CRIME PREDICTION MODEL USING ARTIFICIAL NEURAL NETWORK

Project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

Apurav Sharma (191007)
Srajan Sharma (191015)

UNDER THE GUIDANCE OF

Dr. Harsh Sohal

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

May 2023

TABLE OF CONTENTS

CHAPTER-3: SYSTEM DEVELOPMENT TOOLS

CHAPTER 4: DATASET ANALYSIS

CHAPTER 5: DATA PRE-PROCESSING

CHAPTER 6: MODEL MAKING

DECLARATION

We hereby declare that the work reported in the B.Tech Project Report entitled **“CRIME PREDICTION MODEL USING ARTIFICIAL NEURAL NETWORK ”** submitted at **Jaypee University of Information Technology, Waknaghat, India** is an authentic record of our work carried out under the supervision of **Dr.Harsh Sohal.** We have not submitted this work elsewhere for any other degree or diploma.

Signature of Student
Apurav Sharma
191007

Signature of Student
Srajan Sharma
191015

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Signature of the Supervisor
Dr.Harsh Sohal

Date: 08/05/2023

ACKNOWLEDGEMENT

Besides the hard work of a group, the success of a project also depends highly on the encouragement and guidelines of many others. We take this opportunity to express my sincere and heartfelt gratitude to the people who have been instrumental in the successful completion of this project. Our first and foremost acknowledgement goes to our supervisor and mentor, **Dr Harsh Sohal**, without whose help the completion of this project wouldn't have been possible. It is because of his guidance and efforts that we are able to implement a practical idea based on our field of interest. We would also like to thank our panel members for giving us an opportunity to present our project and for judging our work and providing us feedback which would certainly help me in the future. Last but not the least we would like to acknowledge my institution **Jaypee University of Information Technology** for giving us a platform to give me life and implementation, to the various fields we have studied till date.

LIST OF ACRONYMS AND ABBREVIATIONS

- **ANN** Artificial Neural Network
- **CNN** Convolution Neural Network
- **GIS** Geographic Information System
- **KDE** Kernel Density Estimation
- **KNN** K-Nearest Neighbour
- **LDA** Linear Discriminant Analysis
- **RNN** Recurrent Neural Network
- **RTM** Risk Terrain Modelling
- **VPD** Vancouver Police Department

LIST OF FIGURES

Abstract

One of the biggest and most prevalent issues in our society, crime prevention is a necessary endeavor. Numerous crimes are committed daily in large numbers. This calls for recording all crimes and compiling them into a database that may be accessed later. The current problem faced are maintaining of proper dataset of crime and analyzing this data to help in predicting and solving crimes in future. The objective of this project is to analyze dataset which consist of numerous crimes and predicting the type of crime which may happen in future depending upon various conditions. In this project, we will be using the technique of machine learning and data science for crime prediction of city of Vancouver, Canada crime data set. The city of Vancouver's official open data portal is where the crime statistics were taken from. Later the dataset is analyzed and pre-processed. All of the results arrived during the analyses of the dataset are shared in this report. Also the results arrived the pre-processing of dataset are also shared in this report. Next in order to predict a model neural networks will be used and using these a model will be made which will predict crime with high accuracy. To train the model we analyzed and pre-processed the data which is already shared in this report. After this we go to the next step which is training and implementation of networks. Here we discuss the 4 network we have trained highlighting their input ,output, epoch cycles ,losses ,accuracy and timeline of crime

CHAPTER 1: INTRODUCTION

The frequency and complexity of the crime events area unit kept increasing. Since crime is neither systematic nor random, it cannot be predicted. Crime analysis sometimes includes procedures to identify the perpetrators of incidents in criminal investigations. The early identification of potential suspects helps security forces make the best use of their human and technological resources and plays a key role in crime prediction. Crimes like burglary and arson have reduced, according to the Crime Records Bureau, while murder, sex abuse, gang rape, and other crimes have soared. The program delivers the result with higher chances of trying it, even though we can't forecast the outcome with 100% accuracy.

Depending on the sort of civilization and community, there are differences in the specifics of how crime is committed. Studies on crime prediction in the past have discovered that elements including education, poverty, employment, and climate have an impact on the crime rate. One of Canada's most populous, racially and culturally diverse, and urban cities is Vancouver. Although Vancouver's general crime rate decreased 1.5% in 2017, a problem with high vehicle break-ins and thefts still exists[1].

After the Vancouver Police Department (VPD) deployed a crime predictive model to forecast crimes involving property break-ins, the number of home break-ins in the city of Vancouver fell by 27%. A strategy used by law enforcement to identify crimes that are most likely to occur is called "crime prediction."

Around the world police departments from different regions invest large amounts of money in finding ways to discover crime trends, uncover potential crime plans and develop better policing techniques. In the early days of crime prediction and technology, this was mainly done by observing historical data to find common trends in crime over years, months or even days. Apart from this a lot of undercover officers were used to patrol and be on the lookout for suspicious on goings in the city. These methods were often very expensive and ignored certain factors that affect crime. Crime itself is unpredictable in nature when viewed on its own and it is shown to be dependent on various different factors such as weather, location ,social and economic factors [1] When prediction was done using historical data these factors were ignored and therefore the models did not perform well. With the advancement of deep learning and high-powered computers a new method of crime prediction became more viable, a method that was capable of finding links between various other factors and crime. However, crime prediction is a field that has not received the same level of attention from deep learning as other fields like computer vision, generative modeling etc. Deep neural networks are designed to reduce the need for extensive feature engineering and allow for training over large datasets. The deep entanglement of crime, multiple variables and it's dependency on spatial and temporal factors (location and time of day) make it an ideal candidate for prediction with a deep neural network.[2].

1.1 Problems in the existing system

Various researchers have addressed the problems regarding crime control and have proposed different crime-prediction algorithms. The accuracy of prediction depends on the attributes selected and the dataset used as a reference.

Crime hotspots in London, UK, were predicted using human behavior data generated from mobile phone network activity along with demographic data derived from actual crime data. Weka, an open-source data mining programe, and 10-fold cross-validation were used to compare Decision Tree and Naive Bayesian, two classification algorithms.

The 1990 US Census, 1990 US LEMAS survey, and the 1995 FBI UCR were used to create the socioeconomic, law enforcement, and crime datasets for this study. Various contextual factors, including the driver, weather, vehicle, and road conditions, were taken into consideration when examining the patterns of traffic accidents in Ethiopia. On a dataset of 18,288 accidents, three different classification algorithms—KNN, Naive Bayesian, and Decision Tree—were applied. All three models' prediction accuracy ranged from 79% to 81%.

Accurate and effective analysis of huge crime datasets is a significant obstacle in crime prediction. Data mining is used to swiftly and effectively uncover hidden trends in huge crime datasets. the improved effectiveness and decreased inaccuracies of criminal data mining methods improve the predictability of crime. Based on the knowledge gained from the University of Arizona's Coplink project, a general data-mining framework was created.[3]

The majority of research on crime prediction focuses on locating crime hotspots, or places where crime rates are higher than average. The authors of conducted a comparison of Risk Terrain Modeling and

Kernel Density Estimation (KDE) (RTM) employing limited data, techniques for developing hotspot maps and proposed region-specific forecasting models. For the purpose of predicting crime hotspots, a spatial-temporal model based on histogram-based statistical techniques, Linear Discriminate Analysis (LDA), and KNN were used. An technique for crime occurrence scanning was used to develop an improved Artificial Neural Network (ANN).to foretell Bangladesh's crime hotspots using the Gamma test. a data-driven machine-learning technique based on spatial analysis, visualization, and broken-window theory methods were applied to assess Taiwanese drug-related crime data and forecast new hotspots [4].

1.2 Methodology

Building a model that can make predictions is done through predictive modeling. A machine learning algorithm is used in the procedure, and it learns specific properties from a training set to generate those predictions from the dataset.

The two subfields of predictive modeling are regression and pattern categorization. In order to forecast the values of continuous variables, regression models are built on the investigation of relationships between variables and trends. The goal of pattern classification, in contrast to regression models, is to assign discrete class labels to a specific data value as an output of a prediction. A pattern classification problem in weather forecasting could be the prediction of a sunny, wet, or snowy day. This is an example of a classification model. Pattern classification tasks can be divided into two parts, Supervised and unsupervised learning. In supervised learning, the class labels in the dataset, which is used to build the classification model, are known. In a supervised learning problem, we would know which training dataset has the particular output which will be used to train so that prediction can be made for unseen data[5]

1.3 Introduction to the project

Crime prediction is crucial to society's efforts to reduce crime by assisting law enforcement organisations in developing the best possible patrol plans. Numerous social benefits will result from fewer criminal incidents. Both public safety and economic damage will increase as a result. However, predicting criminal activity is a difficult endeavor. Crime incidents vary in their spatial and temporal distribution depending on the nature. In Vancouver we can see the differences in the spatial distribution of three main categories of criminal activity, namely theft, drug offences, and assault. The likelihood that a specific sort of criminal occurrence will occur in a place in the near future depends on a variety of factors. Demographics and the distribution of various types of services, crime history, human mobility and so on.

Our goal is to pinpoint the areas in a given city with R regions where a specific kind of crime occurrence will occur throughout the upcoming period of time. Various crime-related events are studied, including theft, unauthorised entry, drug offence, offence involving traffic, fraud, and assault. Theft is a crime that comprises removing property belonging to another person without their permission in order to deprive the owner permanently or temporarily. Unlawful entry is when someone enters a structure (such an office, bank, or store) with the intent to commit a crime. Any type of illegal drug or substance sale, dealing, import or export, production, or cultivation is considered a drug offence. Traffic-related offences include those that pertain to the majority of types of road traffic, such as those that involve car license, registration, roadworthiness, or use, bicycle offences, and pedestrian offences. Fraud, according to the Queensland Police, is a sort of conduct that is dishonest, corrupt, or unethical toward a person or an organisation. Assault is the legal term for any act that causes physical or emotional injury to another person. All physical interactions with an individual without their permission fall under this category. In this study, with an aim for short-term crime event prediction we partition a day into total eight intervals and each interval span 3 hours. Crime prediction in finer temporal grain will help the police to design their patrol strategy dynamically and it will increase the probability to reduce crime rate more effectively.

CHAPTER 2: LITERATURE SURVEY

2.1 Machine Learning

2.1.1 Introduction

Machine learning algorithms are a part of artificial intelligence that enable systems or software programmes to become intelligent enough to forecast outcomes and get more accurate without explicit instructions. The fundamental principle behind these algorithms is that they take in input data in the form of text or images and train the system or model using statistical inputs to recognize or predict the output. The outputs may even be updated as fresh data becomes available. The program must scan the dataset for patterns or resemblances before altering or adjusting the system as necessary [6].

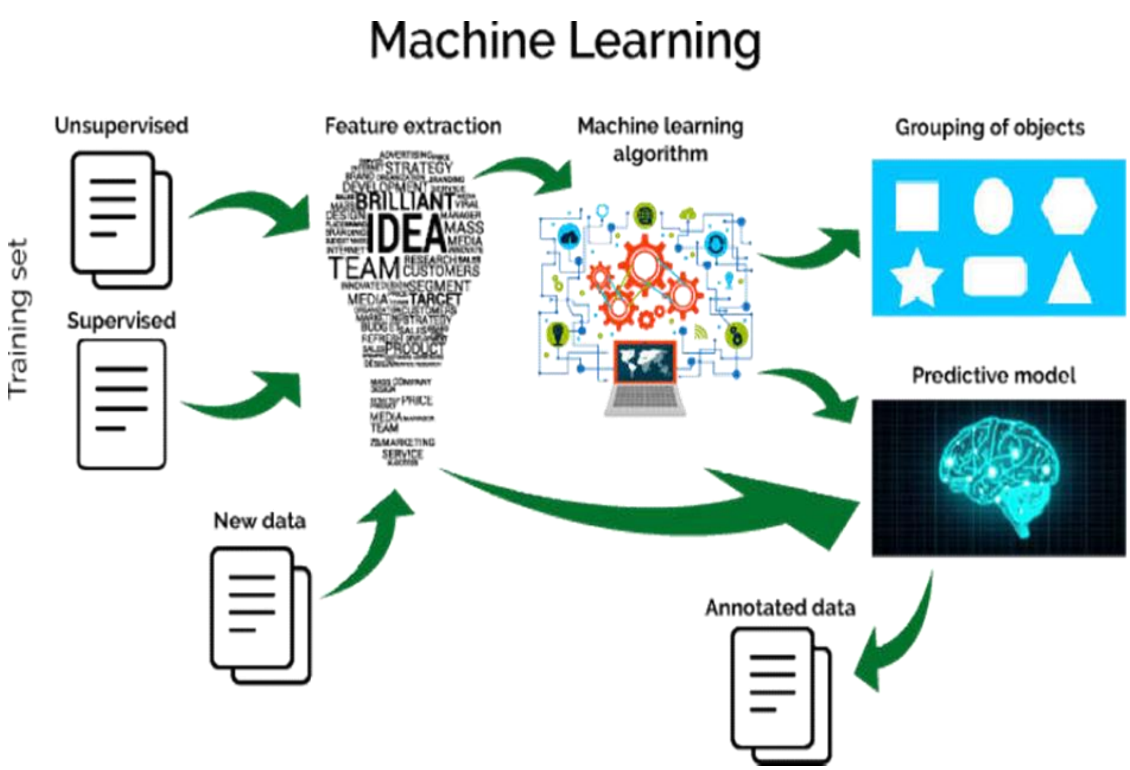


Figure 2.1: Machine Learning[1]

2.1.2 Working of Machine Learning.

The input dataset, which can include photographs, text, tables, and other types of data, is where the machine learning process begins. Additionally, a variety of predefined machine learning techniques are applied to the input data in order to forecast the output and provide acceptable results. These algorithms either classify the input data into groups or look for patterns within the dataset. supervised and unsupervised learning algorithms are two categories for machine learning algorithms [6].

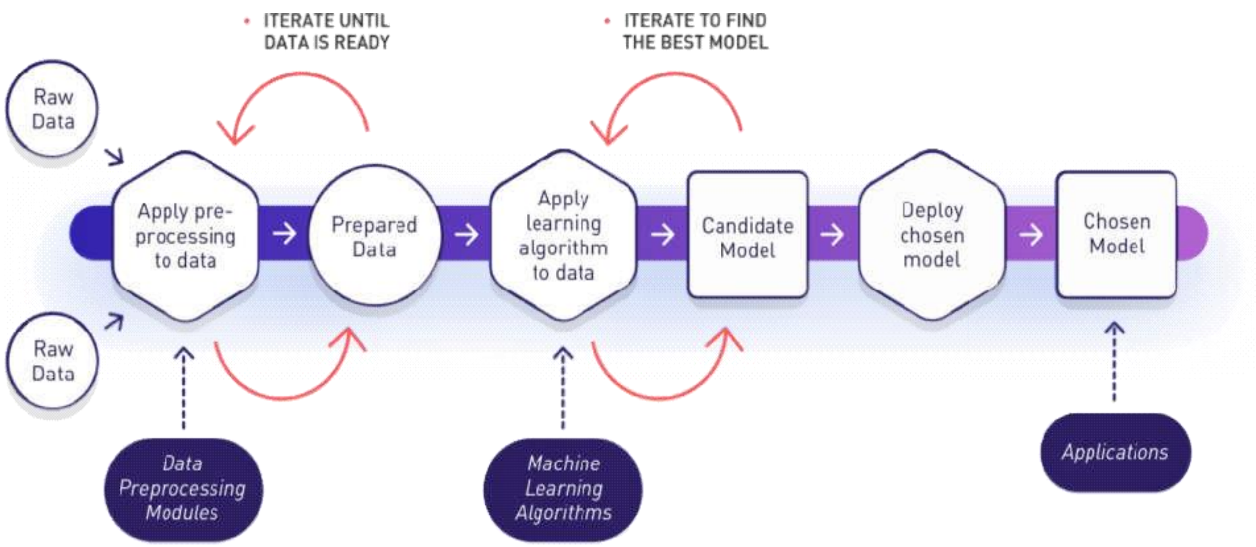


Figure 2.2: Workflow of Machine Learning Algorithm[1]

2.1.3 Types of Machine Learning

Supervised Machine Learning

These algorithms are effective for datasets that have already been educated by past outputs and outcomes utilizing labeled data to forecast the outcome of fresh data. In this instance, the algorithm analyses the known dataset and then generates an inferred function can aid in predicting the values of new data's output. In order to identify faults and be able to correct them and train the model appropriately, it can also analyse the data and the seven outcomes and compare them with the previously stored data.

Unsupervised Machine Learning

This form of machine learning algorithm differs from supervised machine learning algorithms in that the latter are employed when the model has not been trained prior to being classed or labeled. While removing outliers, unsupervised learning methods enable the system to infer a hidden structure or pattern in the unlabeled information and anticipate potential outcomes using such patterns.

Semi-supervised Machine Learning

The benefits of both supervised and unsupervised machine learning algorithms are combined in semi-supervised machine learning algorithms, which yield more effective and potent classifiers. In these kinds of algorithms, the model trains using both labeled and unlabelled data, and it often needs a small amount of labeled data and a lot of unlabelled data that are used simultaneously. This is frequently employed with data that needs both expert and relevant sources for training and learning from it because it helps the model's accuracy and prediction abilities.

Reinforcement Learning

By executing actions and observing the outcomes of those actions, an model learns how to behave in a given environment via reinforcement learning. It is a feedback-based machine learning technique. The agent receives compliments for each positive activity, and is penalized or given negative feedback for each negative action. In contrast to supervised learning, reinforcement learning uses feedback to autonomously train the model without the use of labeled data .The model can only learn from its experience because there is no labeled data.

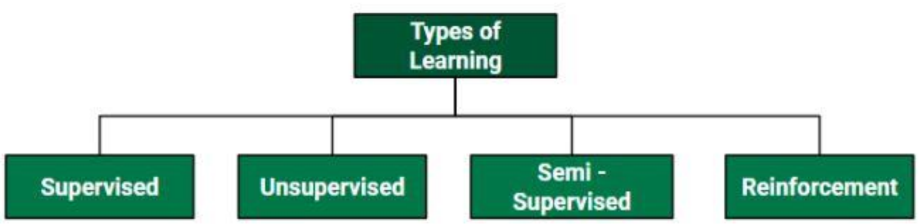


Figure 2.3: Types of Machine Learning [3]

2.2 NEURAL NETWORK

2.2.1 Artificial Neural Network

Artificial Neural Networks, often known as ANNs, are a paradigm for information processing that draws inspiration from how the biological nervous system, including the brain, processes information. It is made up of numerous, intricately linked processing units (neurons) that collaborate to address a particular issue.

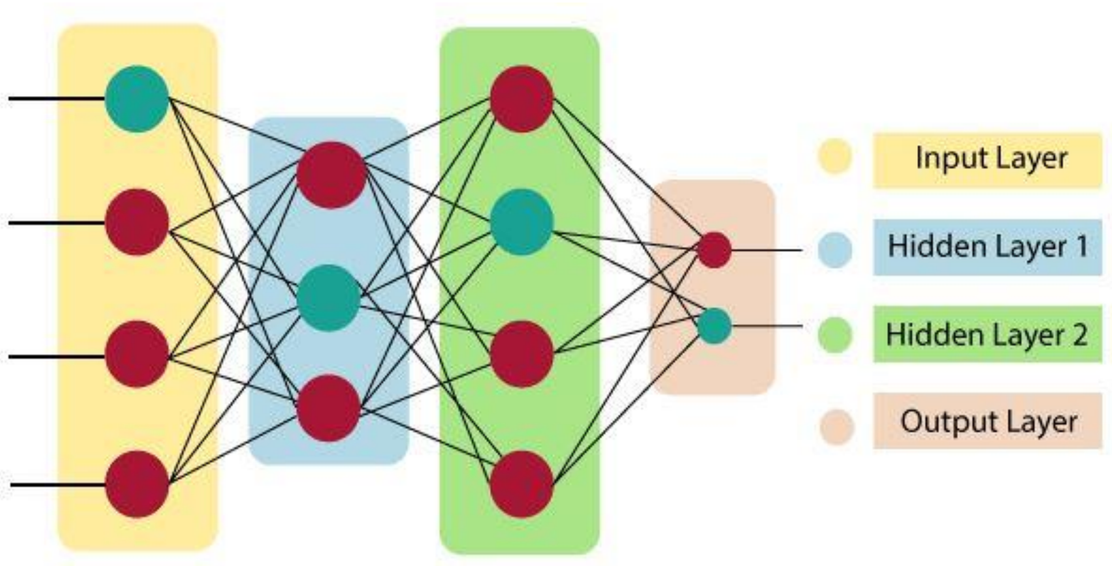


Figure 2.4: Artificial Neural Network[4]

2.2.2 Convolution Neural Network

A Convolution Neural Network (ConvNet/CNN) is a Deep Learning method that can take in an input image, give various elements and objects in the image importance (learnable weights and biases), and be able to distinguish between them. Comparatively speaking, a ConvNet requires substantially less preparation than other classification techniques. ConvNets can learn these filters and attributes, whereas in basic approaches filters are hand-engineered, with adequate training.

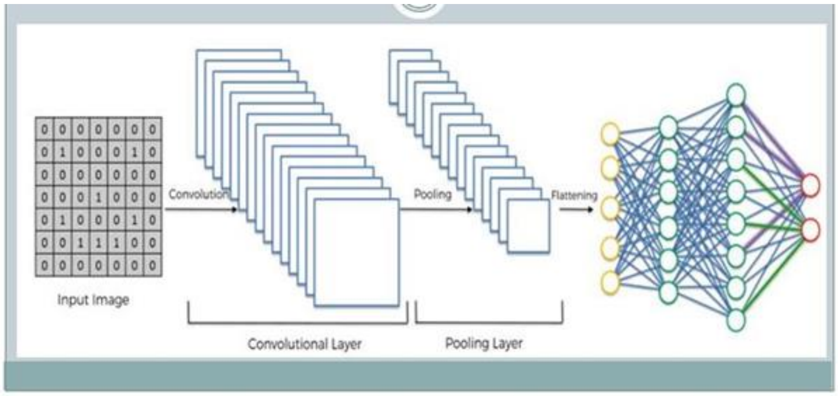


Figure 2.5: Convolution Neural Network[4]

2.2.3 Recurrent Neural Networks

Recurrent neural networks are capable of remembering the past and using what they have discovered to inform their decisions. RNNs remember what they have learned from earlier inputs while producing output, even though they learn similarly during training (s). It belongs to the network. RNNs can receive one or more input vectors and output one or more vectors, with the outputs modified not just by weights given to the inputs, as in a conventional NN, but also by a "hidden" state vector indicating the context based on earlier input(s)/output (s). Therefore, depending on earlier inputs in the series, the same input could result in a different output.

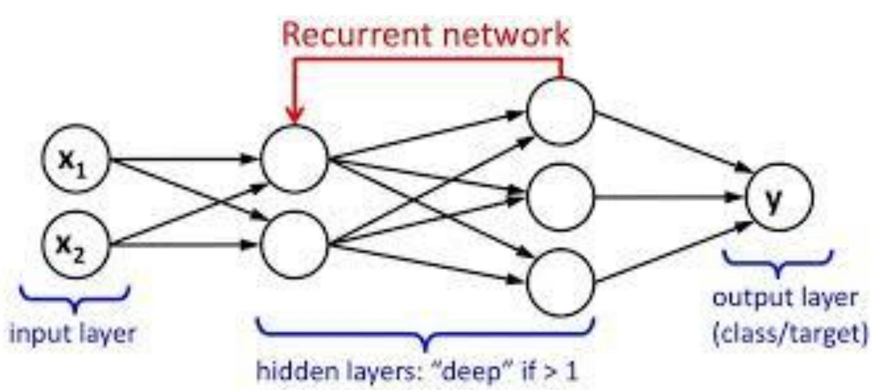


Figure 2.6: Recurrent Neural Network[5]

2.3 ACTIVATION FUNCTION

An ANN needs the activation function to learn and comprehend anything really complex. Their primary function is to transform an input signal into an output signal for a node in an ANN. The subsequent layer of the stack receives this output signal as an input. By calculating the weighted total and then adding bias to it, the activation function determines whether or not a neuron should be stimulated. The goal is to give a neuron's output some non-linearity.

2.3.1 Sigmoid Activation Function (Logistic function)

A sigmoid function is a mathematical function that has a distinctive "S"-shaped curve or sigmoid curve that ranges between 0 and 1. Because of this, it is utilized in models where the output must be a probability prediction This function has the downside of having the potential to cause the neural network to become stuck during training if significant negative input is given.

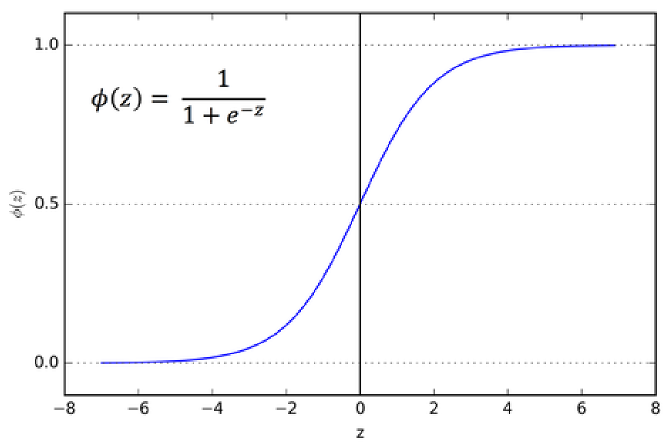


Figure 2.7: Logistic Function[9]

2.3.2 Hyperbolic Tangent Function(tan h)

Hyperbolic Tangent Function is comparable to the Sigmoid but its performance is better because of its nonlinear nature, we can pile layers. The function encompasses (-1,1). The key benefit with this function is that only inputs with zero values are mapped to outputs that are close to zero, while strongly negative inputs will result in a negative output. Therefore, training is less likely to become stuck.

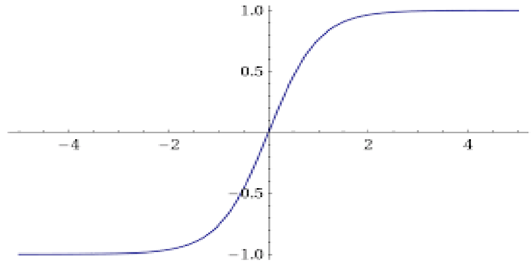


Figure 2.8: Hyperbolic Tangent Function[12]

2.3.3 RmsProp optimizer:

An optimization algorithm/method created for Artificial Neural Network (ANN) training is called RMSProp, or root mean square propagation. The vertical oscillations are limited by the RMSprop optimizer. As a result, we can speed up learning and our algorithm will converge more quickly with greater horizontal steps.

2.4 LOSS FUNCTION

When a prediction or collective group of predictions is provided alongside a label or set of labels, a function known as the cost function, also known as the loss function, IT is used to determine how well the neural network performed. The mean squared error is the most straightforward and often utilized cost function in neural networks out of all the ones that are accessible. Finding the appropriate weights and biases that minimize the cost/loss function is the ultimate goal of training neural networks. We employed an algorithm known as the gradient descent technique for this approach.

2.4.1 Binary Cross entropy

When making yes-or-no decisions, such as when classifying items using multiple labels, the loss function binary cross entropy is applied. The loss reveals the accuracy of your model's predictions. For instance, the model tries to determine if an example belongs to each class in multi-label issues where an example can have numerous labels at once.

$$L(y,\hat{y}) = -\frac{1}{N}\sum_{i=0}^{N}(y * log(\hat{y}_i) + (1-y) * log(1-\hat{y}_i))$$

Figure 2.9: Equation

CHAPTER 3: SYSTEM DEVELOPMENT TOOLS

The algorithms that are being implemented in this project requires some generic system as it requires processing of algorithms.

- Windows 10 (64-bit)
- ANACONDA
- Python
- 4 GB RAM
- Intel(R) Core(TM) i3-3120M CPU @ 2.50 GHz

3.1 WHY PHYTON

Python is a popular programming language that is simple to comprehend and can be quickly read. Additionally, Python provides a variety of packages that simplify even the most complex algorithms or projects. Python offers libraries for practically any file type that are able to be utilised, such as those for working with text, pictures, and audio files. Python is highly adaptable even when working with a new operating system. Due to the Python community's size, getting assistance and advice is much easier.

3.2 WHY ANACONDA?

Anaconda is widely recognised since it comes with all the libraries already installed, saving the user the trouble of having to do it manually otherwise. It offers about 100 packages that can be used for statistical analysis, machine learning, or data science.

3.3 WHY SCKITLEARN?

Usually used for machine learning, the Python library Scikit Learn is capable of showcasing a variety of regression, classification, and clustering algorithms.

3.4 WHY PANDAS?

Pandas is a high-performance Python library that is used in open source. This library has tools for data organisation and data analysis and is simple to use. This library is heavily utilised in the academic, business, and industrial sectors.

3.5 WHY MATPLOTLIB/SEABORN?

Seaborn is a Python data visualisation tool. A high-level interface is provided by Seaborn for creating appealing and educational statistics graphics[21].Matplotlib is typically used for simple plotting.Bars, pies, lines, scatter plots, and other visual representations are frequently used in Matplotlib visualisation.

3.6Why KERAS?

Python-based Keras is an open-source neural network library. It can be used with TensorFlow, Microsoft Cognitive Toolkit, R, Theano, or PlaidML as a foundation. It focuses on being user-friendly, modular, and extensible in order to enable quick experimentation with deep neural networks.

CHAPTER 4: DATASET ANALYSIS

4.1 DATASET USED

For the project we obtained all of our data from the city of Vancouver's open data source. Listed below are all the datasets we used in our project; their specific use cases are discussed in detail in further sections

4.1.1 Crime Data

The raw datasets were retrieved from Vancouver's open data repository. For this study, two datasets—crime and neighborhood—are employed. The VPD has been compiling crime data since 2003, and updates it every Sunday morning. It offers details on the sort of crime that was committed as well as the occasion and setting of the offence. The 22 local regions in the city's Geographic Information System are delineated in the neighborhoods dataset (GIS). The neighborhoods dataset is used for map-making in this project, while the crime dataset is utilized for data analysis. The crime dataset [17] was the first dataset we downloaded from the website. The dataset's columns included the following information:

- Type of crime
- Year
- Day
- Month
- Hour
- Minute
- Block of crime
- Neighborhood of crime
- X Co-ordinate of crime in UTM Zone 10
- Y Co-ordinate of crime in UTM Zone 10
- Latitude
- Longitude

	TYPE	YEAR	MONTH	DAY	HOUR	MINUTE	HUNDRED_BLOCK	NEIGHBOURHOOD	X	Y	Latitude	Longitude
0	Other Theft	2003	5	12	16.0	15.0	9XX TERMINAL AVE	Strathcona	493906.5	5457452.47	49.269802	-123.083763
1	Other Theft	2003	5	7	15.0	20.0	9XX TERMINAL AVE	Strathcona	493906.5	5457452.47	49.269802	-123.083763
2	Other Theft	2003	4	23	16.0	40.0	9XX TERMINAL AVE	Strathcona	493906.5	5457452.47	49.269802	-123.083763
3	Other Theft	2003	4	20	11.0	15.0	9XX TERMINAL AVE	Strathcona	493906.5	5457452.47	49.269802	-123.083763
4	Other Theft	2003	4	12	17.0	45.0	9XX TERMINAL AVE	Strathcona	493906.5	5457452.47	49.269802	-123.083763

Figure 4.1: First five rows of data

The entire dataset consisted of 480724 crimes from 2003 to 2017. However, for some of the them their time and location data was missing as it was protected for privacy reasons. This data was essentially useless for us so we eliminated all of these crimes and that left us with a total of 476290 crimes to work with.

4.1.2 Neighbourhood Data

We then downloaded a second dataset from the Vancouver open data catalogue that gave us a list of neighbourhoods in Vancouver [18]. We added 2 new columns to this dataset called 'Latitude' and 'Longitude' and in here we added the center latitude and longitude for each respective neighborhood. This second dataset consisted of the following columns:

- Map ID
- Neighbourhood Name
- Neighbourhood Center Latitude
- Neighbourhood Center Longitude

	MAPID	NAME	Latitude	Longitude
0	SUN	Sunset	49.218650	-123.091376
1	MP	Mount Pleasant	49.263060	-123.099888
2	RP	Riley Park	49.244679	-123.103239
3	CBD	Downtown	49.279255	-123.119137
4	KITS	Kitsilano	49.265663	-123.166947

Figure 4.2: First five rows of neighbourhood data

4.2 ANALYSIS

From the Figure 4.3 we can depict that theft from the vehicle is the most occurring crimes and also the most common in the last 15 years with over 1.5lakhs reported cases till 2017. It can also be seen that there is not a single case being reported for homicide and vehicle collision or pedestrian struck with fatality from the last 15 years.

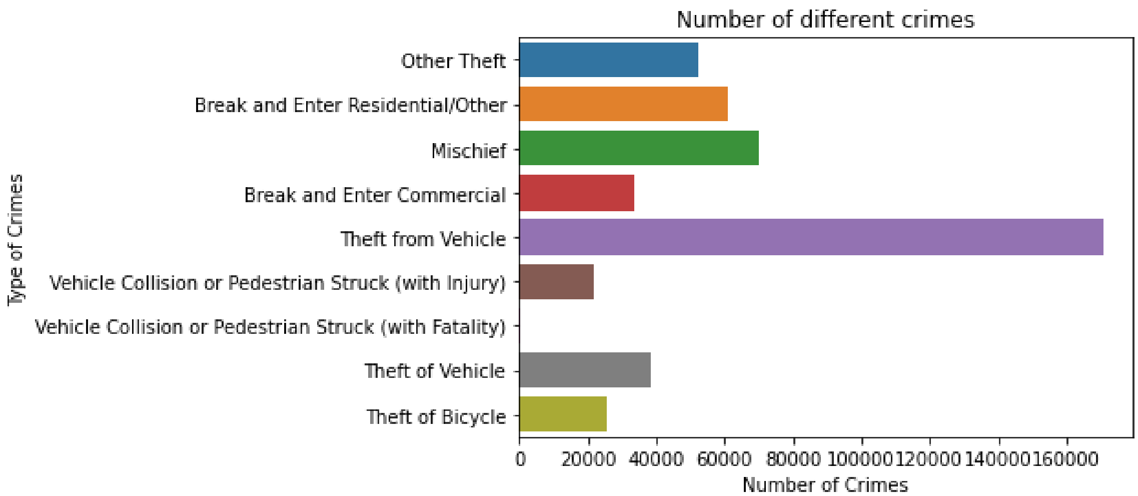


Figure 4.3: No of different crimes

From the Figure 4.4, 2004 was the year with maximum crime case reported. We can see that after that, number of crimes gradually start decreasing till 2011. From 2013 onwards it again starts increasing but rate of increase was low. In 2017, we see a steep fall in the number of cases reported falling under 20,000.

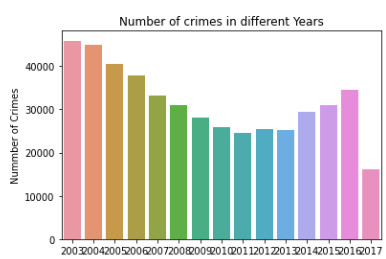


Figure 4.4: Number of Crime per Years**.**

In the Figure 4.5, total number of cases reported in each month in 15 years in shown. We can see that there was almost equal number of cases reported in every month with average cases being around 40000.

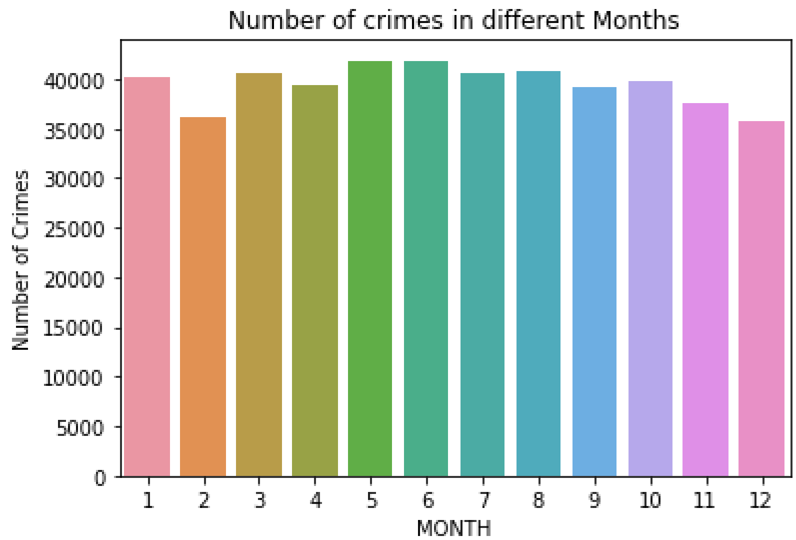


Figure 4.5: Number of crime per month

Figure 4.6, shows number of crime cases reports on each date of every month for 15 years. The last day of every month, month being of 31 days, reported minimum number of cases around 10000. Rest every day average was nearly 16000.

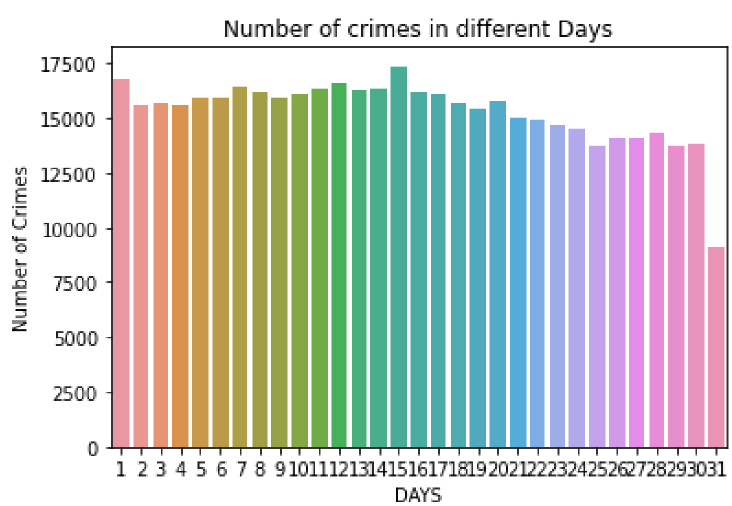


Figure 4.6: Number of crime per Day

The given bar graph fig 4.7 shows that most of the crimes took place during the night hours, with maximum chances being at 6 o'clock in the evening. Morning time as seen from the Figure 4.5, is relatively considered as safe time.

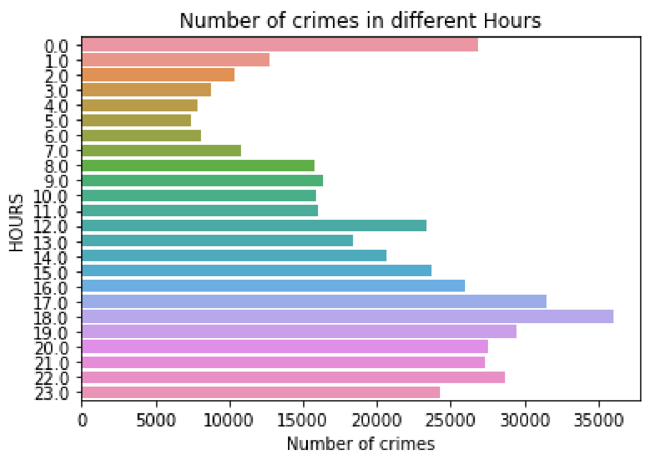


Figure 4.7: Number of crimes per hour

Figure 4.8 shows the number of crimes reported in neighborhood cities of Vancouver. By observing the graph we can conclude that Central Business District can be considered as the most dangerous city among all. The maximum number of cases in Downtown is 90000. The second most dangerous city is West End. Some cities have recorded cases less than 10000 due to which they can be considered as safest among all the neighborhoods of Vancouver.

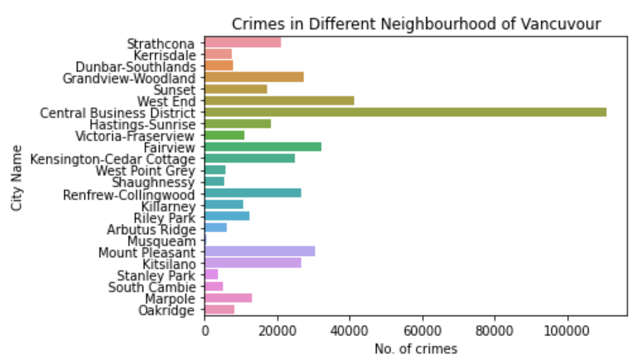


Figure 4.8: Number of crimes in neighborhood of Vancouver

Figure 4.9 consist of a list which also shows the number of crimes in neighborhood of Vancouver.

```
Central Business District    110947
West End                      41352
Fairview                      32161
Mount Pleasant                30536
Grandview-Woodland            27180
Renfrew-Collingwood           26761
Kitsilano                     26699
Kensington-Cedar Cottage      24941
Strathcona                    20919
Hastings-Sunrise              18126
Sunset                        17396
Marpole                       13083
Riley Park                    12521
Victoria-Fraserview           10819
Killarney                     10475
Oakridge                       8037
Dunbar-Southlands              7746
Kerrisdale                     7447
Arbutus Ridge                  6066
West Point Grey                5871
Shaughnessy                    5426
South Cambie                   5212
Stanley Park                   3775
Musqueam                        532
Name: NEIGHBOURHOOD, dtype: int64
```

Figure 4.9: Crimes per neighborhood

CHAPTER 5: DATA PRE-PROCESSING

Data pre-processing is a technique with the help of which we can transform our raw data into something useful. In this case we are having a dataset containing various crimes committed in Vancouver from year 2003 to 2017 and we will be using this dataset to predict crime and help police to in making new and more impactful policing techniques so that the rate of crime can be decreased.

The dataset which is being used to predict crime may contain a lot of garbage values which can make our data bad and we will end up training our machine in a very bad way which can give us wrong outputs for inputs and also become harmful. Here in this case we want to train our model to predict crime so it’s important to have right data so that we can predict crime with precession.

Data pre-processing consists of following steps:

- Data quality assessment
- Data cleaning
- Data transformation
- Data reduction

5.1 DATA QUALITY ASSESMENT:

Data quality assessment is the basic and the first most step for data pre-processing. This includes taking a good look at our dataset and analyzing the quality of dataset. Moreover the dataset should be relevant to our project and also we will have to take care of the consistency of our dataset. Under data quality assessment we look for mixed data values, data outliers and missing data. Every factor is very important and should be considered equally important while pre-processing the dataset.

5.2 DATA CLEANING:

Data cleaning is a part of data pre-processing under which correcting, repairing and removing incorrect or irrelevant data from dataset is done. We searched for all the irrelevant and incorrect data in our dataset like the null values. Figure 5.1 all the values which are found to be null in the dataset.

```
#DATA_PREPROCESSING
crime_data.isnull().sum()
```

```
TYPE              0
YEAR              0
MONTH             0
DAY               0
HOUR              0
MINUTE            0
HUNDRED_BLOCK    13
NEIGHBOURHOOD     0
X                 0
Y                 0
Latitude          0
Longitude         0
dtype: int64
```

Figure 5.1: Null values in dataset

Firstly we stored the data in a variable naming crime data and then we analyzed the dataset with the help of this variable. Now we looked for all the null values in the dataset and then we made a new variable crime_data_1 which stores the relevant information about our project like the day, month, hour year etc. This new variable will be containing only the important information thus removing all the garbage values and making the dataset more efficient and precise for crime prediction. Figure 5.2 contains the initialization of crime_data_1.

```
# getting the data of columns year,month,date,hour,neighbourhood
# which will be used further
crime_data_1=crime_data[['YEAR','MONTH','DAY','HOUR','NEIGHBOURHOOD']]
crime_data_1.shape
```

```
(474028, 5)
```

Figure 5.2: crime_data_1 initialized

5.3 DATA TRANSFORMATION:

Data cleaning had already started the modification of our dataset but data transformation will turn the data into a proper format which is required better analysis of dataset. Under this all our data is combined in uniform format and also the data is normalized into a normalized range so that it can be compared accurately.

In our data pre-processing transformed data by sorting the data by date-time and also we removed duplicates from our new dataset(crime_data_1) which is formed in data cleaning.

.

```
# Sort by date-time and removing duplicates
crime_data_1=crime_data_1.sort_values(by='Date-Time')
crime_data_1=crime_data_1.drop_duplicates()

crime_data_1.head()
```

Figure5.3: Sorting of data and removing the duplicate values

Figure 5.4 shows the all the values we received after sorting the data and removing the duplicates from the dataset.

Date-Time	YEAR	MONTH	DAY	HOUR	Arbutus-Ridge	Downtown	Dunbar-Southlands	Fairview	Grandview-Woodland	Hastings-Sunrise	...	Oakridge	Renfrew-Collingwood	Riley Park	Shaughnessy	South Cambie	Strathcona	Sur
2004-01-01	2004	1	1	0.0	0	1	0	0	0	0	...	0	0	0	0	0	0	
2004-01-01	2004	1	1	0.0	0	0	0	0	0	0	...	0	0	0	0	0	0	
2004-01-01	2004	1	1	0.0	0	0	0	0	0	0	...	0	0	0	0	0	0	
2004-01-01	2004	1	1	0.0	0	0	0	0	0	0	...	0	0	0	0	0	0	
2004-01-01	2004	1	1	0.0	0	0	0	0	0	0	...	0	0	0	0	0	0	

5 rows × 26 columns

Figure5.4: Data received after sorting the data and removing the duplicates from the dataset.

5.4 DATA REDUCTION

In order to make it easy for us to analyze the data we reduce the dataset so that we can analyze it quickly. Also it's not possible for every machine to run a model on very large dataset.

Here we removed the duplicates as shown in figure 5.4 which made our dataset more relevant and also made it easier for our machine to analyze it and also consumed less time.

CHAPTER 6: MODEL MAKING

Till now, we have processed the dataset and we are ready with the final data which will be used for this project. We initially had a dataset which contained the crime records of Vancouver. We removed the null values from it and also we have removed all the duplicates from it. But to predict crime we need to think more practically so, we will be using more datasets. The use of dataset totally depends on the network with which we are dealing. In this project we will be using five neural networks which will have different input parameters and will be used for different scenario's.

For this project, we will be using feed-forward neural networks. We will not be using back-propagation for this project as we don't want the model to correct itself. This project is about predicting the crime so, we will be predicting it with the help of a feed-forward neural network. This network can also be called as a deep neural network because it will be having more than two hidden layers.

Deep neural networks are relatively easy to train and moreover, they consume less time for training. They are designed to reduce the need for extensive feature engineering and they also allow training over a large dataset. This project deals with crime and also it is highly dependent on temporal factors like location and time of day. This makes deep neural networks an ideal neural network for crime prediction.

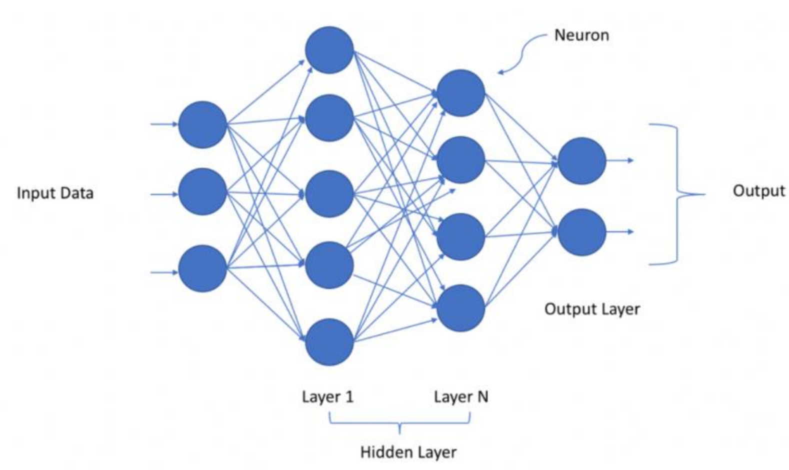


Figure 6.1: Deep neural network

6.1 Network 1

This is our first network in this we will be trying to predict crime without using any complex approach. For this network, we will be using the final dataset which we prepared after data visualization and data pre-processing. We will also be using dataset containing information about distance to graffiti and distance to drinking fountain. We are using these datasets because we wanted to work more practically and we want to achieve higher accuracy. We will be also trying to have a output which can be understood by every person and will not need a machine learning expert to understand the output.

Graffiti dataset contains the information about 8508 points. These points are the exact location of graffiti in Vancouver city. This dataset contains the following columns:

- Latitude
- Longitude
- Map ID
- Latitude
- Longitude
- Name
- Location
- Maintainer
- In Operation
- Pet Friendly
- Photo

Following are the input parameters for network 1:

- Year
- Month

- Day
- Latitude
- Longitude
- Distance to Graffiti
- Distance to drinking fountain

Distance to fountain is another dataset which is being used for this neural network. This dataset contains a list of 241 drinking fountains which are scattered all around the Vancouver city. This dataset contains the following columns:

The following plots shows the accuracy and loss of our network when it was under training.

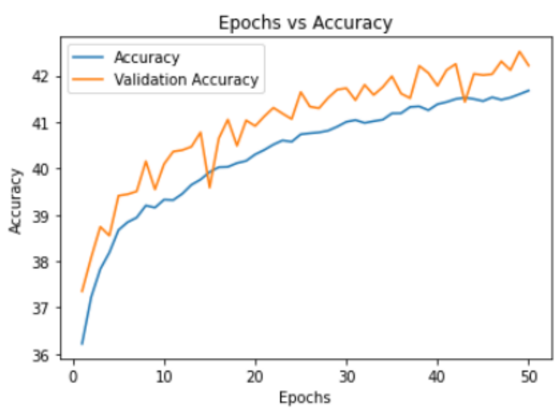


Figure 6.2: Epochs vs. Accuracy for network 1

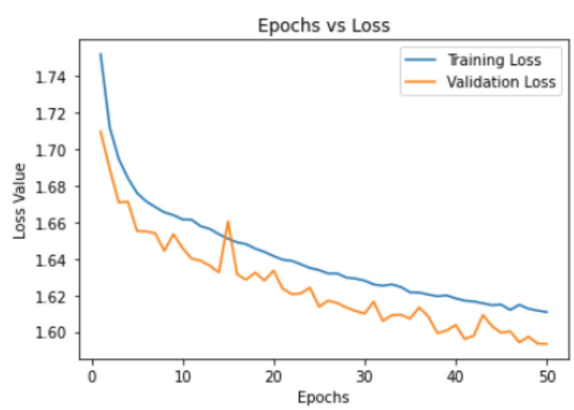


Figure 6.3: Epochs vs. Loss for network 1

```
Epoch 1/50
2017/2017 [==============================] - 10s 5ms/step - loss: 1.7518 - accuracy: 0.3622 - val_loss: 1.7095 - val_accuracy: 0.3735
Epoch 2/50
2017/2017 [==============================] - 10s 5ms/step - loss: 1.7119 - accuracy: 0.3722 - val_loss: 1.6892 - val_accuracy: 0.3808
Epoch 3/50
2017/2017 [==============================] - 11s 5ms/step - loss: 1.6945 - accuracy: 0.3783 - val_loss: 1.6707 - val_accuracy: 0.3874
Epoch 4/50
2017/2017 [==============================] - 11s 5ms/step - loss: 1.6840 - accuracy: 0.3819 - val_loss: 1.6711 - val_accuracy: 0.3855
Epoch 5/50
2017/2017 [==============================] - 11s 6ms/step - loss: 1.6759 - accuracy: 0.3867 - val_loss: 1.6552 - val_accuracy: 0.3941
Epoch 6/50
2017/2017 [==============================] - 12s 6ms/step - loss: 1.6714 - accuracy: 0.3884 - val_loss: 1.6549 - val_accuracy: 0.3944
Epoch 7/50
2017/2017 [==============================] - 13s 6ms/step - loss: 1.6682 - accuracy: 0.3894 - val_loss: 1.6539 - val_accuracy: 0.3951
Epoch 8/50
2017/2017 [==============================] - 14s 7ms/step - loss: 1.6654 - accuracy: 0.3920 - val_loss: 1.6443 - val_accuracy: 0.4016
Epoch 9/50
2017/2017 [==============================] - 15s 7ms/step - loss: 1.6638 - accuracy: 0.3916 - val_loss: 1.6534 - val_accuracy: 0.3955
Epoch 10/50
2017/2017 [==============================] - 14s 7ms/step - loss: 1.6615 - accuracy: 0.3933 - val_loss: 1.6458 - val_accuracy: 0.4010
Epoch 11/50
2017/2017 [==============================] - 15s 7ms/step - loss: 1.6613 - accuracy: 0.3932 - val_loss: 1.6400 - val_accuracy: 0.4036
Epoch 12/50
2017/2017 [==============================] - 15s 7ms/step - loss: 1.6577 - accuracy: 0.3945 - val_loss: 1.6389 - val_accuracy: 0.4040
Epoch 13/50
2017/2017 [==============================] - 15s 7ms/step - loss: 1.6563 - accuracy: 0.3965 - val_loss: 1.6362 - val_accuracy: 0.4047
Epoch 14/50
2017/2017 [==============================] - 16s 8ms/step - loss: 1.6534 - accuracy: 0.3976 - val_loss: 1.6324 - val_accuracy: 0.4078
Epoch 15/50
2017/2017 [==============================] - 16s 8ms/step - loss: 1.6509 - accuracy: 0.3992 - val_loss: 1.6603 - val_accuracy: 0.3959
Epoch 16/50
```

Figure 6.4: Epoch cycles of network 1

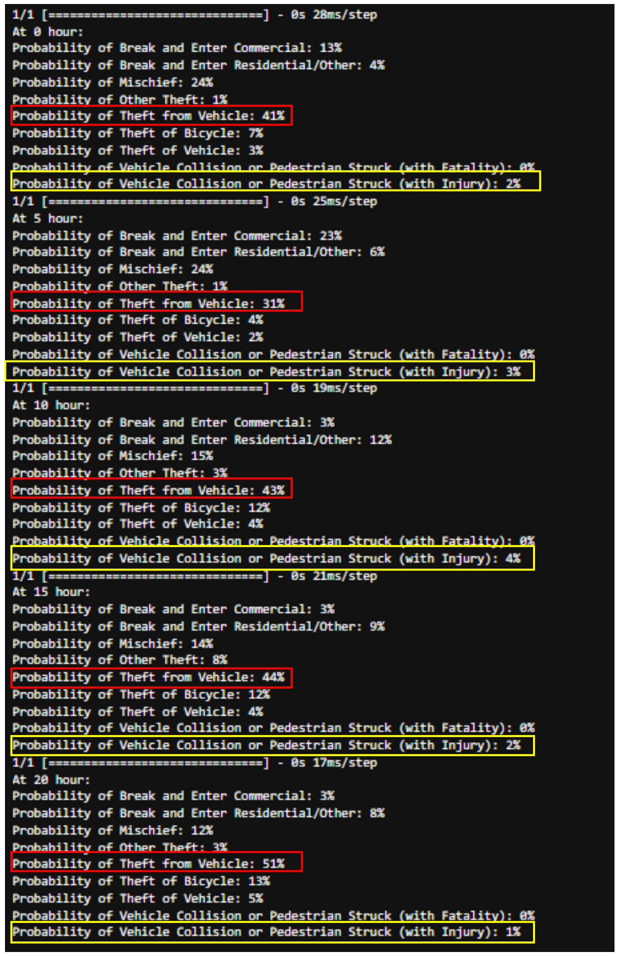
For this network we are having a accuracy of 42.86% and the test loss is 1.61. In this network also we worked with 50 epoch cycles . For the first cycle our accuracy was 36% which got improved to 42.86% by the 50th epoch cycle.

We have achieved an accuracy of 42% which is not at all good but with this network we are able to have our output in the desired form. Following diagram shows the output of this program which completely satisfies our objective but still we need to work on the accuracy of this network.

```
1/1 [==============================] - 0s 28ms/step
At 0 hour:
Probability of Break and Enter Commercial: 13%
Probability of Break and Enter Residential/Other: 4%
Probability of Mischief: 24%
Probability of Other Theft: 1%
Probability of Theft from Vehicle: 41%
Probability of Theft of Bicycle: 7%
Probability of Theft of Vehicle: 3%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 2%
1/1 [==============================] - 0s 25ms/step
At 5 hour:
Probability of Break and Enter Commercial: 23%
Probability of Break and Enter Residential/Other: 6%
Probability of Mischief: 24%
Probability of Other Theft: 1%
Probability of Theft from Vehicle: 31%
Probability of Theft of Bicycle: 4%
Probability of Theft of Vehicle: 2%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 3%
1/1 [==============================] - 0s 19ms/step
At 10 hour:
Probability of Break and Enter Commercial: 3%
Probability of Break and Enter Residential/Other: 12%
Probability of Mischief: 15%
Probability of Other Theft: 3%
Probability of Theft from Vehicle: 43%
Probability of Theft of Bicycle: 12%
Probability of Theft of Vehicle: 4%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 4%
1/1 [==============================] - 0s 21ms/step
At 15 hour:
Probability of Break and Enter Commercial: 3%
Probability of Break and Enter Residential/Other: 9%
Probability of Mischief: 14%
Probability of Other Theft: 8%
Probability of Theft from Vehicle: 44%
Probability of Theft of Bicycle: 12%
Probability of Theft of Vehicle: 4%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 2%
1/1 [==============================] - 0s 17ms/step
At 20 hour:
Probability of Break and Enter Commercial: 3%
Probability of Break and Enter Residential/Other: 8%
Probability of Mischief: 12%
Probability of Other Theft: 3%
Probability of Theft from Vehicle: 51%
Probability of Theft of Bicycle: 13%
Probability of Theft of Vehicle: 5%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 1%
```

Figure 6.5: Output of network 1

We can see that this output is pretty much understandable by everyone and seems to me more interesting as compared to the confusion matrix. This accuracy is bad for this network. This network actually made us motivated and also showed us that we will be able to achieve what we dream of from this project. We can also observe that our project is learning in the right direction. For example, if we look at theft from vehicles then we can see that its percentage is changing with respect to time also its percentage at 15 hours is 44% while its percentage at 20 hours is 51% which is practically true. So, we concluded that we are moving in the right direction and also we are able to make the model understand and work in the right direction. The model is able to distinguish between crimes and can understand that there are more chances of theft from vehicles at 20 hours while the chances of vehicle collision are 1%. On the other hand, the chances of vehicle collision are more at 15 hours. This proves that the model is able to understand the data and can predict. Now, we need to work on the accuracy of our model.



```
1/1 [==============================] - 0s 28ms/step
At 0 hour:
Probability of Break and Enter Commercial: 13%
Probability of Break and Enter Residential/Other: 4%
Probability of Mischief: 24%
Probability of Other Theft: 1%
Probability of Theft from Vehicle: 41%
Probability of Theft of Bicycle: 7%
Probability of Theft of Vehicle: 3%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 2%
1/1 [==============================] - 0s 25ms/step
At 5 hour:
Probability of Break and Enter Commercial: 23%
Probability of Break and Enter Residential/Other: 6%
Probability of Mischief: 24%
Probability of Other Theft: 1%
Probability of Theft from Vehicle: 31%
Probability of Theft of Bicycle: 4%
Probability of Theft of Vehicle: 2%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 3%
1/1 [==============================] - 0s 19ms/step
At 10 hour:
Probability of Break and Enter Commercial: 3%
Probability of Break and Enter Residential/Other: 12%
Probability of Mischief: 15%
Probability of Other Theft: 3%
Probability of Theft from Vehicle: 43%
Probability of Theft of Bicycle: 12%
Probability of Theft of Vehicle: 4%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 4%
1/1 [==============================] - 0s 21ms/step
At 15 hour:
Probability of Break and Enter Commercial: 3%
Probability of Break and Enter Residential/Other: 9%
Probability of Mischief: 14%
Probability of Other Theft: 8%
Probability of Theft from Vehicle: 44%
Probability of Theft of Bicycle: 12%
Probability of Theft of Vehicle: 4%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 2%
1/1 [==============================] - 0s 17ms/step
At 20 hour:
Probability of Break and Enter Commercial: 3%
Probability of Break and Enter Residential/Other: 8%
Probability of Mischief: 12%
Probability of Other Theft: 3%
Probability of Theft from Vehicle: 51%
Probability of Theft of Bicycle: 13%
Probability of Theft of Vehicle: 5%
Probability of Vehicle Collision or Pedestrian Struck (with Fatality): 0%
Probability of Vehicle Collision or Pedestrian Struck (with Injury): 1%
```

Figure 6.6: Output of network 1 with labeling

6.2 Network 2

This will be our model and in this, we will try to increase the accuracy of our project. Till now we have worked hard on the project and we were able to have the output in the desired format. In network 2 we tried to predict the type of crime which is most likely to occur at certain hours of a day at a certain location. But in this network, we will be trying to predict the crime in the neighborhood at a certain hour of the day.

For this network, we will be using the previously final mentioned dataset. The final dataset is pre-processed and is free from null and duplicates which makes it an ideal dataset for all networks.

Before passing the data into the model we have created a crime column in the data. This column indicates that weather a crime happened or not. If the crime happened then the value will be 1 but if not then the value will be 0. Also, if there is no crime at a certain hour then we have added that hour into the dataset, and against it, the value is 0. Let's just assume that there is no crime at 5 hours then we have added that with value 0. This way might help us to improve the accuracy of our model.

Following are the input parameters for this network:

- Year
- Month
- Day
- Hour
- Neighborhood

We expect this network to give us the probability of crime occurring.

The following plots show the accuracy and loss of the model when it was under training.

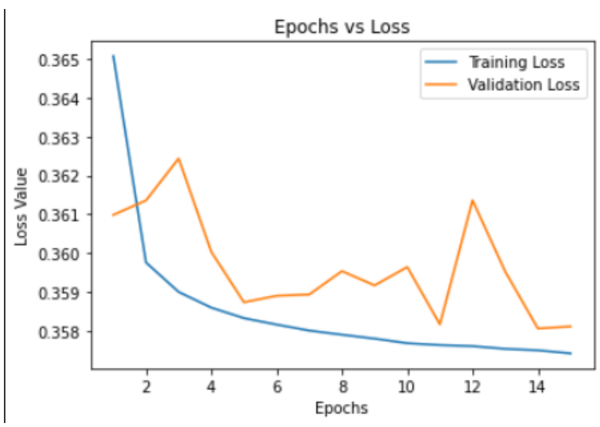


Figure 6.7: Epoch vs. Loss for network 2

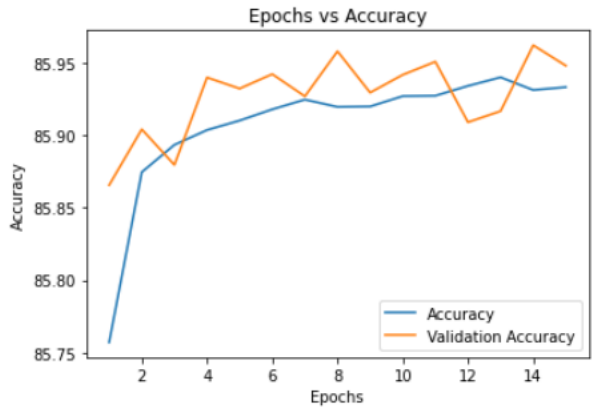


Figure 6.8: Epoch vs. Accuracy for network 2

For this network we are having a accuracy of 85% and the test loss is 0.35. In this network also we worked with 15 epoch cycles. For the first cycle our accuracy was 85.76% which got improved to 85.93% by the 15th epoch cycle. For this network we choose to work with only 15 cycles as we were quite confident that this approach can improve our accuracy also it worked quite well. 50 epoch cycles were consuming a lot of time to train the model but they useful when we start at a low accuracy rate. But that's not the case with this network. It started with 85.76% accuracy which is great.

```
Epoch 1/15
3365/3365 [==============================] - 37s 11ms/step - loss: 0.3651 - accuracy: 0.8576 - val_loss: 0.3610 - val_accuracy: 0.8587
Epoch 2/15
3365/3365 [==============================] - 45s 13ms/step - loss: 0.3598 - accuracy: 0.8587 - val_loss: 0.3614 - val_accuracy: 0.8590
Epoch 3/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3590 - accuracy: 0.8589 - val_loss: 0.3624 - val_accuracy: 0.8588
Epoch 4/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3586 - accuracy: 0.8590 - val_loss: 0.3600 - val_accuracy: 0.8594
Epoch 5/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3583 - accuracy: 0.8591 - val_loss: 0.3587 - val_accuracy: 0.8593
Epoch 6/15
3365/3365 [==============================] - 49s 15ms/step - loss: 0.3582 - accuracy: 0.8592 - val_loss: 0.3589 - val_accuracy: 0.8594
Epoch 7/15
3365/3365 [==============================] - 46s 14ms/step - loss: 0.3580 - accuracy: 0.8592 - val_loss: 0.3589 - val_accuracy: 0.8593
Epoch 8/15
3365/3365 [==============================] - 45s 13ms/step - loss: 0.3579 - accuracy: 0.8592 - val_loss: 0.3595 - val_accuracy: 0.8596
Epoch 9/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3578 - accuracy: 0.8592 - val_loss: 0.3592 - val_accuracy: 0.8593
Epoch 10/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3577 - accuracy: 0.8593 - val_loss: 0.3596 - val_accuracy: 0.8594
Epoch 11/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3576 - accuracy: 0.8593 - val_loss: 0.3582 - val_accuracy: 0.8595
Epoch 12/15
3365/3365 [==============================] - 48s 14ms/step - loss: 0.3576 - accuracy: 0.8593 - val_loss: 0.3614 - val_accuracy: 0.8591
Epoch 13/15
3365/3365 [==============================] - 48s 14ms/step - loss: 0.3575 - accuracy: 0.8594 - val_loss: 0.3595 - val_accuracy: 0.8592
Epoch 14/15
3365/3365 [==============================] - 48s 14ms/step - loss: 0.3575 - accuracy: 0.8593 - val_loss: 0.3581 - val_accuracy: 0.8596
Epoch 15/15
3365/3365 [==============================] - 47s 14ms/step - loss: 0.3574 - accuracy: 0.8593 - val_loss: 0.3581 - val_accuracy: 0.8595
```

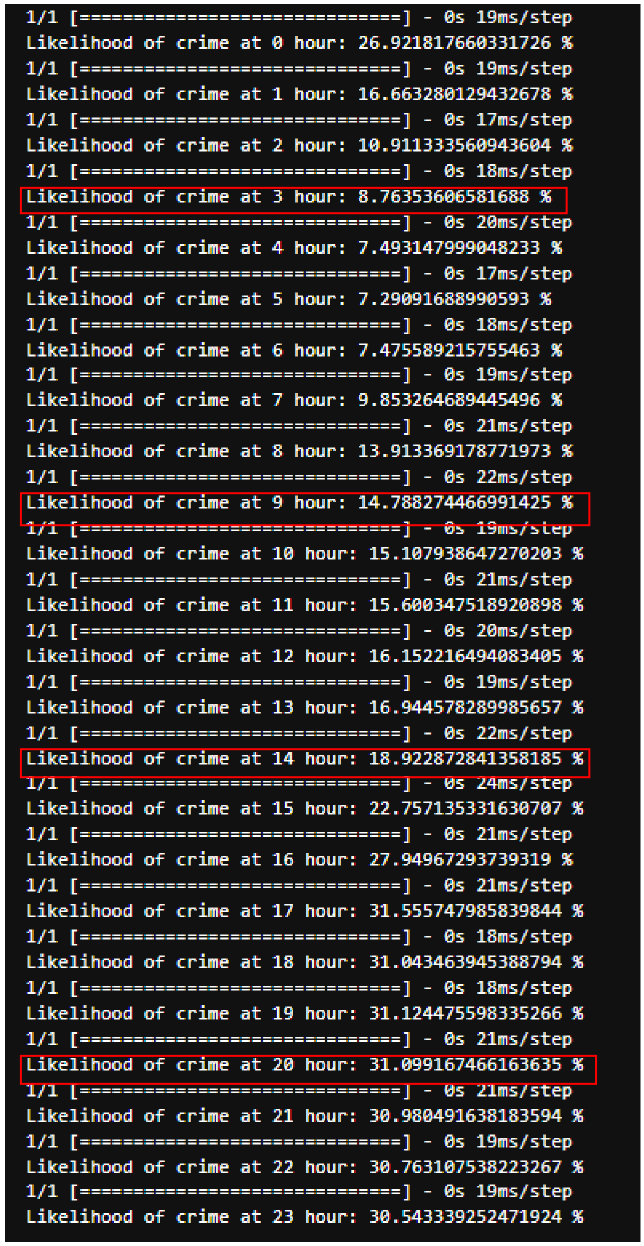
Figure 6.9: Epoch cycles of network 2

Now, coming to the output of this network. Following attached image shows the output of this network.

```
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 0 hour: 26.921817660331726 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 1 hour: 16.663280129432678 %
1/1 [==============================] - 0s 17ms/step
Likelihood of crime at 2 hour: 10.911333560943604 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 3 hour: 8.76353606581688 %
1/1 [==============================] - 0s 20ms/step
Likelihood of crime at 4 hour: 7.493147999048233 %
1/1 [==============================] - 0s 17ms/step
Likelihood of crime at 5 hour: 7.29091688990593 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 6 hour: 7.475589215755463 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 7 hour: 9.853264689445496 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 8 hour: 13.913369178771973 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 9 hour: 14.788274466991425 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 10 hour: 15.107938647270203 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 11 hour: 15.600347518920898 %
1/1 [==============================] - 0s 20ms/step
Likelihood of crime at 12 hour: 16.152216494083405 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 13 hour: 16.944578289985657 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 14 hour: 18.922872841358185 %
1/1 [==============================] - 0s 24ms/step
Likelihood of crime at 15 hour: 22.757135331630707 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 16 hour: 27.94967293739319 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 17 hour: 31.555747985839844 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 18 hour: 31.043463945388794 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 19 hour: 31.124475598335266 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 20 hour: 31.099167466163635 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 21 hour: 30.980491638183594 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 22 hour: 30.763107538223267 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 23 hour: 30.543339252471924 %
```

Figure 6.10: Output of network 2

This is our desired output which we worked for. This output clearly shows the likelihood of crime at a certain hour of day. We can observe that the likelihood of crime at 4 hour is 7.49 which got increased to 15.10 at 10 hour. At night we can observe that the crime increased to 30.76. From this we can conclude that out model is predicting right as these things happen in daily life as well. We have high crime rate in night as compared to early morning or lunch time. This thing about the model can also be observed in the image attached below.



```
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 0 hour: 26.921817660331726 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 1 hour: 16.663280129432678 %
1/1 [==============================] - 0s 17ms/step
Likelihood of crime at 2 hour: 10.911333560943604 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 3 hour: 8.76353606581688 %
1/1 [==============================] - 0s 20ms/step
Likelihood of crime at 4 hour: 7.493147999048233 %
1/1 [==============================] - 0s 17ms/step
Likelihood of crime at 5 hour: 7.29091688990593 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 6 hour: 7.475589215755463 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 7 hour: 9.853264689445496 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 8 hour: 13.913369178771973 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 9 hour: 14.788274466991425 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 10 hour: 15.107938647270203 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 11 hour: 15.600347518920898 %
1/1 [==============================] - 0s 20ms/step
Likelihood of crime at 12 hour: 16.152216494083405 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 13 hour: 16.944578289985657 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 14 hour: 18.922872841358185 %
1/1 [==============================] - 0s 24ms/step
Likelihood of crime at 15 hour: 22.757135331630707 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 16 hour: 27.94967293739319 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 17 hour: 31.555747985839844 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 18 hour: 31.043463945388794 %
1/1 [==============================] - 0s 18ms/step
Likelihood of crime at 19 hour: 31.124475598335266 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 20 hour: 31.099167466163635 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 21 hour: 30.980491638183594 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 22 hour: 30.763107538223267 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 23 hour: 30.543339252471924 %
```

Figure 6.11: Output of network 2 with labeling.

This is now very clear that we can predict crime. Its not necessary that the prediction will be true but we can take security measures based on this prediction. Also, our machine is able to understand the data and performed well. Now we will be working on the model again but with different input parameters to check weather they have any impact on the prediction or not.

6.3 Network 2.1

This network is same as the network 2 but in this we will be working with new input parameters . We want to explore more and as crime is dependent on many factors so we will be including other factors in the input.

Following are the input parameters of this network:

- Year
- Month
- Day
- Hour
- Minute
- Latitude
- Longitude
- Distance from nearest graffiti
- Distance from nearest drinking fountain.

Following plots show the accuracy and loss of this network with respect to epoch cycle.

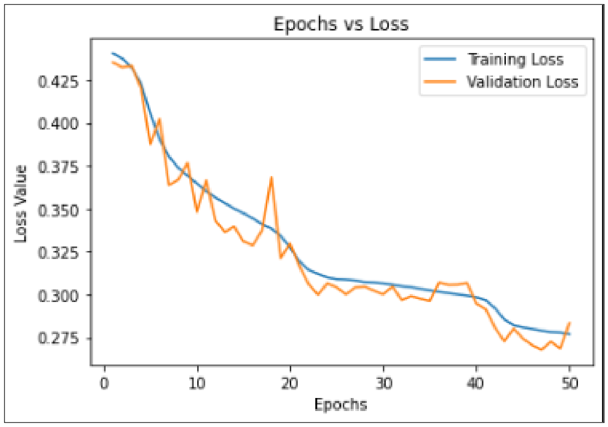


Figure 6.12: Epoch vs Loss for network 2.1

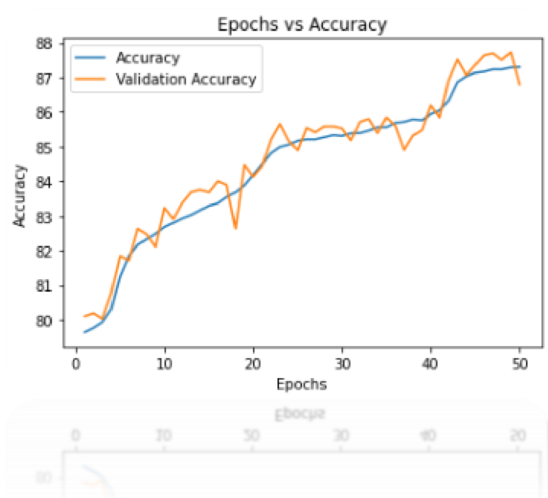


Figure 6.13: Epoch vs Accuracy for network 2.1

Following image shows the output of network on in which we can clearly observe that the output different from the output of network 2. In this network the output ranges is between 78 to 99.

```
1/1 [==============================] - 0s 33ms/step
Likelihood of crime at 0 hour: 98.38404655456543 %
1/1 [==============================] - 0s 40ms/step
Likelihood of crime at 1 hour: 97.9418933391571 %
1/1 [==============================] - 0s 20ms/step
Likelihood of crime at 2 hour: 88.66969347000122 %
1/1 [==============================] - 0s 27ms/step
Likelihood of crime at 3 hour: 78.06049585342407 %
1/1 [==============================] - 0s 25ms/step
Likelihood of crime at 4 hour: 78.37944626808167 %
1/1 [==============================] - 0s 33ms/step
Likelihood of crime at 5 hour: 84.35927033424377 %
1/1 [==============================] - 0s 31ms/step
Likelihood of crime at 6 hour: 91.41652584075928 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 7 hour: 92.79564619064331 %
1/1 [==============================] - 0s 16ms/step
Likelihood of crime at 8 hour: 93.68543028831482 %
1/1 [==============================] - 0s 39ms/step
Likelihood of crime at 9 hour: 94.77003216743469 %
1/1 [==============================] - 0s 25ms/step
Likelihood of crime at 10 hour: 95.76441645622253 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 11 hour: 96.50751948356628 %
1/1 [==============================] - 0s 41ms/step
Likelihood of crime at 12 hour: 97.05377221107483 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 13 hour: 97.48725295066833 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 14 hour: 97.8569507598877 %
1/1 [==============================] - 0s 29ms/step
Likelihood of crime at 15 hour: 98.18381071090698 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 16 hour: 98.47477078437805 %
1/1 [==============================] - 0s 30ms/step
Likelihood of crime at 17 hour: 98.7312376499176 %
1/1 [==============================] - 0s 34ms/step
Likelihood of crime at 18 hour: 98.9533543586731 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 19 hour: 99.14187788963318 %
1/1 [==============================] - 0s 37ms/step
Likelihood of crime at 20 hour: 99.29876327514648 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 21 hour: 99.42710399627686 %
1/1 [==============================] - 0s 37ms/step
Likelihood of crime at 22 hour: 99.5306670665741 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 23 hour: 99.61342215538025 %
1/1 [==============================] - 0s 35ms/step
Likelihood of crime at 24 hour: 99.67915415763855 %
```

Figure 6.14: Output of network 2.1

In this network we can observe that the output having a higher range but its also varying with time. We can observe that the chances of crime at 0 hour are 98.3 which reduces to 78.37 at hour 4. In day time the chances again increase to 95.7 at 10 hour and at 21 hour it is 99.42. Following image justifies this discussion.

```
1/1 [==============================] - 0s 33ms/step
Likelihood of crime at 0 hour: 98.38404655456543 %
1/1 [==============================] - 0s 40ms/step
Likelihood of crime at 1 hour: 97.9418933391571 %
1/1 [==============================] - 0s 20ms/step
Likelihood of crime at 2 hour: 88.66969347000122 %
1/1 [==============================] - 0s 27ms/step
Likelihood of crime at 3 hour: 78.06049585342407 %
1/1 [==============================] - 0s 25ms/step
Likelihood of crime at 4 hour: 78.37944626808167 %
1/1 [==============================] - 0s 33ms/step
Likelihood of crime at 5 hour: 84.35927033424377 %
1/1 [==============================] - 0s 31ms/step
Likelihood of crime at 6 hour: 91.41652584075928 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 7 hour: 92.79564619064331 %
1/1 [==============================] - 0s 16ms/step
Likelihood of crime at 8 hour: 93.68543028831482 %
1/1 [==============================] - 0s 39ms/step
Likelihood of crime at 9 hour: 94.77003216743469 %
1/1 [==============================] - 0s 25ms/step
Likelihood of crime at 10 hour: 95.76441645622253 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 11 hour: 96.50751948356628 %
1/1 [==============================] - 0s 41ms/step
Likelihood of crime at 12 hour: 97.05377221107483 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 13 hour: 97.48725295066833 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 14 hour: 97.8569507598877 %
1/1 [==============================] - 0s 29ms/step
Likelihood of crime at 15 hour: 98.18381071090698 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 16 hour: 98.47477078437805 %
1/1 [==============================] - 0s 30ms/step
Likelihood of crime at 17 hour: 98.7312376499176 %
1/1 [==============================] - 0s 34ms/step
Likelihood of crime at 18 hour: 98.9533543586731 %
1/1 [==============================] - 0s 19ms/step
Likelihood of crime at 19 hour: 99.14187788963318 %
1/1 [==============================] - 0s 37ms/step
Likelihood of crime at 20 hour: 99.29876327514648 %
1/1 [==============================] - 0s 22ms/step
Likelihood of crime at 21 hour: 99.42710399627686 %
1/1 [==============================] - 0s 37ms/step
Likelihood of crime at 22 hour: 99.5306670665741 %
1/1 [==============================] - 0s 21ms/step
Likelihood of crime at 23 hour: 99.61342215538025 %
1/1 [==============================] - 0s 35ms/step
Likelihood of crime at 24 hour: 99.67915415763855 %
```

Figure 6.15: Output of network 2.1 with labeling

6.4 Network 2.2

This is also the same as network 2 but we will be having some different input parameters for this network. For this network we will also be using a Google trends dataset. This dataset contains information about the rate at which word crime is searched in Vancouver city. We used this dataset because it has been found that crime is also dependent on Google searches. To prove this we are going to use this dataset.

Following are the input parameters of this network:

- Year
- Month
- Day
- Hour
- Minute
- Latitude
- Longitude
- Distance from nearest graffiti
- Distance from nearest drinking fountain.
- Google Trend data

Following plots show the accuracy and loss of this network with respect to epoch cycle.

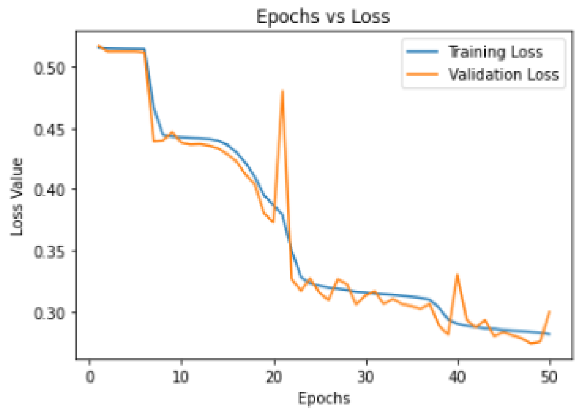


Figure 6.16: Epoch vs Loss for network 2.2

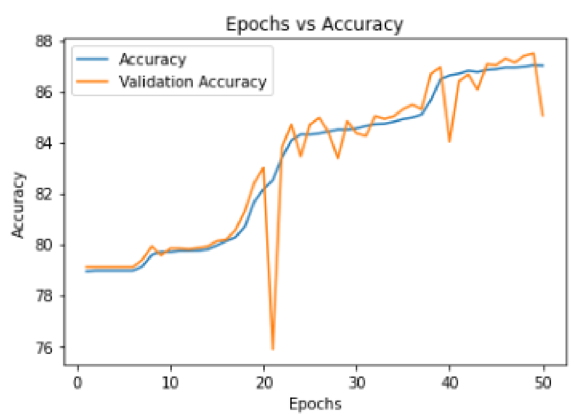


Figure 6.17: Epoch vs Accuracy for network 2.2

For this network we had 50 epoch cycles. In the starting we saw an accuracy of 78% which later got improved to 87%. This network actually proved that Google trend data is having some impact on crime. Searching for crime related activities is not normal and that is why nowadays governments keep track of Google searches. Following images shows the output of our model and from the output only we can observe that its different from our previous networks. In network 3 we had probability in that range of 30 to 50 but for this network the range changes to 89 to 99. We do not have any control on this thing as this totally represents the learning of model but we can observe the data and if we compare it with our learning then we can conclude that the prediction is right.

```
1/1 [==============================] - 0s 76ms/step
Likelihood of crime at 0 hour: 99.46082830429077 %
1/1 [==============================] - 0s 59ms/step
Likelihood of crime at 1 hour: 96.39050960540771 %
1/1 [==============================] - 0s 69ms/step
Likelihood of crime at 2 hour: 89.60691690444946 %
1/1 [==============================] - 0s 88ms/step
Likelihood of crime at 3 hour: 87.88430094718933 %
1/1 [==============================] - 0s 98ms/step
Likelihood of crime at 4 hour: 92.54114031791687 %
1/1 [==============================] - 0s 66ms/step
Likelihood of crime at 5 hour: 96.70429825782776 %
1/1 [==============================] - 0s 75ms/step
Likelihood of crime at 6 hour: 97.59701490402222 %
1/1 [==============================] - 0s 62ms/step
Likelihood of crime at 7 hour: 98.01601767539978 %
1/1 [==============================] - 0s 93ms/step
Likelihood of crime at 8 hour: 98.4180748462677 %
1/1 [==============================] - 0s 73ms/step
Likelihood of crime at 9 hour: 98.73809218406677 %
1/1 [==============================] - 0s 67ms/step
Likelihood of crime at 10 hour: 98.94948601722717 %
1/1 [==============================] - 0s 67ms/step
Likelihood of crime at 11 hour: 99.08435344696045 %
1/1 [==============================] - 0s 80ms/step
Likelihood of crime at 12 hour: 99.17385578155518 %
1/1 [==============================] - 0s 47ms/step
Likelihood of crime at 13 hour: 99.23633337020874 %
1/1 [==============================] - 0s 66ms/step
Likelihood of crime at 14 hour: 99.28176403045654 %
1/1 [==============================] - 0s 90ms/step
Likelihood of crime at 15 hour: 99.31581020355225 %
1/1 [==============================] - 0s 72ms/step
Likelihood of crime at 16 hour: 99.34192895889282 %
1/1 [==============================] - 0s 85ms/step
Likelihood of crime at 17 hour: 99.36236143112183 %
1/1 [==============================] - 0s 62ms/step
Likelihood of crime at 18 hour: 99.37863945960999 %
1/1 [==============================] - 0s 60ms/step
Likelihood of crime at 19 hour: 99.3918240070343 %
1/1 [==============================] - 0s 69ms/step
Likelihood of crime at 20 hour: 99.40269589424133 %
1/1 [==============================] - 0s 65ms/step
Likelihood of crime at 21 hour: 99.41179752349854 %
1/1 [==============================] - 0s 71ms/step
Likelihood of crime at 22 hour: 99.41954016685486 %
1/1 [==============================] - 0s 81ms/step
Likelihood of crime at 23 hour: 99.42622780799866 %
1/1 [==============================] - 0s 100ms/step
Likelihood of crime at 24 hour: 99.43208694458008 %
```

Figure 6.18: Output of network 2.2

Talking about this particular network we are having a very high probability of crime at every hour but the number varies with hours. For 0 hour we are having a probability of 99.46 while this probability changes to 89..6 at hour 2. Also the probability of crime at 10 hour is 98.94 while the probability of crime at 21 hour is 99.41 which is slightly higher than the chances of crime at 10 hour. Following image justifies this discussion.

```
1/1 [==============================] - 0s 76ms/step
Likelihood of crime at 0 hour: 99.46082830429077 %
1/1 [==============================] - 0s 59ms/step
Likelihood of crime at 1 hour: 96.39050960540771 %
1/1 [==============================] - 0s 69ms/step
Likelihood of crime at 2 hour: 89.60691690444946 %
1/1 [==============================] - 0s 88ms/step
Likelihood of crime at 3 hour: 87.88430094718933 %
1/1 [==============================] - 0s 98ms/step
Likelihood of crime at 4 hour: 92.54114031791687 %
1/1 [==============================] - 0s 66ms/step
Likelihood of crime at 5 hour: 96.70429825782776 %
1/1 [==============================] - 0s 75ms/step
Likelihood of crime at 6 hour: 97.59701490402222 %
1/1 [==============================] - 0s 62ms/step
Likelihood of crime at 7 hour: 98.01601767539978 %
1/1 [==============================] - 0s 93ms/step
Likelihood of crime at 8 hour: 98.4180748462677 %
1/1 [==============================] - 0s 73ms/step
Likelihood of crime at 9 hour: 98.73809218406677 %
1/1 [==============================] - 0s 67ms/step
Likelihood of crime at 10 hour: 98.94948601722717 %
1/1 [==============================] - 0s 67ms/step
Likelihood of crime at 11 hour: 99.08435344696045 %
1/1 [==============================] - 0s 80ms/step
Likelihood of crime at 12 hour: 99.17385578155518 %
1/1 [==============================] - 0s 47ms/step
Likelihood of crime at 13 hour: 99.23633337020874 %
1/1 [==============================] - 0s 66ms/step
Likelihood of crime at 14 hour: 99.28176403045654 %
1/1 [==============================] - 0s 90ms/step
Likelihood of crime at 15 hour: 99.31581020355225 %
1/1 [==============================] - 0s 72ms/step
Likelihood of crime at 16 hour: 99.34192895889282 %
1/1 [==============================] - 0s 85ms/step
Likelihood of crime at 17 hour: 99.36236143112183 %
1/1 [==============================] - 0s 62ms/step
Likelihood of crime at 18 hour: 99.37863945960999 %
1/1 [==============================] - 0s 60ms/step
Likelihood of crime at 19 hour: 99.3918240070343 %
1/1 [==============================] - 0s 69ms/step
Likelihood of crime at 20 hour: 99.40269589424133 %
1/1 [==============================] - 0s 65ms/step
Likelihood of crime at 21 hour: 99.41179752349854 %
1/1 [==============================] - 0s 71ms/step
Likelihood of crime at 22 hour: 99.41954016685486 %
1/1 [==============================] - 0s 81ms/step
Likelihood of crime at 23 hour: 99.42622780799866 %
1/1 [==============================] - 0s 100ms/step
Likelihood of crime at 24 hour: 99.43208694458008 %
```

Figure 6.19: Output of network 2.2 with labeling.

From this network we can conclude that Google trend data is having some impact on the crime and also by including this factor we are predicting crime in a more practical way.

CHAPTER 7: CONCLUSION

In this research, we used Vancouver crime data for the last 15 years which was used in two different dataset approaches. Firstly we extracted the crime data from Vancouver's official open data portal and analyzed it using python language. During the analysis of the dataset we got to know about the rate of crime in different parts of Vancouver from 2003 to 2017 and how the crime is varying during all these years considering different hours, days, location, and neighbourhood . Also we made sure that our dataset should be relevant. In order to make our dataset more relevant and increase the precision we pre-processed the dataset and removed all the null values form it. We also transformed the dataset by sorting it according to the date-time. By doing this much work we moved forward towards making the model based on artificial neural networks. After this project, we saw how the machine was able to predict crime with the help of neural networks. The project achieved a maximum accuracy of 87% which is quite good. Our model was able to learn about crime and predicted it with great accuracy. We also used different approaches and used a very practical approach to predict crime. We used different datasets and we also changed the input parameters of networks to study the behavior of crime. In the end, we can conclude that we were able to predict the probability of crime throughout the day. This project clearly shows the potential of neural networks and motivates us to do more research in this field.

CHAPTER 8: FUTURE SCOPE

This project is having a great future. We can obviously work on the accuracy of networks and use more different datasets to predict crime. We can do more research about factors on which crime is dependent and we can include them as well. Talking about the technical advancements in this project, we actually need to develop the frontend side of this project so that users who do not have any knowledge about Python can also use it

Predictive policing: Predictive policing is a concept that uses data analysis and machine learning techniques to identify areas that are at high risk for crime and allocate resources to prevent them. ANNs can play a significant role in this approach by analyzing and predicting crime patterns in real-time

For the purpose of locating the actual criminal, essential manual and digital labour is being put forth. Although not always successful, there is still room for improvement in the prediction system

Overall, crime prediction models have a very broad future application, and improvements in machine learning algorithms and data integration techniques can further improve the precision and efficacy of these models.

REFERENCES

[1] P. Carlen, Women, Crime and Poverty. Open University Press, dec 1988.

[2] A. M. Olligschlaeger, “Artificial neural networks and crime mapping.”

[3] C. Yu, M. W. Ward, M. Morabito, and W. Ding, “Crime forecasting using data mining techniques,” in 2011 IEEE 11th International Conference on Data Mining Workshops, Dec 2011, pp. 779–786.

[4] A. Stec and D. Klabjan, “Forecasting crime with deep learning,” arXiv preprint arXiv:1806.01486, 2018.

[5] “Vancouver open data catalogue.” [Online].Available: https://data:vancouver:ca/datacatalogue/

[6] “Vancouver crime data.” [Online]. Available: https://data:vancouver:ca/datacatalogue/crime-data:html

[7] “Vancouver local area boundary data.” [Online].Available: https://data:vancouver:ca/datacatalogue/localAreaBoundary:htm

[8] “Vancouver graffiti data.” [Online]. Available: https://data:vancouver:ca/datacatalogue/graffitiSites:htm

[9] Y. Khaireddin, “Google trend adder - ele494-project.”[Online].

[10] “Distance from drinking fountain calculator- ele494-project.” [Online]. Available:https://github:com/NasirKhalid24/ELE494-Project/blob/master/Scripts/update drinkingfountain:py

[11] https://www.geeksforgeeks.org/machine-learning/

[12] https://www.section.io/engineering-education/understanding-loss-functions-in-machine-learning/