American International Journal of Computer Science and Information Technology
Volume.7, Number 2;April-June, 2022;
ISSN: 2837-388X| Impact Factor:
https://zapjournals.com/Journals/index.php/aijcsit
Published By: Zendo Academic Publishing

TOWARDS SAFER BANKING: MODELING TRANSACTIONS WITH GENETIC INSIGHTS

Sarah Elizabeth Johnson[1], Michael Anthony Thompson & Emily Michelle Davis

Article Info

Keywords: Credit Card Fraud, Artificial Intelligence, Transaction Modeling, Fraudulent Transactions Banking Security

Abstract

Credit card fraud poses a significant financial threat, with the United States incurring annual losses of approximately $190 billion. Harnessing artificial intelligence (AI) holds immense potential for substantially mitigating these losses. This potential lies in the ability of banks and credit card companies to employ sophisticated methods for predicting and flagging fraudulent transactions, akin to identifying "bad genes." This identification relies on an amalgamation of account data, event specifics, geographic locations, and prior transactions marked as fraudulent. In essence, these elements form the unique "DNA" of transactions (Viaene et al., 2002). This study endeavors to delve into the intricacies of genes, their constituent components, and the fundamental architecture of genes, drawing parallels with transaction modeling. Additionally, this research embarks on a personal exploration of my banking history, dissecting transaction attributes that could have prompted the bank to flag certain activities as potentially fraudulent, enabling preemptive intervention before funds are credited to my account. To facilitate this investigation, two datasets containing transaction information will be leveraged for rigorous analysis. One dataset, dated 2013, offers pertinent but less detailed information, while the other, originating from 2004, presents a comprehensive array of transaction attributes. Both datasets encompass transactions marked as fraudulent and unmarked ones. The earlier dataset encompasses particulars such as transaction type, amount, associated company, and chronological data. Hyperlinks to this dataset are meticulously outlined in the data description and the supplementary appendix (Shaughnessy, 2011; Datasets; Kou et al., 2004; Dasarathy, 2009).

Introduction

The United States loses about $190 billion dollars a year to credit card fraud and artificial intelligence can be used to dramatically reduce that figure. There is a lot of potential for banks and credit card companies to get very witty

[1] Department of Computer Science, Hampton University, 100 E. Queen Street, Hampton, VA 23668, United States of America

to quickly develop a method of predicting fraudulent transactions and identifying them as “bad genes” through information relating to accounts, events, locations, and previous transactions that were reported as fraudulent transactions. Transactions, locations, related accounts and other features can be the “DNA” of transactions (Viaene, S., Derrig, R. A., Baesens, B., &Dedene, G., 2002). This research plansto examine genes, the parts of genes and the formation of genes and model transactions after genes. This paper will also explore my own banking history and the features of those transactions that could have lead the bank to identify the situation behind the transaction as a fraudulent transaction so they could prevent it before it is credited to my account. This research will utilize two datasets with transaction information that would be sufficient for analysis and research. One is from 2013 but with less descriptive information and the other one is from 2004. They both have transactions that are marked as fraudulent and unmarked. The older dataset has information like type, amount, company and data. The hyperlinks to this data are included in description of data and the appendix (Shaughnessy, 2011; Datasets; Kou, Y., Lu, C. T., Sirwongwattana, S., & Huang, Y. P., 2004; Dasarathy, B. V.,2009).

1. Literature Review

Rama Kalyani writes that fraud is unauthorized activity happening on electronic payment systems. Electronic payment systems place high priority on fraud detection and determining if a payment is fraudulent as its being made. Genetic algorithms are evolutionary algorithms that are meant to achieve better solutions by detecting and eliminating fraud. Furthermore, credit card fraud is the use of a card by a cardholder when the owner and card issuer are not aware of its use. Rama Kalyani writes about detecting fraudulent credit card transactions. The solution to detecting fraud is finding the parameters of a transaction that lead to a fraudulent claim. RamaKalyani’s research was also performed using a dataset and the selection of the best solution parameters. Every fraud will not be the same so there will be a variation of solutions. The traditional detection method is currently based on databases and customer education. The problem with the current method is it is inaccurate, not in-time and delayed. Genetic algorithms can be used during the transaction to detect fraud and minimize false positives. The detection must be performed in real time to give the opportunity to the banks to use all of their devices to curb the fraud. The devices the card issuers have at their disposal to make sure a transaction is authorized in real-time are alerting the card holder through text or call and blocking the card. If these devices were used after the transaction was made, there’s a possibility a loss may already be imminent and it’s too late to recover funds, prevent loss of funds or prevent loss of merchandise without payment (Seeja, K. R., & Zareapoor, M., 2014).

The parameters of the transactions used in Ramakalyani’s dataset are customer id, authentication type, current balance, average bank balance, times of overdraft, credit card age, deducted amount, location of credit card use, number of times credit card used with respect to location, average daily overdraft, amount of transaction, credit card type, time of use, card holder income, card holder age, card holder position, card holder profession, marital status of card holder, average daily spending and frequency of card usage. So, genetic algorithms have purposeful use in detecting fraud and minimizing false alerts. The simple method of the genetic algorithm used included selecting the dataset, generating data in the Data, calculating and finding critical values, set threshold, compare data with critical values and then display solutions. Genetic algorithms are repeated until a predetermined number of populations of generations have been formed. Each repetition of the genetic algorithm produces one generation. The system design of the research included a dataset, a fraud and rule set, a rule engine, filtration and prioritization and then the genetic algorithm. The goal of a genetic algorithm is to find better solutions as time progresses (Oreski, Oreski, & Oreski, 2012; Rama Kalyani, 2012; Raj, S.B.E., & Portia, A.A., 2011).

2. Motivations and Problem Statement

The motive for this research is to shift the responsibility of work of solving from the banking institutions employees to an artificial intelligent system and the customer. Other motives include, saving the banking institutions money and predicting instances of fraud so merchants and banks don't lose money from fraudulent transactions. An artificial intelligent system can advance a system in terms of accuracy and time through creating generations of individual solutions to a problem in a very short period of time as an alternative to waiting for years of actual time to go by and experiencing fraudulent transactions. The problem this research is trying to solve is the overall time, money and effort spent on disputing fraudulent transactions. Furthermore, the hypotheses are modeling transactions after mammalian genes is possible, genetic algorithms can eliminate false positives in fraud detection and incorporating a genetic algorithm in fraud detection will allow a fraud detection system to learn (Chiu, C. Tsai , 2004).

3. Methodology

In nature, genes have a phenotype and a genotype. The genotype consists of the genetic makeup. The phenotype is the physical expression of the gene. For example, the brown eyes trait and the blue eyes trait are alleles in the makeup of a gene. Various combinations of those alleles will result in the physical expression of the brown eyes or blue eyes. It is possible to have the brown eyes allele and the blue eyes allele in the makeup of a gene and the eyes come out brown. This project wants to model the transactions after genes in the same way.

The phenotypes in my example are brown eyes and blue eyes. However, the phenotypes in this experiment will be authorized transactions or fraudulent transactions. The genotypes in my example are brown eye allele and blue eye allele. Nevertheless, in this research project, the genotype will consist of the features of this transaction. The goal of this project is to read in all the data from the transactions from the dataset and model them after genes so that they may be mutated and "bread" to predict the genotype of the transaction (Bentley, P. J., Kim, J., Jung, G. H., & Choi, J. U.., 2000)

This research project will attempt to create a program from an algorithm that models transactions after genes through reading the data from each transaction, storing the data from each transaction in a data structure as an array, and labeling the transactions as authorized or unauthorized. Each array is an example of a "transaction gene". The transaction genes labeled unauthorized will be run through well-known genetic algorithms to be "bread" together through crossover to create new genotypes of transactions that may result in fraud. The model for the information currently in the dataset is in Figure 1 (Özçelik, M. H., Duman, E., Işik, M., & Çevik, T, 2010).

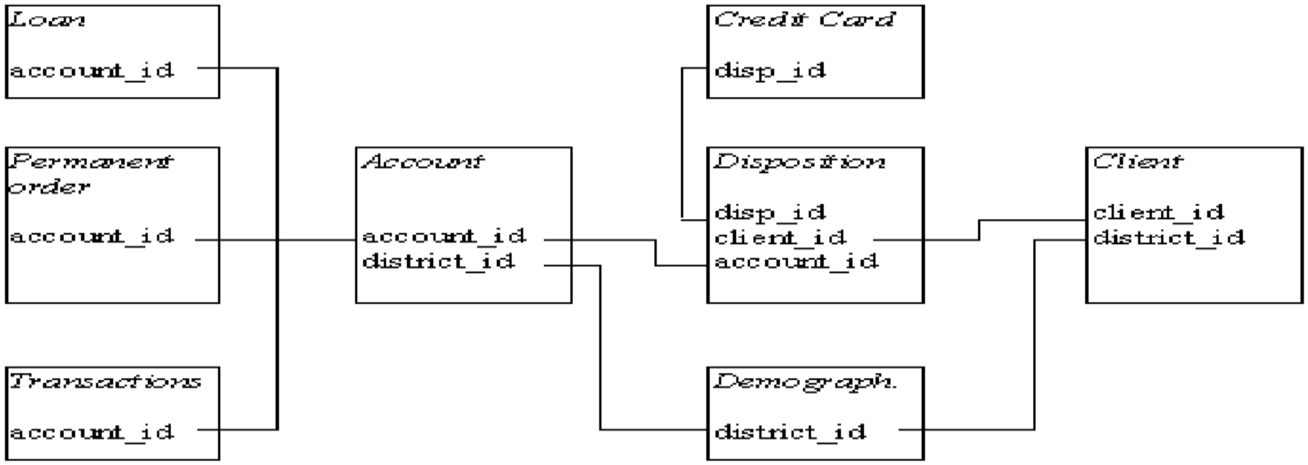


Figure 1. Description of Data

Yet, when data is read into the program, it will be organized in a linear data structure like an array, lists, queue or stack. These data structures are very similar to the models we currently have for genes. The data structure chosen for this research algorithm is a one-dimensional array. Figure 2 is a mammalian gene model and Figure 3 is an array data structure. The components of the mammalian gene are the exons, introns, TATA box, promoter-proximal element and the enhancer yeast UAS.

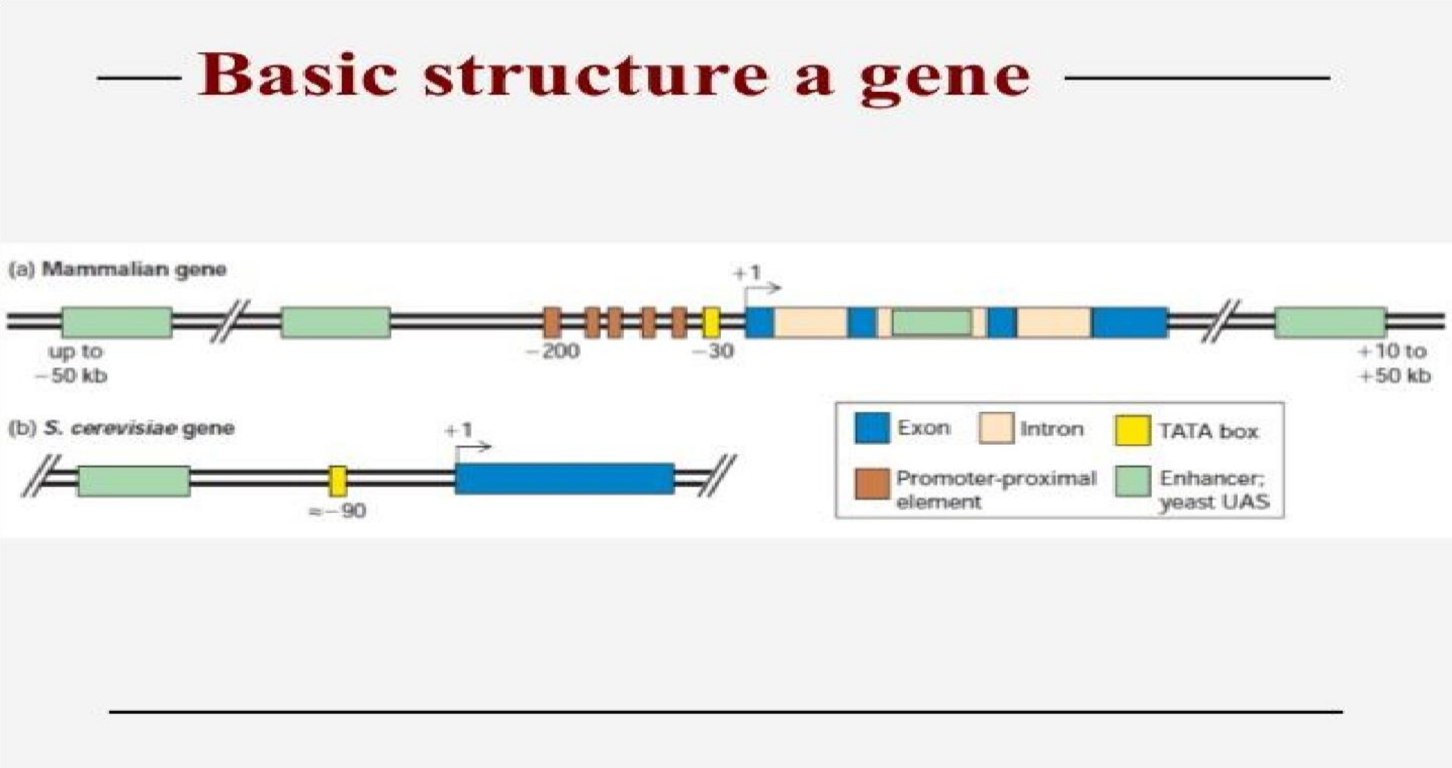


Figure 2. Mammalian Gene Structure

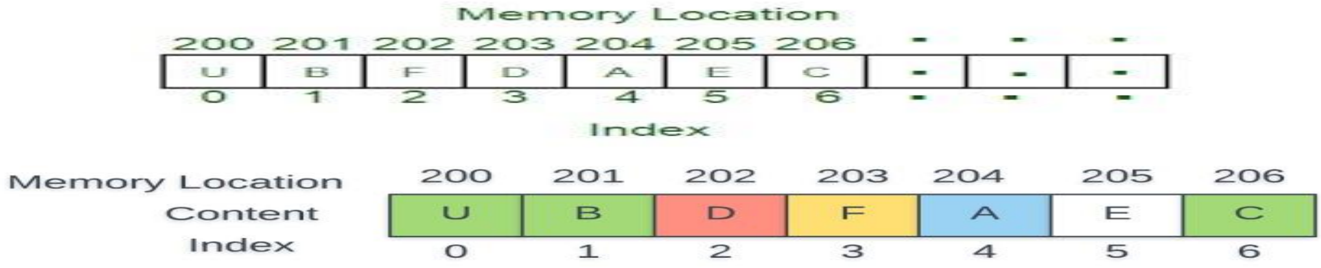


Figure 3. Array Structure

This program will follow a traditional genetic programming (TGP)algorithm. It uses three main steps: Initialization, Fitness Check, and Offspring.

3.1 Initialization

In the initialization step, the program will read through the data and find all fraudulent transaction. The amount of fraudulent transactions in this dataset has been given to us so there should be exactly 492 fraudulent transactions out of 284,000. This is good for us because the population size is given and the solution to the problem is expected to be a fraudulent transaction. The transaction information stored in the form of an array is a transaction gene and each transaction gene is an individual in the population. The code for initialization will be the most important part of the program made following the algorithm of this research. Initialization makes sure the data is in a usable form. Usable form will be a population of 284,000 transactions genes. Those genes will store the information about the transactions in a one-dimensional array.

3.2 Fitness Check

In the fitness check, it will check every transaction for authenticity. This program is looking for fraudulent transaction so a fitness of 1.0 or 100% will be given to all fraudulent transactions in the initialization stage because all of those transactions have already been deemed as fraud. A fitness of 0 will be given to the rest of the transactions. So, after the first fitness check only 492 out of 284,000 transaction will be fit enough to survive and produce offspring. The population of each generation will be 492, the same size as the original population. So all of the genotypes of the authorized transactions will "die out" and a population of fraudulent transactions will survive to produce the next generation of individuals. Each individual must be evaluated by the fitness function. Fit individuals will produce a high value on the fitness check. From the population of the fit individuals, these genes will be mutated to make offspring. The fitness check is a tool to separate the strong from the weak. The strong will be fraudulent transactions and the weak will be authorized transactions since this algorithm is looking to produce information of transactions that will lead to a fraudulent claim so the credit card issuers can take preventative action. The fitness check will consist of a comparison to other fraudulent transactions. Every individual will be closely related to a fraudulent transaction because the parent genes are fraudulent transactions. So the fitness check will compare each fraudulent transaction to the original population. If offspring produced is the same or one data point away from being the same from the original population it will receive a fitness evaluation of 1.0 or 100% and surely be used to produce offspring.

3.3 Offspring

In the Offspring stage, the goal is to create a new population. Each new population will be a generation. Each new generation will go through a fitness check and the unfit individuals or "genes" in the population will die off. Fig 4 is a flowchart of this algorithm. There are many steps and parameters in each individual step to produce useful information.

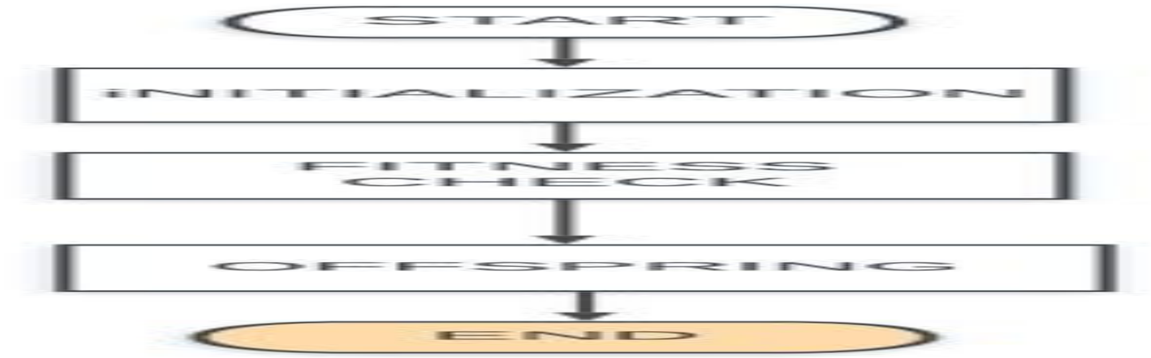
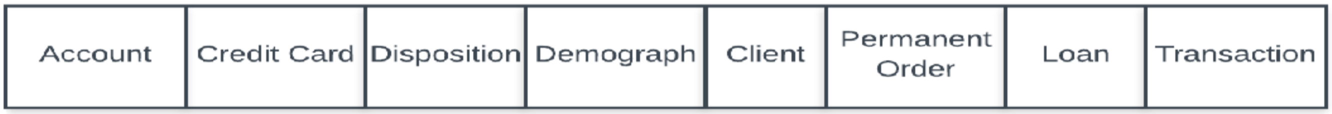


Figure 4. Traditional Genetic Programming Algorithm

3.4 Data Structures

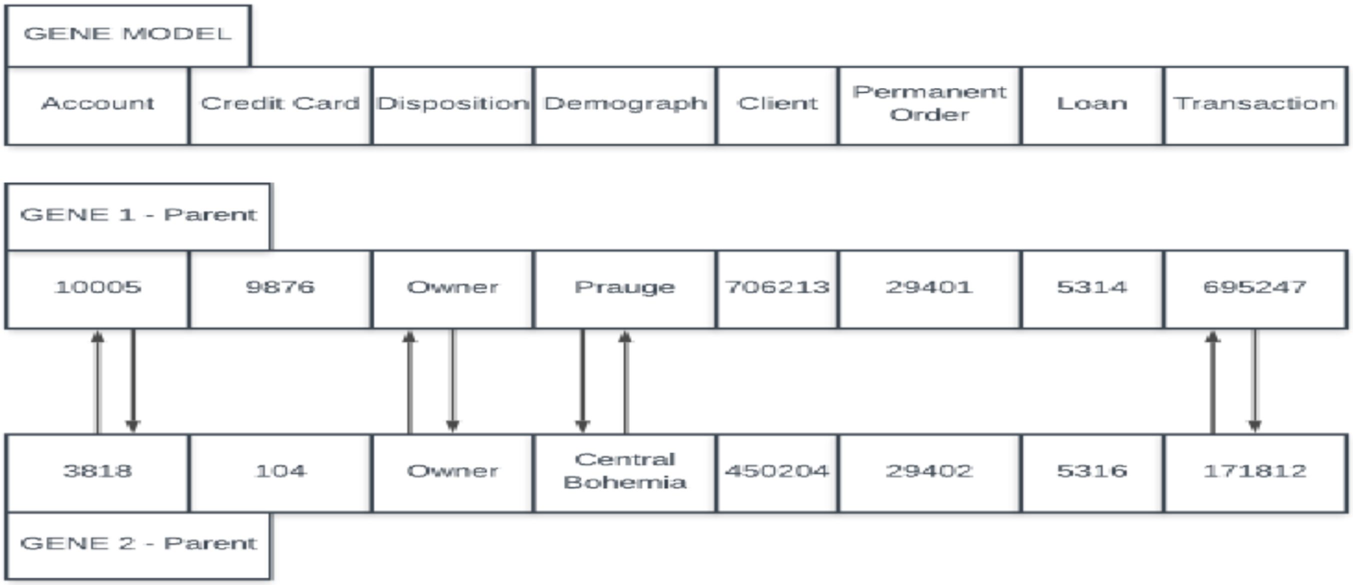
The array data structure is the most similar to the mammalian gene structure. Therefore, the genotype of the transactions will be stored in arrays. Each index of the array stores one piece of information about a transaction. Each index of an array is comparable to a structure in the basic structure of a mammalian gene in Figure 2. When the formation of an array is complete, a new transaction gene has been formed. One array will store all of the categorized information about a transaction. Fig 5 is an example,



Account	Credit Card	Disposition	Demograph	Client	Permanent Order	Loan	Transaction

Figure 5. "Transaction Gene"

The way these genes will produce offspring is through crossover. Each two parent transaction genes will produce two offspring. The offspring produced by the crossover breeding of the previous generation will make up the new generation's population. Figure 6 and Figure 7 shows how the crossover works.

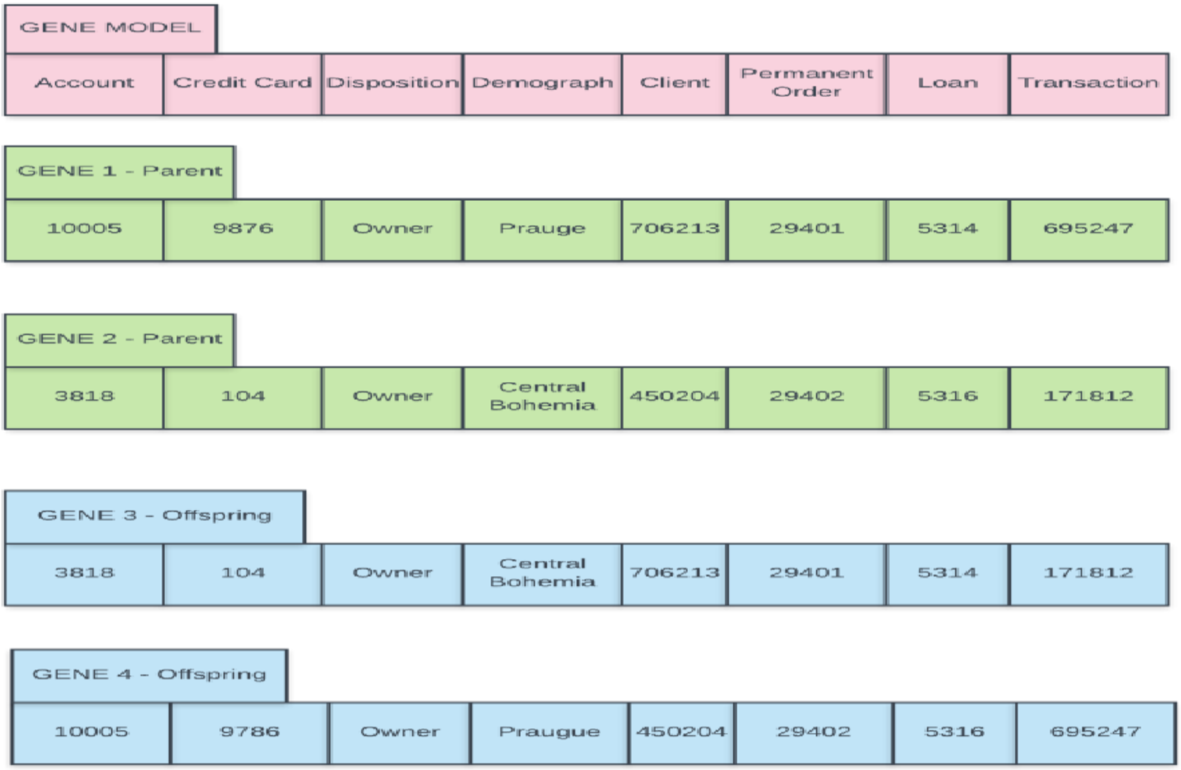


GENE MODEL							
Account	Credit Card	Disposition	Demograph	Client	Permanent Order	Loan	Transaction

GENE 1 - Parent							
10005	9876	Owner	Prauge	706213	29401	5314	695247

3818	104	Owner	Central Bohemia	450204	29402	5316	171812
GENE 2 - Parent							

Figure 6. Before Crossover



GENE MODEL							
Account	Credit Card	Disposition	Demograph	Client	Permanent Order	Loan	Transaction

GENE 1 - Parent							
10005	9876	Owner	Prauge	706213	29401	5314	695247

GENE 2 - Parent							
3818	104	Owner	Central Bohemia	450204	29402	5316	171812

GENE 3 - Offspring							
3818	104	Owner	Central Bohemia	706213	29401	5314	171812

GENE 4 - Offspring							
10005	9786	Owner	Praugue	450204	29402	5316	695247

Figure 7. After Crossover

3.5 Crossover Method

The rules for the crossover breeding method are to choose two random individuals in the population and breed them together through crossover breeding. Crossover breeding involves switching out genetic information from two parents to produce a new individual (Garner, 2011; Onan, A., & Korukoğlu, S., 2017). In this algorithm, three random categories of information will be chosen for two individuals to breed. The genetic information for those two individuals will be switched out for those three randomly chosen categories. This process will be repeated until there is a new population of 984 individuals. At this point, the population is twice the size of the original population because each parent transaction gene produced two child transaction genes. Then, the population of 984 will go through a fitness check and only 492 of 984 will survive and go on to produce the next generation.

4. Description of Data

"The datasets contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2,. V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise."

Another older available one is "German Credit fraud data", which is in ARFF format as used by Weka machine learning. Each account has both static characteristics (e.g. date of creation, address of the branch) given in relation "account" and dynamic characteristics (e.g. payments debited or credited, balances) given in relations "permanent order" and "transaction". Relation "client" describes characteristics of persons who can manipulate with the accounts. One client can have more accounts, more clients can manipulate with single account; clients and accounts are related together in relation "disposition". Relations "loan" and "credit card" describe some services which the bank offers to its clients; more credit cards can be issued to an account, at most one loan can be granted for an account. Relation "demographic data" gives some publicly available information about the districts (e.g. the unemployment rate); additional information about the clients can be deduced from this.”

To make sure a program that can learn autonomously can be produced, the program will have a survey at the end so that the customer fills out the information needed so the program can make new transaction genes or develop resistance to existing bad genes. A transaction gene may be reported as fraud but in another transaction that same gene structure was an authorized transaction. The aim of this software is to match how many times a transaction gene is reported and deemed by the bank as fraud. Therefore, at the end of the program execution, there will be a survey to be filled out so that users can put in information about the transaction that may lead to mutations or new genes that need to be stored and remembered as fraudulent. This allows for the program to continuously learn without the input of the creator and get extra clues on what signals that there is a high probability that a transaction is fraudulent.

The data flow of the program to perform the initialization stage is in Figure 8 below. The program will read in a line of data pertaining to a transaction, organize into an array and store it. Then, identify if it was reported fraudulent or not. If it is reported as fraudulent, it will be written to file that store all the fraudulent transaction genes.

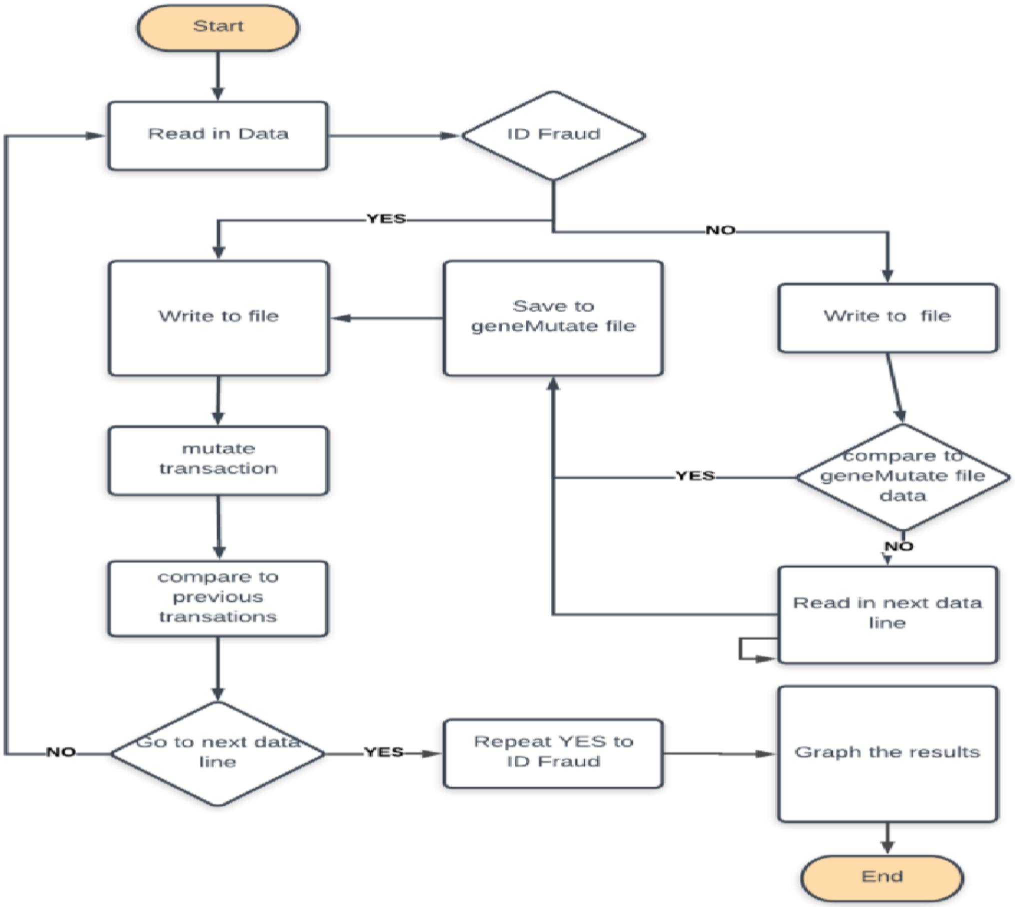


Figure 8. Initialization

After the initialization stage is performed by the program, the program can begin to check the fitness of the transaction genes and produce populations of transaction genes. Each population will go through a fitness check and then the fit individuals of the population will reproduce for five hundred more generations. Figure 9 shows a detailed flowchart how the program goes from initialization, to fitness check, to offspring to produce 500 generations of transaction genes. It shows when transaction genes go through fitness check, what data goes into a crossover or mutation, what is produced by a crossover or mutation and how the program finishes producing generations of populations. The first population of transaction genes to go through the fitness test will all receive a fitness of 1.0 or 100% and each individual in the population will be used to produce offspring. All of the offspring produced by the first population through crossover will go through the fitness check, receive a score and the individuals rated at least .75 or 75% fit will produce the next offspring and those offspring will go through the same process. Precision and accuracy of the fitness evaluation is critical to the success of this algorithm. If the fitness evaluation is strict, it will kill of individuals that may have been fit solutions (transactions that may be fraud). If the fitness evaluation is too yielding, the algorithm will produce false positives (transaction genes that are authorized). A strict fitness evaluation favors the customer because it will allow customers to make transactions by eliminating false positives. A high yielding fitness evaluation favors the credit card issuer because it will surely predict most fraudulent transaction but identify authorized transactions as fraudulent which gives customers the extra hassle of calling their card issuer to authorize there transaction.

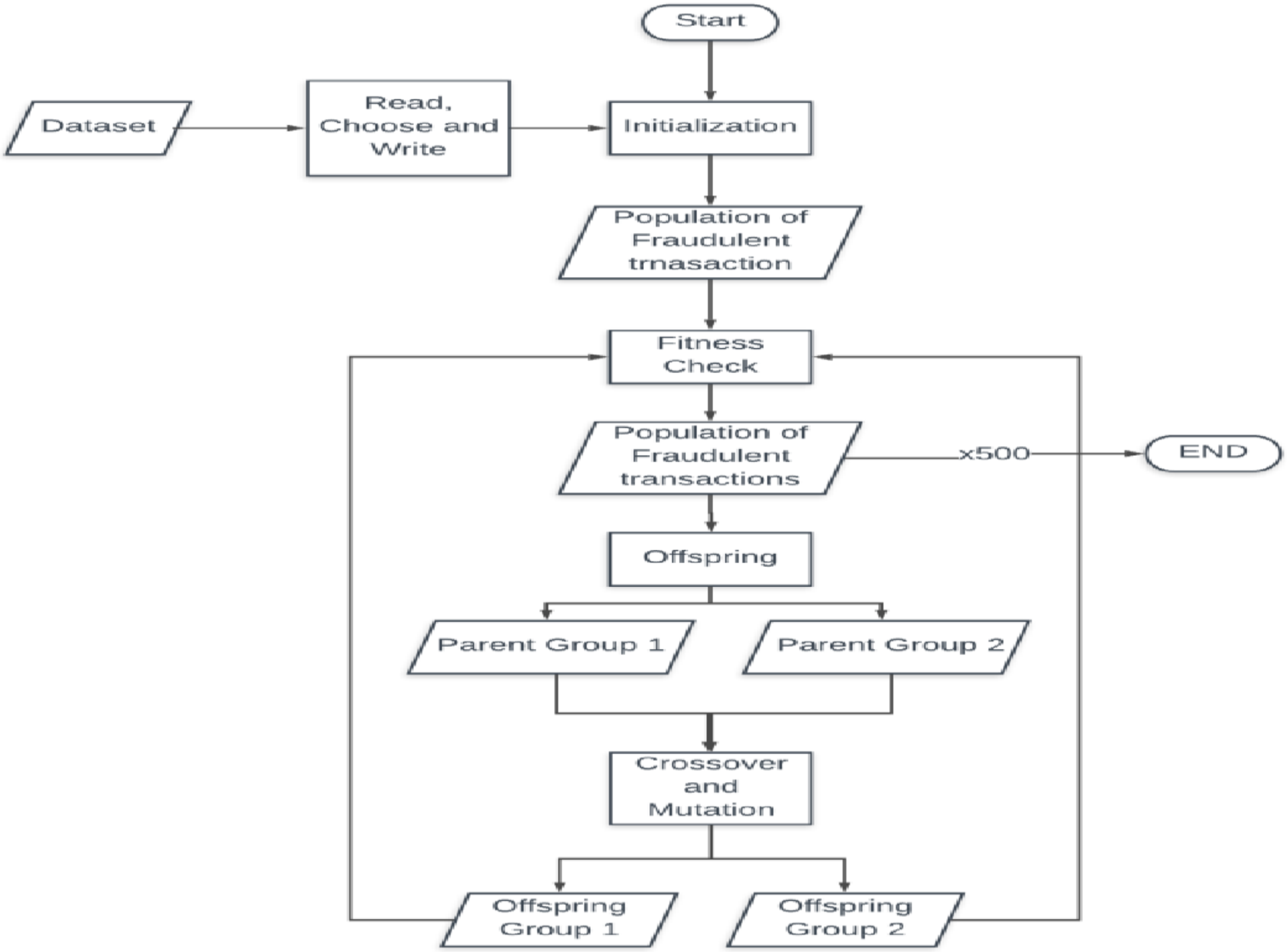


Figure 9. Complete Program Flowchart

5. Expected Results

The future direction of this research is to give financial institutions the option of incorporating a java or python program into their system that classifies fraudulent transactions and predicts the events leading up to other transactions being reported as fraudulent. It is expected that the program will read through the transactions until it gets to a transaction that is fraudulent. When the program gets to a fraudulent transaction, it will create a "transaction gene" with all the features of that transaction in either an array or lists, then it will create random variations of that "transaction genes", finally it will write it to a separate file and keep reading through the transactions, repeating these steps when it reaches another fraud transaction.

In the end, it is expected that a list of "transaction genes" and their genetic variations will be printed in two lists. My hypothesis is that "transaction genes" that were genetically modified will be the same as or very similar to "transaction genes" that are read in later in the program. If the variations of the "transaction genes" read in early in the program are the same as the "transaction genes" that are read in, the goal of my research would be successfully completed.

6. Results

This research has produced an algorithm for a program that can read in data froma dataset, organize it into a usable form, models the transaction information after a mammalian gene in an array data structure, checks the fitness of solutions, produces 500 generations of offspring through crossover breeding, use the offspring to predict a combination of elements that result in fraud and notify banks of how transactions should be blocked. The algorithm is optimal for taking in data from a dataset, identifying instances of fraud, remembering and learning from them. The five hundred generations of offspring produced by this algorithm are array data structures modeled after mammalian genes that store information about a transaction.

7. Analysis

Thorough analysis of the algorithm and results leads to some obvious strengths and weaknesses. The first weakness is an issue with delimiters. If this algorithm was translated to code, one delimiter in the wrong spot can ruin the accuracy of the whole program because that mistake will affect every part of the program. For example, a delimiter may be missing in line 114 for the data and store information about an account in the section for card numbers in the array for a transaction gene. Then, that transaction gene will be bred with other solutions and there will be a bunch of individuals in the population that will have account information where the card number is supposed to be. The second weakness is, after thorough analysis of the dataset, a three-dimensional array might be more useful for the creation of transaction genes because one category of information for one index in the one-dimensional array might have multiple pieces of information.

Isolating each piece of information in the categories would aid in creating more possible solutions and guide the program away from converging on a few possible solutions which is a problem in traditional GP. The use of the one dimensional arrays for the transaction genes may lead the program to finding a few solutions and making slight variations to those few solutions. This will hinder the diversity of the solutions and limit the predictive capabilities of the program. However, a three dimensional array for the transaction means the crossover breeding method will be much more effective at creating different and diverse solutions (Roy, L., Jusak,J., & Cliff, C. Z.,2017).

The first strength of the algorithm is that it simulates passing down generational knowledge in a short period of time. Running this algorithm once will produce five hundred generations of solutions. This many generations of solutions would take much longer if the information of fraudulent transactions were just saved and remembered. The second strength is this algorithm speeds up the process of finding out and predicting solutions and the combination of elements that may lead to a fraudulent transaction.

8. Conclusion

In conclusion, two hypotheses are true and one hypothesis is untested. Transactions can be modeled after mammalian genes to be used in genetic algorithms and incorporating a genetic algorithm into a fraud detection system will allow for the system to learn. There are a few factors of this research that can be done differently to produce similar results. The most notable factor is the data structure used to store the information of transactions. This research chose a one-dimensional array but another person may have found that a linked lists, stack, queue, tree, or hash table would have been more suitable. This research chose a one-dimensional array simple because the conceptual model of a one-dimensional array looks the most like the model of the mammalian gene shown in Figure 2 and 3. In addition, another breeding method may be chosen for creating offspring for populations.

This research chose the crossover breeding method. However, another researcher might find that mutation, tournament selection or elitist selection would be more suitable. The fact that there are replaceable parts to the algorithm developed by this research should be very favorable to the credit card issuers so they have a say in how predictive solutions are developed. After all, they are the ones who get the fraudulent claims and the complaints about authorized transactions being rejected.

9. Future Work

This research is ongoing and has the potential to get much more detailed and fine tuned to the desires and goals of credit card issuers. The next step in this research would be to convert the algorithm to code. This process shall be relatively easy to a good coder. Figure 8 and 9 are very good guides to develop the code for this algorithm. The programmer will need to implements file readers and writers for initialization. In addition, they will need to implement a good equation or inequality for the fitness check based on how strict they want to define an individual as a "fit" solution for breeding. Then, the programmer will need to implement a method for crossover breeding to produce more generations of offspring. Finally, a loop for repeating a fitness check on each individual and repeating the breeding method until five hundred generations has been developed.

There are also additions that can be made to the algorithm to make it more suitable for the use of the banks. The fitness check can give ranks or ratings for individuals instead of just either authorized or fraudulent. For example, instead of the fitness check just selecting fraudulent transaction for breeding. The fitness check can give an individual one of four ranks: "authorized", "alert card holder", "provide more information" or "reject transaction". Individuals who receive an "authorized" rank will be processed without question. In this way, transactions that might seem fraudulent can be in the "alert card holder" rank, these individuals occur in a real transaction they will elicit a call or text from the bank before the transaction goes through. Individuals who receive a "provide more information" rank will elicit a response from the merchant to check for identification and keep the card if identification cannot be provided or allow the transaction to be processed upon proof of identification. Finally, individuals who receive a "reject transaction rank" will elicit a rejection of the transaction at the time of the transaction. Three out of four of these ranks aim to prevent losses at the time of the transaction through predictive qualities of the algorithm produced by this research (Dara, J., &Gundemoni, L.,2006).

Future work can also include autonomy in receiving and using new information. In this research, the dataset included one day of transactions. There are 492 frauds out of over 280,000 transactions. This one day of transactions will produce 500 generations of transactions genes. Each day of transactions can be a new dataset and the genetic algorithms can be used on each on dataset creating new solutions to detecting fraud every day.

References

Bentley, P. J., Kim, J., Jung, G. H., & Choi, J. U. (2000, October). Fuzzy darwinian detection of credit card fraud. In the 14th Annual Fall Symposium of the Korean Information Processing Society (Vol. 14).

Chiu, C. Tsai (2004). A Web Services-Based Collaborative Scheme for Credit Card Fraud Detection,‖*Proceedings of the IEEE International Conference on eTechnology, e-Commerce and e-Service*, 177-181, 2004.

Dara, J., &Gundemoni, L. (2006). Credit Card Security and E-Payment: Enquiry into credit card fraud in E-Payment. Dasarathy, B. V. (2009). *A Special Issue on information fusion in computer security. Information Fusion*, 10(4), 271. Datasets:http://weka.8497.n7.nabble.com/file/n23121/credit_fruad.arff; ______http://sorry.vse.cz/~berka/challenge/pkdd1999/data_berka.ziphttps://web.archive.org/web/20161019192412/http://lisp.vse.cz/pkdd99/berka.htm

Garner, D. (2011). Genetic algorithms for credit card fraud detection. *IEEE Transactions*.&

Kou, Y., Lu, C. T., Sirwongwattana, S., & Huang, Y. P. (2004). Survey of fraud detection techniques. In *IEEE International Conference on Networking, Sensing and Control*, 2: 749-754.

Onan, A., &Korukoğlu, S. (2017). A feature selection model based on genetic rank aggregation for text sentiment classification. *Journal of Information Science*, 43(1), 25-38.

Oreski, S., Oreski, D., &Oreski, G. (2012). Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. *Expert systems with applications*, 39(16), 12605-12617.

Özçelik, M. H., Duman, E., Işik, M., &Çevik, T. (2010, June). Improving a credit card fraud detection system using genetic algorithm. In *2010 International Conference on Networking and Information Technology (pp. 436-440). IEEE.*

Raj, S.B.E., &Portia, A.A. (March 2011) Analysis on Credit Card Fraud Detection Methods‖, *IEEE International Conference on Computer, Communication and Electrical Technology.*

RamaKalyani, K. (2012). Fraud Detection of Credit Card Payment System by Genetic Algorithm. *International Journal of Scientific & Engineering Research*, 3(7), 1-6. Retrieved April 20, 2019.

Roy, L., Jusak,J., & Cliff, C. Z. (2017). Invariant Diversity as a Proactive Fraud Detection Mechanism for Online Merchants.*IEEE Global Communications Conference*, 1–6.

Seeja, K. R., &Zareapoor, M. (2014). FraudMiner: A novel credit card fraud detection model based on frequent itemset mining. *The Scientific World Journal*.

Shaughnessy, H. (March 2011). Solving the $190 billion Annual Fraud Problem: More on Jumio. Forbes.

Viaene, S., Derrig, R. A., Baesens, B., &Dedene, G. (2002). A comparison of state‐of‐the‐art classification techniques for expert automobile insurance claim fraud detection. Journal of Risk and Insurance, 69(3), 373-421.