Detection and analysis of Network Layer Security challenges in cloud using machine-learning Algorithm

MSc Research Project
Cloud Computing

Nilam Choudhari
Student ID: x20154003

School of Computing
National College of Ireland

Supervisor: Aqueel Qazmi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Nilam Choudhari
Student ID:	x20154003
Programme:	Cloud Computing
Year:	2022
Module:	MSc Research Project
Supervisor:	Aqueel Qazmi
Submission Due Date:	31/01/2022
Project Title:	Detection and analysis of Network Layer Security challenges in cloud using machine-learning Algorithm
Word Count:	XXX
Page Count:	27

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	31st January 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Detection and analysis of Network Layer Security challenges in cloud using machine-learning Algorithm

Nilam Choudhari
x20154003

Abstract

Cloud computing is rapidly growing in popularity and use. Many firms are spending their time in this industry, whether it be for personal enrichment or even as a contribution to others. One of the repercussions of Cloud expansion is the increase of various security issues for both businesses and consumers. One way for safeguarding the cloud is by machine learning. Techniques of machine learning were been used in a number of different ways on the Cloud to avoid or discover attacks and also security issues. Security challenges of cloud computing have been investigated at the layer of Infrastructure as a Service (IaaS), since security have emerged as one of the greatest considerations with cloud services in this thesis. Because IaaS involves a wide range of security concerns, the research will be confined to Network Layer Safety. As a consequence, intrusion detection systems are needed to defend cloud services. In our thesis the results shows that multilayer perceptron has been proved best way to classify the attacks as the accuracy is highest. While this is understandable and anticipated, using accuracy as the sole criteria for evaluating a system would be impractical. Another thing to keep in mind is that we must be cautious with false negatives. We probably couldn't afford to get a connection classified as non-harmful when it's actually an attack. We would draw the ROC curve and utilize Area Under Curve as an assessment tool for this.

5v

1 Introduction

Regardless the reality that cloud computing is frequently viewed as a significant and advantageous IT infrastructure transformation, significant security effort is required to address its weaknesses. Because a huge amount of personal and commercial information is stored in the cloud computing environment, cloud security issues and defects must be identified and solved. Because cloud infrastructure uses conventional Network protocols and also virtualization approaches, it may be subject to attacks. With CC, assets may be distributed over a network, letting users to interact with a few of them as required and with variable accessibility. As a result, users' comfort and security are enhanced because they no more need to preserve data with their own. Unfortunately, cloud security is a big concern for cloud users. Because cloud technology is so reliant on user trust, there is concern that businesses will be vulnerable to new threats and risks. Moreover, attackers may breach cloud technology, gaining access to critical data stored in the cloud which is belonged to anyone else. A cloud penetration or assault could have far implications.

Since most invaders are interested in targeting networking with the most customers and also the most strictly controlled public resources, as well as networks holding the most data, they must be extensively investigated and all dangers eradicated whenever possible. According to the Alliance (2017) Cloud Security Alliance (CSA) Alliance, there are seven key dangers to cloud environments (2017). Implementing Intrusion Detection (ID) technologies is indeed the great way to stop attacks and protect networks because these solutions can identify and hence protect against attackers fast. As a consequence, well before benefits can be enjoyed, any concerns regarding cloud security should be addressed. External invader attacks are not really the sole threat to security of cloud. While firewalls generally are excellent in stopping external attackers from penetrating into a cloud environment, Intrusion Detection Systems (IDSs) are more capable at detecting insider attacks. So according Chen and Sion (2010), the costs associated with adopting cryptographic approaches to secure data is not always financially efficient for intrusion detection systems. As a result, employing intrusion detection systems (IDSs) in cloud networks will be a key problem. According to Engen (2010) Engen (2010), work over the previous decade has focused on developing many Machine Learning (ML) algorithms for intrusion detection systems. Most common techniques is one that can train from training samples displaying normal network behavior under varied attacks. IDSs learn to detect breaches through their own, eliminating the requirement for a human to detect the attack. IDSs can detect intrusions by recognizing the common trends and patterns seen in previous, known attacks. According to Engen (2010) Engen (2010), the KDD (Knowledge Discovery and Data Mining) Cup 99 dataset is now one of the greatest and the mostly used datasets for evaluating the performance of the systems or efficiency of intrusion detection systems over the preceding decade. Because there are many distinct types of attacks, researchers employ a range of machine learning algorithms to understand how to recognize these. In prior projects using machine learning inside IDS, researchers employed proven classification algorithms to identify different types of attacks. The majority of this work was completed by scientists who have significant experience with computer systems and related networking security but lacked a fundamental knowledge of machine learning methodologies. As a result, most efforts have essentially used the machine learning approach as a ”resource” to identify attacks, but many essential parts of the utilization and fine-tuning required in employing such tools have yet to receive the attention they require. Users' data is stored and controlled remotely through syatems that are cloud-based, which causes them to be concerned about losing integrity of the information. As a consequence, the majority of the solutions proposed in the literature pertain around assisting in the secrecy, truthfulness, availability, verification, authorization, responsibility, and secrecy of client data through the use of numerous cryptographies, interference prevention and control, and quality assurance processes.

1.1 Research Question

By building system of cyber-attack detection, how effective is the Machine Learning model in autonomously detecting the intrusions in network and revealing shortcomings of cloud computing security? What are the potential methods for balancing the dataset that is imbalanced with the use of classifiers of machine learning in order to improve accuracy rate of performance? What are all the parameters for calculating the reliability of small attack detection and the influence of the existence of classifications?

1.2 Structure of the Document

In the paper, we will understand methods and discuss various results. In the 2nd section we will find different papers and we will discuss the methodology of those papers and also its results. This section will let us know different techniques that were used earlier. Section 3 will introduce the proposed methodology in detail.This section will have few subsections which explain the process of implementation and Machine learning algorithms used. Section 4 will discuss the implication of the paper, limitation of methodology.

2 Related Work

Security of Cloud computing has been a major issue in recent years. Because the issue is so important, research on this topic in the previous years has already been completed. This section of the literature review will cover a variety of previously completed experiments.

2.1 Cloud computing overview

The most basic element of the Cloud is its component-centered nature, which is identified as providing a range of different advantages, including Customizability, extensibility, reusability, scalability, and substitutability are all important considerations. Alternate solution adoptions, runtime element replacements, and dedicated connections are all included. Which is been emphasized in the work of Mladen Vouk (2008) M.Vouk et al. The study of Mladen Vouk (2008)M. Vouk et al. (2008) examined the key importance of Cloud Computing, which therefore founded the key differences discernible further when comparing Grid Computing with the Cloud. In Armbrust et al. (2010) M. Armbrust et al.(2010), they have tried clarify terms to reduce the confusion, Easy figures were provided to measure and understand cloud and conventional computing differences. They have tried to identify the hurdles that are both the technical ones as well as non-technical once. They have also mentioned many opportunities of the cloud computing field. Various definitions were provided to clear the concepts of cloud computing in the field of computer science. The definition of the cloud computing by M. Armbrust and Rh (2009) M. Armbrust and Rh(2009) framework implemented by IT that is virtualization centred in which various resources are applied via internet that includes different applications,resources such as data and also resources based on infrastructure,distributed as a service by one or many different providers.According to the study by M. Armbrust and Rh (2009) M. Armbrust and Rh(2009) these services are scalable,can be used on demand and pay as you go services. In relation to same topic Zhang et al. (2010)Q. Zhang and R. Boutaba (2010) that Cloud computing makes use of technology based on virtualisation in order to meet the goal of delivering resources related to computing as a precious function. When comparing Autonomic Computing technology, grid Computation, and the Cloud, a many different elements are acknowledged as equivalent; however, there are a number of differences between the three.

The world of computation has grown enormously large and complex. Cloud computing is a new technology in the world of IT. Cloud computing provides IT skills in the form of services.These services which are based on cloud are available on the demand, they are scalable, device agnostic, and trustable. The concept of virtualization underpins cloud computing. Virtualization does the separation of the hardware from the software and provides advantages such as multiple servers and real - time migration. In this paper Dayalan (2019) Dayalan,Mutthu (2019), they have provided an overview of cloud computing technology and discussed its innovative products, ”virtualization.” In addition, we present research challenges in cloud computing.

2.2 Cloud computing Security

In direct comparison to more traditional computing, work by Buyya et al. (2013). Buyya, C. Vecchiola, and S. T. Selvi (2013) presented the debate that Cloud Computing offers a considerable number of extra security issues when compared to more traditional computing. Others, such as Sadiku et al. (2014) M. N. O. Sadiku, S. M. Musa, and O. D. Momoh (2014), recognize Cloud security from the standpoint of data security, recognizing that any organization takes priority and strongly values data among its key, most basic assets. As a result, various works, including those by R. Buyya, C. Vecchiola, and S. T. Selvi (2013), M. N. O. Sadiku, S. M. Musa, and O. D. Momoh (2014) propose that the Confidentiality, Integrity, and Availability (CIA) of data be assured by providers of these kind of services; this will help further to enable Cloud security. . Furthermore, the variety of limitations and designs between Cloud service providers and consumers results in Cloud weaknesses Z. Erl (2015) Z. Erl, T., Puttini, R. Mahmood(2015).

2.2.1 Requirements of intrusion detection systems in Cloud

A number of critical attacks have an impact on the availability, confidentiality, and integrity of Cloud resources and services. These intrusions can take any form of a backdoor stream, flooding, insider, user root, or virtual machine attacks. There is indeed a recommendation that the network is the ultimate foundation of the Cloud, which means that any network weak spot can affect overall security of Cloud, as stated in Modi et al. (2013) C. Modi and C. N. Modi and Rajarajan (2012) M. Rajarajan(2013), C. N. Modi and and M. Rajarajan(2012) . As a result, Detection of network intrusion is regarded as one of the most crucial issues of the security in the Cloud technology. network attacks such as DNS toxicity, DoS or DDoS attacks, insider attacks, IP spoofing, man-in-the-middle, and port scanning, as discussed in C. Modi and M. Rajarajan(2013), C. N. Modi and and M. Rajarajan(2012). The detection and prevention of intrusions and attacks is one of the most difficult security challenges in cloud computing. To detect and prevent malicious network layer activities,the paper has suggested us a security framework that combines a network intrusion detection system (NIDS) into the infrastructure of the cloud. This framework is implemented using snort and classifier called Bayesian machine learning techniques. To verify our approach, we use KDD experimental intrusion datasets to assess the efficiency and detection efficiency of our NIDS. The results demonstrate that the model proposed here has a higher rate of detection with limited false positives at a low cost of computation. Patel and Srivastava (2013) K. Patel and R. Srivastava(2013)

2.3 Overview of IDs

2.3.1 Defination related to IDS

Attack detection identifies intrusion indicators in accordance with incident supervision and examining what is observed to take place across ICT systems. Nevertheless, attacks are reported to be toxic to or infraction of security features or components of the system where availability, confidentiality, and integrity are jeopardized. In the Network Attacks are carried out either in or out,In intruders widely etract and misuse the permissions bestowed upon them, and out attackers gaining access to the device through the web. Intrusion Detection Systems are systems that are seen to perform automatic analysis and monitoring (IDSs). Furthermore, IDSs are used by network and system admins for a variety of reasons, as mentioned below in relation to the work of Bace (2001) R. Bace(2001):

- To assist in preventing the emergence of illegal behavior by raising the levels of risk associated with identification and punishment.
- To facilitate security protocols by further comprising the attacks and security intrusions that it did not prevent.
- Keep the prevention and resolution of attacks in mind.
- Documentation is used to advise the institution about the presence of risks.
- Providing a good service quality, especially in the case of enterprise that is complex, by controlling security system and quality of design.
- To improve the identification, improvement, and correction of root causes through the delivery of valuable data relating to attacks identified.

As per T. W. Purboyo and Kuspriyanto (2011) T. W. Purboyo (2011) and J. Arshad and Xu (2013) J. Arshad (2013) When it comes to system quality assessment, presence, privacy, and honesty are recognized as among the most influential and pushing security methods for application. In order to ensure such approaches, intrusion detection systems (IDSs) must be accurate and efficient in detecting instances of intrusion.

Raghunath and Mahadeo (2008)B. R. Raghunath and S. N. Mahadeo research(2008) provides the network intrusion detection system (NIDS), which uses a set of data mining methods to detect attacks on networks of computer and systems automatically. This study focuses on contributions like unsupervised technique of detection that allocates a rating to each network that demonstrates how odd the connection is, and (ii) an association pattern analysis-based module that sums those internet connections that the anomaly detection module ranks highly anomalous.

L. Dali et al.(2015) mentions numerous types of intrusion detection systems which can be used in a wide range of devices to improve system security. There has been a lot of research into deploying a general security framework that is independent of virtualization technology. However, the safety, confidentiality, and privacy protection are still in their infancy. Different solutions, taking into consideration a variety of factors and systems, may be required to address security, privacy, and confidentiality concerns.

2.3.2 Classification of IDs

- Network based In the following paper,designing and implementation of a IDPS system with multimode operation which employs a variety of countermeasures against network attacks. It initially logs malicious files, however if there is increase in number of malicious packets per second , the attacker's IP address is blocked, and at last, it stops the service if still it is unable to block the attacker, this is done to ensure that the attack is not succeeded.This is stated by R. M. Yousufi and Potdar (n.d.) R. M. Yousufi and M. B. Potdar(2017) Furthermore, it has been suggested that cracking down the network into separate pieces, particularly through the use of switches, is identified as the most useful approach when attempting to achieve huge network security. As a result, the independent parts of the network are safe and secure by the use of information security, such as intrusion detection systems (IDSs) and firewalls, as suggested by Scarfone and Mell (n.d.) K. Scarfone and P. Mell (2007)

Malware detection is a significant issue. Using deep neural networks, Al-Maksousy et al. (2018) H. H. Al-Maksousy and C. Wang (2018) design an effective real-time system for detecting and classifying malware related to network behavior. We demonstrate that dividing the system into two neural networks, detecting and analysing, is the key to improving accuracy. As a result, this approach enables the development of a monitoring system which is real-time with CPU usage which is low. Finally, for this task, we present a comparative analysis of four machine learning classifiers.

- Host based An IDS which is seen as being host-based appears to exist on network endpoints because it is placed on a host to enable its collection and close monitoring of suspicious data and occasions observed across it. Because the design of HIDSs is done to enable features and functions on specific hosts, such as web servers or mail, HIDSs show a significant preciseness and durability. Furthermore, any attack and how it affects processes and consumers can be formed, as shown and focused in the work of Scarfone and Mell (n.d.) K. Scarfone and P. Mell(2007)

  Intrusion detection is the way to detect and respond to malicious activity directed at resources of computing and communication, and it has moved into the masses of information assurance as the attacks has increased dramatically. An intrusion detection system (IDS) does the monitoring and collection of data from a targeted system that require protection, operations and coincides the information gathered, and activates responses when proof of an intrusion is identified.Y. j. Ou and j. Ou (2010) Y. -j. Ou and Y. Zhang (2010) developed and constructed a host-based intrusion detection system that combines different detection technologies, one of which is technology that uses log file analysis and the other is technology related to BP neural networking, in this paper.

  As an added capability, HIDSs can perform thorough examination on encoded packets and thus demonstrate audio features all over switched networks, positioning them as an useful augment to NIDSs. Moreover, they have had the capacity to spot attacks that NIDSs would not have identified otherwise, owing primarily to their oversight of host-local events; HIDSs are still unable to identify threats much further than their boundaries. Nonetheless, HIDS management is difficult, as evidenced by the fact that all monitoring hosts' information necessitates management and configuration, as mentioned in the work of S. Roschke and Meinel (2009) F. Cheng, and C. Meinel (2009)

- Distributed based The article addresses some issues with the use of intrusion detection systems (IDS), particularly those involving evasion attacks. The key characteristics of this sort of attack are introduced, and the attack possible options are examined. The features of network and host-based intrusion detection systems are evaluated by comparing, and some elements of an IDS architecture with a distributed approach are examined. Based on this discussion, the article suggests the use of client-based distributed network-based IDS systems in the sensing of evasion attacks. The developed framework is evaluated in comparison to host-based intrusion prevention systems (HIPS). Implementation issues are also addressed in the work of Basicevic et al. (2005)F. Basicevic and and V. Kovacevic(2005)

2.4 Machine learning in IDs

In today's world, Artificial Intelligence (AI) is recognized as the key technology in a variety of more innovative applications, such as the detection of attempted fraudulent activity in the financial sector, by use of a robot with the ability to identify and react to feelings, and even supplying software systems with the best advice that can function as a human expert. In fact, there is a school of thought that such technologies would not exist if no knowledge was gained from the completion of AI studies. According to Mitchell (2006) T.M. Mitchell, Machine-Learning (ML), which is regarded as a basic element of AI, is defined as a computational mechanism that allows computer machine to learn from analogy, illustrations, and expertise. As a result, the outputs of learning could be guided as intellectual ability in order to solve a specific problem. All operations and their information must be examined in intrusion detection in order to emphasize trends in behavior, whether usual or disturbing. Nonetheless, it is argued that the data sample of operations, referred to as the training dataset, must include a sufficient number of observations related environmental under investigation in order to highlight the trend as a whole. As a result, new information instances could only be classified using the learned framework based on their similarity to normal human behavior (anomaly identification) or known attack signs (misuse identification) as per Sommer and Paxson (2010) R. Sommer and V. Paxson(2010)

In the reference paper Al-Maksousy et al. (2018) specific combination of two neural network learning algorithms is used also In the reference paper et al. (2016) a Multilayer Perceptron (MLP) is a machine learning algorithm capable of classifying large amounts of data and finding patterns in complex datasets. In this paper the MLP algorithm is used to perform intrusion detection based on the Knowledge Discovery and Datamining (KDD) dataset.

In my research project, I have used multiple supervised learning machine learning algorithms also neural network algorithms and compared the accuracy to see which one gives best performance. I have also done model tunning to improve the accuracy score which I got due to base models.

3 Methodology

CRISP-DM is an acronym for cross-industry data mining process. The CRISP-DM is process for methodically arranging a data mining operation. It is, undoubtedly, a reliable and attempted method. We don't allege it like our own. We have not created it. Even so, whenever it arrives to using analytics to solve difficult business issues, experts are supporters for its rationality, flexibility, and utility. It is only the priceless thread that almost every customer interaction is linked. The model illustrates a fictitious sequence of steps. Several of the activities could be finished in any order, and some practices will regularly need to be repeated. The system doesn't quite try to catch all imaginable paths as it progresses through the Knowledge discovery process. The general structure of the CRISP-DM methodological approach is depicted in Figure 1.

The Crisp DM is a standard process model with 6 phases included that naturally describes the data science life cycle which is commonly used worldwide. In my research project, I have chosen CRISP-DM process as my research question was how I can build machine learning models to detect intrusions in the network in cloud security. By using CRISP DM, firstly I got a clear idea of the business understanding, here in which the 1st phase of my project was to understand how we can build a machine learning model which can detect the intrusions and how to balance the data with help of machine learning in order to improve the accuracy rate of performance and classifications of the attacks.Further, with help of CRISP-DM process, the phases which were followed were Data understanding, Data preparation, Modelling, Evaluation and Deployment by which all processes were done by me in a simple and accurate manner to achieve the business planning.

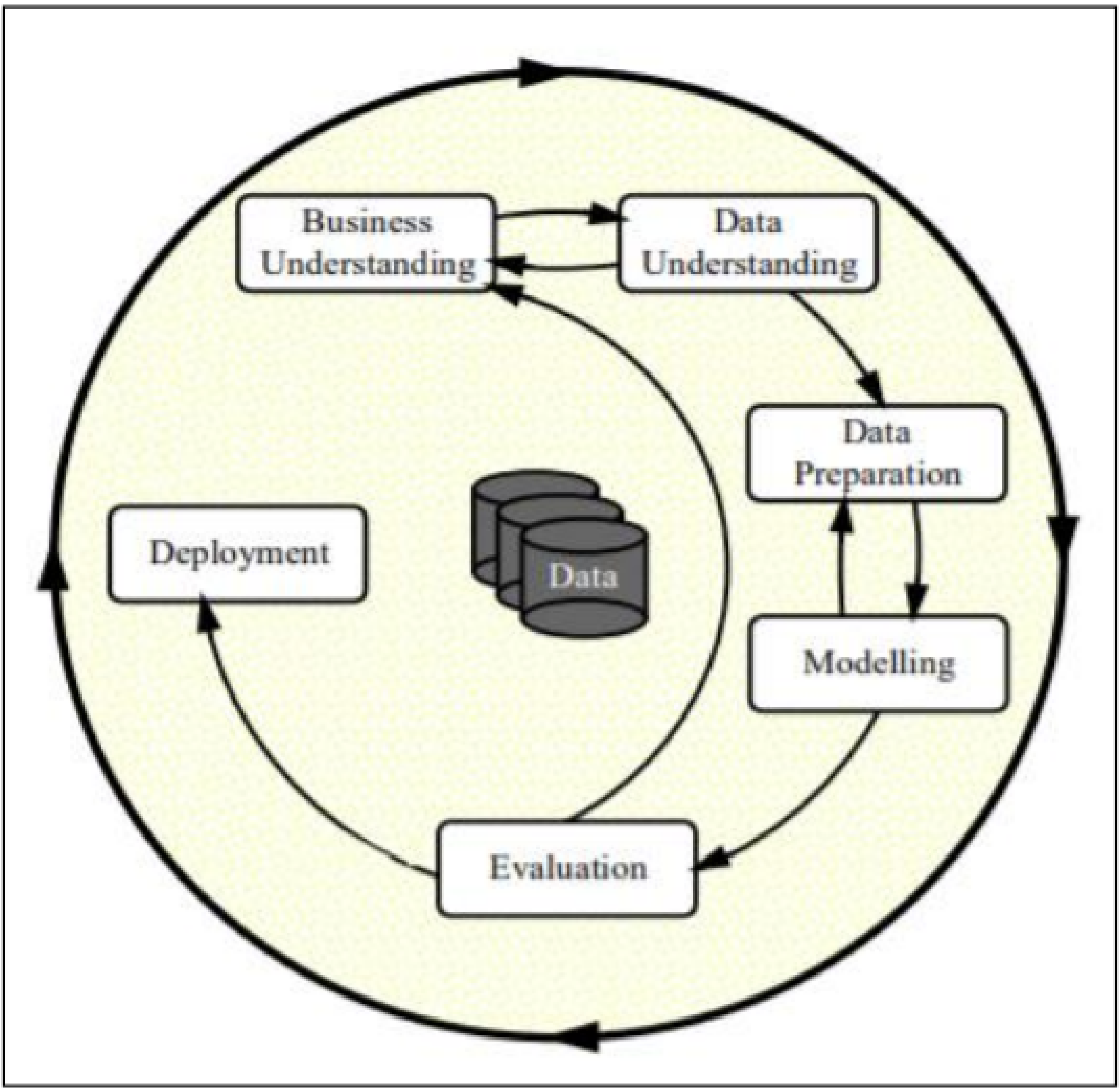


Figure 1: CRISP DM Methodology Pondel and Korczak (2017)

3.1 Business understanding

This phase focuses on obtaining a knowledge and competence of the project's goals and requirements, and also converting that data into a data mining problem explanation and a project initiation plan to achieve the targets. The emergence of Cloud Computing (CC) and the numerous applications that support it has resulted in rapid changes in information technology. CC enables the distribution of resources all over a network, allowing people to engage with these resources as and when needed, with adaptable access. As a consequence, because consumers are not essentially collecting and analyzing individually, accessibility and protection are enhanced. Companies need to concentrate more on research and innovation of their own goods while spending very little time and energy developing reliable storage amenities, buying equipment for that intent, or staff training to follow due process whether they can buy into cloud services provided by specialized cloud service providers. The many functionalities and benefits of cloud computing, such as increased efficiency, reduced costs, increased availability, stability, and versatility to handle and extent the processes, start making it very appealing to a wide range of businesses and organizations in a wide range of industries. Sadly, cloud security is a major concern among users of cloud. Because cloud utilisation is overly dependent on user trust, there is problem that organizations may become vulnerable to different hazards and vulnerabilities. Furthermore, there is indeed a risk that cloud technology will be breached by invaders or hacking, granting them access to essential data stored in the cloud that belongs to others. An invasion or attack can have serious consequences for cloud usage. Because most intruders are likely to hit networks with more users, the much more usable and digitalized assets, and the networks usually contains the most data, it is investigated thoroughly and all dangers should be remedied where possible. The Cloud Security Alliance (CSA) lists 7 main risks to cloud structures . Utilizing Intrusion Detection (ID) methodologies is the best way to avoid threats and protect structures so these systems can quickly recognize and thus defend against hackers. As a result, any concerns about cloud security must be discussed before any advantages can be realized.

Objective

- This thesis will achieve the following objectives by attempting to address demanding research issues of existing services of cloud security by developing an intellectual and customizable danger monitoring system based on techniques of machine learning. It involves Conducting an evaluation of the current cloud security devices to identify specific vulnerabilities and possible solutions

- Investigate the shortcomings of existing machine learning Intrusion detection system models for detection of attacks for which we use standard machine learning techniques.

- Investigate the impact of high-dimensional (unbalanced) data between the classes on machine learning models accuracy rate when applied to IDS.

- Investigate the accuracy at which minor attacks are discovered, as well as the impact because there are sub-minor categories

- Create a one-of-a-kind machine learning structure that identifies data imbalance in both directions between and within classes

- Examine the impact of extraction of features on prediction of the accuracy and how it can be used to increase the efficiency of models used during detection of network attack with an unbalanced dataset. Figure 4 depicts the entire model-building workflow.

3.2 Data understanding

The data understanding phase starts with collection of data and progresses through acts to become acquainted with the data, identify quality problems, example can find knowledge from big data, or reveal interesting subgroups to create assumptions about incomplete details. Data and business comprehension are inseparably connected concepts. At the very minimum, For the data to be developed, some understanding of the currently offered information is required. The extraction problem, and the strategy of project.

- Collect Preliminary Information: This is one of the most important steps because the statistics must be fully realized. Kaggle, a freely available resource, was used to collect the data. From Kaggle, KDD Cup'99 Dataset is used. This dataset contains a wide range of incursions simulated in a military network context.

- Describing Data: Collected Data consisted of 42 features in which there are different 23 categories of output labels out of which 22 classes represent different type of bad connections and one class named 'normal' represent good connection.

In Data understanding EDA (Exploratory Data analysis) was done from which we got several understandings of data. We can see various graphs and outcomes from the data by the help of EDA done below. In the beginning phase checking for the missing values in the dataset was done from which we can see there were no missing values in the dataset as shown in Figure 2

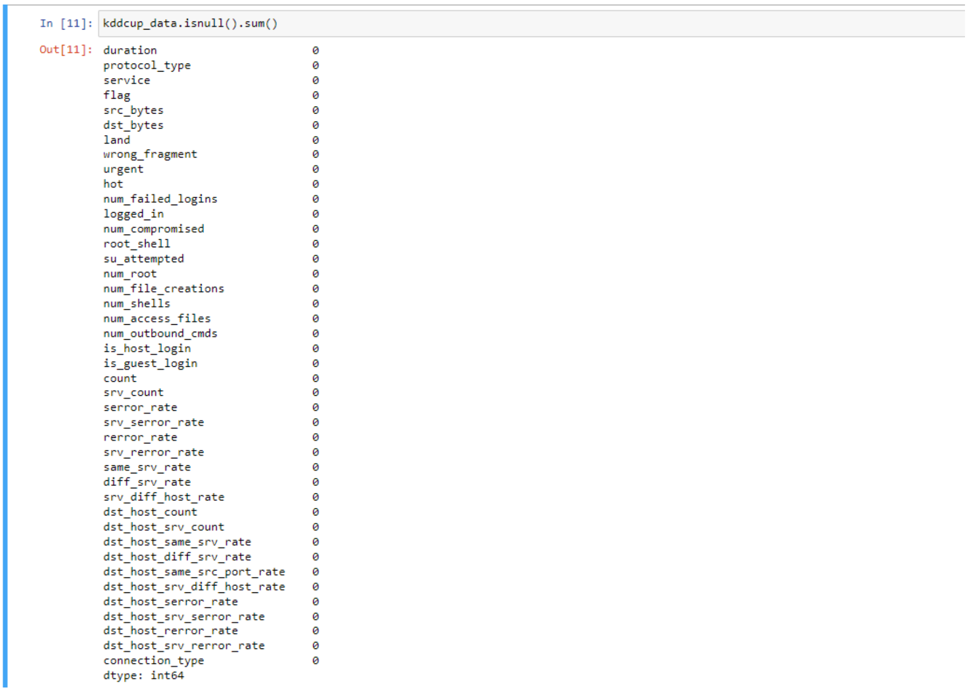
We have checked the attack types with respective counts as from which it was observed that number of normal (no attack) counts was higher and Neptune (Actual attack) was one of the attacks with higher number of counts shown in Figure 3

We can see that the number of normal connections is significantly larger the number of attacks. Though this is expected and makes sense, it would not be feasible to use accuracy as the only metric while evaluating a model. We can see the same in the Figure ?? given below.

3.3 Data preparation

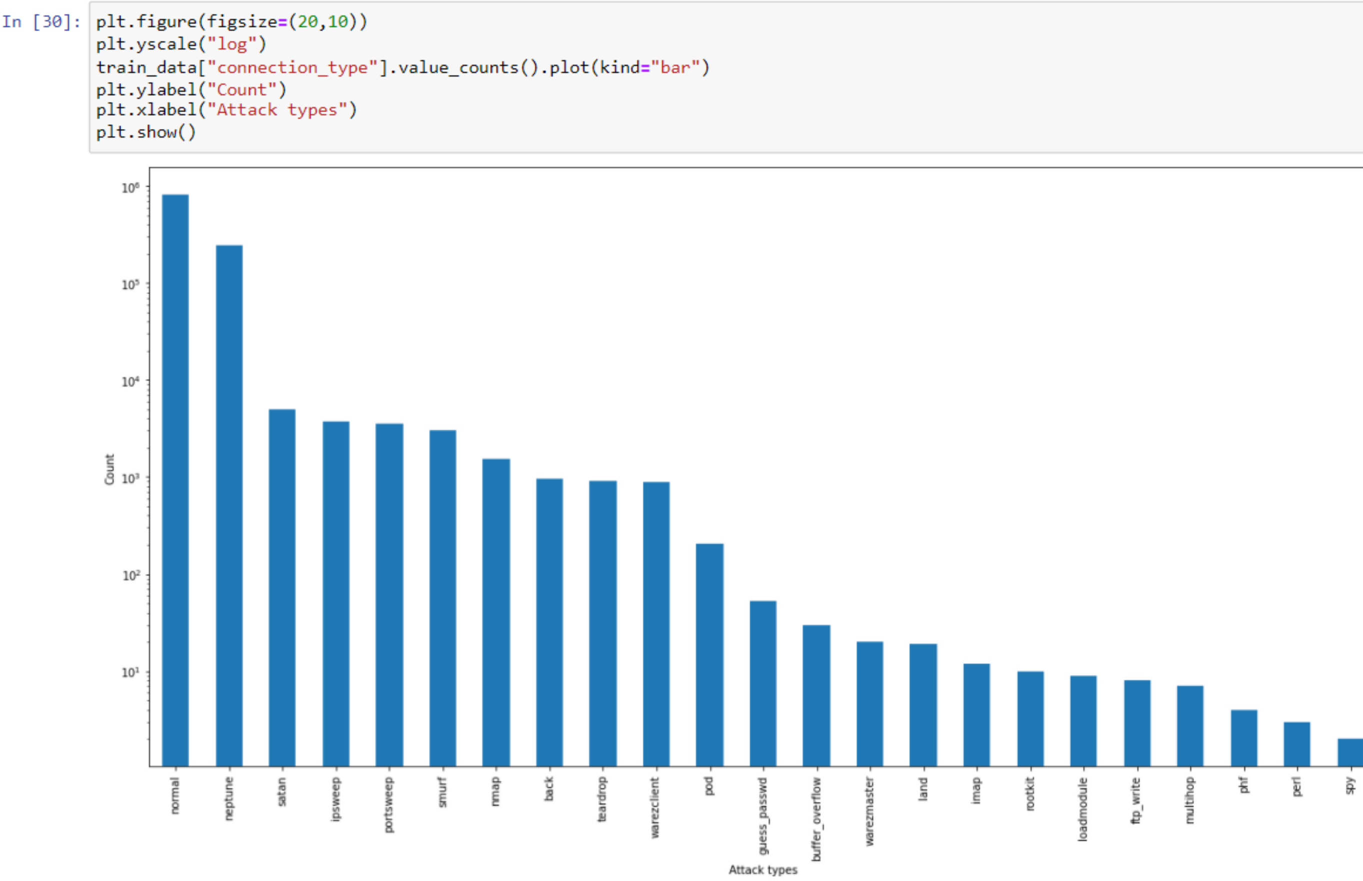
The steps included in data pre-processing covers all the activities that culminate in the creation of the ultimate training dataset as from the original information. P Data pre-processing activities of the dataset are likely to occur several times and in no specific order. Extraction of databases, objects, and characteristics, data purification, attribute generation, and data transformation, also a All task includes gathering data for modelling techniques. The data pre-processing workflow is depicted in Figure 5.

Select your data: This is the stage of the cycle in which we select the information that would be used in the evaluation. For additional processing, I chose the publicly accessible KDD cup dataset. More precisely, from 1999, it has been stated that the most often used dataset for evaluating identification techniques is the KDD Cup '99 Dataset. This was designed in accordance with the data acquired by DARPA 1998 TCP/IP. In terms of the



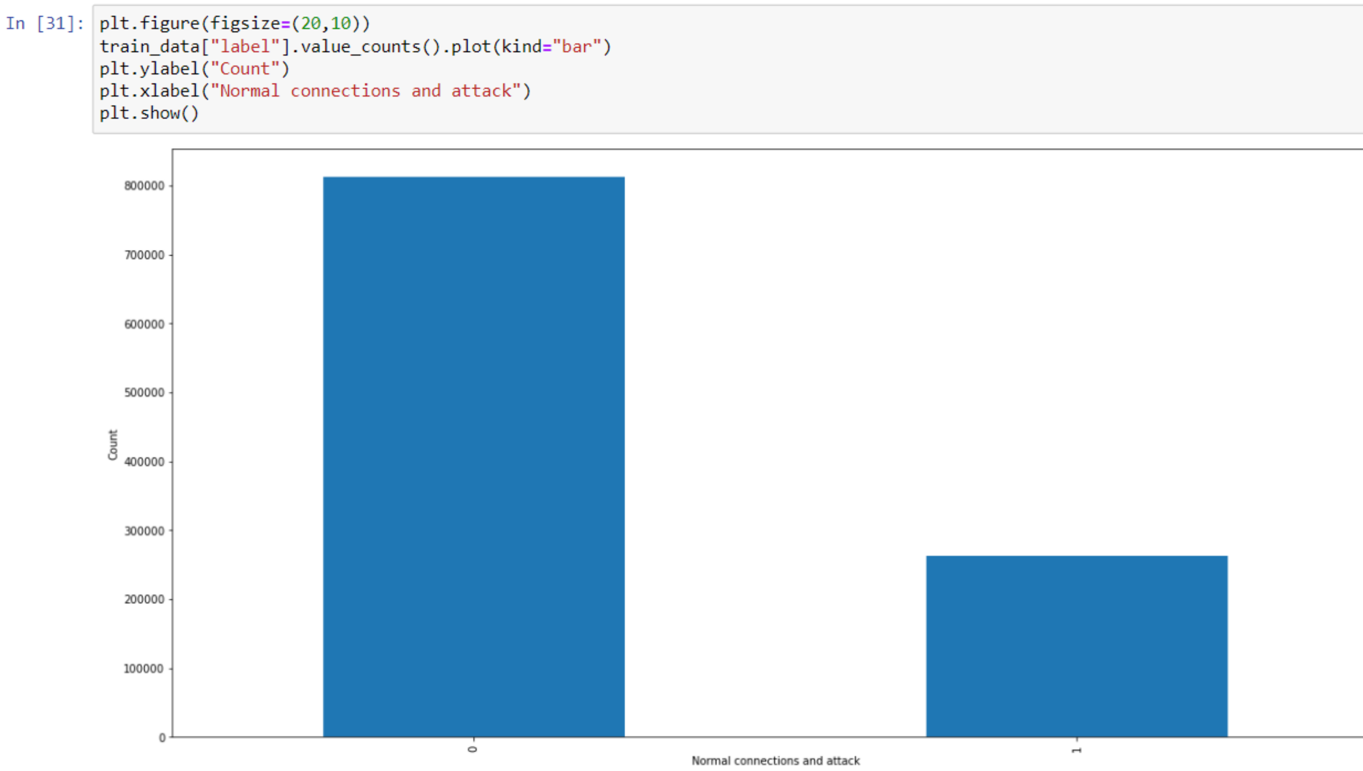
```
In [11]: kddcup_data.isnull().sum()

Out[11]: duration                       0
         protocol_type                  0
         service                        0
         flag                           0
         src_bytes                      0
         dst_bytes                      0
         land                           0
         wrong_fragment                 0
         urgent                         0
         hot                            0
         num_failed_logins              0
         logged_in                      0
         num_compromised                0
         root_shell                     0
         su_attempted                   0
         num_root                       0
         num_file_creations             0
         num_shells                     0
         num_access_files               0
         num_outbound_cmds              0
         is_host_login                  0
         is_guest_login                 0
         count                          0
         srv_count                      0
         serror_rate                    0
         srv_serror_rate                0
         rerror_rate                    0
         srv_rerror_rate                0
         same_srv_rate                  0
         diff_srv_rate                  0
         srv_diff_host_rate             0
         dst_host_count                 0
         dst_host_srv_count             0
         dst_host_same_srv_rate         0
         dst_host_diff_srv_rate         0
         dst_host_same_src_port_rate    0
         dst_host_srv_diff_host_rate    0
         dst_host_serror_rate           0
         dst_host_srv_serror_rate       0
         dst_host_rerror_rate           0
         dst_host_srv_rerror_rate       0
         connection_type                0
         dtype: int64
```

Figure 2: Null values



```
In [30]: plt.figure(figsize=(20,10))
         plt.yscale("log")
         train_data["connection_type"].value_counts().plot(kind="bar")
         plt.ylabel("Count")
         plt.xlabel("Attack types")
         plt.show()
```

Figure 3: Types of attacks



```
In [31]: plt.figure(figsize=(20,10))
         train_data["label"].value_counts().plot(kind="bar")
         plt.ylabel("Count")
         plt.xlabel("Normal connections and attack")
         plt.show()
```

Figure 4: Normal VS Attacks

KDD subset, the training data contains 5 million records, while the test set has about 4 million records disperse across 41 various aspects; but at the other side, only 24 kinds of attack are also included for each of the training data records, while only 14 kinds are placed in the test data. All trained datasets are labeled, with either the sort of assault or with the fact that it is normal.

3.4 Cleaning of data

The column 'connection-type' was misplaced in the KDD dataset.So made the coloum appropriate as it is my 'target-variable' coloum.I renamed the coloum no.41 as 'connection-type' and dropped the column 42.As shown in figure 6.

Then there was an extra '.' in the 'connection type' column after the connection type. For better visualization and interpretation, I remove the '.'. As shown in the below figure 7.

Next, I checked for the duplicate rows and removed them from the training dataset. Originally, we had 4898431 rows in the dataset. After removing the duplicates, we then had 1074992 rows. This shows that there were too many redundant rows in the data set. Since this data is a TCP dump, having same values is very much expected as shown below in the figure ??

Here, we are dealing with binary classification i.e. whether a connection is an attack or not. In our data set, columns are labelled as either 'normal' or as the attack type. So we need to denote normal connections as one class type and all the attack types as another class. The new class label would be 0 if the connection is normal and 1 if it is an attack.This is shown in figure ??

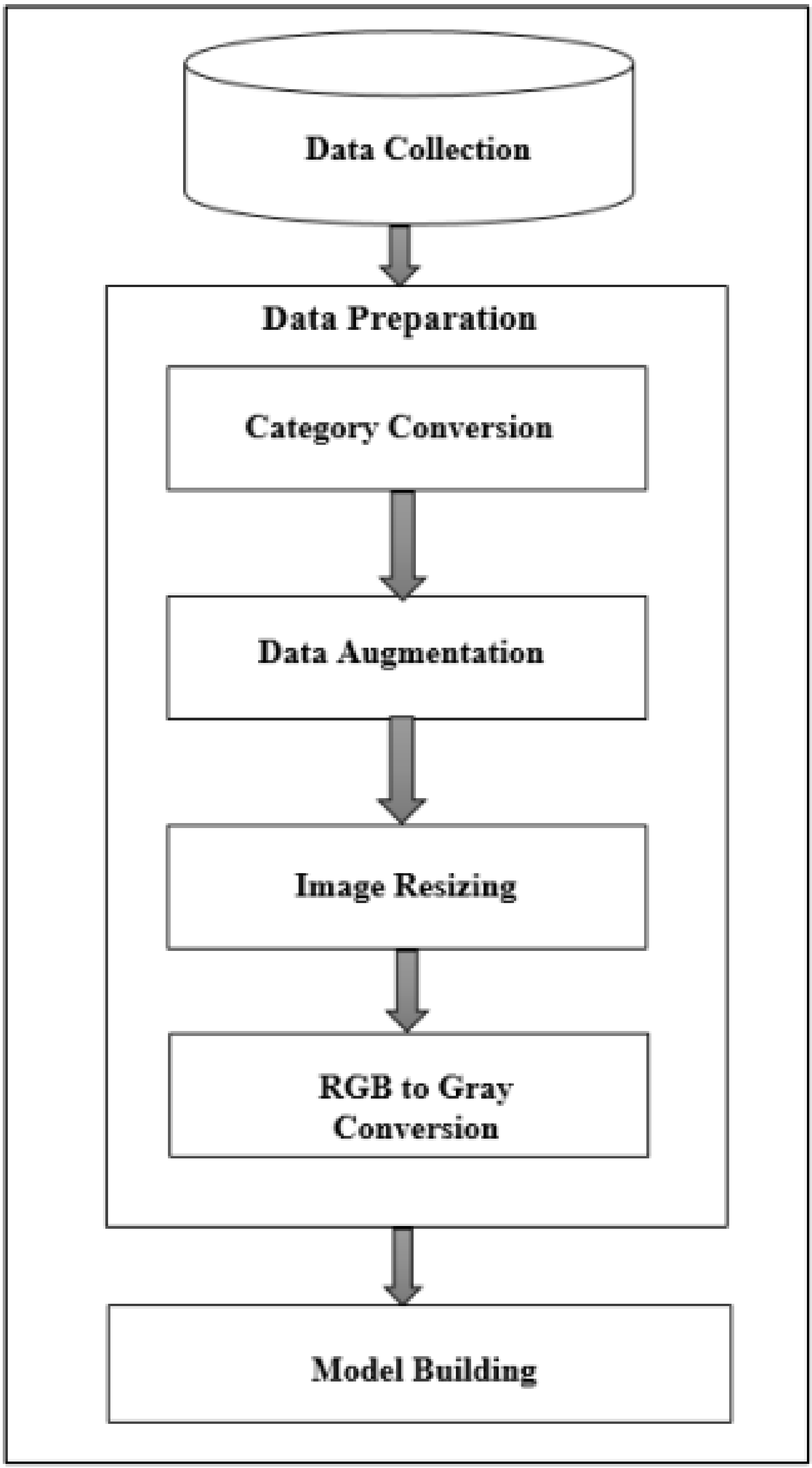
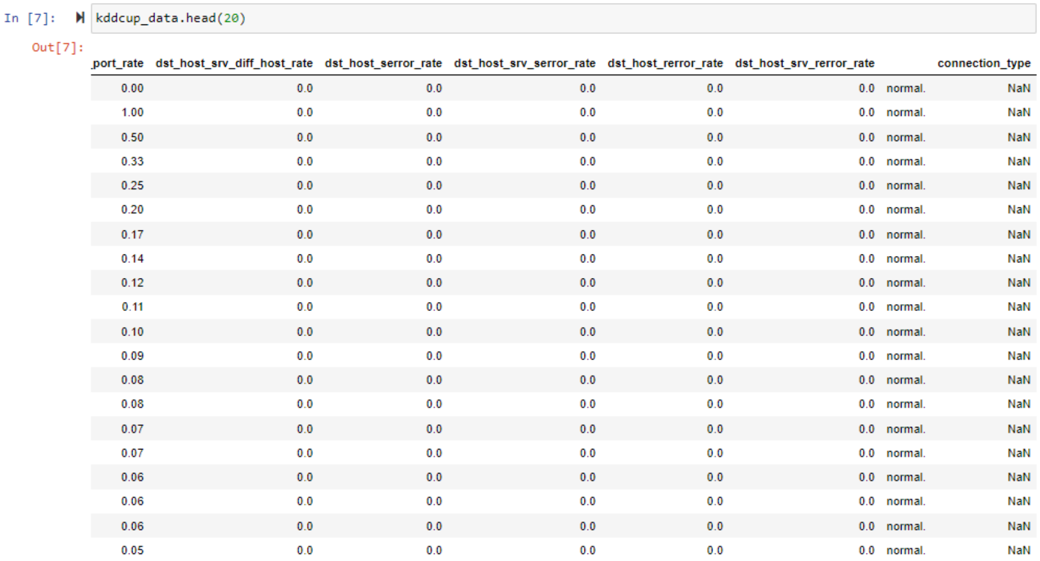


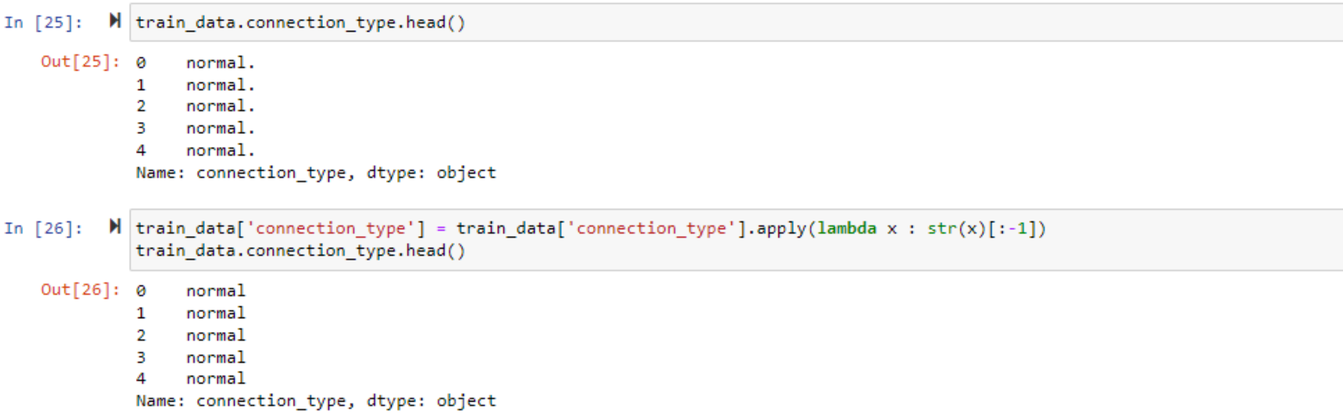
Figure 5: Data preparation Flow



```
In [7]:  kddcup_data.head(20)
Out[7]:
```

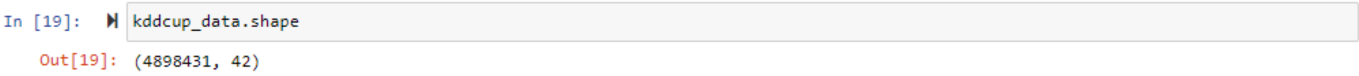
port_rate	dst_host_srv_diff_host_rate	dst_host_serror_rate	dst_host_srv_serror_rate	dst_host_rerror_rate	dst_host_srv_rerror_rate		connection_type
0.00	0.0	0.0	0.0	0.0	0.0	normal.	NaN
1.00	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.50	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.33	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.25	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.20	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.17	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.14	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.12	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.11	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.10	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.09	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.08	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.08	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.07	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.07	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.06	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.06	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.06	0.0	0.0	0.0	0.0	0.0	normal.	NaN
0.05	0.0	0.0	0.0	0.0	0.0	normal.	NaN

Figure 6: Coloum name mislabelled



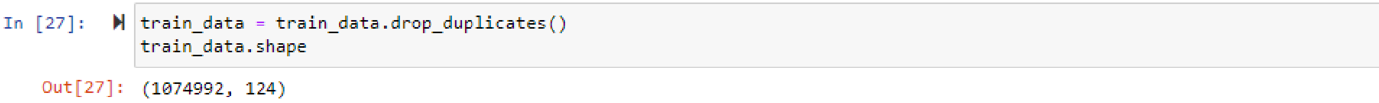
```
In [25]:  train_data.connection_type.head()

Out[25]: 0    normal.
         1    normal.
         2    normal.
         3    normal.
         4    normal.
         Name: connection_type, dtype: object

In [26]:  train_data['connection_type'] = train_data['connection_type'].apply(lambda x : str(x)[:-1])
          train_data.connection_type.head()

Out[26]: 0    normal
         1    normal
         2    normal
         3    normal
         4    normal
         Name: connection_type, dtype: object
```

Figure 7: Data correction



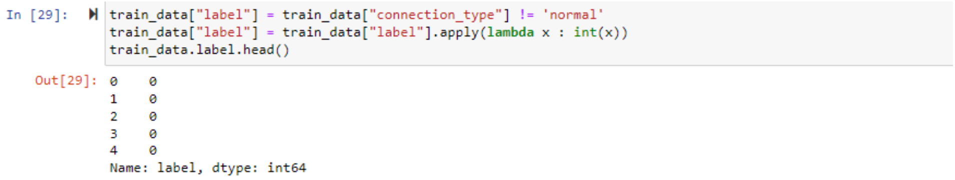
```
In [19]:  kddcup_data.shape

Out[19]: (4898431, 42)
```

Figure 8: Data with Duplicate



```
In [27]:  train_data = train_data.drop_duplicates()
          train_data.shape

Out[27]: (1074992, 124)
```

Figure 9: Data after duplicates removal



```
In [29]:  train_data["label"] = train_data["connection_type"] != 'normal'
           train_data["label"] = train_data["label"].apply(lambda x : int(x))
           train_data.label.head()

Out[29]:  0    0
          1    0
          2    0
          3    0
          4    0
          Name: label, dtype: int64
```

Figure 10: 1 Hot Encoding

3.5 Modelling

After dividing the data into training and testing sets, the first stage is to model it. First, we must decide which model to train, therefore I choose to train data on the following pretrained models:

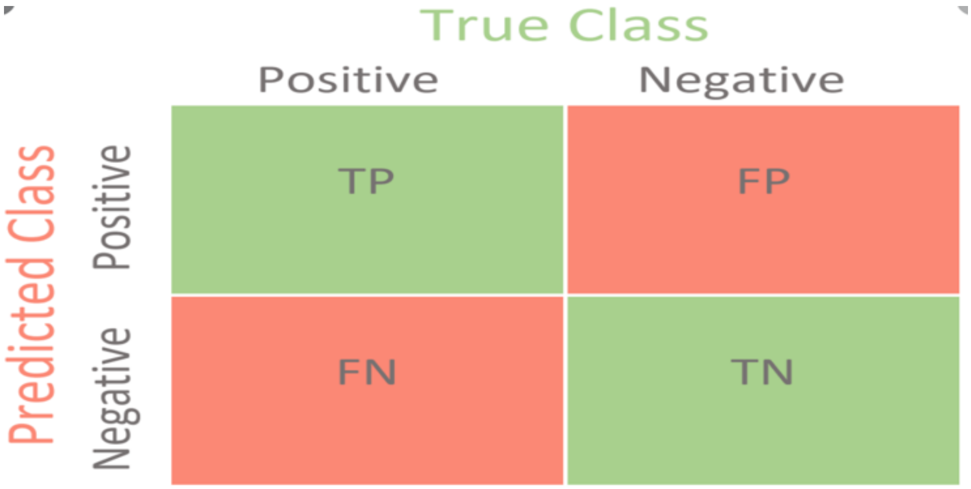
3.5.1 Random Forest

Random forest is a supervised machine learning algorithm that is commonly used to solve classification and regression issues. It constructs decision trees from several samples and uses the majority vote for classification and the average for regression.One of the most essential characteristics of the Random Forest Algorithm is that it can handle data sets with both continuous and categorical variables,as in regression and classification. The random forest method generates decision trees from data samples, then obtains predictions from each of them before voting on the best answer.From a randomly selected portion of the training data, the Random forest classifier generates a collection of decision trees.It consists of a set of decision trees (DT) derived from a randomly selected subset of the training set,which then accumulates votes from various decision trees to get the final prediction. Working of Random Forest: Step 1: In the First Phase, Begin by selecting random samples from a specified training dataset. Step 2: Following that, this algorithm will create a decision tree for each sample of training data. The forecast result from each decision tree will then be obtained. Step 3: In this third phase, every expected outcome will be voted which will lead to the outputs or predictions. Step 4: Finally, as the final forecast result, we can choose the one with the most votes.

3.5.2 Logistic regression

We can apply logistic regression if the output of a factor is categorized or binary. While logistic regression functions as a classification, it is based on linear modeling underneath the surface. For classifying in logistic regression, researchers employ the logistic sigmoid function or the sigmoid function. The graph of this logistic regression model is an S-shaped curve. Logistic regression is employed in both binary and multiclass classification. The binary classification convention is to have two classes, 0 and 1. We can assess the model's performance after training it with data using the logistic function by studying the confusion matrix as shown in figure 11

The confusion matrix is a cross-tabulation of the expected situation or even what you anticipated the name to be vs the client's condition and what the true labeling was. If indeed the circumstance was positive but the theory anticipated that it would be positive, it falls into the true positive category. If the condition was positive but our model anticipated it to be zero, this is referred to as a false detection, also known as a



		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Figure 11: Logistic Regression confusion matrix

Type 1 mistake. If the criterion was positive and our model expected it to be negative, it is referred to as a genuine negative; if our model projected the result of the failure to be positive, it is referred to as a false negative, also known as a Type 2 mistake. Different governing equations, such as precision, adjusted odds ratio, negative likelihood ratio, and so on, are derived using this matrix.

3.5.3 Gradient Boosting algorithm

Gradient boosting is indeed a machine learning approach that is utilized in problems such as classification and regression problems. It returns a forecast model is a representation of an aggregation of weak estimation techniques, often decision trees. [1] Whenever a decision tree is used as the weak learner, the final algorithm is known as gradient-boosted trees; it typically surpasses random forest. [1] A gradient-boosted trees model is constructed in the same stage-wise manner as previous boosting methods, but still, it extrapolates the other approaches by enabling optimum of any discrete loss function. Three main components of Gradient boosting: Loss Function-The loss function's duty is to evaluate whether product is good at making predictions using the provided data. This may vary based on the nature of the situation. For instance, if we're attempting to estimate a bodyweight based on certain input data, the loss function would be one that assists us in determining the difference between the estimated and measured weights. If, on the other hand, we're trying to anticipate whether a person would like a particular movie based on personality, we'll need a loss function to determine how effective our system is at categorizing people that did or didn't choose certain movies. Weak Learner - A weak learner is someone who identifies our information but does so ineffectively, possibly no better than random chance. In other words, it has a high percentage of mistake. Usually, these really are decision trees. Additive model - The continuous and progressive strategy of building the trees each step at a time is referred to as the additive model. You need to be nearer to our final version with each cycle. In other words, with each iteration, the result of the loss function must be reduced.

3.5.4 Perceptron and Multilayer Perceptron

In this paper, an intrusion detection system was designed employing a variety architectures of perceptron learning. Training process and testing are the two stages of perceptron algorithms. Using a learning rule, the weights w as shown in fig are repeatedly changed throughout the training phase. Each feature in a training data is displayed to the networks numerous times, and the weights of network are changed slightly after each image is applied. Following training, the system will deliver the highest possible identification accuracy depending on an optimum set of weighting factor as mentioned by F. Palenzuela et al. (2016). A single - layered perceptron with a single weight vector separating the input sequence from the output layers. However, a multi - layer perceptron including one layer of the network (known as a hidden layer) sandwiched between two separate weight matrices (wa and wb) through which the input sequence must pass befre a categorization judgment is made. This buried layer of neurons has the advantage of producing a system with improved classification results since more nuanced distinctions can be formed throughout the classification stage.This is shown in Figure 12 and Figure 13.

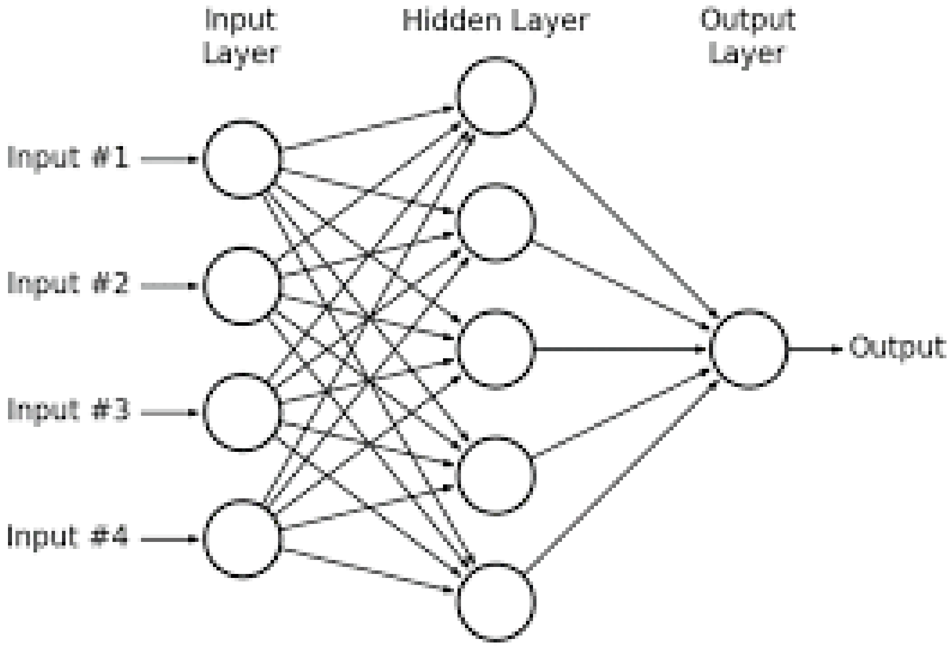


Figure 12: Multilayer Layer Perceptron

3.6 Evaluation

- Analyzing the results: At this stage, we must assess the usefulness of our models in terms of fulfilling the business goals that inspired the process of data-mining . We must investigate why the concept could be inappropriate for business use.
- Investigating the operation: Now that we've gone so over information and constructed models, take some time to think on our approach. It is an opportunity to highlight issues which we may have overlooked and to draw attention to flaws in our work while we still have time to fix them before release.

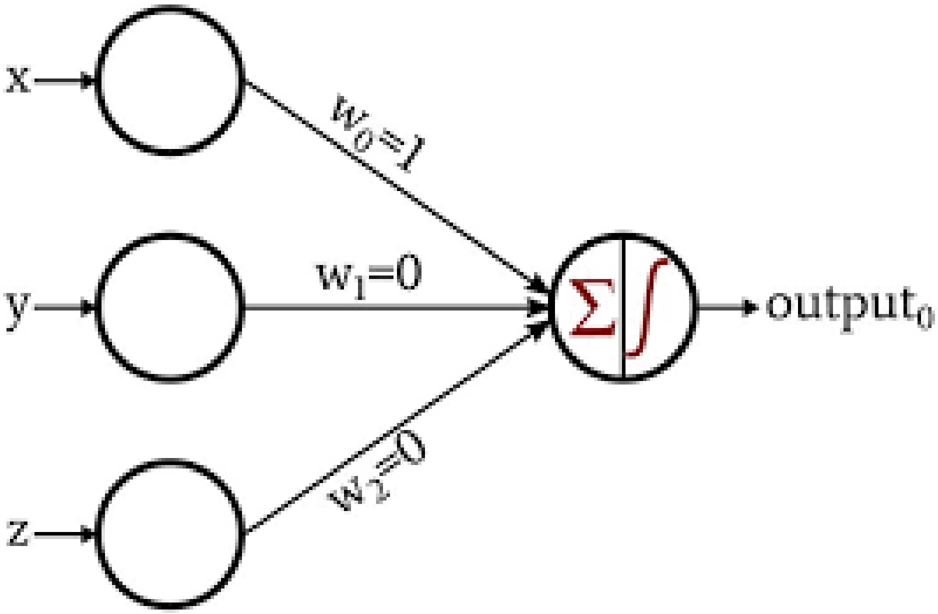


Figure 13: Perceptron

- Creating a strategy for the following phases: The evaluation process concludes with our recommendations for the following stage. The design may be ready for usage, or we may determine that it is better to repeat some of the steps and strive to improve it.

3.7 Deployment

Data mining brings benefits whenever it pertains to deployment. It makes no difference how clever your results are, or how well your models fit the reality, if you don't use them to improve actual way business is conducted in this final step of a process of Cross-Industry Standard for Data Mining process. The deployment phase consists of four tasks. They are as follows:

- Planning of Deployment
- Monitoring and maintenance planning
- Final results summarization
- Final outcomes examination

4 Results and Analysis

Here intrusion detection system has been used to deal with increasing level of attacks. These systems use machine learning algorithms here which initially learn from the data so that they can evaluate network traffic and see what is dangerous and what is not and determine if it is attack or normal. With no intervention of humans or tedious programming IDS systems are able to detect weather the attack is normal or harmful.

The technique of categorizing a set of data into various class is known as classification. Furthermore, algorithms used for classification can conduct binary classification, in which the result can only pick two values. In my project I have two binary values which are normal and attack hence I have used all the classification algorithms.

For data with lot of features, its generally the case that many of those features will be unrelated or weakly related to your target variable. The problem is that even a noisy feature appears to be informative in general sense. In my project I have chosen these variables wisely, which are depending upon the importance of the variables on target variable. Therefore, on that basis I have trained my models on the training data that is seen data. Now, if we run the models with unseen data, the unseen data must have same variables and features as the data which is used by me to initially train the model. Surely there will be minor difference between the outputs as the data varies. Since it all depends on the variables and features which will be present in the unseen data which would be similar to the training data.

This section will discuss the results of many algorithms upon training sample.It is vital to analyze and evaluate the data collected after offering our notion during research. The accuracy of all the given models upon training data is shown in Table 1.

Table 1: Comaprison of Accuracy

Model	**Accuracy**
Logistic Regression	91.90
Random Forest	92.58
Gradient Boosting algorithm	92.60
Perceptron	81.53
Multi-layer Perceptron	93.006

I have also utilized the ROC curve and AUC to evaluate the outcomes. They are used to select the appropriate cut-off that will have the lowest false positive rate and the highest sincere rate (true positive). As can be observed, the ROC curve for Logistic Regression, Random Forest, and Multiperceptron is superior. If a packets are identified as an attack but is not categorized as such, it might cause damage. So, after reviewing the ROC curve, I determined that the Multi perceptron will be the best classification technique to utilize.Figure 14 shows the ROC curve.

4.1 Model Tunning

Tuning is the attempt to improve performance of the model without over fitting or increasing variance.It is done to optimize the performance of the model.It is shown in the figure 15

- **Random Forest**

- Important variable tunning - In this method we have taken the important variables of the dataset and then tunned the Random Forest model to see if there is any improvement in the accuracy og the model.

- Gini Entropy method A gini impurity quantifies the likelihood that any piece of the dataset would be mislabeled whenever arbitrarily classified. he Gini Index has a

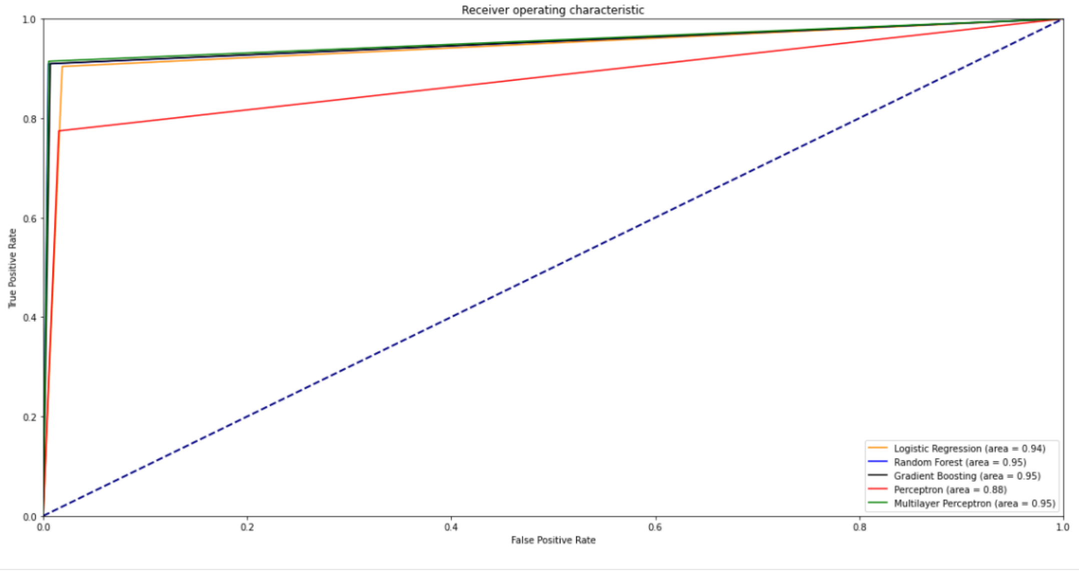
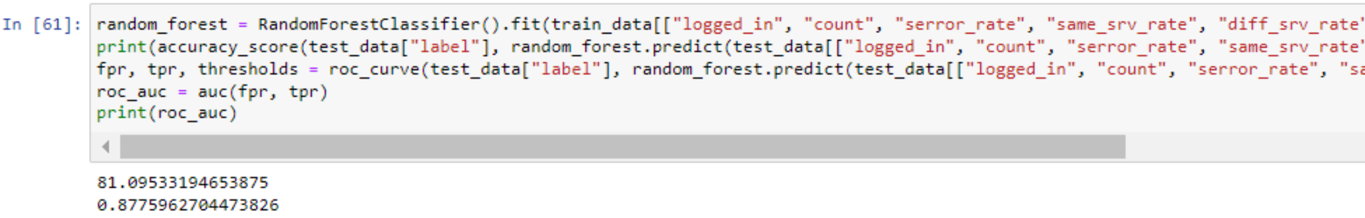


Figure 14: ROC Curve

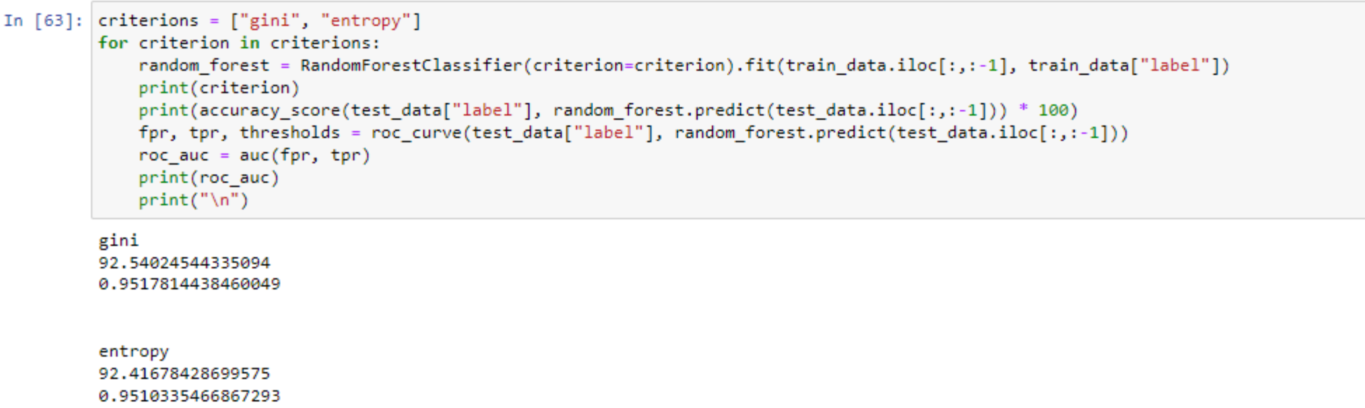


```
In [61]: random_forest = RandomForestClassifier().fit(train_data[["logged_in", "count", "serror_rate", "same_srv_rate", "diff_srv_rate"
         print(accuracy_score(test_data["label"], random_forest.predict(test_data[["logged_in", "count", "serror_rate", "same_srv_rate"
         fpr, tpr, thresholds = roc_curve(test_data["label"], random_forest.predict(test_data[["logged_in", "count", "serror_rate", "sa
         roc_auc = auc(fpr, tpr)
         print(roc_auc)

         81.09533194653875
         0.8775962704473826
```

Figure 15: Randome Forest Important variable tunning

lower limit of 0. This occurs whenever the node is pure, which indicates that all of the items included within the node are from the same class. As a result, this node will never be divided again. As a result, the characteristics with the lowest Gini Index are picked as the best split. Furthermore, it has the greatest value whenever the likelihood of the 2 classifications is equal.

Entropy is a data metric that represents the instability of the characteristics with respect to the aim. The characteristic with the least entropy, comparable to the Gini Index, determines the optimal split. It reaches its highest value whenever the chance of the 2 classifications is equal, and a node is pure whenever the entropy is at lowest value, which is zero.Figure shows the method 16



```
In [63]: criterions = ["gini", "entropy"]
         for criterion in criterions:
             random_forest = RandomForestClassifier(criterion=criterion).fit(train_data.iloc[:,:-1], train_data["label"])
             print(criterion)
             print(accuracy_score(test_data["label"], random_forest.predict(test_data.iloc[:,:-1])) * 100)
             fpr, tpr, thresholds = roc_curve(test_data["label"], random_forest.predict(test_data.iloc[:,:-1]))
             roc_auc = auc(fpr, tpr)
             print(roc_auc)
             print("\n")
```

```
gini
92.54024544335094
0.9517814438460049


entropy
92.41678428699575
0.9510335466867293
```

Figure 16: Randome Forest Gini Entropy tunning

- Split and Leaf method In this method the dataset is divided and then accuracy is calculated.Split in a decision tree meansdividing node into multiple sub nodes for a pure node and leaf is decision that is taken after all the features are computed.Figure 17 shows the result

  In the split leaf method the better accuracy that we found is for 5 1 ratio as shown in figure 18

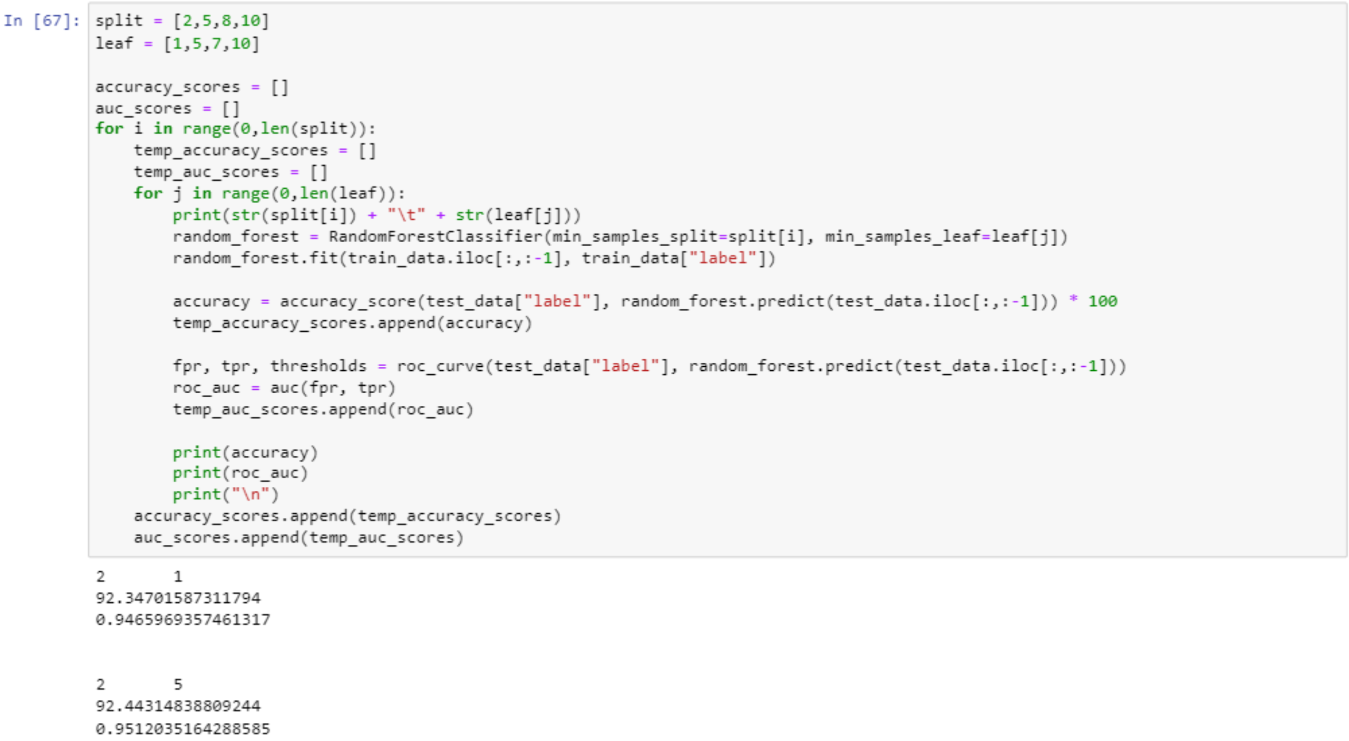
  In split and leaf we have also calculated accuracy curve and the Area under curve graphs as shown in figure 19 and Figure 20

- **Gradient Boosting Algorithm Tunning**

- Important variable tunning - Same like Randon forest we have taken important variables in the dataset and tunned the model.Figure 21 is below.

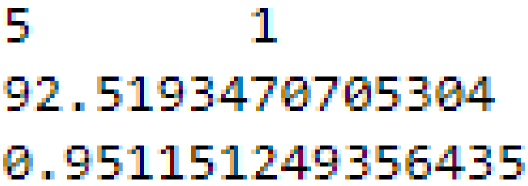
5 Conclusion and Future Work

The study presented in this thesis first gave an introduction of the basic ideas of Cloud Computing, followed by a literature review with the goal of assessing the cloud's security weaknesses. There seem to be a number of security flaws in Cloud Computing, that have a significant impact on all of its layers. To narrow the scope of this research project, it



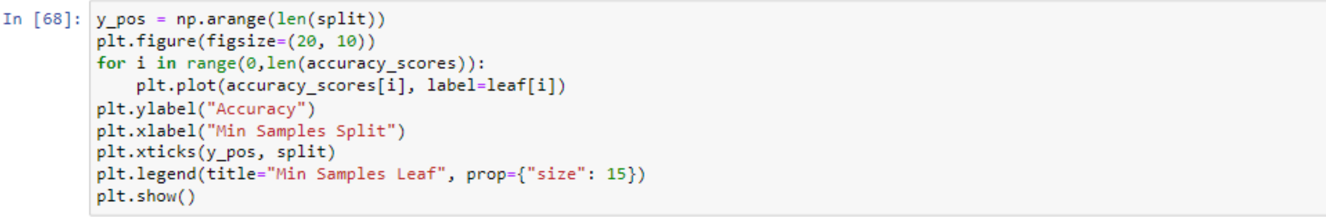
```
In [67]: split = [2,5,8,10]
         leaf = [1,5,7,10]

         accuracy_scores = []
         auc_scores = []
         for i in range(0,len(split)):
             temp_accuracy_scores = []
             temp_auc_scores = []
             for j in range(0,len(leaf)):
                 print(str(split[i]) + "\t" + str(leaf[j]))
                 random_forest = RandomForestClassifier(min_samples_split=split[i], min_samples_leaf=leaf[j])
                 random_forest.fit(train_data.iloc[:,:-1], train_data["label"])

                 accuracy = accuracy_score(test_data["label"], random_forest.predict(test_data.iloc[:,:-1])) * 100
                 temp_accuracy_scores.append(accuracy)

                 fpr, tpr, thresholds = roc_curve(test_data["label"], random_forest.predict(test_data.iloc[:,:-1]))
                 roc_auc = auc(fpr, tpr)
                 temp_auc_scores.append(roc_auc)

                 print(accuracy)
                 print(roc_auc)
                 print("\n")
             accuracy_scores.append(temp_accuracy_scores)
             auc_scores.append(temp_auc_scores)

         2       1
         92.34701587311794
         0.9465969357461317


         2       5
         92.44314838809244
         0.9512035164288585
```

Figure 17: Randome Forest Split leaf tunning



```
5       1
92.519347070530 4
0.95115124935643 5
```

Figure 18: Randome Forest Spli leaf



```
In [68]: y_pos = np.arange(len(split))
         plt.figure(figsize=(20, 10))
         for i in range(0,len(accuracy_scores)):
             plt.plot(accuracy_scores[i], label=leaf[i])
         plt.ylabel("Accuracy")
         plt.xlabel("Min Samples Split")
         plt.xticks(y_pos, split)
         plt.legend(title="Min Samples Leaf", prop={"size": 15})
         plt.show()
```

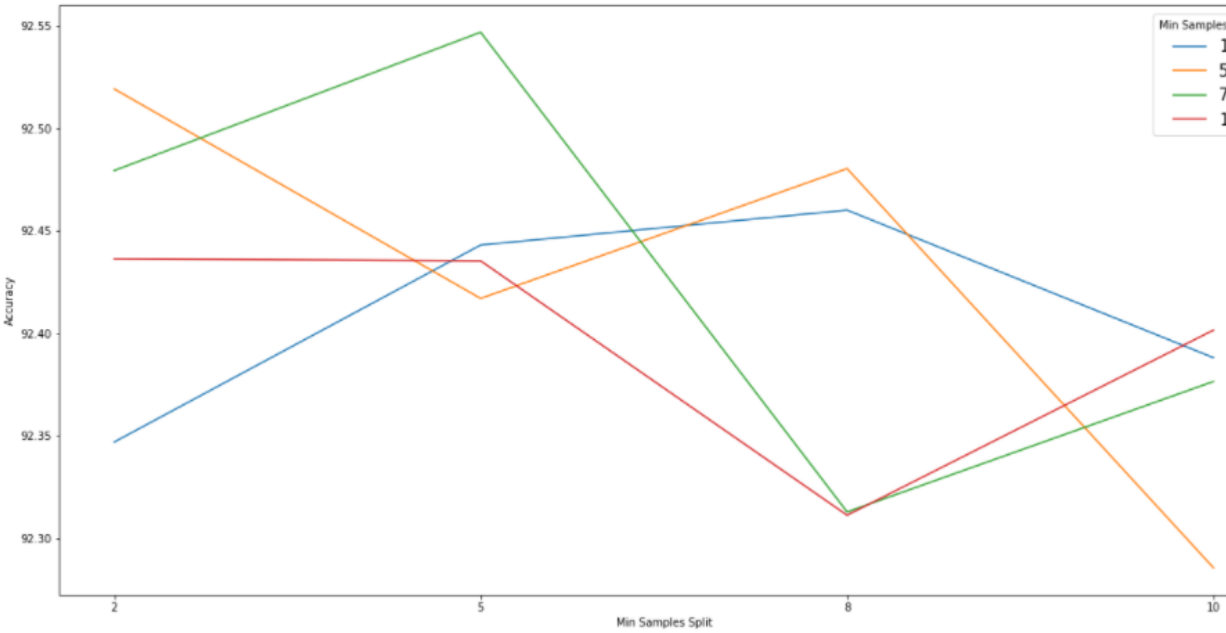
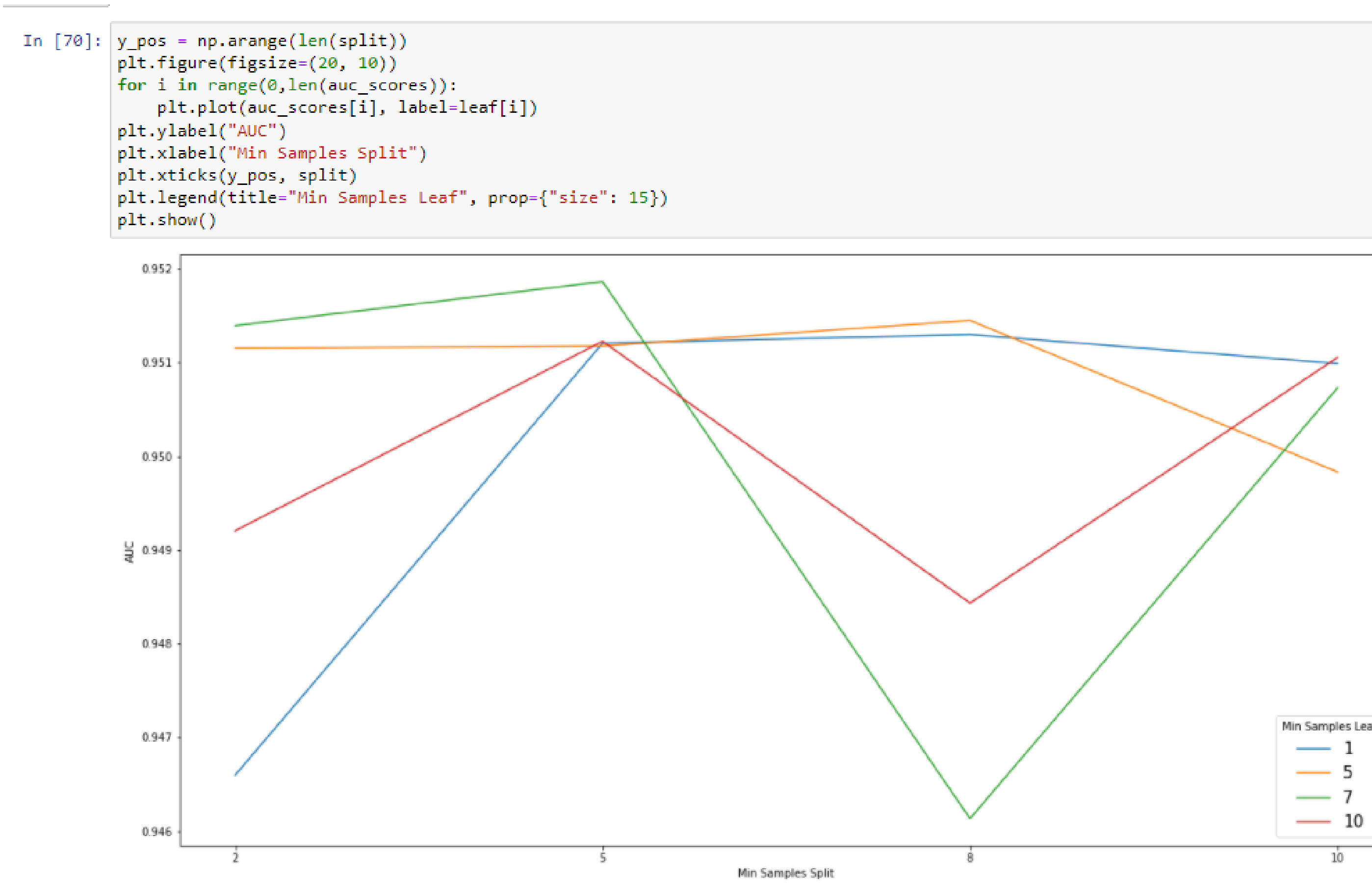
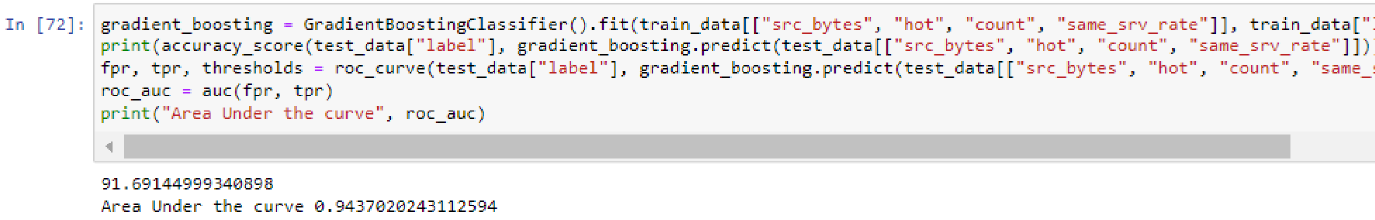


Figure 19: Split and leaf Accuracy curve



```
In [70]: y_pos = np.arange(len(split))
         plt.figure(figsize=(20, 10))
         for i in range(0,len(auc_scores)):
             plt.plot(auc_scores[i], label=leaf[i])
         plt.ylabel("AUC")
         plt.xlabel("Min Samples Split")
         plt.xticks(y_pos, split)
         plt.legend(title="Min Samples Leaf", prop={"size": 15})
         plt.show()
```

Figure 20: Split and leaf AUC Curve



```
In [72]: gradient_boosting = GradientBoostingClassifier().fit(train_data[["src_bytes", "hot", "count", "same_srv_rate"]], train_data["
         print(accuracy_score(test_data["label"], gradient_boosting.predict(test_data[["src_bytes", "hot", "count", "same_srv_rate"]])
         fpr, tpr, thresholds = roc_curve(test_data["label"], gradient_boosting.predict(test_data[["src_bytes", "hot", "count", "same_
         roc_auc = auc(fpr, tpr)
         print("Area Under the curve", roc_auc)

         91.69144999340898
         Area Under the curve 0.9437020243112594
```

Figure 21: Gradient Boosting important variable tunning

was decided to concentrate on the IaaS layer, which is a component of the networking security layer. The selection of IaaS could be driven by the fact that almost all subsequent layers are constructed on top of it. Significantly, the choice to prioritize network security is predicated on its practical significance as a core part of the Cloud, which means that any flaws in the network have a dramatic and direct influence on the overall protection of the Cloud. As has already been emphasized all through the research, a variety of various incursions and assaults can have an effect on overall network security, implying that Cloud network security could be improved by implementing more standard defence mechanisms, such as IDSs and firewalls. We're working classification that is binary, which means determining if a connection is an attack or it is not an attack. Columns in our data set are labelled as 'normal' or as the sort of attack. As a result, we must designate normal connections like one category and all attack kinds as another. If the connection is OK, the updated class label would be 0, otherwise it would be 1. Depending on this binary classification we used algorithms like Logistic regression, Random Forest, Gradient boosting neural network algorithms like Perceptron and Multi perceptron. After a range of diverse factors, the KDD Cup 99 dataset is chosen as the primary dataset being used because the constraints present in other datasets would not be an issue with this dataset. Importantly, this dataset has limitations, such as the fact that certain machine-learning algorithms may function inadequately in the present particular features of this collected data. The reality that KDD Cup 99 is an unbalanced dataset, with a considerable excess of cases in one category over the instances in the other, has been the most heavily influenced in this regard. An imbalanced dataset can pose substantial issues in attack categorization, whether it be at the multi-class or binary class stage, for both exhibiting certain form of subjectivity in regard to the core categories and, as either a result, contributing to minor category misclassifications. The bulk of studies in this field have focused solely on binary class misclassifications; this really is primarily due to the intricacy of multi-class classification, that necessitates the survey of various classes.

There are number of important features that we have given to Random Forest to check the performance based on the important features. Also, given Gradient boosting algorithm for checking its performance. Hence as seen the accuracy of Logistic Regression is 91.90, Gradient boosting is 92.60, Random Forest method is 92.58, perceptron is 81.53 and Multi perceptron is 93.00. Accuracy of the Multi perceptron to classify the attack is good. We have also used ROC curve and AUC here for evaluating the results. They are used to determine best cut-off which has the lowest false positive rate and the greatest genuine positive rate which is true positive. As seen ROC curve is better for Logistic Regression, Random Forest and Multiperceptron.If in our classification if any packet is attack and not classified as attack then it can harm the system. So, I have checked the ROC curve and decided the Multi perceptron will be the best classification algorithm to be used. Even though this thesis performed a thorough and rigorous investigation into the possible utilisation machine learning methods in the identification of attacks in an unbalanced IDS-related dataset, more research might be performed if time had allowed. Following is a list of additional study directions that are suggested based on the outcomes of this thesis' research:

- A need to collect additional useful data and make it publicly available due to a lack of Cloud-specific assault datasets.

- Through the use of deep learning, the research undertaken in the case of this project can be validated, verified, and improved.

- The effect of selecting features must be examined using other frequently utilized classifications.

References

Al-Maksousy, H. H., Weigle, M. C. and Wang, C. (2018). Nids: Neural network based intrusion detection system, *2018 IEEE International Symposium on Technologies for Homeland Security (HST)*, pp. 1–6.

Alliance, C. S. (2017). The treacherous 12 - top threats to cloud computing + industry insights, *cloud secur. alliance* (p.60).

Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. and Zaharia, M. (2010). A view of cloud computing, *Commun. ACM* **53**: 50–58.

Bace, R. (2001). Nist special publication on intrusion detection systems, *Nist Spec. Publ.* p. pp. 116 of 146.

Basicevic, F., Popovic, M. and Kovacevic, V. (2005). The use of distributed network-based ids systems in detection of evasion attacks, *Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference/E-Learning on Telecommunications Workshop (AICT/SAPIR/ELETE'05)*, pp. 78–82.

Buyya, R., Vecchiola, C. and Selvi, S. T. (2013). Mastering cloud computing: Foundations and applications programming.

C. N. Modi, D. R. Patel, A. P. and Rajarajan, M. (2012). Integrating signature apriori based network intrusion detection system (nids) in cloud computing, **6**: pp. 905–912.

Dayalan, M. (2019). Cloud computing – research issues, challenges, architecture, platforms and application.

Engen, V. (2010). Machine learning for network based intrusion detection: an investigation into discrepancies in findings with the kdd cup'99 data set and multi-objective evolution of neural network classifier ensembles from imbalanced data (doctoral dissertation, bournemouth university.

et al., F. P. (2016). Multilayer perceptron algorithms for cyberattack detection, pp. pp. 248–252.

J. Arshad, P. T. and Xu, J. (2013). A novel intrusion severity analysis approach for clouds, *Futur. Gener. Comput. Syst.,* **vol. 29**: pp. 416–428.

M. Armbrust, a Fox, R. G. A. J. and Rh (2009). Above the clouds: A berkeley view of cloud computing, *Univ. California, Berkeley, Tech. Rep. UCB* pp. 07–013.

Mitchell, T. M. (2006). "the discipline of machine learning," mach. learn.,, Vol. vol. 17, p. pp. 1–7.

Mladen Vouk, Sam Averitt, M. B. A. K. A. P. H. S. E. S. S. S. J. T. (2008). Powered by vcl ' - using virtual computing laboratory (vcl) technology to power cloud computing, **6**: 1–10.

Modi, C., Patel, D., Borisaniya, B., Patel, A. and Rajarajan, M. (2013). A survey on security issues and solutions at different layers of cloud computing, *J. Supercomput.* **63**(2): 561–592.
URL: *https://doi.org/10.1007/s11227-012-0831-5*

Patel, K. and Srivastava, R. (2013). Classification of cloud data using bayesian classification, **2**: pp. 2–7.

Pondel, M. and Korczak, J. (2017). A view on the methodology of analysis and exploration of marketing data, *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 1135–1143.

R. M. Yousufi, P. L. and Potdar, M. B. (n.d.). A network-based intrusion detection and prevention system with multi-mode counteractions.

Raghunath, B. R. and Mahadeo, S. N. (2008). Network intrusion detection system (nids), *2008 First International Conference on Emerging Trends in Engineering and Technology*, pp. 1272–1277.

S. Roschke, F. C. and Meinel, C. (2009). "an extensible and virtualization-compatible ids management architecture," 5th int. conf. inf. assur. secur. ias 2009, Vol. vol. 2, p. pp. 130–134.

Sadiku, M. N. O., Musa, S. M. and Momoh, O. D. (2014). Cloud computing: Opportunities and challenges, *IEEE Potentials* **33**: 34–36.

Scarfone, K. and Mell, P. (n.d.). A guide to intrusion detection and prevention systems (idps) recommendations of the national institute of standards and technology.

Sommer, R. and Paxson, V. (2010). "outside the closed world: On using machine learning for network intrusion detection", p. pp. 305–316.

T. W. Purboyo, B. R. and Kuspriyanto (2011). Security metrics: A brief survey, *Int. Conf. Instrumentation, Commun. Inf. Technol. Biomed. Eng ., no. November* p. p. 4.

Y. j. Ou, Y. Lin, Y. Z. and j. Ou, Y. (2010). "the design and implementation of host-based intrusion detection system," 2010 third international symposium on intelligent information technology and security informatics, pp. pp. 595–598.

Z. Erl, T., P. R. . M. (2015). Cloud computing: Concepts, technology architecture, **Prentice Hall PTR.**

Zhang, Q., Cheng, L. and Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges, *Journal of Internet Services and Applications* **1**: 7–18.