International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-2, Issue-5, May 2014

Credit Card Fraud Detection by Improving K-Means

Mahesh Singh, Aashima, Sangeeta Raheja

***Abstract*— Today, Internet has become the essential component of life. As telephone, fridge it becomes the important feature. Now a days people don't have much time and they wish to do shopping sitting at home. Credit card is purchase now and pays later. As Credit card has the power to purchase the things, its frauds also increased. In this paper, we have included the operations performed to validate the Credit card number. It will be done as a combination of Luhn algorithm and K-Means. Luhn Algorithm will be applied if a credit card number is not accepted by K-Means Algorithm. K-Means is then enhanced to addition of epochs. Epochs are the maximum number of iterations and error value is calculated.**

***Index Terms*— Clustering, Data Mining, luhn, K-Mean.**

I. INTRODUCTION

Credit Card numbers are (most times) 13 to 16 digit numbers which are protected by a special numerical condition, called Luhn check. The Luhn algorithm or Luhn method, also known as the "modulus 10" or "mod 10" algorithm, was made in the 1960s as a method of validating identification numbers. It is a simple checksum method used to validate a variety of account numbers, such as credit card numbers and Canadian Social Insurance Numbers. Many of its notoriety comes from credit card companies' adoption of it shortly after its creation in the late 1960s by IBM scientist Hans Peter Luhn (1896–1964). The algorithm is in the public domain and is in wide use today. It is not intended to be a cryptographically safe hash function; it protects from random error, not mlicious attack. Most credit cards and many government identification numbers use the algorithm as a simple method of distinguishing valid numbers from collections of random digits. For fraud detection, Luhn algorithm is to be checked against each card. To avoid this, K-mean algorithm is applied with some extension and epochs.

Manuscript received May 20, 2014.

Mahesh Singh, Assoc. Member IEEE AITM Students Branch, Branch Code: 09831, Asst. Professor CSE Department. Advanced Institute of Technology & Management, Palwal

Aashima, Student, M.Tech (CSE), Advanced Institute of Technology & Management, Palwal.

Sangeeta Raheja, Student, M.Tech (CSE), Advanced Institute of Technology & Management, Palwal

Credit card details

II. DATA MINING

Data mining refers to deriving or "mining" knowledge from large amount of data. The term is actually confusion. Mining of gold from rocks or sand is termed as gold mining rather than rock or sand mining. Thus, data mining must be more properly named "knowledge mining from data," which is unfortunately long. Nevertheless, mining is a realistic term characterizing the process that finds a small set of inestimable nuggets from a great deal of raw material. Thus, such a confusion that carries both "data" and "mining" became a popular choice. Many other terms carry a same or slightly unlike meaning to data mining, such as knowledge extraction from data, knowledge mining, data/pattern analysis, data excavation, and data dredging.

A. STEPS IN DATA MINING PROCESSING

Data mining is the key activity in a larger process called knowledge discovery in databases (KDD). A generic description of Data Mining is given in Figure 1.1.

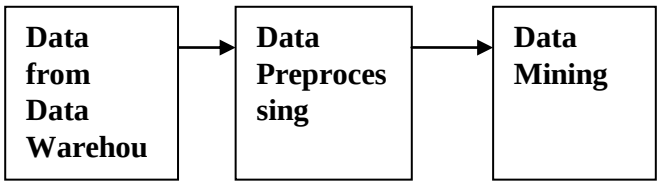


Figure 3.1 Data Mining Procedure

Data can come from several different origins and in a variety of patterns. The purpose of preprocessing is to transform the raw data into an appropriate form for data mining. Preprocessing typically involves steps like fusing data from multiple origins, selecting the data compatible for the mining task and cleaning data; e.g. handling missing values and outliers. The output of preprocessing is a standard data matrix; i.e. a vector of objects (tuples or instances) where each instance is a set of attribute values. If there are n

instances and each instance has p attributes, the standard data matrix thus has n rows and p columns. Data mining uses the preprocessed data to produce models, typically used for either description or prediction.

Data Source: Data is central to all data mining activity. First of all, data must be available and in a suitable pattern. In most real-world, larger scale, applications the necessary data is initially stored in several different relational databases and data warehouses. For data mining purposes it is, however, often assumed that the data has been preprocessed into a standard data matrix. Some data sets, however, do not fit well into the table pattern. One example is a time series where consecutive values correspond to measurements taken at consecutive times. If a time series is stored as a two-variable matrix the ordered aspect of the data is lost, something that would probably lead to a poor model. Each attribute (column in the standard data matrix) represents a specific property of the objects; i.e. each property is described by the values in that column. Obviously it is important to distinguish between different measures. Most data mining methods require the data to be in some specific pattern. Thus, one important step before applying the data mining methods is to understand what the data represents and possibly convert it to a specific pattern.

Data Preprocessing: Today's real-world databases are highly acceptable to noisy, missing, and conflicting data due to their typically huge size (often several gigabytes or more) and their similarly origin from multiple, heterogeneous origins. Low quality mining results can be due to Low-grade data.

Data preprocessing techniques are many. Data cleaning can be applied to get rid off noise and correct disproportionateness in the data. Data is combined by data integration from multiple sources into a logical data store, such as a data warehouse. Data transformations, such as normalization, may be practiced. For example, normalization may improve the accuracy and performance of mining algorithms including distance measurements. *Data* reduction can reduce the data size by collecting, removing redundant features, or clustering, for instance. These methods are not correlative; they may work together.

For example, data cleaning can involve transformations such as by transforming all entries for time field to a common pattern; this involves transformations to wrong correct data. Data processing methods, when processed before mining, can extensively progress the overall quality of the patterns mined and/or the time required for the real mining.

B. *DATA MINING TECHNIQUES*

The kinds of designs that can be discovered depend upon the data mining tasks employed. By and large, there are two ways of data mining tasks: descriptive data mining tasks that describe the general properties of the existing data, and predictive data mining tasks that tried to do predictions based on inference accessible data.

In some cases, we may have no idea regarding what kinds of patterns in their data may be interesting, and hence may wish to search for several different kinds of patterns in parallel. Thus, it is important to have a data mining system that can mine many kinds of patterns to accommodate different user expectations or applications. Furthermore, data mining systems should be able to discover patterns at various levels of abstraction.

Data mining systems should also allow customers to specify hints to guide or focus the search for interesting patterns. Because some patterns may not hold for all of the data in the database, a measure of certainty or "trustworthiness" is usually associated with each discovered pattern. Data mining functionality, and the kinds of patterns they can found, are described below.

a) Association Analysis

Association analysis is the discovery of what are commonly called association rules. It studies the occurences of items occurring together in transactional databases, and on the basis of threshold called support, identifies the recurrent item sets. Other threshold, confidence, which is the restrictive probability than an item appears in a transaction when another item appears, is used to locate association rules. Association analysis is commonly used for market basket analysis. Such as, it could be useful for the Our Video Store manager to know what movies are often rented together or if there is a relationship between renting a certain type of movies and buying popcorn or pop. The founded association rules are of the form: U V [s,c], where U and V are conjunctions of attribute value-pairs, and s (for support) is the probability that U and V appear together in a transaction and c (for confidence) is the conditional probability that V appears in a transaction when U is present. For example, the hypothetic association rule:

RentType(Y, "game") ^ Age(Y, "13-19") Buys(Y, "pop") [s=2% ,c=55%]

would indicate that 2% of the transactions considered are of customers aged between 13 and 19 who are renting a game and buying a pop, and that there is a certainty of 55% that teenage customers who rent a game also buy pop.

b) Classification and Prediction

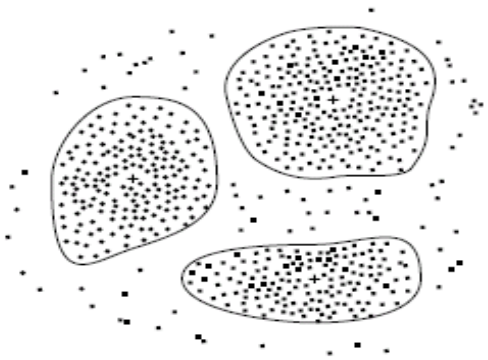
Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts, for the reason of being able to use the model to predict the class of objects whose class label is unknown. The Extracted model is based on the analysis of a set of training data (i.e., data objects whose class label is known).

"How is the derived model presented?" The derived model may be represented in various forms, such as classification (IF-THEN) rules, decision trees, mathematical method, or neural networks . A decision tree is a flow-chart-like tree structure, where each node denotes a test on an feature value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions. Decision trees can easily be converted to classification rules. A neural network, used for classification, is typically a collection of neuron-such as processing units with weighted connections between the units. There are many other technes for constructing classification models, like naïve Bayesian classification, maintain vector machines, and *k*-nearest neighbor classification. Whereas classification predicts categorical (discrete, unordered) labels, prediction models continuous-valued functions. That is, it is used to predict unavailable numerical data *values* rather than class labels. Although the term *prediction* may refer to both numeric prediction and class label forecast, we use it to refer primarily to numeric prediction. Regression analysis is a statistical

method that is most often used for numeric prediction, although other methods exist as well. Prediction also encompasses the identification of distribution trends based on the available data. Classification and prediction may have to be preceded by relevance analysis, which attempts to identify properties that do not contribute to the classification or prediction process. These properties can then be excluded.

c) Cluster Analysis

Not like classification and prediction, which analyze class-labeled data objects, clustering analyzes data objects without consulting a known class label.



Cluster Analysis

A 2-D plot of customer data with respect to departments in a company , showing three data clusters. Each cluster "center" is marked with a "+".

The Internet has taken its place beside the telephone and the television as an important part of people's lives. Customers rely on the Internet to shop, bank and invest online. Many online shoppers use credit cards to pay for their purchases. As credit card becomes the famous mode of payment, cases of

```
function cLuhn(string ACC) {
    int sum := 0
    int leng := length(ACC)
    int parity := leng modulus 2
    for i from 0 to leng - 1 {
        int ddv := integer(ACC[i])

        if i modulus 2 = parity
            ddv := ddv × 2
        if ddv > 9
            ddv := digit - 9
        sum := sum + digit
    }
    return (sum modulus 10) = 0
}
```

fraud associated with it are increasing. In this paper, we model the series of operations in credit card transaction processing using a K-Means and Luhn Algorithm and show how it can be used for the detection of frauds. A Luhn Algorithm is instructed with behavior of cardholder. If an incoming credit card transaction is not accepted by the K-mean with sufficient high probability, it is considered to be fraudulent then confirmation is given by Luhn Algorithm. We present detailed experimental results to show the effectiveness of our approach.

III. LUHN ALGORITHM

The Luhn Algorithm is the checksum procedure used by payment verification systems and mathematicians to verify the sequential integrity of real credit card numbers. It's used to help get order to seemingly random numbers and used to prevent erroneous credit card numbers from being cleared for use. The Luhn Algorithm is not used for direct credit card number generation from scratch, but rather used as a simple computational way to distinguish valid credit card numbers from random collections of numbers put together. The validation method also works with most debit cards as well. The one thing to keep in mind is that validity in terms of passing the Luhn test only means that it is mathematically valid for computational compliance purposes. It does not assure that the credit card number sequence is indeed a working number that is backed up with a valid credit card account at the card issuer's end. It is not unexceptional for one to artificially generate a mathematically valid credit card number that passes the Luhn validation check, but still ultimately does not pass as a fake credit card number with no actual substance. The Luhn algorithm only varifies the 15-16 digit credit card number and not the other critical components of a genuine working credit card account such as the expiration date and the commonly used Card Verification Value (**CVV**) and Card Verification Code (CVC) numbers (used to prove physical possession of the debit or credit card).

The algorithm proceeds in three steps. Firstly, every second digit, starts with the next-to-rightmost and proceeding to the left, is doubled. If that answer is greater than nine, its digits are summed (which is equivalent, for any number in the range 10 to 18, of subtracting 9 from it). Therefore, a 2 becomes 4 and a 7 becomes 5. Secondly, all the digits are added. Finally, the answer is divided by 10. If the remainder is zero, the original value is valid.

Example

Consider the example identification number

121-215-214.

1 1 1

2 4 4

1 1 1

2 4 4
1 1 1
5 10 1
2 2 2
1 2 2
4 4 4
Sum: 20

The sum of 20 is divided by 10; the remainder is 0, so the number is valid.

Limitation of Luhn Algorithm:

For every credit card number, it has to be applied

IV. FOR CLUSTERING AMENDING K-MEAN CLUSTERING TECHNIQUE

Clustering is a technique of Data Mining which aims at grouping a set of data objects into a specific number of clusters according to some similarity/dissimilarity measure. K-means is a well known and widely used algorithm used for clustering, but it has certain drawbacks. In my work I have tried to remove one of the main limitations of k-means algorithm. My algorithm proves to be better than the original k-means algorithm and a further research done in this direction.

The idea makes k-means more efficient, especially for dataset containing many clusters. Since, in each cycle, the k-means algorithm calculates the distances between data point and all centers, this is computationally very costly especially for huge datasets. Therefore, we do can benefit from previous iteration of k-means algorithm. For each data point, we can keep the distance to the closest cluster. At the next iteration, we compute the distance to the previous closest cluster. If the new distance is less than or equal to the previous distance, the point lies in its cluster, and there is no need to calculate its distances to the other cluster centers. This saves the time required to calculate distances to k−1 cluster centers.

Following fig. explains the idea.

Fig.(a) represents the dataset points and the initial 3 centroids.

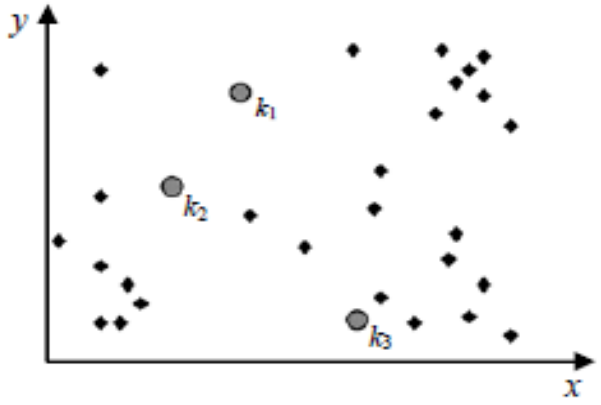


Figure (a) : Initial Centroids to a dataset

Fig.(b) shows points classified over the initial 3 centroids, and the new centroids for the next cycle.

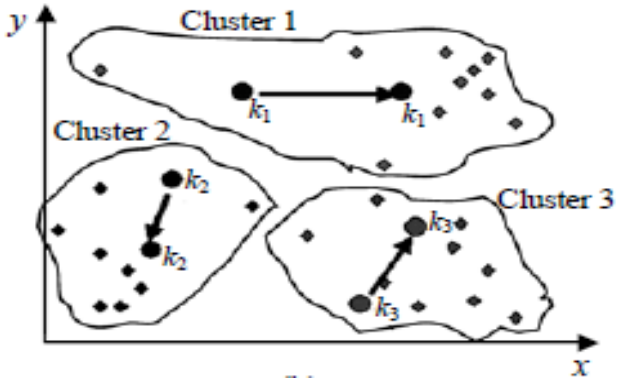


Figure (b) : Recalculating the position of the centroids

Fig. (c) shows the final clusters and their centroids.

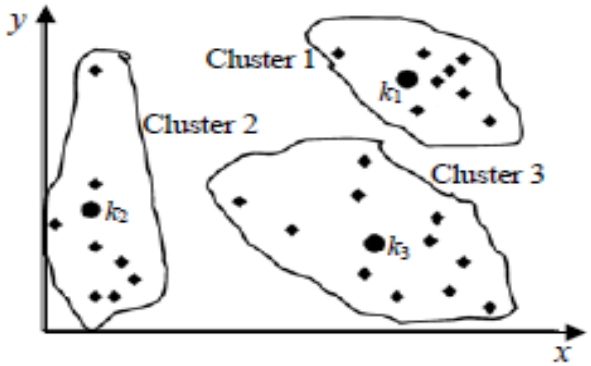


Figure (c) : Final patternions of the Centroids

When we examine Fig. (b), in Clusters 1, 2 we note that, the most points become closer to their new center, only one point in Cluster 1, and 2 points in Cluster 2 will be redistributed (their distances to all centroids must be computed), and the final clusters are presented in Fig.(c). Based on this idea, the proposed algorithm saves a lot of time.

In the proposed method, we write functions. The basic function of the k-means algorithm, that finds the nearest center for every data point, by calculating the distances to the *k* centers, and for every data point keeps its distance to the closest center.

An Efficient Amend k-Mean Clustering Algorithm :

Function KMean ()

//assign each point to its nearest cluster

1. For i = 1 to n
2. For j = 1 to k
3. Compute squared Euclidean distance $d^2(xi, mj)$;
4. endfor
5. Search the closest centroid mj to xi;
6. mj = mj+xi; nj = nj+1;
7. MSE = MSE + $d^2(xi, mj)$;
8. Clusterid[i] = number of the closest centroid;
9. Pointdis[i] = Euclidean distance to the closest centroid;
10. endfor
11. For j = 1 to k
12. mj = mj/nj;
13. endfor

K-mean can be amended by fixing the epochs. After epochs have been applied, best result should be considered depending on the error value.

Epochs are the maximum number of iterations which we will fix to 20, which tells us that if till 20 iterations that it will not go to 21 it will display result calculated ton 20th iteration.
Error value is the diff. between the means of random numbers and means of distances calc. , we will compare the error value with threshold value if error value is less than threshold value then means will be considered equal.

V. COMBINATION OF K-MEAN AND LUHN ALGO

After forming the clustering, the cards which does not belong to any of the clusters which are invalid, on them Luhn algorithm can be applied.
This will give the best result. Luhn Algorithm

VI. CONCLUSION

The Luhn algorithm is widely used on the Internet to validate of credit card numbers, but this algorithm
suffers from weaknesses, as confirmed by tests. . As the end clustering results of the k-mean clustering method are highly dependent on the selection of initial centroids , so there should be a systematic method to determine the initial centroids which makes the k-mean algorithm to converge in global optima and unique clustering results. This requirement is fulfilled by the proposed algorithm. Besides solving the problem of non-unique results, our proposed algorithm is also widely applicable to different types to problems. The problems with uniform as well as the problems with non-uniform distribution of data points are better addressed by our proposed algorithm.
K-means algorithm can be amended by using Epochs and confirming the number of clusters which will improve the efficiency of Credit card fraud detection system.

VII. REFERENCES

[1] Jiawei Han, MichelineKamber; Data Mining: Concepts and Techniques

[2] M. Halkidi, Y.Batistakis, M. Vazirgiannis; Clustering algorithms and validity measures : 0-7695-1218-6/01 2001 IEEE

[3] Rui Xu; Survey of Clustering Algorithms : IEEE Tansactions on Neural Networks, Vol 16, No. 3, May 2005

[4] Tian Zhang, Raghu Ramakrishnan, and MironLivny; BIRCH: An Efficient Data Clustering Method for Very Large Databases: Technical report, Computer sciences Dept., Univ. of Wisconsin Madison, 1996.

[5] Vladimir Estivill-Castro; Why so many clustering algorithm- A Position Paper: SIGKDD Explorations: Vol 4, Issue 1

[6] HesamIzakian;Clustering Categorical data using a Swarm-based method: 978-1-4244-5612-3/09 2009 IEEE

[7] AristidisLikas; the global k-means clustering algorithm: Pattern Recognition 36(2003).

[8] Rodrigo G.F. Soares; An Evolutionary approach for the Clustering data Problem : 978-1-4244-1821-3/08 2008 IEEE

[9] Yinghua Zhou, Hong Yu; A Novel k-means Algorithm for clustering and outlier detection: Second International conference on future information technology and management engineering: 2009 IEEE

[10] SudiptoGuha; ROCK: A robust clustering algorithm for categorical attributes: 0-7695-0071-4/99 1999 IEEE

[11] ĐinhQuangHuy and ĐinhMạnhTường :LINK-CONNECTED: A New Approach Of Clustering Algorithm For Categorical Attributes : Department of Computer Science and Engineering, Harbin institute of Technology, P.R.China, 2005.

[12] Maria Halkidi, YannisBatistakis, MichalisVazirgiannis; On Clustering Validation Techniques: Journal of Intelligent Information Systems, Vol. 17, pp. 107–145, 2001.

[13] T,Chiu, D.Fang, J.Chen, Y.Wang : A Robust and Scalable Clustering Algorithm for Mixed type attributes in large Database environment : Int.Conf. on Knowledge Discovery and Data Mining, pp. 263-268, 2001.

[14] Li, G. Biswas; Unsupervised Learning with Mixed Numeric and Nominal Data: IEEE Transaction On Knowledge and Data Engineering, Vol. 14, no. 4, 2002.

[15] SushmitaMitra, Sankar K. Pal and PabitraMitra; Data Mining in Soft Computing Framework: A Survey: in IEEE Transactions on Neural Networks, Vol. 13, No. 1, January 2002.

[16] P. Pantel; Clustering by Committee: Ph.d. dissertation, Department of Computing Science, University of Alberta, 2003

Mahesh Singh, Assoc. Member IEEE AITM Students Branch, Branch Code: 09831, Asst. Professor CSE Department. Advanced Institute of Technology & Management, Palwal

Aashima, Student, M.Tech (CSE), Advanced Institute of Technology & Management, Palwal

Sangeeta Raheja, Student, M.Tech (CSE), Advanced Institute of Technology & Management, Palwal