A thesis on Credit Card Fraud Detection Using Artificial Neural Network

Project Work (CA 6151) report submitted to

National Institute of Technology Durgapur

in partial fulfilment for the award of the degree of

Master of Computer Applications

in

Computer Science

by

Abhishek Singh

(16/CA/6004)

Under the supervision of

Dr. Nanda Dulal Jana

Assistant Professor

Department of Computer Science and Engineering

National Institute of Technology Durgapur

Spring Semester, 2018-19

May, 2019

CERTIFICATE OF APPROVAL

This forgoing thesis is hereby approved as a creditable study of Technology subjects carried out and presented in a proper manner of satisfactory to warrant its acceptance as a prerequisite with degree for which it has been submitted. It is to be understood that by this approval, the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn there in but approve the thesis only for the purpose for which it has been submitted.

BOARD OF THESIS EXAMINORS

1. ________________________

2. ________________________

3. ________________________

4. ________________________

DECLARATION

I certify that

(a) The work contained in this report has been done by me under the guidance of my supervisor.

(b) The work has not been submitted to any other Institute for any degree or diploma.

(c) I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

(d) Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

Date: May, 2019 (Abhishek Singh)
Place: Durgapur (16/CA/6004)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY DURGAPUR

DURGAPUR-713209, INDIA

CERTIFICATE

This is to certify that the project report entitled "**A thesis on Credit Card Fraud Detection Using Artificial Neural Network** " submitted by **Abhishek Singh** (Roll No. 16/CA/6004) to National Institute of Technology Durgapur towards partial fulfilment of requirements for the award of degree of Master of Computer Applications in Computer Science is a record of bonafide work carried out by him under my supervision and guidance during Spring Semester, 2018-19.

Dr. Tandra Paul	Dr. Nanda Dulal Jana
Department of Computer Science and Engineering	Department of Computer Science and Engineering
National Institute of Technology Durgapur	National Institute of Technology Durgapur
Durgapur-713209, India	Durgapur-713209, India

Abstract

In present time, internet has become more popular in almost every domain of life. E-commerce , online shopping and transactions are increasing day by day. People are more used to online shopping and payment due to which frauds related to online payments are on the rise. The main modes of the online payment are credit cards, debit cards, net banking. The fraud related to Credit card is increasing nowadays because people prefer using them as an easy mode of payment. Many credit card fraud identification systems have been introduced till now. In this paper we will try to detect fraud transaction through the Artificial Neural Network. As we will see that Artificial Neural Network When trained properly can work as a human brain, though it is impossible for any machine driven techniques to imitate the human brain to extent at which brain works, yet Neural Network and brain, depend for there working on the neurons, which is the small functional unit in brain as well as ANN.

Acknowledgements

First and foremost, I would like to thank my research supervisor Dr. Nanda Dulal jana without his assistance and dedicated involvement in every step throughout the process, this thesis would never have been accomplished. Their guidance helped me in all the time of project and writing of this thesis. I would like to thank him very much for the support, patience, motivation and understanding over this duration of the project.

I wish to acknowledge Dr. Tandra Pal, HOD of CSE, for providing supporting environment for my work. I would also like to show gratitude and respect to other faculty members of our department for their encouragement.

Last but not the least, I would like to thank my family, my parents and my friends for their unconditional support.

Abhishek Singh
16/CA/6004
National Institute Of Technology

Contents

report subfigure comment graphicx

Chapter 1

Intoduction

The advent of credit cards and their increasing functionality have not only given people more personal comfort, but have also attracted malicious characters interested in the handsome rewards to be earned. Credit cards are nice target for fraud, since in a very short time a lot of money can be earned without taking many risks. This is because often the crime is only discovered a few weeks after the date. Credit card fraud can be defined as "unauthorized account activity by a person for which the account was not intended. Operationally, this is an event for which action can be taken to stop the abuse in progress and incorporate risk management practices to protect against similar actions in the future". In simple terms, credit card fraud is defined when an individual uses another individual's credit card for personal benefit while the owner of the card and the card issuer are not aware of the fact that the card is being misused. And the person using the card has not at all having the connection with the card holder or the issuer and has no intention of making the repayments for the purchase they done.

Chapter 2

DIFFERENT TYPES OF FRAUD TECHNIQUES

There are three classes of frauds namely card related, merchant related and internet frauds. Some of them are listed below

2.1 Card Related Frauds

2.1.1 Lost/Stolen Card:

This type of fraud occurs when the fraudster simply steals a customer's card. In this case, the customer might feel he has lost his card, but actually this card might have been acquired by an attacker.

2.1.2 Account Takeover:

This type of fraud occurs when the valid customer's personal information is taken by fraudsters. The fraudsters takes control of a legitimate account by either providing the customer's account number or the card number. The fraudster then contacts the card issuer, as the genuine card holder, to ask the mail to redirect to a new address. The fraudster reports card lost and asks for a replacement to be sent.

2.1.3 Cardholder-Not-Present (CNP):

CNP transactions are performed only on the internet that is remotely, in such kind of frauds neither the card nor the cardholder is present at the point-of-sale. This take many types of transactions such as orders made over the phone or Internet, by mail order or fax. In such transactions, retailers are unable to physically check user or identity of the card holder which makes the user unknown and able to disguise their true identity, The details of the credit card are normally copied without the cardholder's knowledge, collected from the receipts thrown by the customer or obtained by skimming process. Frequently obtained card details are generally used with fabricated personal details to make fraudulent CNP purchases. This means that while the three or four digit card security code (CVV number) on the back of cards can help prevent fraud where card details have been obtained, but when the card is stolen it won't be helpful.

2.1.4 Erasing the Magnetic strip:

This is the type of fraud where the fraudsters erase the magnetic stripe by using the powerful electromagnet. The fraudsters then tampers with the details on the card so that they match the details of a valid card, which they may have attained, for example, when the fraudster begins to use the card, the cashier will swipe the card through the terminal several times, before realizing that the metallic strip does not work. The cashier will then proceed to manually input the card details into the terminal. This kind of fraud is having high risk because the cashier will be looking at the card closely to read the numbers.

2.1.5 Phishing:

Phishing is a type of fraud designed to steal a person's identity. It is usually committed via spam e-mail or pop-up windows. Phishing works by a malicious person sending lots of false emails. The emails looks like they have come from a website or company you trust, for example your bank. The message tells you to provide the company with your personal details including your payment card details. They can claim that the reason for this is a database crash or something like this. To make the email look even more authentic, the fraudster might put a link to a website that look exactly like the real one but in fact that is not the real one and that is the fake one. These copies are often called “Spoofed websites”. When you are on the spoofed site they can ask you for even more personal details that will be directly transmitted to the person who made that website.

2.2 Merchant Related Frauds

2.2.1 Merchant Collusion:

This type is done when a merchant purposely passes on his customer’s personal information to fraudsters.

2.2.2 Triangulation:

Here, the fraudster creates a fake website and operates from there. Many discounts are given to the customers through this website due to which users are attracted to such websites. They purchase items and there they enter their personal information. Then this information is obtained by the fraudsters and they use it to perform illegimitate transactions.

2.3 Internet Frauds

2.3.1 False Merchant Sites:

In this type, the website asks the customers to enter their personal details if they want to access the content of the website. In this way, these fraudsters collect many credit card number which they use later for performing fraudulent transactions.

2.3.2 Keystroke Loggers:

“Keystroke logger” is a spyware which infects a user’s computer unknown to him. This spyware tracks all the details typed by the user and gives this information to the fraudster who thus obtains all the personal details.

2.3.3 Cell phone camera Scan:

When a customer is paying his bills, a fraudster may be roaming somewhere near him. The customer may be under the assumption that the attacker is busy chatting on his phone, but actually he is taking digital image of the computer’s details such as card number, expiry date, etc. This type of fraud is possible because of powerful cameras used these days.

2.3.4 Site Cloning:

Site cloning is where fraudsters close an entire site or just the pages from which the customer made a purchase. Customers have no reason to believe they are not dealing with the company that they wished to purchase goods or services from because the pages that they are viewing are identical to those of the real site. The cloned site will receive these details and send the customer a receipt of the transaction through the email just as the real company would do. The customer suspects nothing, while the fraudsters have all the details they need to commit credit card fraud.

Chapter 3

PROBLEMS WITH CREDIT CARD FRAUD DETECTION

One of the biggest problems associated with fraud detection is the lack of both literature providing experimental results and of real world data for academic researchers to perform experiments on. This is because fraud detection is often associated with sensitive financial data that is kept confidential for reasons of customer privacy.

We now enumerate some of the properties a fraud detection system should have in order to perform good results.

- The system should be able to handle skewed distributions, since only a very small percentage of all credit card transactions is fraudulent, To solve this problem, often the training sets are divided into pieces where the distribution is less skewed.
- The ability to handle noise. This is simply the presence of errors in the data, for instance incorrect dates. Noise in actual data limits the accuracy of generalization that can be achieved, no matter how extensive the training set is. One way to deal with this problem is by cleaning the data.
- Overlapping data is another problem in this field. Many transactions may resemble fraudulent transactions, when actually they are legitimate. The opposite also happens, when a fraudulent transaction appears to be normal.

- The system should be able to adapt themselves to new kinds of fraud. Since after a while successful fraud techniques decrease in efficiency, due to the fact that they become well known. Then a "Good" fraud tries to find new and inventive ways of doing his job.

- There is a need for good metrics to evaluate the classifier system. As an example, the overall accuracy is not suited for evaluation on a skewed distribution, since even with a very high accuracy, almost all fraudulent transactions can be misclassified.

This means that there should be a decision layer on top of the fraud detection system. The decision layer decides what action to take when fraudulent behaviour is detected via the fraud detection system, taking into account factors like the amount of the transaction and the quality of the customer doing the transaction

Chapter 4

ARTIFICIAL NEURAL NETWORKS

4.1 Definition

Artificial Neural Network (ANN)(1) is an efficient computing system whose central theme is borrowed from the analogy of biological neural networks. ANN acquires a large collection of units that are interconnected in some pattern to allow communication between the units. These units, referred to as nodes or neurons, are simple processors which operate in parallel. Every neuron is connected with other neuron through a connection link. Each connection link is associated with a weight that has information about the input signal. This is the most useful information for neurons to solve a particular problem because the weight usually excites or inhibits the signal that is being communicated. Each neuron has an internal state, which is called an activation signal. Output signals, which are produced after combining the input signals and activation rule, may be sent to other units.

Neural Network exists in many ways and diffeent forms. The type of neural network we will discuss in this paper is the Feed Forward Multilayer Perceptron. A feedforward multilayer perceptron consists of different layers of perceptrons that are interconnected by a set of weighted connections We can distinguish three types of layers:

4.1.1 Input

Input layer Receives input from an input stream, which can be database or some device or something else.

4.1.2 Hidden Layer

Is hidden from the outside world and receives input only from the input layer or another hidden layer

4.1.3 Output Layer

Connects the network to the outside world again and provides the final output of the network.

A feed forward multilayer perceptron has no cycles and there is full connectivity between the perceptions of two consecutive layers. Signals can be propagated in two directions: function signals are propagated forwards, i.e. from input laver through the hidden layer(s) to the output layer and error signals are propagated backwards, i.e. from output layer through the hidden layer(s) to the input layer.

4.2 Biological Neuron

A nerve cell (neuron) is a special biological cell that processes information. According to an estimation, there are huge number of neurons, approximately 10^{11} with numerous interconnections, approximately 10^{15}

4.2.1 Working of a Biological Neuron

As shown in the diagram (2), a typical neuron consists of the following four parts with the help of which we can explain its working

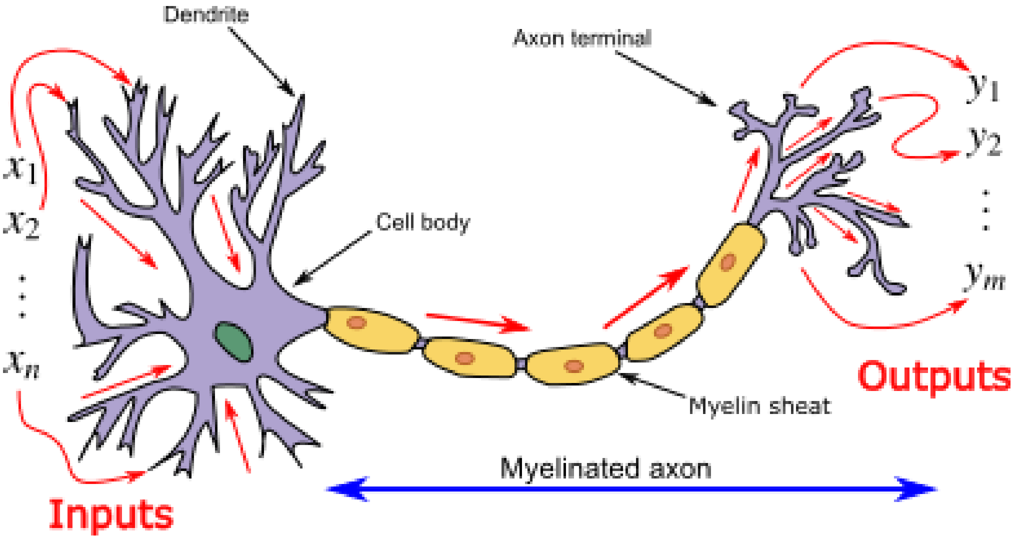


FIGURE 4.1: Biological Neuron

- **Dendrites:** They are tree-like branches, responsible for receiving the information from other neurons it is connected to. In other sense, we can say that they are like the ears of neuron.
- **Soma:** It is the cell body of the neuron and is responsible for processing of information, they have received from dendrites.
- **Axon:** It is just like a cable through which neurons send the information.
- **Synapses:** It is the connection between the axon and other neuron dendrites.

4.3 Biological neural Network(BNN) v/s Artificial Neural Network(ANN)

Before taking a look at the differences between Artificial Neural Network (ANN) and Biological Neural Network (BNN), let us take a look at the similarities based on the terminology between these two.

Biological neural Network (BNN)	**Artificial Neural Network (ANN)**
Soma	Node
Dendrites	Input
Synapse	Weights or Interconnections
Axon	Output

The following table shows the comparison between ANN and BNN based on some criteria mentioned.

Criteria	**BNN**	**ANN**
Processing	Massively parallel, Slow but superior than ANN	Massively Parallel, Fast but inferior than BNN
Size	10^{11}neurons and 10^{15}interconnections	$10^{2}$to 10^{4}nodes (mainly depends on the type of application and network designer)
Learning	They can tolerate ambiguity	Very precise, structured and formatted data is required to tolerate ambiguity
Fault Tolerance	Performance degrade with even partial damage	It is capable of robust performance, hence has the potential to be fault tolerant
Storage Capacity	Stores the information in the synapse	Stores the information in continuous memory locations

4.4 Model of Artificial Neural Network

The following diagram represents the general model of ANN followed by its processing.

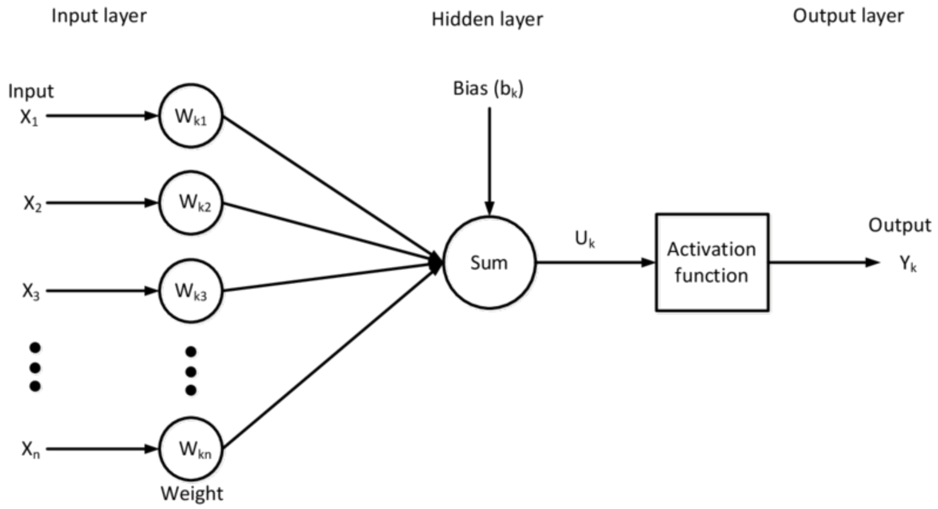


Figure 4.2: ANN Model

For the above general model of artificial neural network, the net input can be calculated as follows

$$Y_{in} = x_1 * w_1 + x_2 * w_2 + x_3 * w_3 + ... + X_n * w_n$$

i.e., net input

$$y_{in} = \sum_{i}^{n} x_i * w_i$$

The output can be calculated by applying the activation function over the net input.

$$Y = F(Y_{in})$$

Output = function (net input calculated)

Processing of ANN depends upon the following three building blocks

- Network Topology
- Adjustments of Weights or Learning
- Activation Functions

4.5 NETWORK TOPOLOGY

A network topology (3) is the arrangement of a network along with its nodes and connecting lines. According to the topology, ANN can be classified as the following kinds

4.5.1 Feedforword Network

It is a non-recurrent network having processing units/nodes in layers and all the nodes in a layer are connected with the nodes of the previous layers. The connection has different weights upon them. There is no feedback loop means the signal can only flow in one direction, from input to output. It may be divided into the following two types

- **Single layer feedforward network:** The concept is of feedforward ANN having only one weighted layer. In other words, we can say the input layer is fully connected to the output layer.

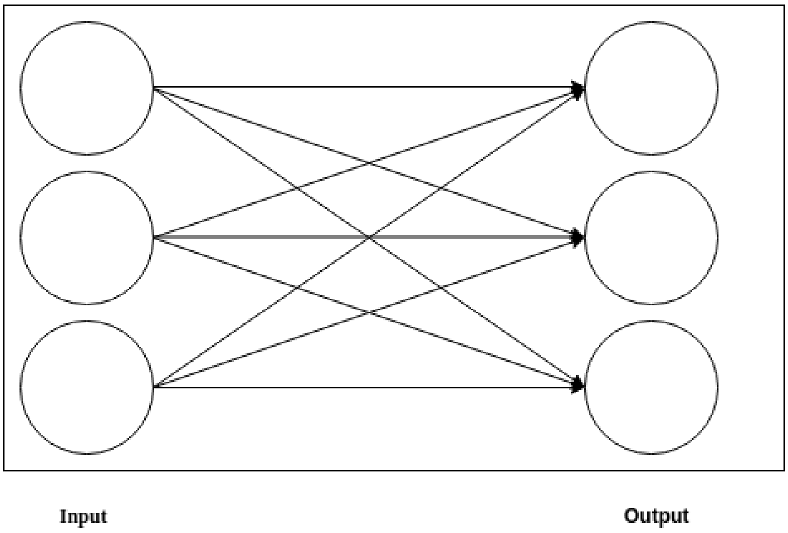


FIGURE 4.3: Single Layer Feed Forward Network

- **Multi layer feedforward network:** The concept is of feedforward ANN having more than one weighted layer. As this network has one or more layers between the input and the output layer, it is called hidden layers.

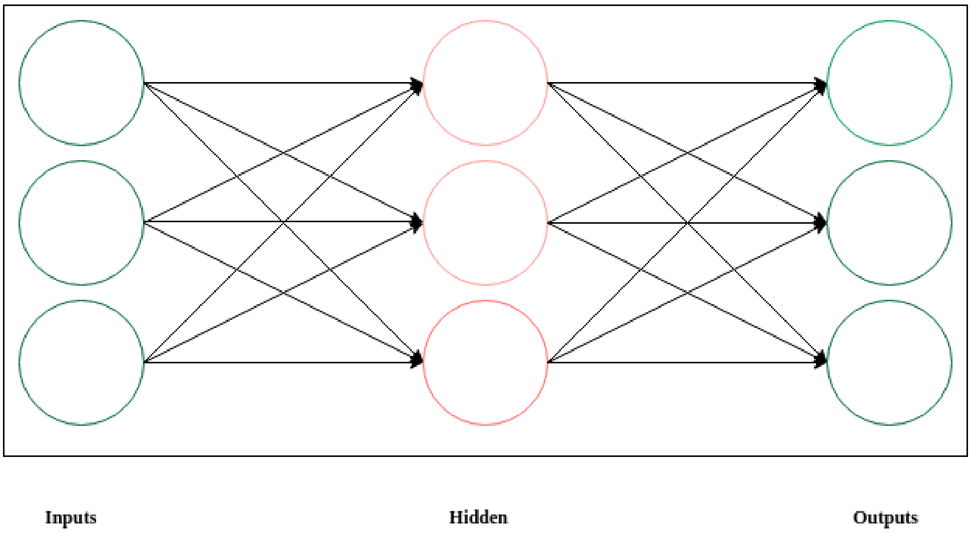


FIGURE 4.4: Multilayer Feedforward Network

4.5.2 Feedback Network

As the name suggests, a feedback network has feedback paths, which means the signal can flow in both directions using loops. This makes it a non-linear dynamic system, which changes continuously until it reaches a state of equilibrium. It may be divided into the following types

- **Recurrent networks** They are feedback networks with closed loops. Following are the two types of recurrent networks.
- **Fully recurrent network** It is the simplest neural network architecture because all nodes are connected to all other nodes and each node works as both input and output.

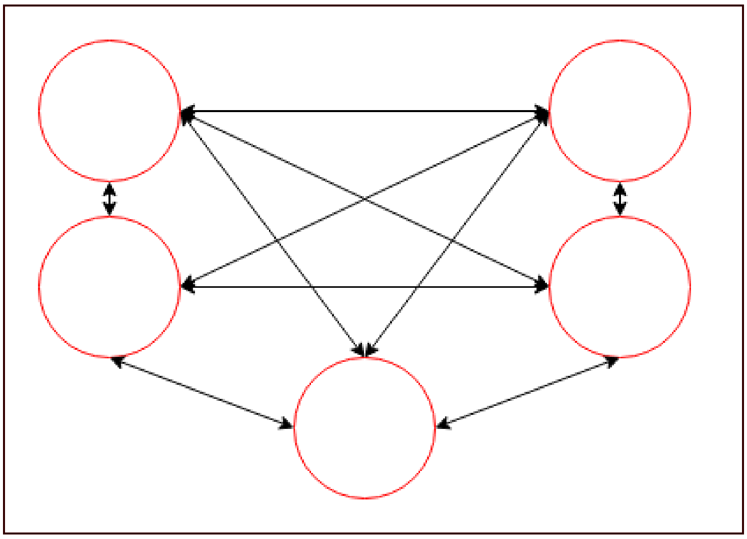


FIGURE 4.5: Fully Recurrent Network

- **Jordan network** It is a closed loop network in which the output will go to the input again as feedback as shown in the following diagram.

4.6 Adjustments of Weights or Learning

Learning, in artificial neural network, is the method of modifying the weights of connections between the neurons of a specified network. Learning in ANN can be

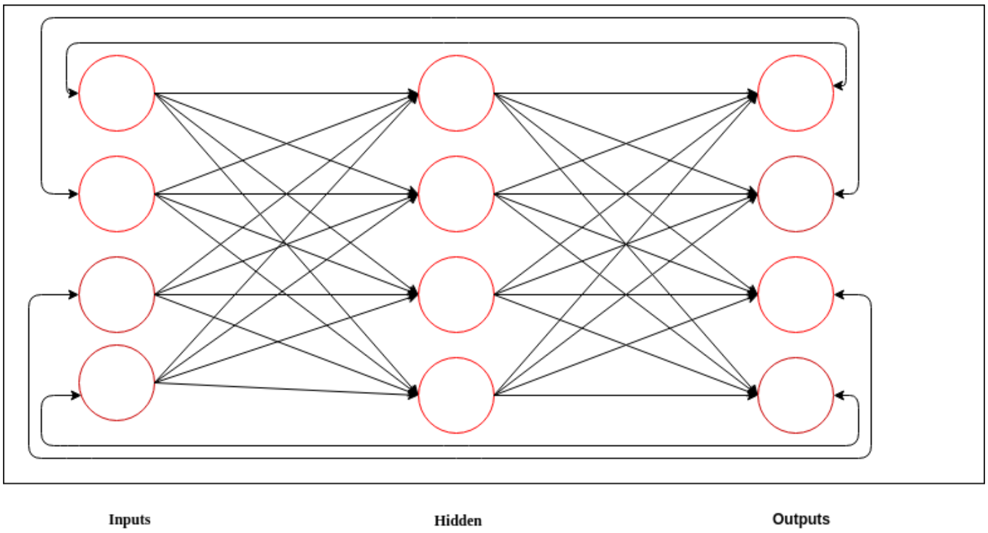


FIGURE 4.6: Jordan Network

classified into three categories namely supervised learning, unsupervised learning, and reinforcement learning.

4.6.1 Supervised Learning

As the name suggests, this type of learning is done under the supervision of a teacher. This learning process is dependent.

During the training of ANN under supervised learning, the input vector is presented to the network, which will give an output vector. This output vector is compared with the desired output vector. An error signal is generated, if there is a difference between the actual output and the desired output vector. On the basis of this error signal, the weights are adjusted until the actual output is matched with the desired output.

4.6.2 Unsupervised Learning

As the name suggests, this type of learning is done without the supervision of a teacher. This learning process is independent.

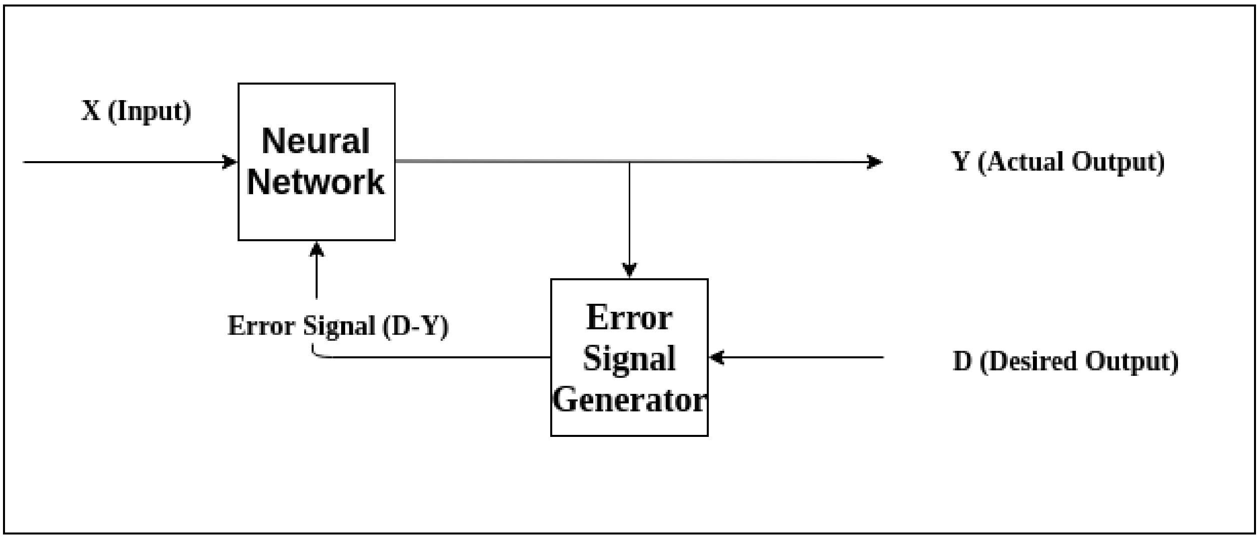


Figure 4.7: Supervised Learning

During the training of ANN under unsupervised learning, the input vectors of similar type are combined to form clusters. When a new input pattern is applied, then the neural network gives an output response indicating the class to which the input pattern belongs. There is no feedback from the environment as to what should be the desired output and if it is correct or incorrect. Hence, in this type of learning, the network itself must discover the patterns and features from the input data, and the relation for the input data over the output.

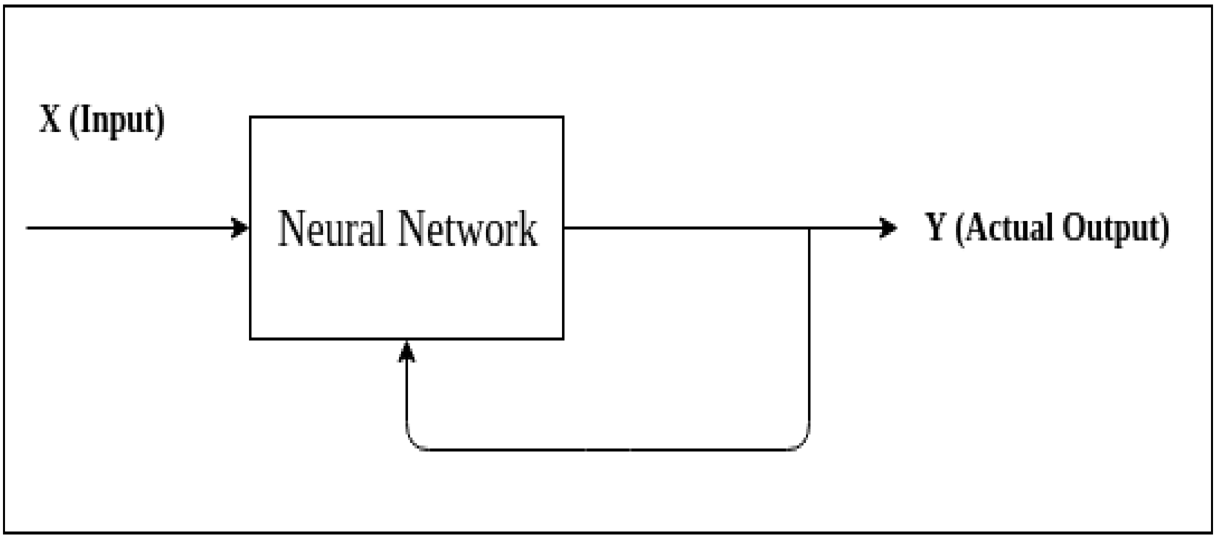


Figure 4.8: Unsupervised Learning

4.6.3 Reinforcement Learning

As the name suggests, this type of learning is used to reinforce or strengthen the network over some critic information. This learning process is similar to supervised learning, however we might have very less information.

During the training of network under reinforcement learning, the network receives some feedback from the environment. This makes it somewhat similar to supervised learning. However, the feedback obtained here is evaluative not instructive, which means there is no teacher as in supervised learning. After receiving the feedback, the network performs adjustments of the weights to get better critic information in future.

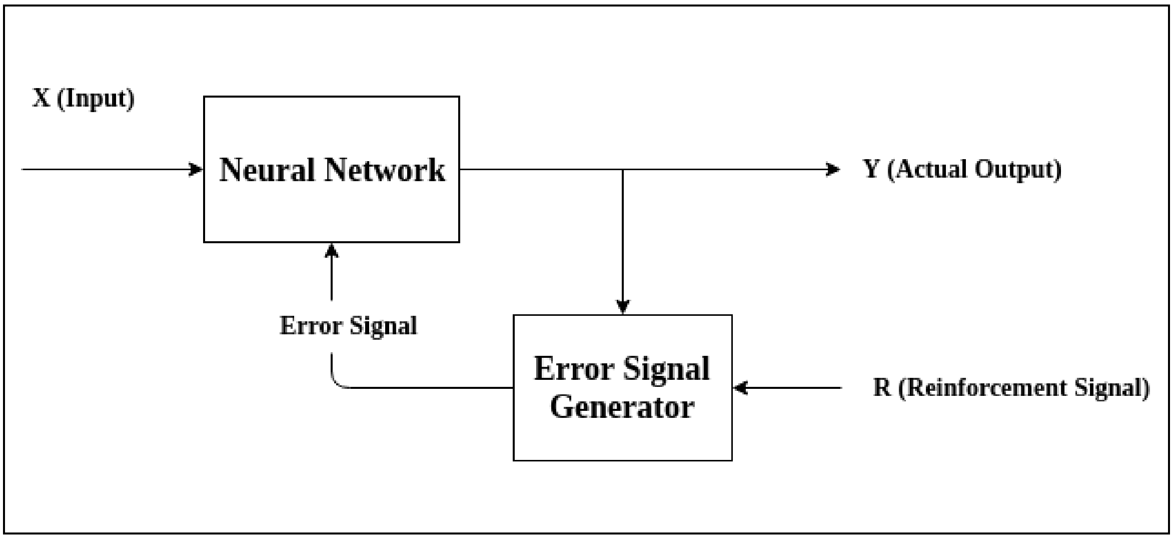


Figure 4.9: Reinforcement Learning

4.7 Activation Functions

It may be defined as the extra force or effort applied over the input to obtain an exact output. In ANN, we can also apply activation functions over the input to get the exact output. Followings are some activation functions of interest

4.7.1 Linear Activation Function

It is also called the identity function as it performs no input editing. It can be defined as

$$f(x) = x$$

4.7.2 Sigmoid Activation Function

It is of two type as follows

- **Binary sigmoidal function** This activation function performs input editing between 0 and 1. It is positive in nature. It is always bounded, which means its output cannot be less than 0 and more than 1. It is also strictly increasing in nature, which means more the input higher would be the output. It can be defined as

  $$f(x) = \frac{1}{1 + e^{-x}}$$

- **Bipolar sigmoidal function** This activation function performs input editing between -1 and 1. It can be positive or negative in nature. It is always bounded, which means its output cannot be less than -1 and more than 1. It is also strictly increasing in nature like sigmoid function. It can be defined as

  $$f(x) = \frac{2}{1 + e^{-x}} - 1 = \frac{1 - e^{x}}{1 + e^{x}}$$

4.8 Learning

The type of learning we will discuss is commonly called supervised learning or error correction learning. The algorithm that we have used to do this is called Back Propagation of Error signals or in short Backprop. Every iteration of the algorith consists of two pases:

4.8.1 Forward Pass:

Forward pass every perceptron calculates the weighted linear combination of all its inputs and applies to the result of this summing junction an activation function. The result of the activation function provides the perceptron of its output value.

4.8.2 Backward Pass:

At the output layer of the network we calculate the error with respect to the desired output value for a certain pattern. This error is propagated backwards through the network enforcing a correction on the weights of all connections in the network. This technique is based on the observation that all perceptrons in the network have a shared responsibility for the error that has been calculated at the output layer.

4.9 Optimization Algorithms Used:

4.9.1 Adagrad:

Adagrad (4) is an algorithm for gradient-based optimization that does just this: It adapts the learning rate to the parameters, performing smaller updates (i.e. low learning rates) for parameters associated with frequently occurring features, and larger updates (i.e. high learning rates) for parameters associated with infrequent features. For this reason, it is well-suited for dealing with sparse data.

Pros:

- Well Suited for dealing with sparse data
- Significantly improves the robustness of SGD
- Lesser need to manually tune learning rate

Cons:

- Accumulates squared gradients in denominator. Causes the learning rate to shrink and become infinitesimally small.

4.9.2 Adadelta:

Adadelta (5) is a more robust extension of Adagrad that adapts learning rates based on a moving window of gradient updates, instead of accumulating all past gradients. This way, Adadelta continues learning even when many updates have been done. Compared to Adagrad, in the original version of Adadelta you don't have to set an initial learning rate. In this version, initial learning rate and decay factor can be set, as in most other Keras optimizers.

4.9.3 Adam:

Adaptive Moment Estimation (Adam) (6) is another method that computes adaptive learning rates for each parameter. In addition to storing an exponentially decaying average of past squared gradients similar to momentum. Whereas momentum can be seen as a ball running down a slope, Adam behaves like a heavy ball with friction, which thus prefers flat minima in the error surface.

4.9.4 SGD:

The word 'stochastic' means a system or a process that is linked with a random probability. Hence, in Stochastic Gradient Descent (7), a few samples are selected randomly instead of the whole data set for each iteration. In Gradient Descent, there is a term called “batch” which denotes the total number of samples from a dataset that is used for calculating the gradient for each iteration. In typical Gradient Descent optimization, like Batch Gradient Descent, the batch is taken to be the whole dataset. Although, using the whole dataset is really useful for getting to the minima in a less noisy or less random manner, but the problem arises when our datasets get really huge. Suppose, you have a million samples in your dataset, so if you use a typical Gradient Descent optimization technique, you will have to use all of the one million samples for completing one iteration while performing the Gradient Descent, and it has to be done for every iteration until the minima is reached. Hence, it becomes computationally very expensive to perform. This problem is solved by Stochastic Gradient Descent. In SGD, it uses only a single sample, i.e., a batch size of one, to perform each iteration. The sample is randomly shuffled and selected for performing the iteration.

Chapter 5

Environment Setup And Performance Metrices

5.1 Dataset:

The datasets contains transactions made by credit cards in September 2013 by european cardholders. This dataset presents transactions that occurred in two days, where there are 492 frauds out of 284,807 transactions. It contains only numeric input variables which are the result of a PCA transformation. Due to confidentiality issues, original features are not provided 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-dependant cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

5.2 Environment:

5.2.1 Jupyter Notebook:

The Jupyter Notebook is an interactive computing environment that enables users to author notebook documents that include: - Live code - Interactive widgets - Plots - Narrative text - Equations - Images - Video.

5.2.1.1 Components:

The Jupyter Notebook combines three components:

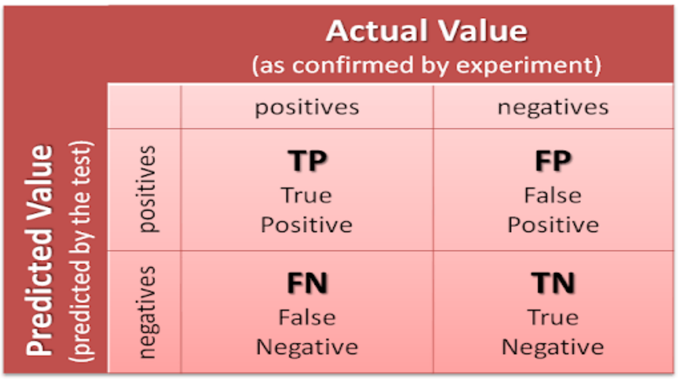
- **The Notebook Web Application:** An interactive web application for writing and running code interactively and authoring notebook documents.
- **Kernals:** Separate processes started by the notebook web application that runs users' code in a given language and returns output back to the notebook web application. The kernel also handles things like computations for interactive widgets, tab completion and introspection.
- **Notebook Documents:** Self-contained documents that contain a representation of all content visible in the notebook web application, including inputs and outputs of the computations, narrative text, equations, images, and rich media representations of objects. Each notebook document has its own kernel.

5.2.2 Operating System:

This project is done on Linux Operating Systems.

5.3 Confusion Matrix:

A confusion matrix(8) is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.



		Actual Value (as confirmed by experiment)	
		positives	negatives
Predicted Value (predicted by the test)	positives	**TP** True Positive	**FP** False Positive
	negatives	**FN** False Negative	**TN** True Negative

Figure 5.1

Definition of the Terms:

- **True Positive (TP):** Observation is positive, and is predicted to be positive.
- **False Negative (FN):** Observation is positive, but is predicted negative.
- **True Negative (TN):** Observation is negative, and is predicted to be negative.
- **False Positive (FP):** Observation is negative, and is predicted to be positive.

5.3.1 Classification Rate/Accuracy:

Classification Rate or Accuracy is given by the relation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

5.3.2 Recall:

Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (small number of FN).

Recall is given by the relation:

$$Recall = \frac{TP}{TP + FN}$$

5.3.3 Precision:

To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labeled as positive is indeed positive (small number of FP). Precision is given by the relation:

Recall is given by the relation:

$$Precision = \frac{TP}{TP + FP}$$

- **High recall, low precision:** This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives.
- **Low recall, high precision:** This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP)

5.3.4 F-Measure:

Since we have two measures (Precision and Recall) it helps to have a measurement that represents both of them. We calculate an F-measure which uses Harmonic Mean in place of Arithmetic Mean as it punishes the extreme values more. The F-Measure will always be nearer to the smaller value of Precision or Recall.

$$F - measure = \frac{2 * recall * precision}{recall + precision}$$

5.4 Performance Metrices:

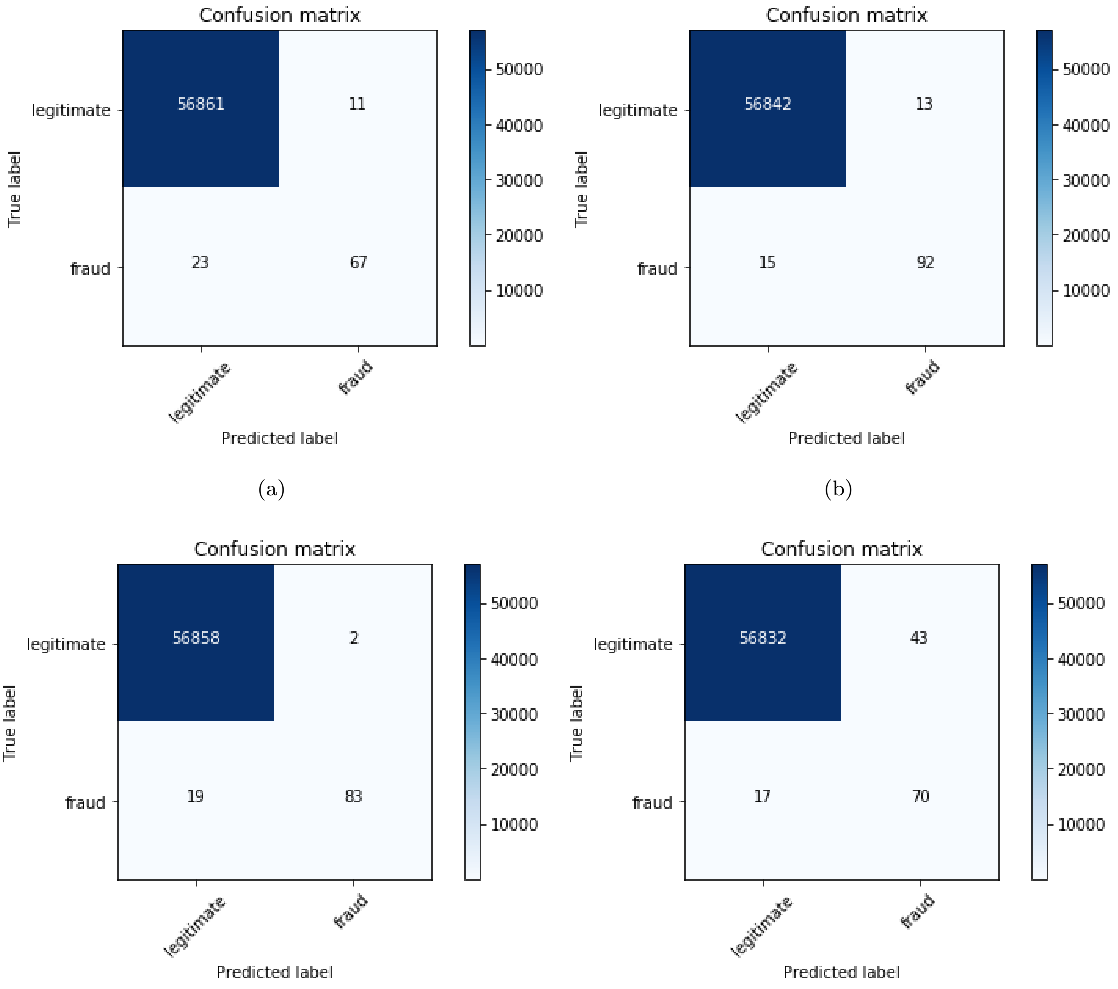
Performance Measures:

Performance Measures				
Algorithms		**Precesion**	**Recall**	**F-Score**
	ADAM	0.87	0.84	0.85
	SGD	0.88	0.86	0.87
	ADAGARD	0.98	0.81	**0.89**
	ADADELTA	0.62	0.8	0.7

Accuracy:

	Accuracy
ADAM	99.94
SGD	99.95
ADAGARD	**99.96**
ADADELTA	99.89

Confusion Metrices:



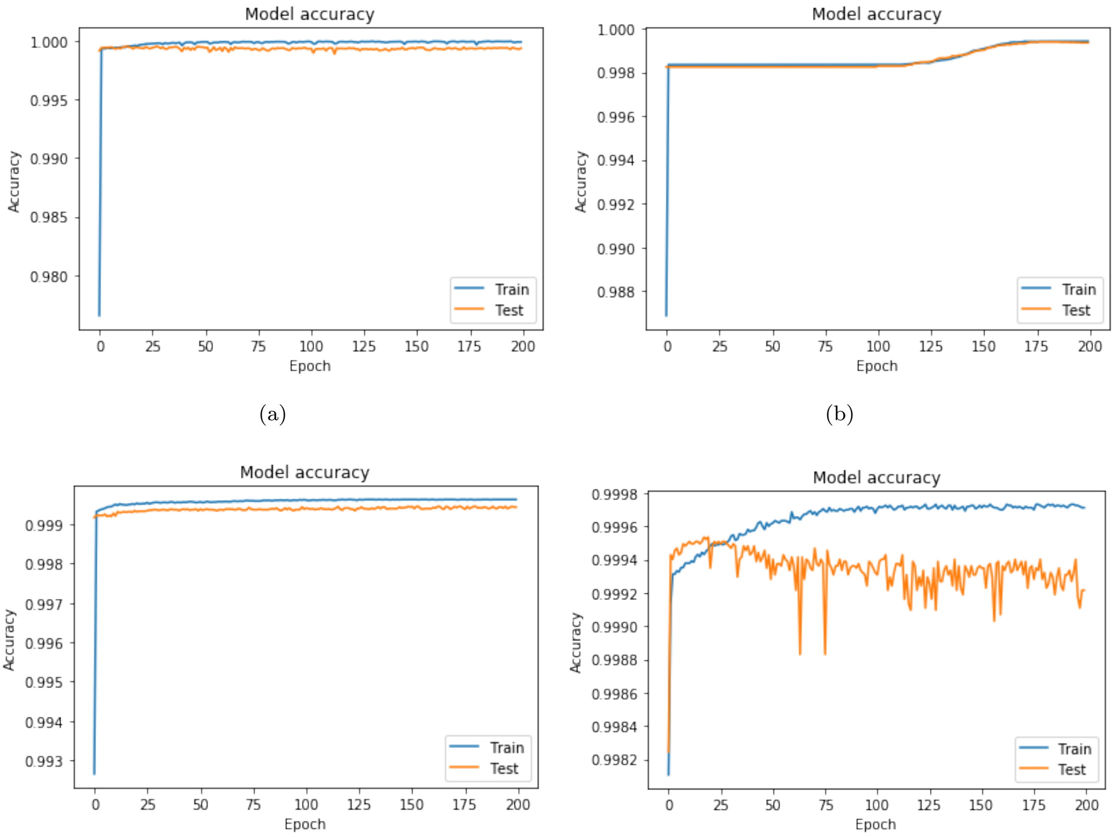
(a)

(b)

(c)

(d)

FIGURE 5.2: Confusion matrix comparison among 5.2(a) ADAM(6), 5.2(b) SGD(7), 5.2(c) ADAGRAD(4), 5.2(d) ADADELTA(5)



(a)

(b)

(c)

(d)

Figure 5.3: Model Accuracy Comparison among 5.3(a) ADAM(6), 5.3(b) SGD(7), 5.3(c) ADAGRAD(4), 5.3(d) ADADELTA(5)

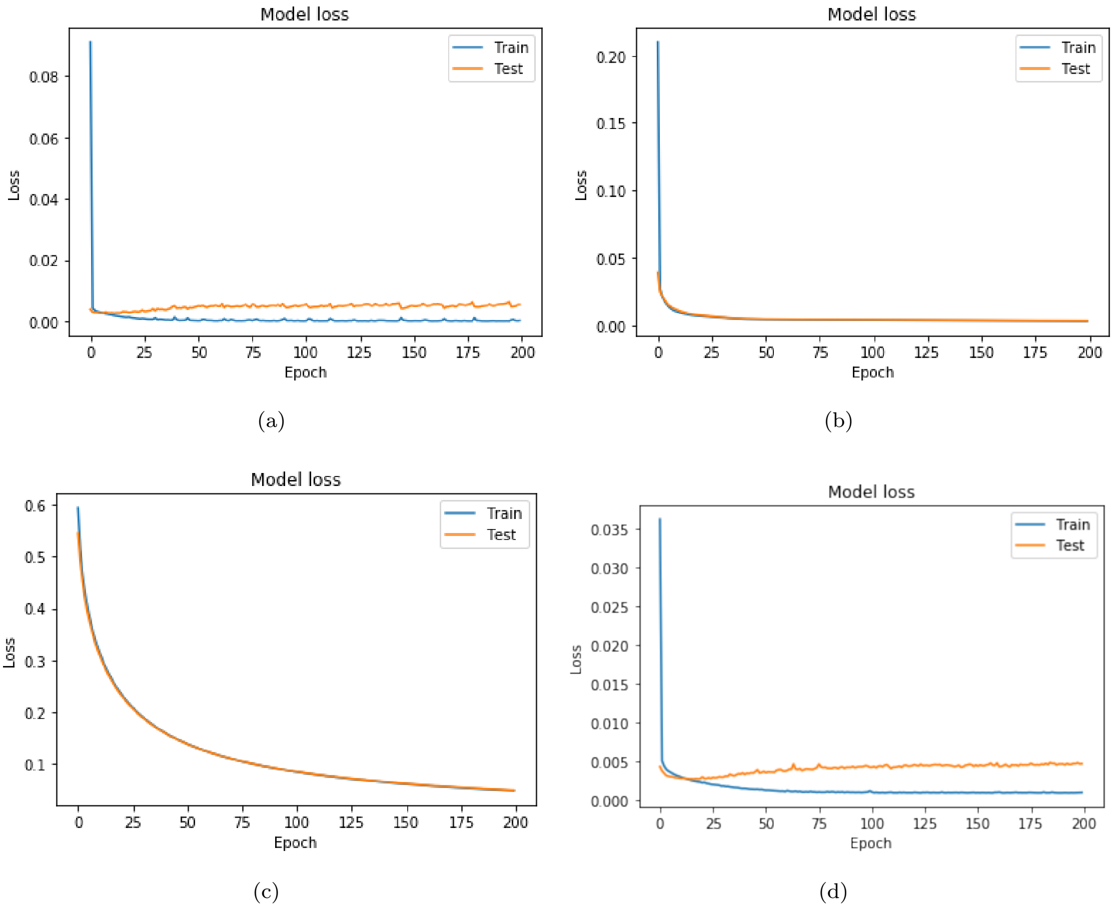


FIGURE 5.4: Model Loss Comparison among 5.4(a) ADAM(6), 5.4(b) SGD(7), 5.4(c) ADAGRAD(4), 5.4(d) ADADELTA(5)

Chapter 6

Result Discussion

6.1 Performance Comparisons

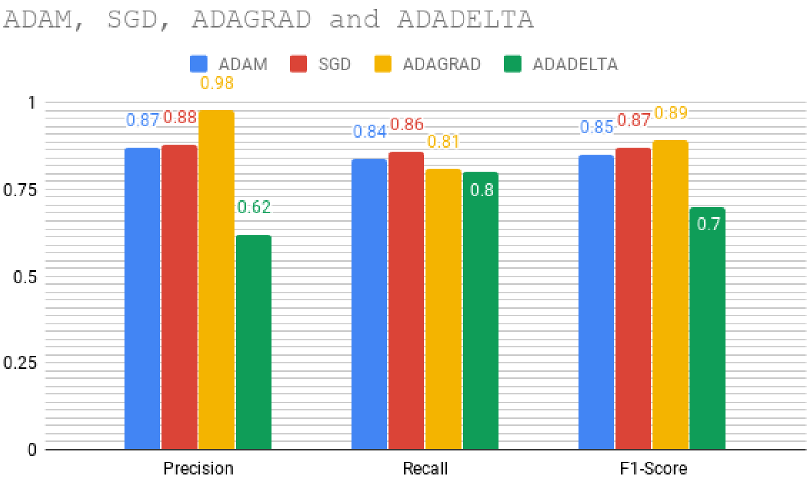


Figure 6.1: Precision, Recall and F-score comparison among ADAM, SGD, ADAGRAD, ADADELTA

As it is given in the above graph that ADAGRAD is having the heighest F-score so this optimizer algorithm is giving the best performance among these four algorithms. After this SGD, ADAM, ADADELTA are having better performances in decreasing order.

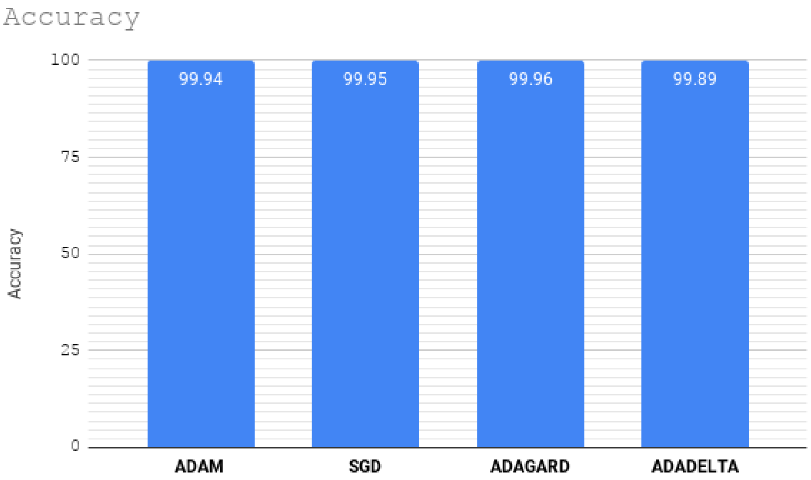


Figure 6.2: Accuracy Comparison among ADAM, SGD, ADAGRAD, ADADELTA

Bibliography

[1] https://www.tutorialspoint.com/artificial$_{n}eural_{n}etwork/artificial_{n}eural_{n}etwork_{b}asic_{c}oncept$

[2] https://en.wikipedia.org/wiki/Biological$_{n}euron_{m}odel$

[3] https://www.tutorialspoint.com/artificial$_{n}eural_{n}etwork/artificial_{n}eural_{n}etwork_{b}uilding_{b}lock$

[4] http://ruder.io/optimizing-gradient-descent/index.htmladagrad

[5] http://ruder.io/optimizing-gradient-descent/index.htmladadelta

[6] http://ruder.io/optimizing-gradient-descent/index.htmladam

[7] http://ruder.io/optimizing-gradient-descent/index.htmlstochasticgradientdescent

[8] https://www.geeksforgeeks.org/confusion-matrix-machine-learning/