Concordia Institute for Information System Engineering (CIISE)

Concordia University

INSE 6180

Implementing and Analyzing Data Mining Techniques to detect Credit Card Fraud

Submitted to:
Prof. Arash Mohammadi

Submitted by-
Tanmya Rampal (40197691), Ekta Patel (40229205), Alka Singh Rathour (40217141)

Table of Contents-

Implementing and Analyzing Data Mining Techniques to detect Credit Card Fraud

Tanmya Rampal (40197691), Ekta Patel (40229205), Alka Singh Rathour (40217141)
GitHub: https://github.com/Tanmya/Machine-Learning-6180.git

***Abstract*—Due to the quick development of computerization and digitization techniques, a vast amount of data is now accessible in research, business, industry, and many other fields. Such data could be a valuable resource for decision-making and knowledge discovery. We used a dataset of Credit Card fraud to analyze and predict fraud using Decision Trees and Multilayer Perceptron Artificial Neural Networks. Furthermore, we tuned the hyperparameters using GridSearchCV to achieve higher AUC-ROC and came to the conclusion that the Neural Networks model is more accurate than Decision Trees with 78% AUC score. Furthermore, we discussed the possible attacks and detection mechanisms for Neural Networks. We implemented k-anonymity Generalization to prevent these attacks from taking place.**

***Index Terms*—Decision Trees, Neural Networks, GridSearchCV, K-anonymity, Credit Card Fraud**

I. Introduction

A credit card is a small, rectangular piece of plastic or metal that is issued by a bank or other financial institution and enables its holder to borrow money to pay for products and services at businesses that accept credit cards. Credit cards impose the need that cardholders repay the borrowed funds, plus any applicable interest and any other agreed-upon charges, in full or over time, either by the billing date or at a later date. Credit card firms and financial institutions use a variety of rules, tools, procedures, and practices to prevent and identify identity fraud.

This approach is known as credit card fraud detection. Identifying fraudulent purchase attempts and rejecting them instead of executing the order is the process of credit card fraud detection. The majority of merchants use a combination of many of the many tools and strategies that are available for identifying fraud [2].

Payment cards are simple to use since identifying your account and authorizing the transaction simply need sending a small number of digits to the bank.
They are additionally exposed due to their simplicity. On a few straightforward numbers that must be communicated with the persons you are trading with; it is quite difficult to enforce strict data security.

Credit card scams may take many different forms and employ many different techniques, such as impersonating someone else by using their instalment card, and this is only the beginning. Similarly, the causes of the card update's false portrayal. Some want for nothing in exchange for goods, while others are intended to profit from accounts.

With the emergence of the Internet and its widespread use, the market for credit card payments has increased tremendously over time. The majority of firms and sectors have changed their operations to include online services for e-commerce, connection, and accessibility for their clients [7, 8].

Fraud detection has mostly been digital and automated in recent years due to the explosion of data and the rise in payment card transactions. When specific cases of credit card fraud are discovered, the majority of contemporary systems use artificial intelligence (AI) and machine learning (ML) to manage data analysis, predictive modelling, decision-making, fraud alerts, and remedial action.

II. Problem Statement

Nowadays, majority of the people use credit cards to acquire items that they desperately need but can't now afford. Credit cards are used to satisfy demands, and the related fraud is growing, therefore it is necessary to create a model that fits well and forecasts with more accuracy.

On the dataset collected from credit card defaults, we want to use decision trees and artificial neural network algorithms. In addition, we will train and test the model to evaluate and compare the effectiveness of both the algorithms.

Credit card Fraud is classified in the following manner:

1. Fraud using counterfeit cards is a sort of fraud in which the perpetrator duplicates all of the data from the magnetic strip of the actual card, making it seem and function exactly like the original card. Fraudulent usage of this card.
2. Merchant Collision: In this sort of fraud, merchants exchange cardholder information with a third party or the fraudster without the cardholder's consent.
3. Website fake fraud: The perpetrator will add malicious code to the site to carry out their fraudulent activity.
4. Card id theft is a sort of fraud in which the cardholder's identity is taken and fraud is committed.

5. Takeover of the account: In this case, the fraudster will seize total control of the account holder and commit fraud.
6. Fraudulent mail or phishing can result in a credit card being issued but not being received. This technique is known as mail fraud or postal fraud.
7. Card imprinting, either electronically or manually: In this type of fraud, the con artist steals data from the card's magnetic strip, utilizes the credentials, and then executes the scam.
8. This sort of credit card does not require the actual card to be present while making a purchase.

III. Machine Learning terminology

A. *Decision Trees:*

A decision tree is a diagram representing the potential consequences of several connected options. Based on their costs, probabilities, and advantages, it enables a person or organization to compare potential courses of action. They may be utilized to spark casual conversation or to create an algorithm that predicts the optimal option based on arithmetic [17, 18].

Typically, a decision tree begins with a single node and branches out into potential possibilities. Each of those outcomes connects to more nodes, each of which branches out into different possibilities. This shapes it like a tree. Chance nodes, decision nodes, and end nodes are the three main categories of nodes. The likelihood of specific outcomes is displayed by a chance node, which is symbolized by a circle.

B. *Neural Networks:*

Using a technique that resembles how the human brain works, a neural network is a collection of algorithms that aims to identify underlying relationships in a piece of data.
Neural networks are systems of neurons, whether they are natural or artificial, in this sense.
In the field of finance, neural networks support the creation of procedures including time-series forecasting, algorithmic trading, securities categorization, credit risk modelling, and the creation of custom indicators and price derivatives [4].

The functioning of a neural network is comparable to that of the human brain. A mathematical function known as a "neuron" in a neural network gathers and categorizes data in accordance with a certain design. The network closely resembles statistical approaches like regression analysis and curve fitting.

C. *Types of Neural Networks:*

i. Feed-Front Neural Networks:

One of the most straightforward neural network types is the feed-forward network. It transmits information in a single route through input nodes, processing it in this single direction until it reaches the output mode. The most often used feed-forward neural network type for facial recognition systems may contain hidden layers for functioning.

ii. Neural networks with recurrence:

Recurrent neural networks are a more advanced sort of neural network that feed information back into the network from a processing node's output. The network gets better as a consequence of theoretical "learning" Each node keeps a record of previous operations, and subsequent processing iterates over these stored operations.

iii. Constraining Neural Networks:

Data is categorized in many layers of convolutional neural networks, often known as ConvNets or CNNs. These networks consist of an input layer, an output layer, and several hidden convolutional layers between them. The layers provide feature maps that catalogue regions of an image that are further subdivided until they produce useful outputs.
These networks are very useful for image identification applications since these layers may be combined or connected fully.

Convolutional neural networks operate normally, whereas deconvolutional neural networks operate the opposite. The network is used to detect objects that a convolutional neural network would have classified as relevant. During the convolutional neural network execution phase, these objects would have most likely been eliminated. For image processing or analysis, this kind of neural network is also often used.

iv. Modular Neural Network:

A modular neural network is made up of multiple networks that work independently of each another. During an analytical procedure, there is no interaction between these networks.
Instead, these procedures are carried out so that labor-intensive, intricate computer procedures may be carried out more effectively. The objective of network independence, like other modular sectors like modular real estate, is to have each module accountable for a certain aspect of a larger overall image.

D. *Applications of Neural Networks:*

With applications in trade, business analytics, financial operations, corporate planning, and product maintenance, neural networks are widely employed.

Business applications including forecasting and marketing research solutions, fraud detection, and risk assessment have all seen a significant increase in the use of neural networks. A neural network analyses price data and finds chances for trading decisions based on the study of the data. Networks have the ability to recognize patterns and subtle nonlinear interdependencies that other technical analysis techniques cannot. The accuracy of neural networks in predicting stock prices varies, according to studies. While some models are accurate 50% to 60% of the time, others are accurate 70% of the time when predicting stock values. Some have said that the maximum an investment can expect is a 10% increase in efficiency [4].

E. *Advantages:*

Store entire information on the network, the ability to work with insufficient knowledge, good fault tolerance, ability to handle complex and non-linear data, it is a self-learning algorithm.

IV. PROPOSED SYSTEM

A. *Dataset Description*

Our dataset consists of 25 columns and 30000 rows, with the heatmap given below we can see the correlation of all attributes with each other.

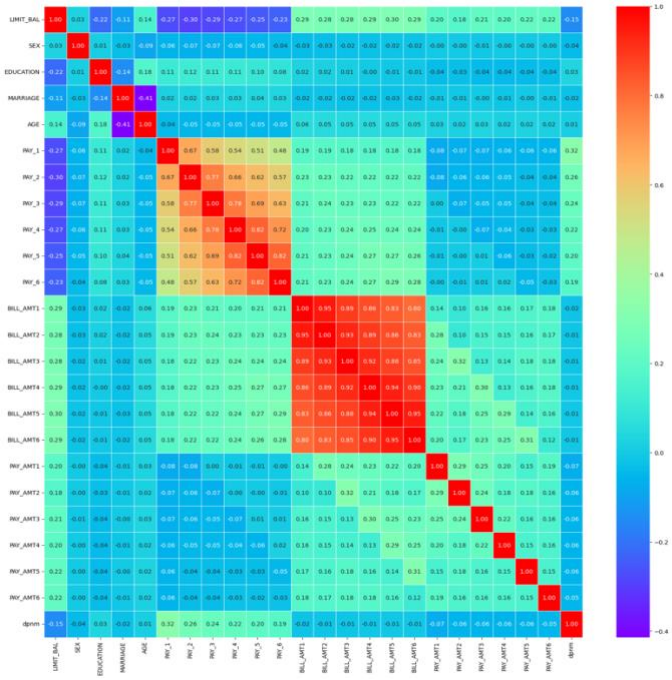

Fig1. Heatmap

The account balance increases as age increases which can be seen in the jointplot below.

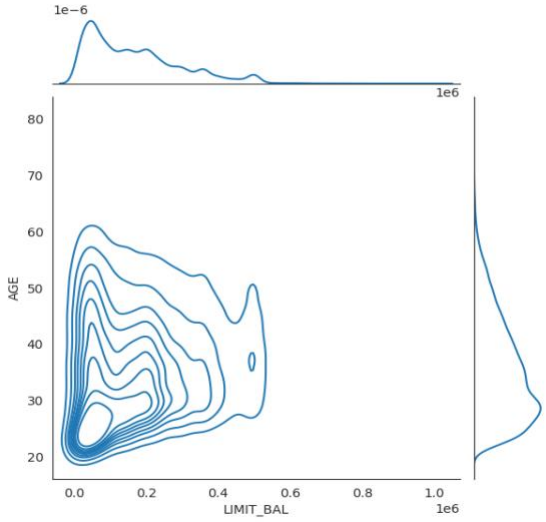

Fig2. Graph between Age and Balance

B. *Data Cleaning and Pre-processing*

Since it is a large dataset, it is bound to have multiple duplicated rows, irrelevant columns and columns having null values. It is essential to drop or delete such rows and columns to have a better training model.

Therefore, we carried out the following method to drop those rows and similarly dropped one column.

```
print(f"There are {dataset.duplicated().sum()} duplicate rows.")
dataset = dataset.drop_duplicates()
print("Removed the duplicated rows.")
```

```
There are 35 duplicate rows.
Removed the duplicated rows.
```

Fig3. Data cleaning

Standardization –

Data transformation is one of the primary measures in the section of data processing. We frequently come across various forms of variables in the similar data set. An important problem is that the range of the variables can vary greatly. Utilizing the first scale can give more weight to the wide-range variables. To solve this issue, we need to implement the feature rescaling method to independent variables or features of data in the data pre-processing step [1].

The outcome of standardization (or z-score normalization) is that features are rescaled to guarantee that the mean is 0 and standard deviation is 1, and features with the distribution value between 0 and 1 are useful for optimization algorithms used in machine learning that weight inputs. For example, neural networks and logistic regression. Given below is the equation of Standardization.

$$X_{stand} = \frac{x - mean(x)}{standard\ deviation\ (x)} \quad (1)$$

After successfully completing data cleaning and preprocessing, we have the following features divided into independent variables and dependent variables.

Independent variables -

1. Limit_bal: The given credit
2. Sex: Person's Gender (here, 1 is male & 2 is female).
3. Education: Level of Education achieved by that person (here, 1 is graduate school, 2 is university, 3 is high school, 4 is others).
4. Marriage: Marital Status of the person (here, 1 is married, 2 is single, 3 is others).
5. Age: Age of the person
6. Pay_1 to pay_6 : Status of repayment
7. Bill_amt1 to bill_amt6: The bill Statement of that person in months
8. Pay_amt1 to pay_amt6: Payment done in the previous month

Dependent variable-

1. dpnm: default payment next month.(here, yes is 1 and no is 0)

C. *Implementing Decision Trees*

We chose to implement decisions trees for this model as it is a supervised machine learning algorithm and is great to work for classification. Decision trees are highly helpful for data analysis and machine learning as they break down complex data into more docile piece [17].

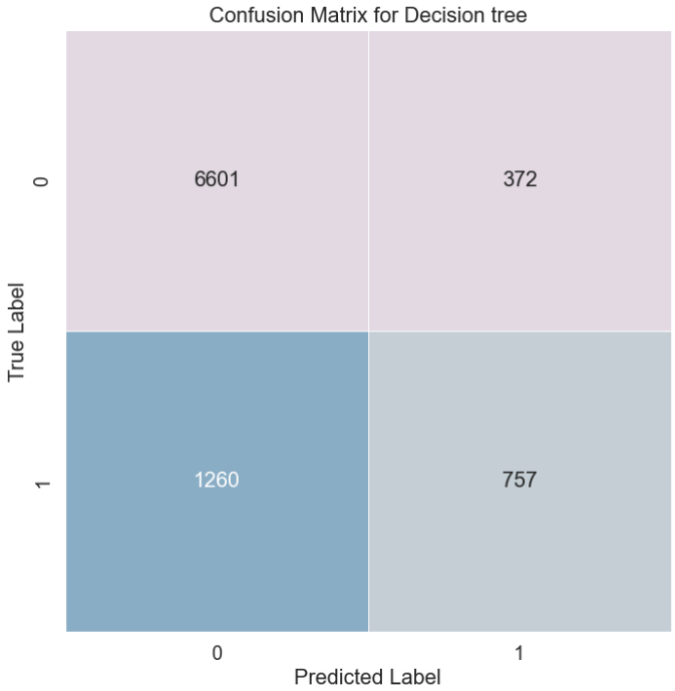

Fig4. Confusion matrix for Decision Tree

Accuracy: It is the ratio of total number of accurately classified attributes to the total number of attributes [15].

$$Accuracy = \frac{TP + TN}{P + N} \quad (2)$$

Confusion Matrix: The Confusion matrix depicts the following values achieved by the model; True positive, true negatives, false positives, and false negatives [16].

$$Confusion\ Matrix = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

Precision: It is calculated by the following formula. This helps in understanding how beneficial the results we achieved are.

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

Recall score: It is calculated by the following formula. This helps in understanding how comprehensive the results we achieved are.

$$Accuracy = \frac{TP}{TP + FN} \quad (5)$$

F1 Score: With precision and recall, we can calculate our F1 score.

$$F1\ score = \frac{2 * Recall * Precision}{Recall + Precision} \quad (6)$$

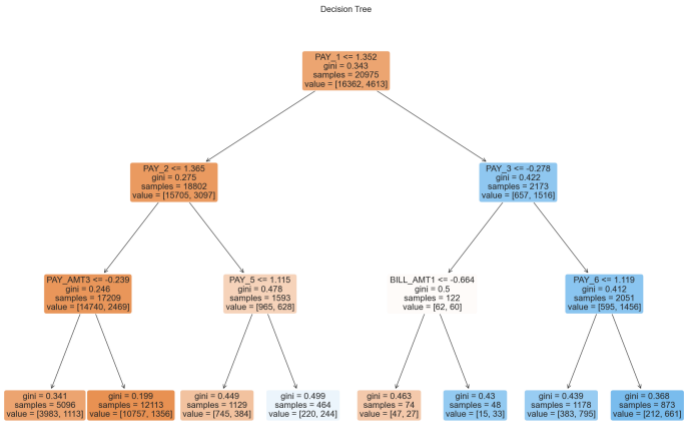

Fig5. Decision Tree

With the help of the confusion matrix, we got our results. The accuracy of the model achieved was 81.8%, whereas the precision score was 67.1%, the recall score was 37.5% and the F1 score was 48.1%.

D. *Implementing Neural Networks*

We implemented the multilayer perceptron. A fully connected multilayer neural network is named a Multilayer Perceptron (MLP). It has 3 layers containing a hidden layer. If it has more than 1 hidden layer, it is called deep Artificial neural Network. Multilayer Perceptron isa feedforward artificial neural network. Number of layers and neurons give the hyperparameters of an ANN and have to be tuned. To get the appropriate values for these, cross-validation techniques should be used. Backpropagation is used to adjust weights. Deeper neural networks can process data better [8].

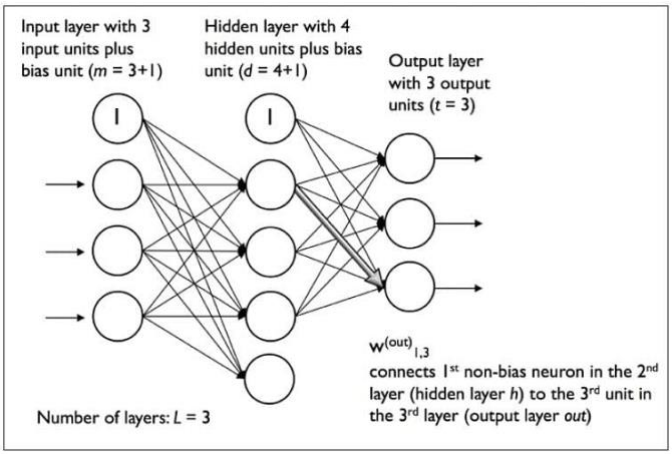

Fig6. Multilayer Perceptron

With the given confusion matrix, we were able to achieve these values for our MLP neural network; The accuracy of the model achieved was 81.4%, whereas the cross-validation score was 81.8%, precision score was 64.5%, the recall score was 38.1% and the F1 score was 47.9%.

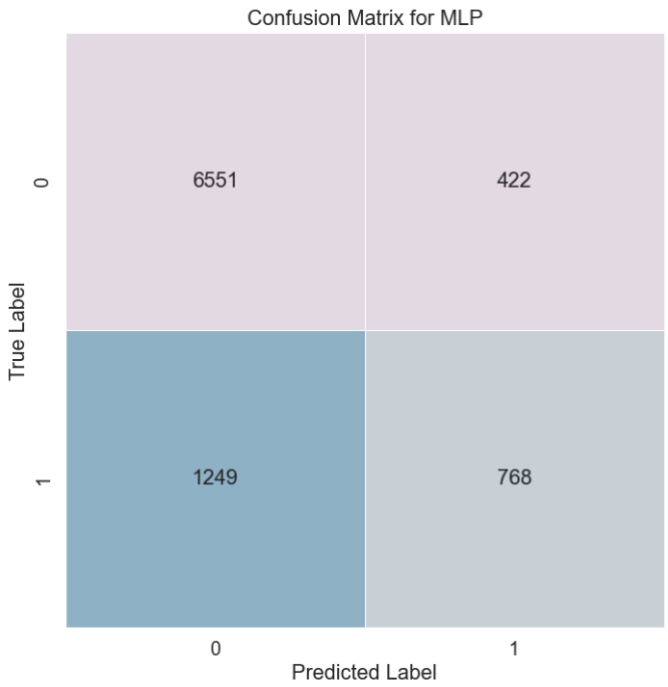

Fig7. Confusion matrix for Neural Network

There was not a significant difference between the two models. Thus, we decided to implement GridSearchCV in order to tune the hyperparameters and achieve higher accuracy and precision.

GridSearchCV tries all blends of the values passed into the dictionary and assesses the model for all combination using the cross-validation procedure. Therefore, after utilizing this function, we get an accuracy/loss for each combination of hyperparameters and can pick the one that provides the suitable results.

E. Setting Hyperparameters with GridSearchCV

GridSearchCV is the procedure of performing hyperparameter tuning to decide the appropriate values for a provided model. As stated earlier, the performance of a model varies with the value of the hyperparameters. As we know, there is no way to understand in advance the best values for hyperparameters, so ideally, we should try all possible values to know the optimal ones. This manual operation can take a significant amount of time and resources, which is why we use GridSearchCV to program hyperparameter tuning [19].

GridSearchCV is a function provided in the model_selection package of Scikit-learn (or SK-learn). So, an essential point to note is that we must have the Scikit learning library installed on the computer. This feature helps to browse the previously defined hyperparameters and fit the estimator (model) to the training set. So finally, we can select the best parameters from the listed hyperparameters.

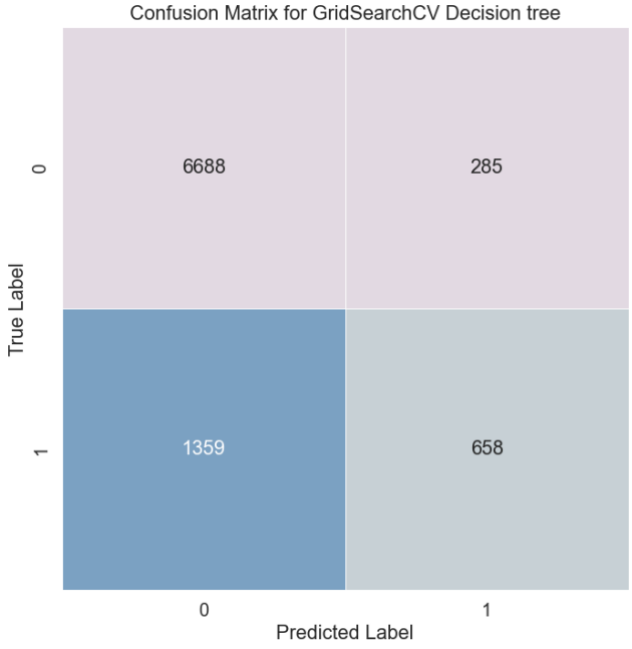

Fig8. Confusion matrix for GridSearchCV Decision Tree

The AUC-ROC curve is a performance measure of classification challenges at different threshold setup. ROC is a probability curve and AUC portrays the degree or measure of separability. It indicates how well the model is able to distinguish between classes [14, 20].

The AUC provides an overall measure of performance throughout all feasible classification thresholds. One way to interpret AUC is the probability that the model grades a positive random sample higher than a negative random sample.

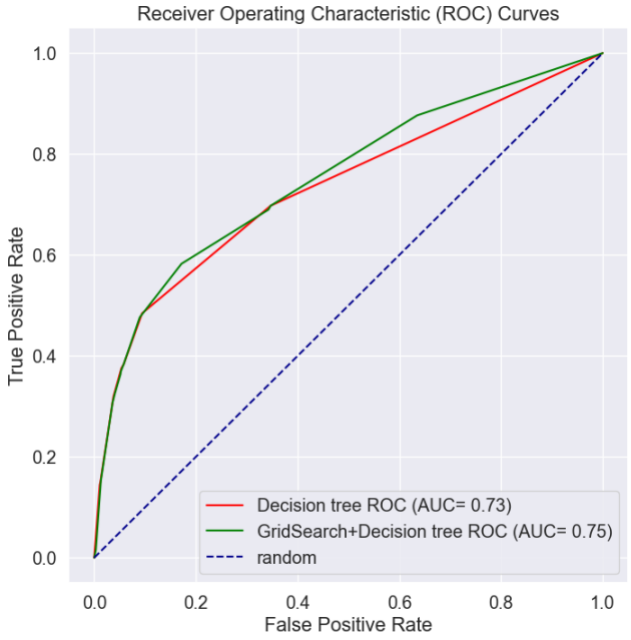

Fig9. ROC Curve (1)

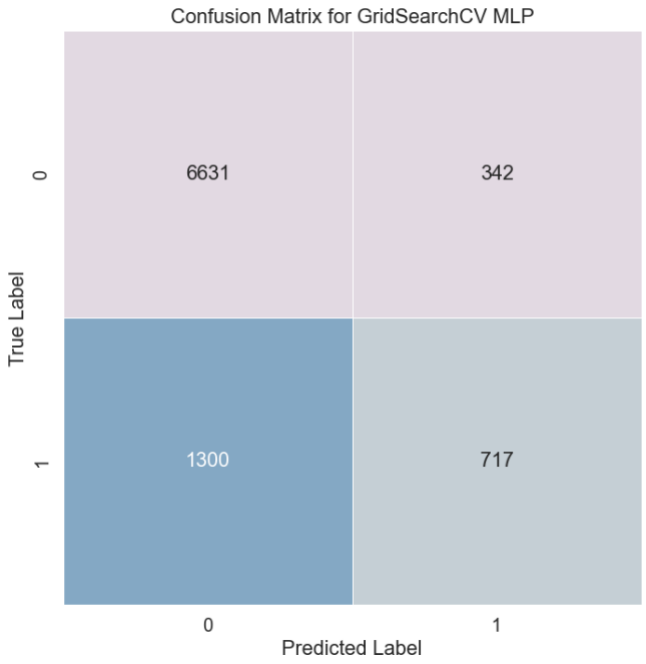

Fig10. Confusion matrix for GridSearchCV Neural Network

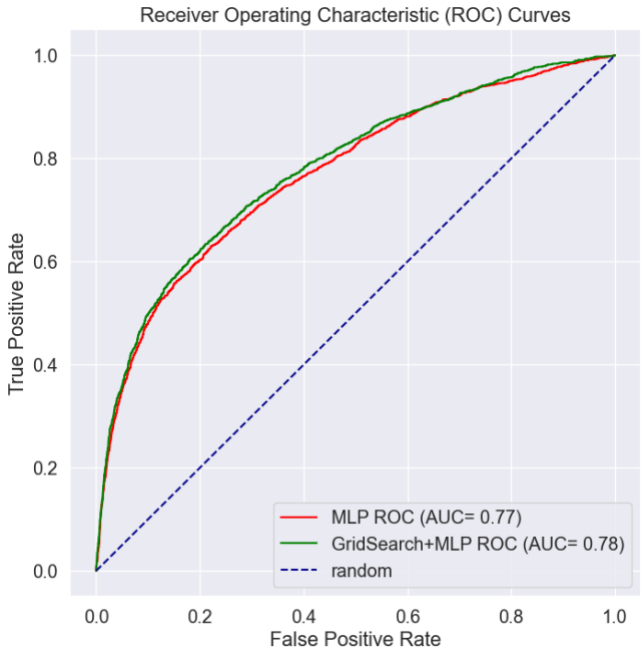

Fig11. ROC Curve (2)

F. Comparing the results of Decision Trees and Neural Networks

On comparison we have seen that, there is very slight difference in the Accuracy of both the models with and without Hyperparameter tuning.

However, there is significant difference in Area Under the ROC curve of Decision Trees and Neural Networks. Neural Networks has a higher Area under the ROC curve with and without hyperparameter tuning when compared to Decision Trees.

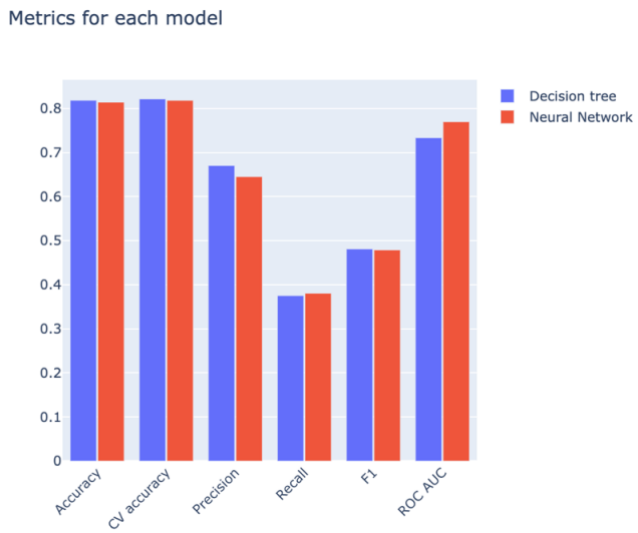

Fig12. Decision Trees vs Neural Networks (1)

In the graph above, we can see the comparison between Decision Trees and Neural Networks without GridSearchCV and the bar plot below shows the comparison with GridSearchCV.

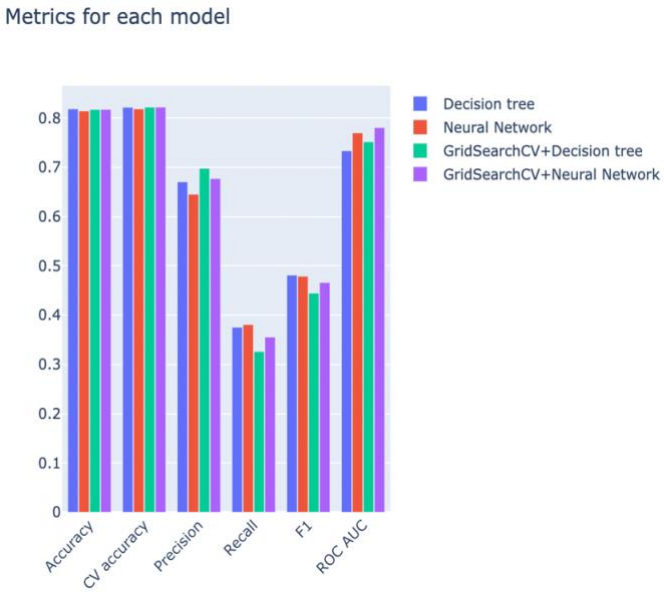

Fig13. Decision Trees vs Neural Networks (2)

The accuracy achieved of the Decision Trees model using GridSearchCV was 82.2%, whereas the cross-validation score was 81.8%, precision score was 69.8%, the recall score was 32.6% and the F1 score was 44.5%.

The accuracy achieved of the Neural Networks model using GridSearchCV was 82.2%, whereas the cross-validation score was 81.8%, precision score was 67.7%, the recall score was 35.5% and the F1 score was 47.6%.

V. ATTACKS ON NEURAL NETWORKS

White box assault

In this case, the entire model's information is at hand. Armed with this knowledge, attackers attempt to change malware files (detected by the model) into adversarial samples that perform identically to the malicious ones but are incorrectly categorized as benign. When the ML detector is a component of the client program and can be obtained via code reversing, this attack is practicable in real life. Such a situation was specifically identified for the Cylance antivirus program by researchers at Skylight [10].

Gray-box assault**:**

A substantial amount of memory and processing resources are often needed for complex ML models. As a result, the servers of the security firm may host the cloud-based ML classifiers. In this instance, the client programmers are only computing and sending file features to these servers. The cloud-based malware classifier replies with forecasts for the specified characteristics. Despite without having access to the model, the attackers still understand how features are built, so they can use the security product to scan any file and obtain labels for it [13].

Black-box assault:

In this instance, the cybersecurity business handles feature processing and model prediction. Either the security provider obtains files in another method, or the client apps just submit raw files. As a result, there is no information available on feature processing. The transmission of information from a user machine is subject to stringent legal constraints. This strategy also involves limiting traffic. This indicates that the virus detection procedure often cannot be run simultaneously for all user files. Consequently, a black-box system assault remains the most challenging [3].

Malware Injection:

Neural network models are inadequately explainable and have a decent generalization capability. By embedding malware in neurons, the malware can be delivered covertly, with minor or no impact on the performance of neural network

Extraction Attack:

An ML model extraction attack arises when an adversary obtains black-box access to some target model f and attempts to learn a model that closely approximates, or even matches model [2, 12].

Backdooring:

Backdoor attacks are effective at such applications since the attacker can leave some poisoned data on the web for the victims to download and use in training. It is not easy to mitigate such attacks as in the big data setting [9].

A. Attack Detection:

Malware Injection

It can be detected using “traditional methods” like static and dynamic analysis. It is still extremely difficult to detect as Neural networks are complex with multi-layered neurons [11].

Extraction Attack

1. Extracting the model demands running multiple queries on the target model

2. The surveyed samples are purposely produced and/or chosen to excerpt the maximum amount of information. Samples sent by an Attacker should have a distinctive distribution that differs from the distribution of samples sent in benign requests.

Backdooring

It is possible to detect this by outlier detection.

VI. PREVENTION

A. K-Anonymity

K-anonymity is based on the notion that by merging sets of data with alike attributes, determining information about any of the people who contribute that data can be concealed. K-anonymization is frequently referred to as the ability to "hide in the crowd." Data from individuals is aggregated into a bigger group, which means that group information can match each individual member, revealing the identity of the person or persons to be veiled issue.

The k in k-anonymity refers to a variable, k relates to the count of times each amalgamation of values occurs in a data set. If k = 2, the data is said to be 2-anonymous. This means that the data points have been generalized to the point that there are at least two sets of each data combination in the data set. For example, if a dataset contains the positions and ages of a group of people, the data should be generalized to include at least two occurrences of each age/position pair.

B. Implementing Generalization

In the chosen dataset, Age, Sex, and balance Limit are the possible QIDs and in order to prevent attacks we can generalize the data making it k-anonymous.
Thus, we implemented generalization on Age.

```
0         <25
1         <30
2         >40
3         >40
4         >40
         ...
29995     >40
29996     >40
29997     >40
29998     >40
29999     >40
Name: age_group, Length: 29965, dtype: category
Categories (4, object): ['<25' < '<30' < '>40' <
'>60']
```

Fig14. K-Anonymity Generalization

Now, if we drop the AGE column, it will be generalized. Similarly, we can implement this with sex column and thus prevent attacks on our data.

VII. Conclusion

In conclusion, we can say that the Neural network have a better AUC score when compared to Decision Trees for this model with the use of GridSearchCV for hyperparameter tuning with an AUC score of 78%. Multilayer Perceptron Neural Networks are widely used because of their in-depth analysis of data using forward and backpropagation. It is known that AUC score is a better predictor of model accuracy than the machine learning metrics. Further, we learned about the attacks that can be carried out on Neural Networks and their prevention methods such as Backdoor Attack, Malware Injection, Information Extraction attack, etc.

Consequently, we implemented Generalization method on the QID 'Age' to make it k-anonymous which can prevent attacks from happening on the model. This can be done on other QIDs as well to make the model more secure.

References

Basic format for books:

[1] Agrawal, S. and Agrawal, J., 2015. Survey on anomaly detection using data mining techniques. Procedia Computer Science, 60, pp.708-713.

[2] Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U. and Oprea, A., 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)* (pp. 2633-2650).

[3] Su, J., Vargas, D.V. and Sakurai, K., 2019. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, *23*(5), pp.828-841.

[4] Taylor, J.G. and Taylor, J.G. eds., 1996. Neural networks and their applications.

[5] Delamaire, L., Abdou, H. and Pointon, J., 2009. Credit card fraud and detection techniques: a review. *Banks and Bank systems*, *4*(2), pp.57-68.

[6] Raj, S.B.E. and Portia, A.A., 2011, March. Analysis on credit card fraud detection methods. In *2011 International Conference on Computer, Communication and Electrical Technology (ICCCET)* (pp. 152-156). IEEE.

[7] Bhattacharyya, S., Jha, S., Tharakunnel, K. and Westland, J.C., 2011. Data mining for credit card fraud: A comparative study. *Decision support systems*, *50*(3), pp.602-613.

[8] Chan, P.K., Fan, W., Prodromidis, A.L. and Stolfo, S.J., 1999. Distributed data mining in credit card fraud detection. *IEEE Intelligent Systems and Their Applications*, *14*(6), pp.67-74.

[9] Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H. and Zhao, B.Y., 2019, May. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)* (pp. 707-723). IEEE.

[10] Juuti, M., Szyller, S., Marchal, S. and Asokan, N., 2019, June. PRADA: protecting against DNN model stealing attacks. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)* (pp. 512-527). IEEE.

[11] Wu, B., Yang, X., Pan, S. and Yuan, X., 2022, May. Model Extraction Attacks on Graph Neural Networks: Taxonomy and Realisation. In *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security* (pp. 337-350).

[12] Li, T., 2021. *Model Extraction and Adversarial Attacks on Neural Networks Using Side-Channel Information*. Rochester Institute of Technology.

[13] Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U. and Oprea, A., 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)* (pp. 2633-2650).

[14] Fan, J., Upadhye, S. and Worster, A., 2006. Understanding receiver operating characteristic (ROC) curves. *Canadian Journal of Emergency Medicine*, *8*(1), pp.19-20.

[15] Carvalho, D.V., Pereira, E.M. and Cardoso, J.S., 2019. Machine learning interpretability: A survey on methods and metrics. *Electronics*, *8*(8), p.832.

[16] Zhou, J., Gandomi, A.H., Chen, F. and Holzinger, A., 2021. Evaluating the quality of machine learning explanations: A survey on methods and metrics. *Electronics*, *10*(5), p.593.

[17] Podgorelec, V., Kokol, P., Stiglic, B. and Rozman, I., 2002. Decision trees: an overview and their use in medicine. *Journal of medical systems*, *26*(5), pp.445-463.

[18] Quinlan, J.R., 1990. Decision trees and decision-making. *IEEE Transactions on Systems, Man, and Cybernetics*, *20*(2), pp.339-346.

[19] Kartini, D., Nugrahadi, D.T. and Farmadi, A., 2021, September. Hyperparameter tuning using GRIDSEARCHCV on the comparison of the activation function of the ELM method to the classification of pneumonia in toddlers. In *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)* (pp. 390-395). IEEE.

[20] Paper, D. and Paper, D., 2020. Scikit-Learn Regression Tuning. *Hands-on Scikit-Learn for Machine Learning Applications: Data Science Fundamentals with Python*, pp.189-213.