

Genetic Algorithm with Logistic Regression for Detection of Credit Card Fraud

Akash Prasad

Research Student

Department of Information Technology

B. K. Birla College of Arts, Commerce, & Science (Autonomous), Kalyan, India

Abstract— Credit card frauds are increasing day by day due which many big companies and certain organisation suffers a huge amount loss to due such frauds. Due to such increasing frauds certain researchers has started different Machine learning algorithms to analyse various transaction of credit card details to detect the fraudulent transactions. In this Research paper we have develop a novel technique in which we analyse the certain credit card transactions details of the merchants and extract certain features. In this model we have use the logistic regression to analyse the datasets and using genetic algorithm we have train our model and we got best and optimized results.

Keywords: Optimized, Genetic Algorithm, Logistic Regression, fraudulent Transactions

I. INTRODUCTION

In this modern era of digitalization had reaches at top heights in which every data is processed in form of bits so it can be processed by computer or any other digital device. Due to digitalization most of the data are not secure, due to which many hackers misuse them for their own profit. As due to number of increase in hackers or criminals many of the private and public sectors have to face the financial loss due to such credit transactions frauds. Credit card is generally refers to a card which is assign to the customer for purchasing the goods or items with in than credit limit. So the Credit card fraud is a type of a fraud in which obtaining goods or services by the criminals or access the account details tries to make the payment from that account. Various Machine Learning algorithm are proposed by researchers in order to detect the fraudulent transactions. The Machine learning algorithms detects pattern in financial operations in which they identify the fraudulent and legitimate transactions. Many Algorithms of Machine learning are evolved such as Fuzzy logic, Support vector Machines, random forest, are used to detect the credit card fraudulent transactions The main aim of this paper is develop a credit card fraud detection using genetic algorithm with logistic Regression.

Genetic Algorithm is search based optimization technique based on the principles of Natural selection it used to find the optimal results for certain difficult problems. The advantage of using genetic Algorithm against certain statistical approaches models such as logistic regression can find most best or optimal solution and also increase its accuracy. This makes the Genetic Algorithm applicable in combinatorial and many mixed integer problems. Thus the approach makes the Genetic Algorithm more accurate that required efficient searching of subset of features of high dimensional classification problems.

II. LITERATURE REVIEW

A detailed study is done on the machine learning algorithms to detect the credit card fraud transactions from reviewing certain Research papers In[1]Dornadula V.,&S.G The researcher propose that they have use the Smote technique on certain machine learning algorithms as local outlier factor, Isolated forest, logistic regression, decision tree and random forest to detect the fraud transactions and they got better results. In[2] Zareapoor M.&Shamsolmoali The researcher proposed that they have used bagging ensemble algorithm on real life credit card details and they found that bagging ensemble against decision tree works well and it able to detect the fraudulent transactions. In[3] Rtayli N. & Enneya N The researcher have proposed that they have use the credit card Identification method based on the features selected from certain algorithms to detect the risk of fraud in credit card.[4]In 2012 Ramakalyani K & Umdevi D The researcher proposed that genetic algorithm is used to able to find fraud transactions as genetic algorithm is used to provide optimal results .In[5] S.P Maniraj ,Aditya Sani Shadab The research paper proposed that they have used certain machine learning algorithms like logistic Regression random forest and KNN Algorithm and they found random Forest classifier was able to detect the fraudulent transactions as compare to other algorithms. In[6]I Altyer Taha & Malebary S.J The researcher have proposed a Bayesian base Hyperparameter optimization is integrate all the parameters of a light gradient boasting machine for detection of fraud in credit cards and they have achieve highest accuracy. In[7] Panigrahi S kundu & Majumdar A.K In this research paper the author proposed a model that combines the evidences from current and past behavior. It contains four components as rule-based filter, Dempster -Shafer adder, transaction history and Bayesian learner and combination of these components provide them good results to discriminate between fraud and legitimate transactions In[8] Mahmoudi N,&Duman E In this research paper the author proposed a modified fisher discrimination is implemented which makes tradition function more sensitive to important instances to detect the fraud transactions. In[9] Amusam D.G Olabode A.O In this research paper the author has designed a hybrid counter propagation neural network and genetic algorithm to increase the falsely alarm time to detect the fraudulent transactions. In[10] Asmaa Alotaibi, Masheal Alqhtani In this research paper the researcher proposed the 2 methologies Clonal Selection Algorithm and the genetic Algorithm to overcome the misclassification of clonal algorithm in credit card frauds.

III. METHODOLOGY

A. Genetic Algorithm:

Genetic algorithm are based on the concepts of natural selection and genetics. Genetic algorithm adapt the process of the natural selection, in which certain species can adapt their changes in that environment and reproduce the next species and go to next generation, it means that they simulate the survival of the fittest among the other individual of that particular generation for solving the particular problem, due to which Genetic Algorithm has the ability to deliver the optimized or best results. This makes the algorithm more attractive and it can be used in solving difficult problems.

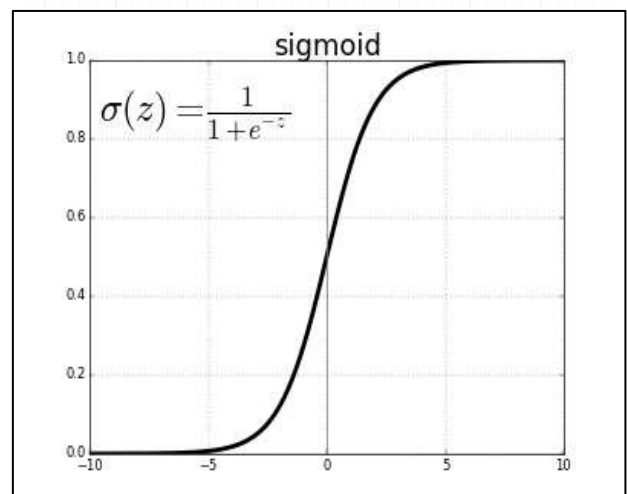
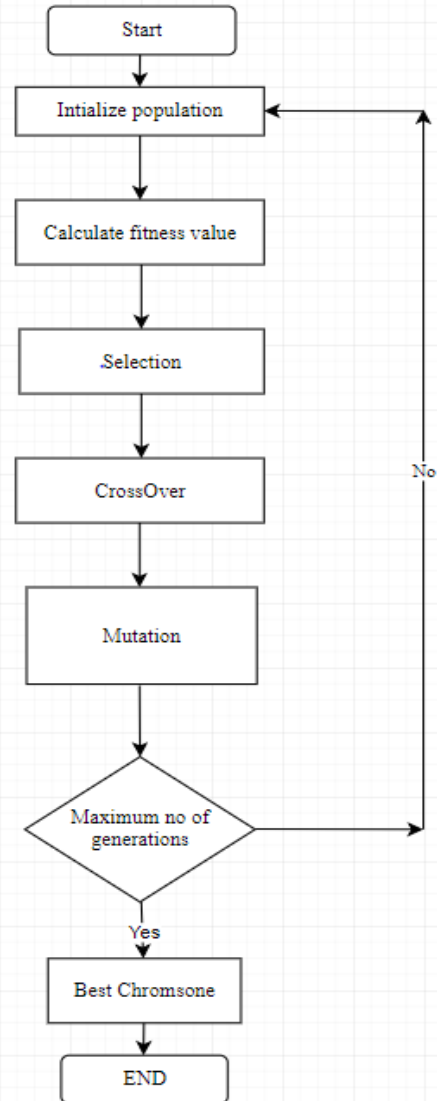
B. Basic Terminology of Genetic Algorithm:

- 1) Population: -It is a subset of all the possible solutions for that given problem.
- 2) Chromosome: -A Chromosome is one of the solution of the given problem.
- 3) Gene: -A gene is an element position of that chromosome.
- 4) Allele: -It is the value of a gene takes for particular chromosome.
- 5) Fitness Function:-The Fitness function is a function in which takes the solution of the input and produced the desired output. Some cases the objective is same while in others it may be different.

C. Basic Structure of Genetic Algorithm:

- 1) Initial Population: Initial the population randomly based on the scores for the simulation and creates the first generation of the chromosome.
- 2) Fitness Function-The Fitness Function is calculated based on the particular objective function that give a summary of entire population, as a single merit in which how a given solution is achieving the set aims.
- 3) Selection: -Selection is the stage of a genetic algorithm in which individuals are chosen from the initial population for their reproduction. The Selection of the individuals is done based on fitness Values. If the selection is repeated until there are enough selected individuals then such selection is known as roulette wheel selection. If the selection of best individuals repeatedly on the bases of random values is known as tournament selection.
- 4) Crossover: -The Crossover is a genetic operator which combines the genetic information of two parents to generate the new offspring. Crossover is a way to generate new solutions from the existing population.
- 5) Mutation:-Mutation is a genetic operator used to maintain the genetic diversity from a particular generation. It is used to alter two or more gene values in a chromosome from its initial stage, and it change the solution entirely from previous solution.
- 6) Termination: -The algorithm is terminate either when the maximum number of generations is produced or a satisfactory fitness level has acquired by the new generation.

D. Flow Diagram of Genetic algorithm



- 1) Sigmoid Curve: -A sigmoid function have a property that they can map the entire number into a small range such as between 0 to 1 so the use of sigmoid curve is to convert the real into one and zero that can be interpreted as probability.

Function is

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

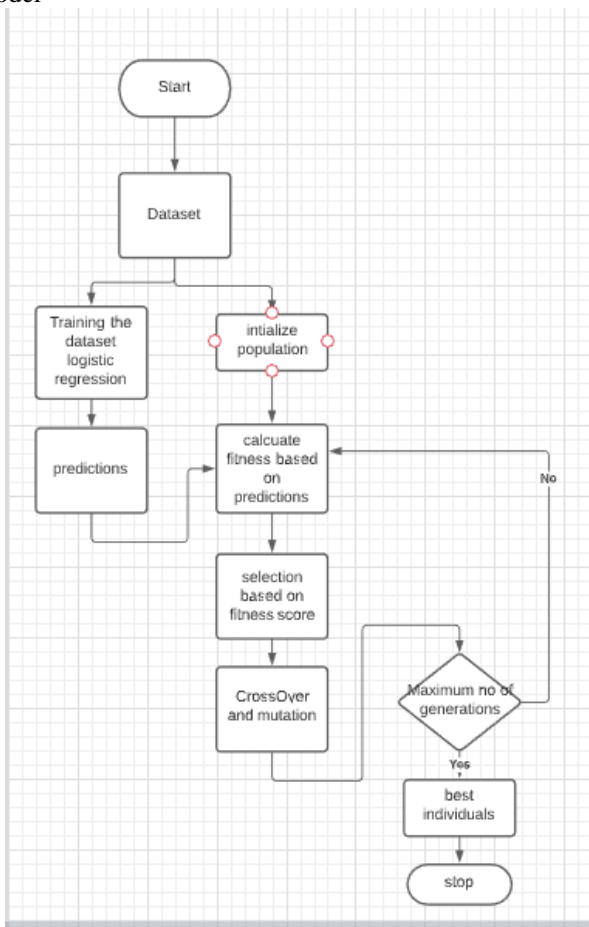
P is the probability of 1 (proportion of 1s the mean of Y), e is the natural logarithm a and b are the parameters.

The value of a yields P when X is zero and b adjust itself then it shows that how the probability changes quickly with changing the X values.

IV. EXPERIMENTS

A. System Design:

The proposed system is used to detect the fraudulent transactions that occurred in merchants with help of genetic algorithm from predictions done by the logistic regression model



B. Dataset:

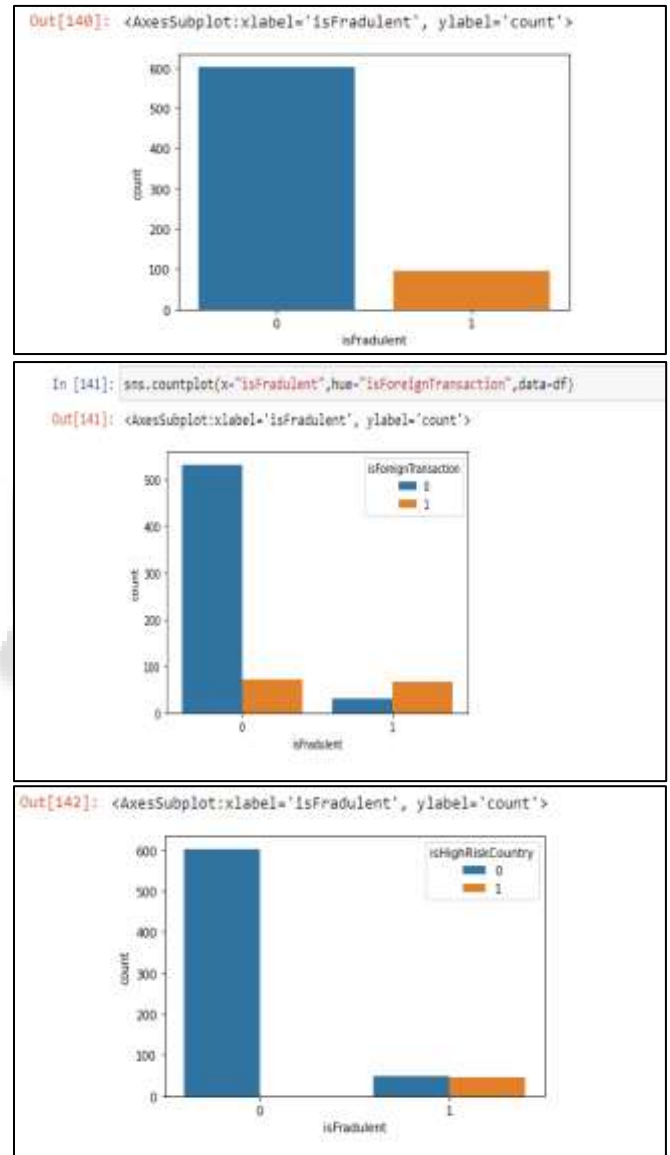
The Dataset comprises of transactions of credit cards made by the merchants collected from Kaggle dataset., contains 150 transactions out which 23 transaction made by merchants are fraudulent transactions and 150 transactions are legitimate transactions. The features are

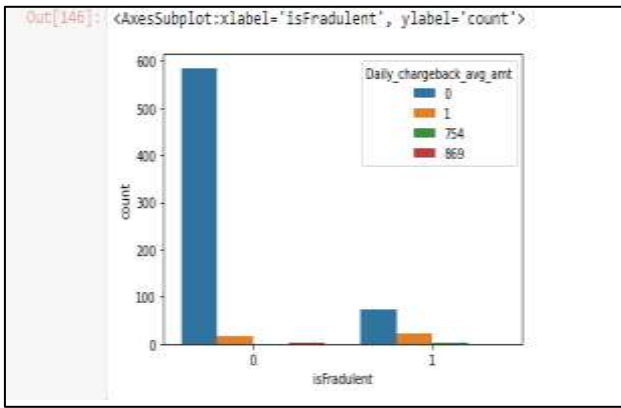
Features no	description
1	Average Amount/transaction/day
2	Transaction amount
3	Is Foreign Transaction-Any foreign transactions

	have taken place
4	Is High Risk Country-Is country is risk in which credit card fraud takes more
5	Daily_chargeback_avg_amt-Any extra charges on the average amount transaction
6	is Fraudulent-Is transaction is made by merchants is fraudulent or not

C. Data Analysis:

We have plot certain bar charts to know the relationship between the features so to find the dependent and independent variables.





D. Algorithm:

First, we use clustering methods on the datasets and plot some bar plots to identify the relationship between the independent variables and the target variables

We have use Jupyter notebook application for training and testing the both algorithms

Jupyter Notebook: The Jupyter notebook is an open source web application development environment that allows to create and share the documents that contain various equation visualization machine learning algorithms.

```
X_train=["Average
Amount/transaction/day","Transaction_amount", "isForeign
Transaction", "isHighRiskCountry", "Daily_chargeback_avg_
amt"]
```

```
Y_train=["isFraudulent"]
```

Then we perform logistic regression on dataset and train the model and predict the values.

Step 1: Import the logistic regression module in sklearn
 from sklearn.linear_model import LogisticRegression

Step 2: Declare an object of the logistic regression
 logmodel=LogisticRegression()

step 3: Fitting the logistic regression model
 logmodel.fit(X_train, y_train)

step 4: predicting values
 pred=logmodel.predict(X_test)

Lastly we perform the genetic algorithm based on the predictions done by the logistic regression.

E. Genetic Algorithm

Pseudo code:

Step 1: Initialization of population from the dataset

```
def initialization_of_population(size,n_feat):
```

```
    population = []
```

```
    for i in range(size):
```

```
        chromosome = np.ones
```

```
        chromosome[:int (0.3*n_feat)] =False
```

```
        np.random.shuffle(chromosome)
```

```
        population.append(chromosome)
```

```
    return population (n_feat,dtype=np.bool)
```

```
        chromosome[:int(0.3*n_feat)]=False
```

```
        np.random.shuffle(chromosome)
```

```
        population.append(chromosome)
```

```
    return population
```

step 2: calculate the fitness function based on predictions done by the logistic model

```
def fitness_score(population):
```

```
    scores = []
```

for chromosome in population:

```
    logmodel.fit(X_train.iloc[:,chromosome],y_train)
```

```
    predictions
```

```
logmodel.predict(X_test.iloc[:,chromosome])
```

```
    scores.append(accuracy_score(y_test,predictions))
```

```
scores, population = np.array(scores), np.array(population)
```

```
inds = np.argsort(scores)
```

```
return list(scores[inds][::-1]), list(population[inds,:][::-1])
```

Step 3: Selection of parents based on fitness scores

```
def selection(pop_after_fit,n_parents):
```

```
    population_nextgen = []
```

```
    for i in range(n_parents):
```

```
        population_nextgen.append(pop_after_fit[i])
```

```
    return population_nextgen
```

Step 4: performing crossover and mutation for generation of new chromosome

```
def crossover(pop_after_sel):
```

```
    population_nextgen=pop_after_sel
```

```
    for i in range(len(pop_after_sel)):
```

```
        child=pop_after_sel[i]
```

```
child[3:7]=pop_after_sel[(i+1)%len(pop_after_sel)][3:7]
```

```
    population_nextgen.append(child)
```

```
def mutation(pop_after_cross,mutation_rate):
```

```
    population_nextgen = []
```

```
    for i in range(0,len(pop_after_cross)):
```

```
        chromosome = pop_after_cross[i]
```

```
        for j in range(len(chromosome)):
```

```
            if random.random() < mutation_rate:
```

```
                chromosome[j]= not chromosome[j]
```

```
        population_nextgen.append(chromosome)
```

```
    return population_nextgen return population_nextgen
```

step 5: create the function to calculate no of generations

```
def
```

```
generations(size,n_feat,n_parents,mutation_rate,n_gen,X_train,
```

```
            X_test, y_train, y_test):
```

```
    best_chromo= []
```

```
    best_score= []
```

```
population_nextgen=initilization_of_population(size,n_feat)
```

```
for i in range(n_gen):
```

```
    scores, pop_after_fit
```

```
fitness_score(population_nextgen)
```

```
    print(scores[:2])
```

```
    pop_after_sel = selection(pop_after_fit,n_parents)
```

```
    pop_after_cross = crossover(pop_after_sel)
```

```
    population_nextgen
```

```
mutation(pop_after_cross,mutation_rate)
```

```
    best_chromo.append(pop_after_fit[0])
```

```
    best_score.append(scores[0])
```

```
return best_chromo,best_score
```

Step 6: passing values to the function

```
chromo,score=generations(size=20,n_feat=5,n_parents=10,
mutation_rate=0.10,
```

```
n_gen=10,X_train=X_train,X_test=X_test,y_train=y_train,y
_test=y_test)
```

```
logmodel.fit(X_train.iloc[:,chromo[-1]],y_train)
```

```
predictions = logmodel.predict(X_test.iloc[:,chromo[-1]])
```

```
print ("Accuracy score after genetic algorithm is="
      "+str(accuracy_score(y_test,predictions)))
```

F. Formula:

In our proposed system we have to evaluate accuracy, recall and precision, but these parameters are considered as base parameters for evaluate any model.

The Mathew Correlation coefficient (MCC) is a machine learning is used to check the balance of the (two-class) classifiers, It takes true and false values that is why it is regarded as balance measure.

$$Accuracy = \frac{T.P + T.N}{T.P + F.P + F.N + T.N}$$

$$precision = \frac{T.P}{T.P + F.N}$$

$$Recall = \frac{T.P}{T.P + F.P}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (FN + TN) \times (TP + FN) \times (FP + TN)}}$$

V. RESULTS

We have experimented 2 models as logistic regression and Genetic Algorithm. The results are tabulated format which shows a slightly differences in Accuracy, precision, and recall.

Confusion matrix:

```
array([[416, 1],
       [ 19, 54]], dtype=int64)
```

Logistic Regression Classification Report

	precision	recall	f1-score	support
0	0.96	1.00	0.98	417
1	0.98	0.74	0.84	73
accuracy			0.96	490
macro avg	0.97	0.87	0.91	490
weighted avg	0.96	0.96	0.96	490

Accuracy of logistic regression:- 0.9591836734693877
MCC(Mathew correlation coefficient):0.831695

Genetic Algorithm classification report

Confusion matrix:

```
array([[417, 0],
       [ 19, 54]], dtype=int64)
```

	precision	recall	f1-score	support
0	0.96	1.00	0.98	417
1	1.00	0.74	0.85	73
accuracy			0.96	490
macro avg	0.98	0.87	0.91	490
weighted avg	0.96	0.96	0.96	490

Accuracy of genetic algorithm: - 0.9612244897959183
MCC (Mathew correlation coefficient):-0.841124402

VI. CONCLUSION

Credit card fraud transaction is major problem in the financial sectors in which the merchants and customers suffers a great loss In this paper we develop a novel technique for fraud detection in credit cards in which train the genetic algorithm on the predict values that generate from logistic regression and after training the genetic Algorithm with such statistical modes provided best and optimized results and also increases its accuracy. Hence genetic algorithm with statistical approaches is used to provided best and optimized results for such complex problems.

ACKNOWLEDGMENT

A special gratitude is conveyed to our prof Swapna Augustine Nikale Department of Information Technology of B.K Birla College of Arts, Science and Commerce (Autonomous) Kalyan Thane Maharashtra India

REFERENCES

- [1] Dornadula, V., & S.G. (2019). Credit card fraud detection using machine learning algorithms. *Procedia Computer Science*, 165, 631– 641. <https://doi.org/10.1016/j.procs.2020.01.057>
- [2] Zareapoor, M., & Shamsolmoali, P. (2015). Application of credit card *Procedia Computer Science*, 48, 679 – 685. <https://doi.org/10.1016/j.procs.2015.04.201>
- [3] Rtayli, N., & Enneya, N. (2020). Selection Features and Support Vector Machine for Credit Card Risk Identification. *Procedia Manufacturing*, 46, 941– 948. <https://doi.org/10.1016/j.promfg.2020.05.012>
- [4] Ramakalyani, K., & Umadevi, D. (2012). Credit card fraud detection by genetic algorithm. *International Journal of Scientific and Engineering Research*, 3(7), 1– 5. <http://www.isjer.org>
- [5] S P Maniraj, Aditya Saini, Shadab Ahmed, & Swarna Deep Sarkar. (2019). Credit Card Fraud Detection using Machine Learning and Data Science. *International Journal of Engineering Research And*, 08(09), 110– 115. <https://doi.org/10.17577/ije.rtv8is090031>
- [6] Taha, A. A., & Malebary, S. J. (2020). An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine. *IEEE Access*, 8, 25579– 25587. <https://doi.org/10.1109/access.2020.2971354>
- [7] Panigrahi, S., Kundu, A., Sural, S., & Majumdar, A. K. (2009). Credit card fraud detection: A fusion approach using Dempster– Shafer theory and Bayesian learning. *Information Fusion*, 10(4), 354–363. <https://doi.org/10.1016/j.inffus.2008.04.001>
- [8] Mahmoudi, N., & Duman, E. (2015). Detecting credit card fraud by Modified Fisher Discriminant Analysis. *Expert Systems with Applications*, 42(5), 2510– 2516. <https://doi.org/10.1016/j.eswa.2014.10.037>
- [9] D.G, A., A.O, O., O.S, O., A.O, F., & M.O, O. (2019). Hybrid Design using Counter Propagation Neural Network-Genetic Algorithm Model for the Anomaly Detection in Online Transaction. *International Journal of Advances in Scientific Research and Engineering*, 5(9), 107–114. <https://doi.org/10.31695/ijasre.2019.33512>

- [10] Alotaibi, A., Alqhtani, M., A., & Batarfi, O. (2019). Identifying Credit Card Fraud Using Genetic and Clonal Selection Algorithms. *International Journal of Innovative Science, Engineering & Technology*, 2348(7968), 2348–7968. <https://www.ijset.co>
<https://datascienceplus.com/genetic-algorithm-in-machine-learning-using-python/>
<https://www.quora.com/What-is-a-sigmoid-function-in-neural-networks>
<http://faculty.cas.usf.edu/mbrannick/regression/Logistic.html>
<https://www.nosimpler.me/accuracy-precision/>
<https://pubs.rsc.org/en/content/articlelanding/ay/2015/c5ay00168d#!divAbstract>

