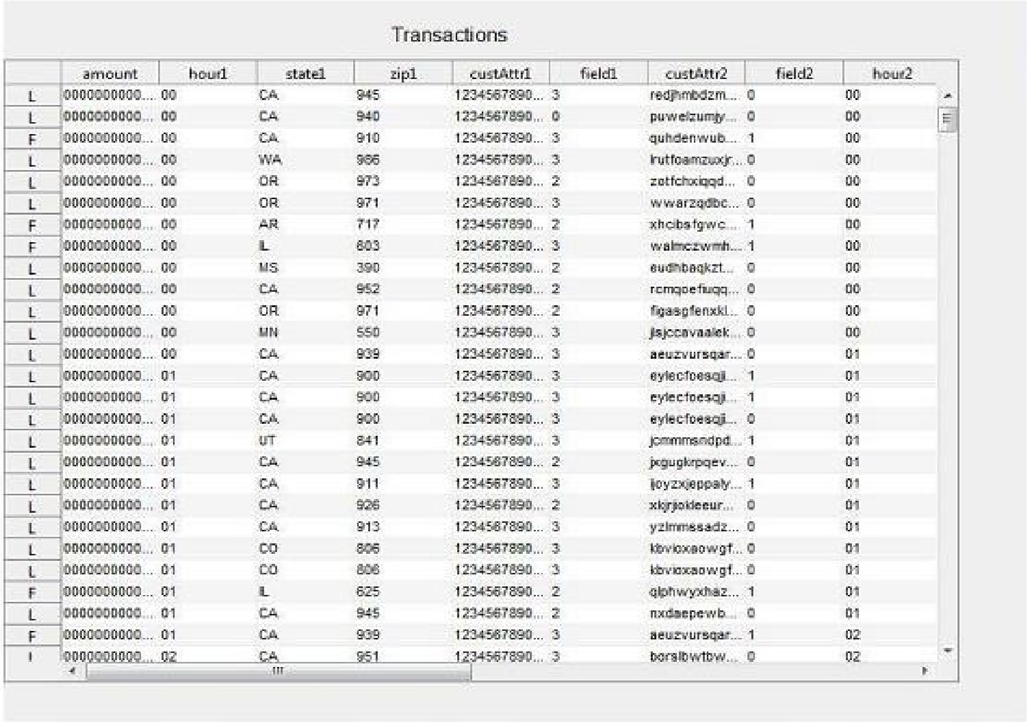
**Fig 3.5 GA simulation**

The selected features after GA-KNN approach are 1 3 4 5 6 7. These features are then used for training SVM. Svmtrain() function takes the input arguments as training data and the class vector. The linear kernel function used finds the maximum hyperplane for the separation. The dot product of the support vectors along with the bias intercept gives the optimal hyperplane. $W^TX + b$. The parameters of SVM train output is as shown below:

| FIELDS | VALUE |
|---|---|
| KERNEL FUNCTION | LINEAR KERNEL |
| ALPHA | 0.4623 TO -.4623 |
| SUPPORT VECTOR INDICES | 1:18 |
| BIAS | 0.2746 |

**Table 4.2 SVM output parameters**

After the successful training of SVM, the test data can be applied to the trained classifier to perform classification. Svmclassify() takes input arguments of test data to output the classification results. The table below shows the output of classification stating each transaction is either legal or fraud indicating it on the row names as F or L.



Transactions

| | amount | hour1 | state1 | zip1 | custAttr1 | field1 | custAttr2 | field2 | hour2 |
|---|---|---|---|---|---|---|---|---|---|
| L | 0000000000... | 00 | CA | 945 | 1234567890... | 3 | redjhmbdzm... | 0 | 00 |
| L | 0000000000... | 00 | CA | 940 | 1234567890... | 0 | puwelzumjy... | 0 | 00 |
| F | 0000000000... | 00 | CA | 910 | 1234567890... | 3 | quhdenwub... | 1 | 00 |
| L | 0000000000... | 00 | WA | 986 | 1234567890... | 3 | lrutfoamzuxjr... | 0 | 00 |
| L | 0000000000... | 00 | OR | 973 | 1234567890... | 2 | zotfchxiqqd... | 0 | 00 |
| L | 0000000000... | 00 | OR | 971 | 1234567890... | 3 | wwarzqdbc... | 0 | 00 |
| F | 0000000000... | 00 | AR | 717 | 1234567890... | 2 | xhcibsfgwc... | 1 | 00 |
| F | 0000000000... | 00 | IL | 603 | 1234567890... | 3 | walmczwmh... | 1 | 00 |
| L | 0000000000... | 00 | MS | 390 | 1234567890... | 2 | eudhbaqkzt... | 0 | 00 |
| L | 0000000000... | 00 | CA | 952 | 1234567890... | 2 | rcmqoefuqq... | 0 | 00 |
| L | 0000000000... | 00 | OR | 971 | 1234567890... | 2 | figasgfenxkl... | 0 | 00 |
| L | 0000000000... | 00 | MN | 550 | 1234567890... | 3 | jlsjccavaalek... | 0 | 00 |
| L | 0000000000... | 00 | CA | 939 | 1234567890... | 3 | aeuzvursqar... | 0 | 01 |
| L | 0000000000... | 01 | CA | 900 | 1234567890... | 3 | eylecfoesqji... | 1 | 01 |
| L | 0000000000... | 01 | CA | 900 | 1234567890... | 3 | eylecfoesqji... | 1 | 01 |
| L | 0000000000... | 01 | CA | 900 | 1234567890... | 3 | eylecfoesqji... | 0 | 01 |
| L | 0000000000... | 01 | UT | 841 | 1234567890... | 3 | jcmmmsndpd... | 1 | 01 |
| L | 0000000000... | 01 | CA | 945 | 1234567890... | 2 | jxgugkrpqev... | 0 | 01 |
| L | 0000000000... | 01 | CA | 911 | 1234567890... | 3 | ljoyzxjeppaly... | 1 | 01 |
| L | 0000000000... | 01 | CA | 926 | 1234567890... | 2 | xkjrjoideeur... | 0 | 01 |
| L | 0000000000... | 01 | CA | 913 | 1234567890... | 3 | yzlmmssadz... | 0 | 01 |
| L | 0000000000... | 01 | CO | 806 | 1234567890... | 3 | kbvioxaowgf... | 0 | 01 |
| L | 0000000000... | 01 | CO | 806 | 1234567890... | 3 | kbvioxaowgf... | 0 | 01 |
| F | 0000000000... | 01 | IL | 625 | 1234567890... | 2 | qlphwyxhaz... | 1 | 01 |
| L | 0000000000... | 01 | CA | 945 | 1234567890... | 2 | nxdaepewb... | 0 | 01 |
| F | 0000000000... | 01 | CA | 939 | 1234567890... | 3 | aeuzvursqar... | 1 | 02 |
| I | 0000000000... | 02 | CA | 951 | 1234567890... | 3 | borslbwtbw... | 0 | 02 |

**Fig 3.6 SVM classification results**

The performance measures of the proposed approach have been evaluated relevant to Accuracy, specificity and sensitivity. It has been found out that the proposed approach shows good percent of performance. The fraud here is considered to be the positive class and legal is the negative class. Therefore T, N, TP, TN, FP, and FN are defined as follows:
T represents number of fraud transactions
N represents number of legal transactions
TP represents number of fraud transactions predicted as fraud
TN represents number of legal transactions predicted as legal
FP represents number of legal transactions predicted as fraud
FN represents number of fraud transactions predicted as legal

Sensitivity: It is defined as the true positive rate, implies proportion of positives which are correctly identified as positive.

Sensitivity = TP ÷ (TP+FN)

Specificity: It is defined as the true negative rate, implies proportion of negatives which are correctly identified as negative.

Specificity = TN ÷ (TN+FP)

Accuracy: It indicates number of corrected assessments have been detected from the while assessments.

Accuracy = TP +TN ÷ (TP +TN +FP +FN)

## V. CONCLUSION AND FUTURE WORK

The problems faced by the existing methods are the lack of publicly available datasets and therefore a novel approach is used for detecting frauds on imbalanced dataset of UCSD dataset in the form of a hybrid approach involving genetic algorithm and support vector machines. In order to evaluate the proposed model, UCSD-FICO Data mining contest 2009 data set is used. The proposed fraud detection model are to be evaluated using anonymised dataset and able to handle class imbalance. The proposed model involves the stages like preprocessing, clustering, feature selection using genetic algorithm and finally support vector machine classification. These stages are successfully implemented to the dataset and created a good model for detecting fraud. SVM shows good accuracy in the proposed approach by classifying the test data to fraud and legal respectively. This accuracy is achieved through the feature selection process which have been modelled using K Nearest Neighbour approach.

## REFERENCES

[1] T. M. Cover and P. E. Hart, “Nearest neighbor pattern classification,” IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21–27, 1967.

[2] S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick, “Credit card fraud detection using Bayesian and neural net- works,” in Proceedings of the 1st International NAISO Congress on Neuro Fuzzy Technologies, pp. 261–270, 1993.

[3] C. Cortes and V. Vapnik, “Support-vector networks,” Machine Learning, vol. 20, no. 3, pp. 273–297, 1995. [4] Babatunde Oluleye, Armstrong Leisa, Jinsong Leng and Diepeveen Dean, “A Genetic Algorithm Based

[4] Feature Selection”.Internation Journal for Electronics Communication and Computer Engineering, Vol.5, Issue4, May16 2015

[5] Abhinav Srivastava, Amlan Kundu, Shamik Sural and Arun K. Majumdar, "CreditCard Fraud Detection Using Hidden Markov Model" IEEE, Transactions On Dependable And Secure Computing, Vol. 5, No

[6] 1. , January-March 2008

[7] L. Breiman, “Random forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[8] T. M. Cover and P. E. Hart, “Nearest neighbor pattern classification,” IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21–27, 1967.

[9] Mubeena Syeda, Yan-Qing Zhang and Yi Pan, “ Parallel Granular Neural Networks Detection for Fast Credit Card Fraud Detection” ,Vol-2, pp.220-226,IEEE,2002.

[10] Lean Yu Shouyang Wang, Kin Keung Lai “Credit risk assessment with a multistage neural network ensemble learning approach”, Expert Systems with Applications, Volume 34(2), Elsevier-Feb 1, 2008.

[11] K.R seeja and Masoumeh Zareapoor “FraudMiner: A Novel CreditCard Fraud Detection Model Based on Frequent Itemset Mining” Hindawi Publishing Corporation, Scientific World Journal Volume 2014.

[12] Andreas L. Prodromidis and Salvatore J. Stolfo; "Agent-Based Distributed Learning Applied to Fraud Detection"; Department of Computer Science-Columbia University; 2000. [12] Carolyn Mair, Gada Kadoda, Martin Lefley, and others; "An investigation of machine learning based prediction systems"; The journal of System Software 53; 2000; pp. 23-29.

[13] R. Wheeler, S. Aitken; "Multiple algorithms for fraud detection"; KnowledgeBased Systems 13; 2000; pp. 93-99.

[14] Dean W. Abbott, I. philip Matkovsky, John F. Elder; "An Evaluation of High-end Data Mining Tools for Fraud Detection; Elder Research – San Diego. CA 9212; oct. 1998.

[15] Bentley, P., Kim, J., Jung. G. & J Choi. 2000. Fuzzy Darwinian Detection of Credit Card Fraud, Proc. of 14th Annual Fall Symposium of the Korean Information Processing Society.

[16] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules in large databases,” in Proceedings of the 20th International Conference on Very Large DataBases, pp.487–499, 1994.

[17] E. Aleskerov and B. Freisleben, “CARD WATCH: a neural network based database mining system for credit card fraud detection,” in Proceedings of the Computational Intelligence for Financial Engineering, pp. 220–226, 1997.

[18] N. O. Francisca, “Data mining application in credit card fraud detection system,” Journal of Engineering Science and Technol- ogy, vol. 6, no. 3, pp. 311–322, 2011.

P. K. Chan, W. Fan, A. L. Prodromidis, and S. J. Stolfo, “Distributed data mining in credit card fraud detection,” IEEE Intelligent Systems and Their Applications, vol. 14, no. 6, pp. 67– 74, 1999.

[19] T. S. Quah and M. Sriganesh, “Real-time credit card fraud detection using computational intelligence,” Expert Systems with Applications, vol. 35, no. 4, pp. 1721–1732, 2008.

[20] Mubeena Syeda,Yan-Qing and Yi-Pan,“Parellel Granular Network For Credit Card Fraud Detection”. IEEE 2002. [20] D.WHITLEY,“Genetic Algorithm And Neural Network.”2003.

[21] Min Pei, Erik D Goodman, William F Punch and Ying Ding, “Genetic Algorithms for Classification and Feature Extraction”.

## AUTHORS

**First Author** – Nadisha Abdulla- M.Tech Final Year Student, Mar Athanasius College of Engineering, Kerala, India, Nadisha39@gmail.com

**Second Author** – Dr. Surekha Mariam Varghese, Head of the Department CSE, Mar Athanasius College of Engineering, Kerala, India, Surekha.laju@gmail.com