Turkish Journal of Computer and Mathematics Education Vol.12 No. 11 (2021), 1762- 1776
Research Article

Pre-Processing of KDD'99 & UNSW-NB Network Intrusion Datasets

Inadyuti Dutt [a], Samarjeet Borah [b], Indra Kanta Maitra [c]

[a] Part-time Research Scholar, [b] Professor, Sikkim Manipal Institute of Technology, Sikkim Manipal University, Sikkim, India
[c] Controller of Examination, St. Xavier's University, Kolkata 700160, India

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 10 May 2021

Abstract: An IDS is essential for securing the network. The IDS not only traverses the header portion of the network packet but also inspects the data portion or the payload of the network packet. Therefore, if at all any malicious code is present in the data portion, it would be detected by the IDS and the packet will be denied access to the network or removed accordingly. Various datasets on computer and network intrusions are available publicly and are extensively used in developing successful intrusion detection systems. In this development process pre-processing plays an important role to make the data ready for the prediction process. This paper presents and analyses on two widely used datasets - KDD'99 and UNSW-NB 15 in intrusion detection. Pre-processing of both the datasets is performed considering missing, redundant and noisy data along with the data cleaning process using Weka data miner tool. A brief discussion on a newly created dataset is also presented in the paper.

Keywords: Intrusion Detection, Dataset, Pre-processing, KDD'99, UNSW-NB 15, Analysis

1. Introduction

Security of interconnected networks is at frequent risk because of over-reliance of the user community on these services. The data that is used by the user community are constantly transmitted to a widespread of network available to them. The network underlying these facilities faces lot of security threats due to the intrusive activities either initiated by illegitimate users or legitimate users who somehow manage to misuse their access rights. Therefore, it becomes necessary to find correct measures or deploy tools that would constantly protect the network. Researchers and network security professionals across the world have devised variety of software as well as hardware devices to address this issue. Firewalls are widely used to monitor the internet traffic invading the internal system of an organization. However, it falls short in limiting the misuse of the internal traffic of the system. For this reason, organizations have been using Intrusion Detection System (IDS) that would monitor both the internal and external traffics of the network. IDS are software tools that dynamically monitor, record the network packets and apply recognition algorithms to identify intrusions in a network (**Hofmeyr S. A. et al. 2000**). Several intrusion detection systems are available in literature which are basically designed based on publicly available **datasets (Wang K. et al., 2004, Dutt I. et al., 2020, Borah S. et al., 2018, Dutt I. et al., 2015, Panigrahi R. et al., 2019, Dutt I. et al., 2019, Borah S. et al., 2011)**.

1.1 IDS Datasets

The network is able to transmit enormous quantity of data that can be used as datasets for intrusion detection and additional analysis. The network data is transformed into meaningful structure referred as connections or connection records. Each connection has numerous features that are specific to particular domain knowledge. This paper entails the two publicly available and widely used standard datasets. The datasets are **KDD'99 (Stolfo et al., 1999, Tavallaee M. et al., 2009, Gunes Kayacık A. et al., 2005)** and UNSW-NB 15 **(Moustafa N. 2015, Moustafa N. et al., 2019)** These datasets have been taken into consideration in many research works for developing intrusion detection systems. KDD'99 is the Knowledge Discovery in Databases, well-known benchmark dataset for intrusion detection. UNSW-NB 15 is prepared using IXIA PerfectStorm **(14)** tool of the "Cyber Range Lab of the Australian Centre for Cyber Security" in order to create real-time day to day activities and simulated attacks in accordance to the current network scenarios. In addition to these two datasets, real-time data has been captured for a three weeks' duration to create an additional dataset. As this paper is representing a portion of a research project on network intrusion detection through resource consumption, emphasis has been given on capturing such kind of data. Primarily, data on network bandwidth, CPU and memory usages are considered for the same. Following sections of the papers presents a brief description on the datasets along with the pre-processing results.

2. Public Datasets

The public datasets that are most widely used are considered here. The descriptions of them are as follows:

2.1 KDD'99 Dataset

KDD'99 dataset is used for evaluating the intrusion detections. This set of data, created by **(Stoflo et al. 1999)** and has primarily raw network data of "DARPA'98 IDS" evaluation program **(Lippmann R et al. 2000)**. "DARPA'98" consisted of network traffic data with around 5 lakhs of connection records. KDD training-dataset comprises of around 4,900,000 single-connection records. This dataset comprises of connection records. With each such record consisting of information related to a session between a "source" and "destination IP address" over a pair of ports. These records can be either labelled as "normal" or "attack" types. There are 22 types of attacks that may be present in any of the connection records. There are 41 features in each connection record with 31 continuous and 7 discrete-valued respectively. The features are again grouped into basic, traffic and content features. The basic features have originated from the header information whereas the traffic and content features have been extracted from the "payload portion" of the network packets.

The attacks that are considered are based on the attack types. Following are the types of attacks seen in the simulated environment:

a) "Denial of Service Attack (DoS)": This attack takes place when a masquerader attempts to make the system unnecessarily busy so that the legitimate user is unable to access the system. The attacks that fall under this category are "back", "land", "Neptune", "smurf", "pod" and "teardrop".
b) "User to Root Attack (U2R)": This takes place when an intruder tries to invade any authorized user account by sniffing passwords or any programming errors done by the legitimate user. The main intention of the intruder is to access the root. In this category there are attacks namely "Loadmodule", "buffer overflow", "rootkit" and "perl".
c) "Remote to Local Attack (R2L)": This intrusion takes place when the intruder attempts to access the host machine remotely. These attacks are namely "phf", "guess_passwd", "warezmaster", "imap", "multihop", "ftp_write", "spy" and "warezclient".
d) Probing Attack: Such intrusion occurs when the intruder tries to obtain information about the network by abusing the host machine and looking deliberately for exploits. The attacks that fall under this category are satan, nmap, portsweep, ipsweep.

Features of KDD'99 are categorised into 3 groups:

a) "Basic features": These have been captured from the header portion of the TCP/IP connection records. Features in this category are from (*f1–f9*). These are simple features that are basically derived from the flow measurements on routers. Some are derived from the comparison of values and others from status of the connections.
b) "Content features": They have been derived from the "payload" section of the network packet. These features *(f10-f22)* need the entire packet to be captured and thus are costlier than the packet headers. These features become cumbersome to be obtained as they need to be inspected in the data portion and then needs to be reassembled. Moreover, if the communication has been encrypted end to end, then the payload portion could be decrypted on the host and not in the network. Therefore, the extraction of the payload portion at all the host machines needs more effort.
c) "Traffic features": The traffic features primarily take care of these aspects i) the features are computed within a window interval and considers ii) either the "same host" features where the connections are within the window interval of 2 seconds and to the "same destination host" as the "current connection" iii) or the "same service" features where the "connections within the window interval of 2 seconds" and have the "same service" as that of the "current connection". These kinds of attributes that rely on the window time interval of 2 seconds are referred to as "time-based" traffic features (*f23–f31)*. However, there are certain attacks that do not exhibit patterns within the 2 seconds of time interval and cannot be estimated with the "same host" or "same features" connections. Therefore, for these cases, "connection window" of "100 connections" to the "same destination" regardless of the "time window" of 2 seconds have been taken into consideration. The features related to this category are called "connection-based" traffic features.

KDD dataset faces lot of challenges in terms of data redundancies as there are multiple records for a single instance. This causes biasness towards some records that are more frequent than others thereby providing it difficult for the learning algorithm to predict the correctly the attacked as well as normal data. These infrequent records become harmful for attacks like U2R and R2L where the intruder sends small number of records for accessing the system directly or remotely.

For considering the KDD dataset, the redundant records have been removed for the better detection of infrequent records. Following is the table (Table 1) that represents the statistics of the redundant records and the total unique records that are considered for the proposed work:

Table 1: Redundant and Total Records

Category	Original Records	Unique Records
Attacked Records	39,25,650	2,62,178
Normal Records	9,72,781	8,12,814
Total Records	48,98,431	10,74,992

2.2 UNSW–NB15 Dataset

This dataset has 49 features that can be grouped as: "Basic", "Content" and "Time". There are some more features that are regarded as the "general purpose" and "connection-based features". They constitute both the "header" and the "payload" section of the pack. These can be categorized as mentioned below:

a) "Flow-based Features": They constitute the first five features of the packet flow i.e. "direction", "inter-arrival time" and "inter-packet length". These have taken into consideration only those packets that go through a network link. These include features from *f1* to *f5* (Table 2).

Table 2*:* Flow-Based Features

Fr. No.	Feature (s)	Description
f1	"Srcip"	"Source IP address"
f2	"Sport"	"Source port number"
f3	"Dstip"	"Destination IP address"
f4	"Dsport"	"Destination port number"
f5	"Proto"	"Transaction protocol"

The next features are "*Basic", "Content" and "Traffic"*-based features that are gathered from the payload portion of the packet. They constitute the *f6* - *f35* features of the data packets.

b) "Basic Features": These represent protocol connections and constitute the features from *f6* to *f18* (Table 3).

Table 3: Basic Features

Fr. No.	Feature (s)	Description
f6	"State"	"The state and its dependent protocol ACC, CLO, else (-)"
f7	"Dur"	"Record total duration"
f8	"Sbytes"	"Source to destination bytes"
f9	"Dbytes"	"Destination to source bytes"
f10	"Sttl"	"Source to destination time to live"
f11	"Dttl"	"Destination to source time to live"
f12	"Sloss"	"Source packets retransmitted or dropped"
f13	"Dloss"	"Destination packets retransmitted or dropped"
f14	"Service"	"http, ftp, ssh, dns,..., else(-)"
f15	"Sload"	"Source bits per second"
f16	"Dload"	"Destination bits per second"
f17	"Spkts"	"Source to destination packet count"
f18	"Dpkts"	"Destination to source packet count"

c) "Content Features": These features mostly encapsulate the contents of TCP/IP and some features of TCP/IP (Figure 4).

Table 4: Content Features

Fr. No.	Feature (s)	Description
f19	"Swin"	"Source TCP window advertisement"
f20	"Dwin"	"Destination TCP window advertisement"
f21	"Stcpb"	"Source TCP sequence number"
f22	"Dtcpb"	"Destination TCP sequence number"
f23	"Smeansz"	"Mean of the flow packet size transmitted by the src"
f24	"Dmeansz"	"Mean of the flow packet size transmitted by the dst"
f25	"Trans_depth"	"the depth into the connection of http request/response transaction"
f26	"Res_bdy_len"	"The content size of the data transferred from the server's http service"

"Time Features": These features are dependent on time say "arrival time" between packets, "start/end packet time" and "round trip time" of "TCP protocol" (Table 5).

Table 5: Time Features

Fr. No.	Feature (s)	Description
f27	"Sjit"	"Source jitter (mSec)"
f28	"Djit"	"Destination jitter (mSec)"
f29	"Stime"	"Record start time"
f30	"Ltime"	"Record last time"
f31	"Sintpkt"	"Source inter-packet arrival (mSec)"
f32	"Dintpkt"	"Destination inter-packet arrival time (mSec)"
f33	"Tcprtt"	"The sum of 'synack' and 'ackdat' of the TCP"
f34	"Synack"	"The time between the SYN and the SYN_ACK packets of the TCP"
f35	"Ackdat"	"The time between the SYN_ACK and the ACK packets of the TCP"

There are twelve additional attributes in this dataset that constitute features *(f36 - f47)*. The "general purpose features" *(f36 - f40)* and "connection-based features" *(f41 - f47)*. The "general purpose features" are used to shield the protocol services. The "connection-based features" allow to capture the flow for 100 numbers record connections sequentially that were last time featured.

d) "General Purpose Feature":

Table 6: General Purpose features

Fr. No.	Feature (s)	Description
f36	"is_sm_ips_ports"	"If source equals to destination IP addresses and port numbers are equal, this variable takes value 1 else 0"
f37	"ct_state_ttl"	"Number for each state according to specific range of values for source/destination time to live"
f38	"ct_flw_http_mthd"	"Number of flows that has methods such as Get and post in http service"
f39	"Is_ftp_login"	"If the ftp session is accesses by user and password then 1 else 0"
f40	"ct_ftp-cmd"	"Number of flows that has a command in ftp session"

e) "Connection-Based Features":

Table 7: Connection-Based Features

Fr. No.	Feature (s)	Description
f41	"ct_srv_src"	"Number of connections that contain the same service and source address in 100 connections"
f42	"ct_srv_dst"	"Number of connections that contain the same service and destination address in 100 connections according to the last time"
f43	"ct_dst_ltm"	"No. of connections of the same destination address in 100 connections according to the last time"
f44	"ct_src_ltm"	"No. of connections of the same source address in 100 connections according to the last time"
f45	"ct_src_dport_ltm"	"No. of connections of the same source address (1) and the destination port in 100 connections according to the last time"
f46	"ct_dst_sport_ltm"	"No. of connections of the same destination address and the source port in 100 connections according to the last time"
f47	"ct_dst_src_ltm"	"No. of connections of the same source and the destination address in 100 connections according to the last time"

UNSW-NB 15 exhibits nine types of attacks that are discussed below:

a) *Fuzzer Attacks*: These attacks take place when the intruder tries to find out security defects in any platform whether it can be operating system, application or any networking environment by incorporating massive volume of data to jeopardize the system

b) *Analysis Attacks*: These types of attack can occur via ports, emails or web scripts so as to penetrate the web applications, the e-mail user system or the HTML files.

c) *Backdoor Attacks*: These attacks take place when an intruder attempts to invade a system illegitimately using unauthorized remote access and keeps itself unnoticed till the last execution of a command.
d) *Denial of Service Attacks*: The attack of this type tries to disrupt the system by continuously accessing the computer resources so as to stop the legal user to access the system.
e) *Exploit Attacks*: The attack tries to take advantage of a technical snag, error or a bug thereby resulting victim user of the system remains unaware of the behaviour.
f) *Generic Attacks*: This attack can take place "against every block-cipher using a hash function to cause a collision without respect to the configuration of the block-cipher".
g) *Reconnaissance Attacks*: These are considered to be probes that try to capture information related to a system or a network so as to evade the security controls.
h) *Shellcode Attacks*: This type of codes tries to penetrate a system from the shell in order to compromise the host machine.
i) *Worm Attacks*: Worms try to replicate themselves so that they may proliferate to other computer resources. They may often use the network to spread themselves and eventually compromise the target computer.

3. A Real-Time Dataset

Real-time network traffic has been captured for generating a dataset for detecting intrusion. The incoming files for three weeks are considered for the experiment. The performance logs of the system based on Network bandwidth, CPU and Memory usages are captured after every five minutes of time period. The real-time traffic with mean sample size of incoming files was 30,377. For a specific day, maximum number of files considered was 30,390. The average population size is approximately 2,20000. Unknown "HTTP traffic files" are also reflected which can be used by the researchers for performing vulnerability tests. Mean/average sample size of the HTTP files is appx. 3,000/day. The USB driver, audio driver and graphics drivers are disabled to create the intruded data set.

Considering the sample set of observations at a time interval of 5 minutes - T_n *(*T_1*,* T_2*,* T_3 *to* T_4*)* for system resource features such as CPU, RAM and network bandwidth usage are as in Table 8:

Table 8: Observations on Uses of System Resources

System Resource Usage	T_1	T_2	T_3	T_4
Network Bandwidth usage	90	100	90	100
CPU usage	120	110	90	110
RAM usage	110	120	100	120

Uses information given above are in percentage. Table 9 characterizes reduced observations (dividing by a constant say10) from the above observation table.

Table 9: Set of Observations After Reduction

System Resource Usage	T_1	T_2	T_3	T_4
Network Bandwidth usage	0.9	0.10	0.9	0.10
CPU usage	0.12	0.11	0.9	0.11
RAM usage	0.11	0.12	0.10	0.12

On capturing the suspicious traffic, the packets have been segregated into two portions – header and the payload respectively. The header portion of the packet considers the "*packet_sequence_number", "source_port", "destination_port", "source_address", "destination_address", "length_of_the packet"* and "*protocol_type"*, whereas the data portion includes the raw-data in binary representation.

4. Pre-Processing of Datasets

Pre-processing of data becomes crucial in eradicating the noisy, missing and inconsistent data available in the dataset. Since the records of the dataset are collected from multiple and heterogeneous sources, the quality of the data deteriorates and therefore needs to be pre-processed. Several factors affect the quality of the data. These factors comprise accuracy, completeness, consistency, timeliness, believability and interpretability.

Data can be inaccurate for number of reasons. It may be due to the human or computer errors that had occurred during the data entry. Errors might also occur during data transmission. There may be duplicate records that need to be cleaned. Incomplete data can be of different forms. It may be due to the unavailability of the data during the time of entry. Or, otherwise it may be due to the malfunctioning of equipment that captured the data. Moreover, the data might be overlooked or may be misunderstood. In this section, the same has been discussed using KDD'99 and UNSW-NB 15 datasets as an example. The task is performed using WEKA data mining tool.

4.1 Missing Data

Data can be missing due to the varied data mining scenarios that happened while capturing the data. Missing data could depict the unavailability of the data itself, the absence of the event or the inapplicability of the field in the context of experiment. The policies that have been used to overcome the missing values are:

a) *Ignore the records with missing values*: This becomes feasible when the class label is missing. In this case, the standard KDD99 records had missing values in the class label which was accomplished by matching the most relevant attribute values with the records missing the class labelled.

b) *Filling up values manually based on the domain knowledge acquired*: This can be more time-taking and perhaps become infeasible for the datasets that are quite large.

c) *Using "WEKA* (Waikato Environment for Knowledge Analysis)" *Tool feature*: Missing data that were unavailable and could not be assumed by matching with the most relevant ones was removed with the help of WEKA, by choosing the *RemoveWithValues* Filter option.

4.2 Redundant Data

KDD99 have lot of redundant records for a particular record instance. This may cause biasness during actual detection of intrusion as it may be quite difficult for the learning algorithm to arrive at a particular conclusion that an attack has been identified or not. The records that are infrequent in nature are rather harmful for attack detections like U2R and R2L where the intruders try to send few packets for accessing the system directly or remotely. Duplicate records have been removed using WEKA's unsupervised *RemoveDuplicates* filter.

4.3 Noisy Data

Noisy data are random error or variance that can be handled using the statistical techniques with measures of central tendency. WEKA tool was used for analysing the noisy or meaningless data that often contribute to Extreme Values and Outliers. *InterQuartileRange* filter was used to find the extreme and outlier values Figure 1.

Figure 1: Represents the Extreme Values for duration attribute of KDD dataset

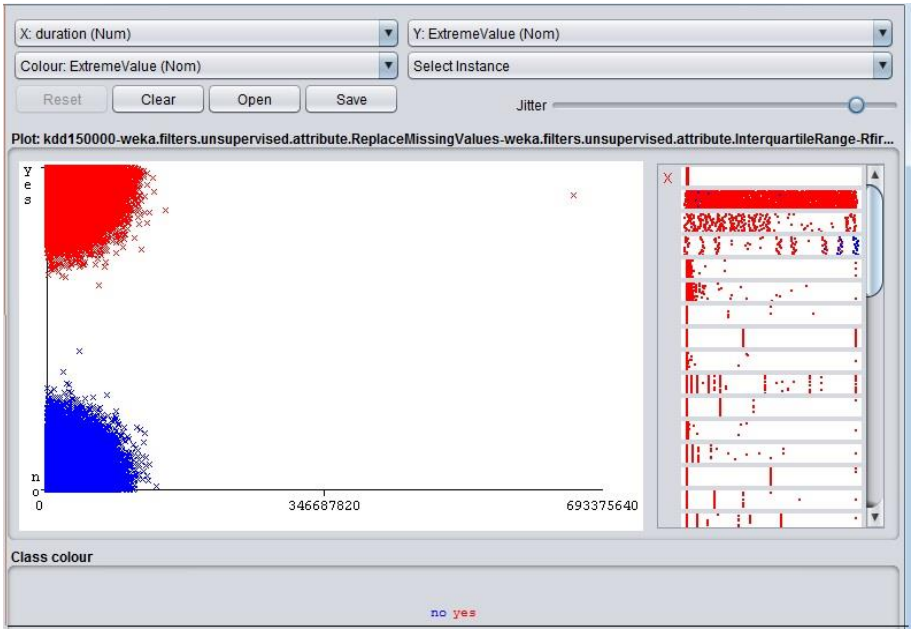


Figure 2: Represents the Extreme Values for src_bytes attribute of KDD dataset

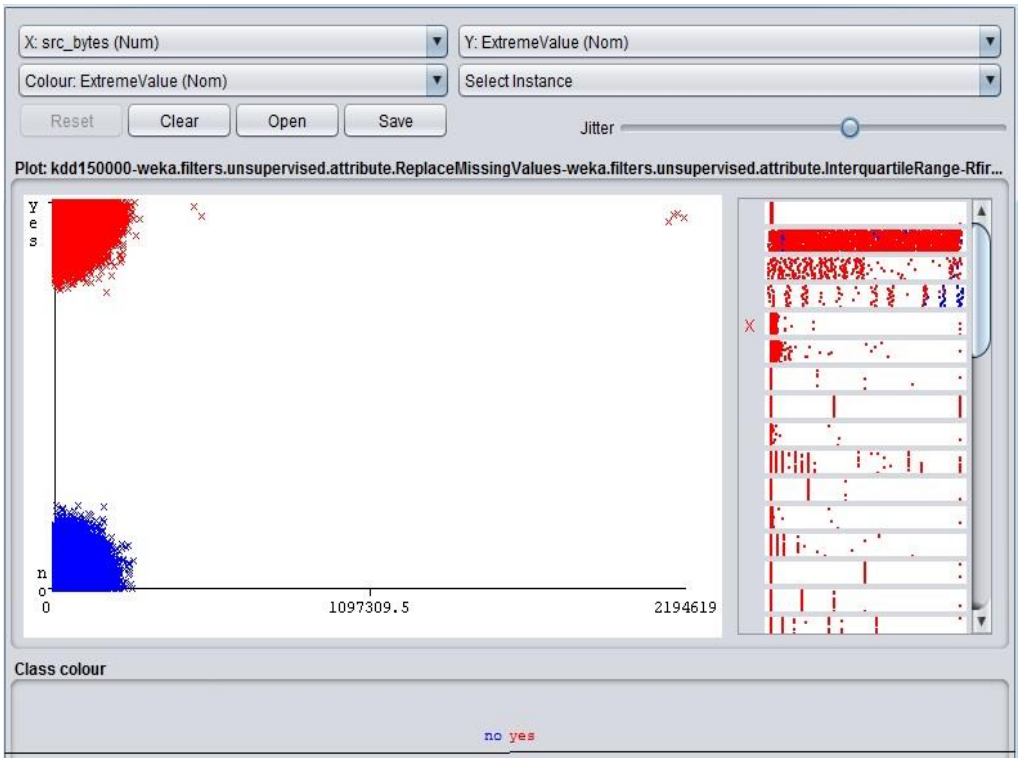


Figure 3: Represents the Outlier Values for duration attribute of KDD dataset

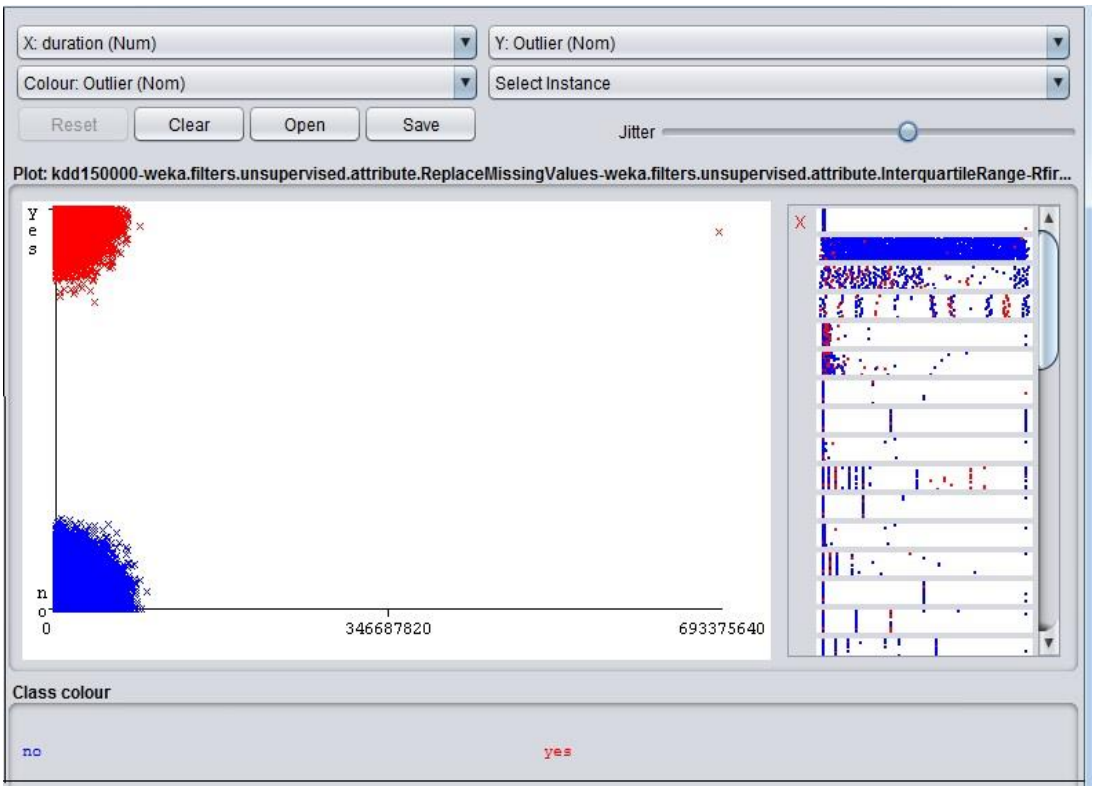
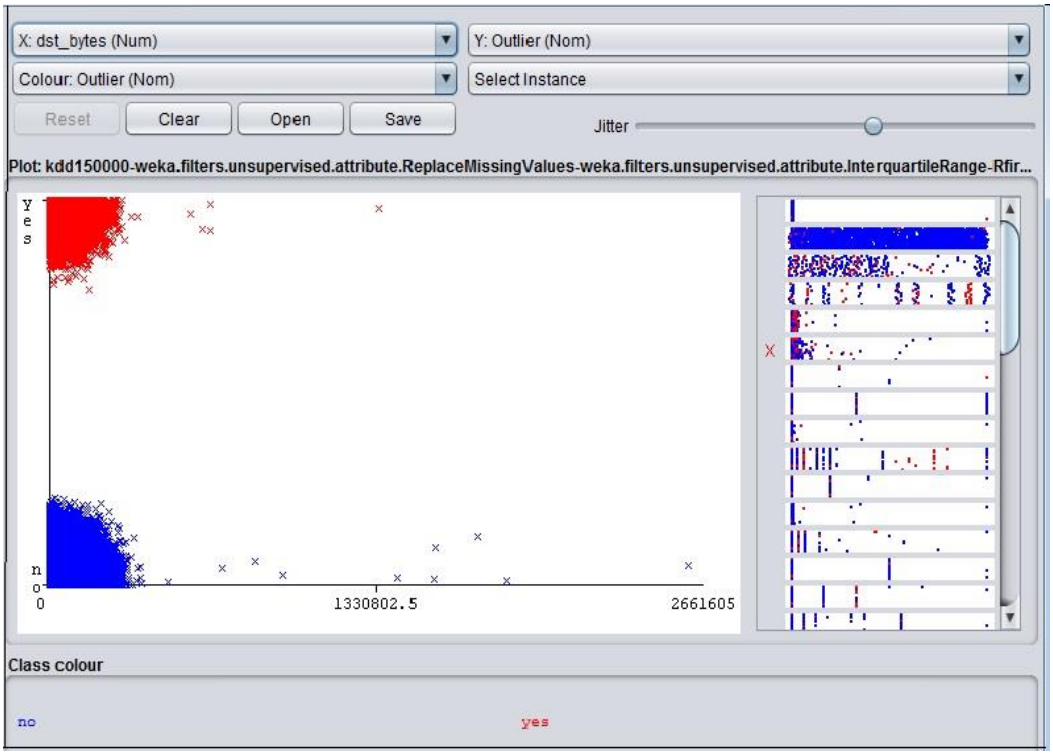
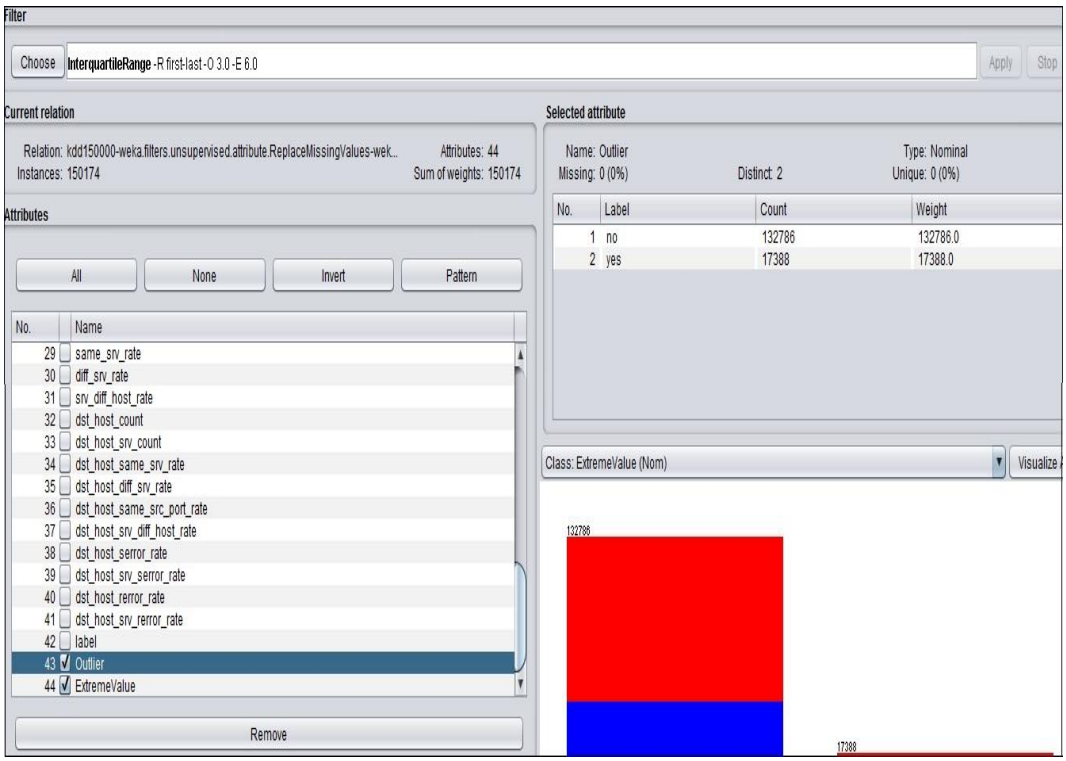


Figure 4: Represents the Outlier Values for dst_bytes attribute of KDD dataset



Then the extreme values and outliers were removed with *RemoveWithValues* Filter option in WEKA (Figure 5).

Figure 5: Represents the removal of Extreme and Outlier values of KDD dataset



Noisy data in UNSW-NB 15 dataset are visualized with the help of extreme values and outlier values. Figure 6 and Figure 7 show the extreme values for the "dpkts" and "spkts" features of the dataset:

Figure 6: Represents the Extreme Values for dpkts attribute of UNSW-NB15 dataset

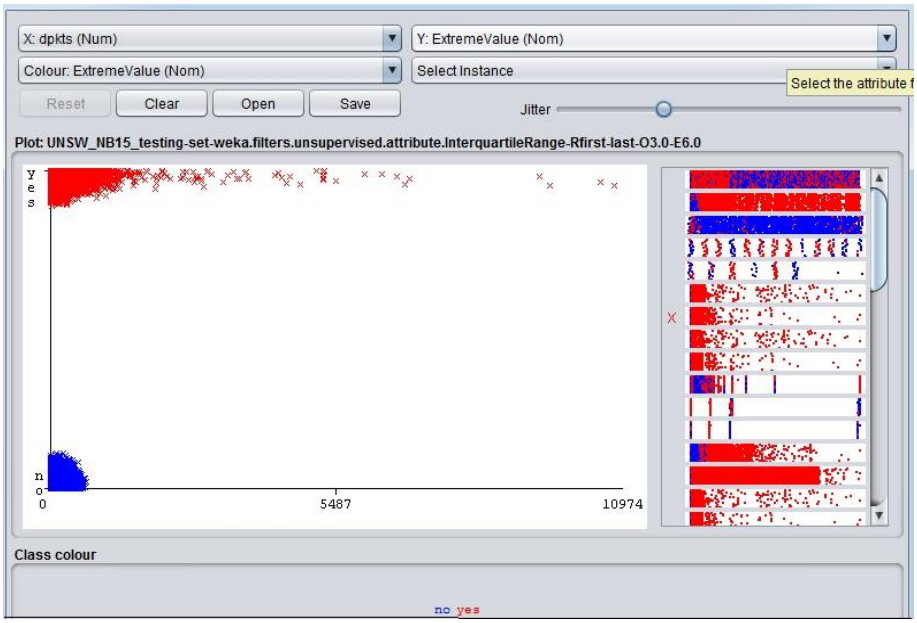
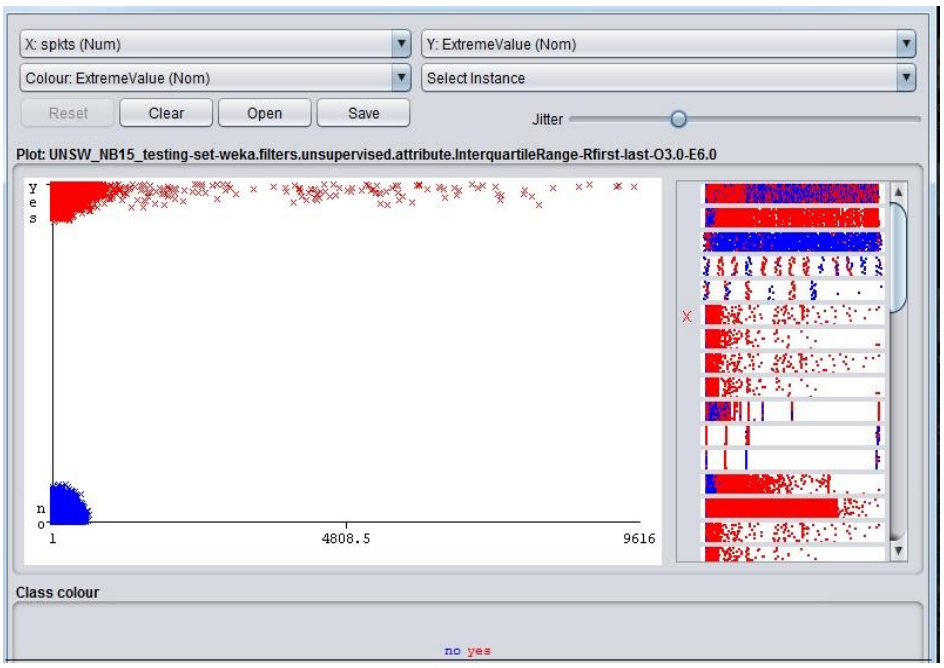


Figure 7: Represents the Extreme Values for spkts attribute of UNSW-NB15 dataset



Following Figure 8 and Figure 9 show the Outlier Values of the features "sload" and "dload".

Figure 8: Represents the Outlier Values for sload attribute of UNSW-NB 15 dataset

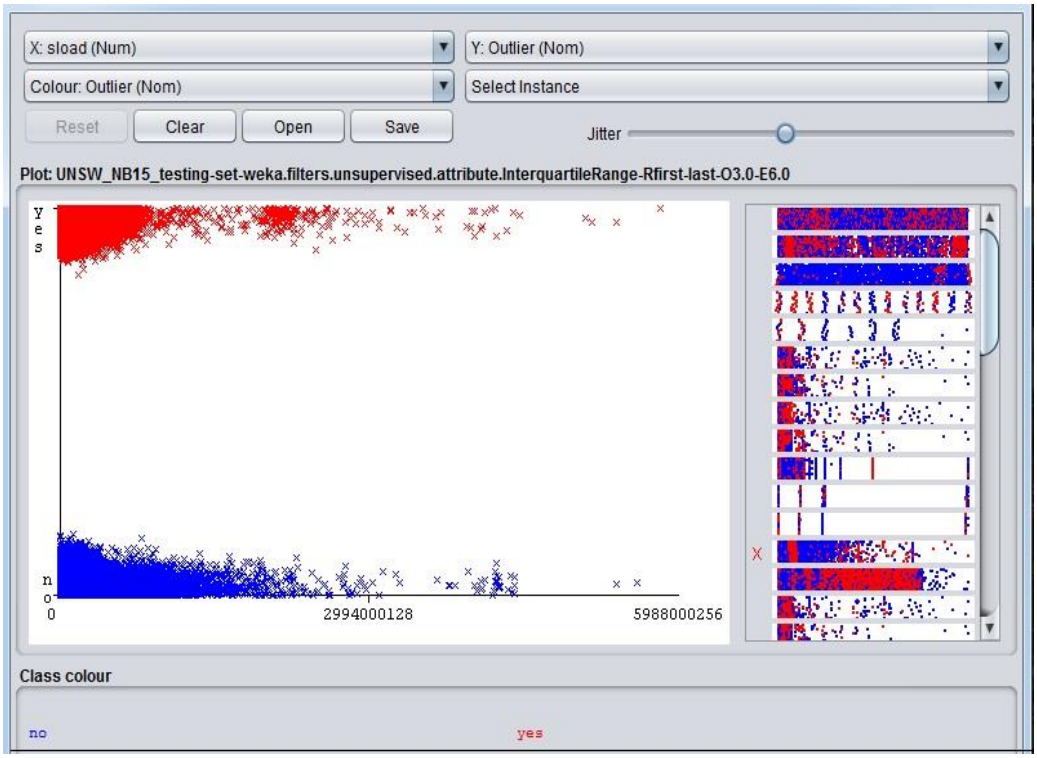
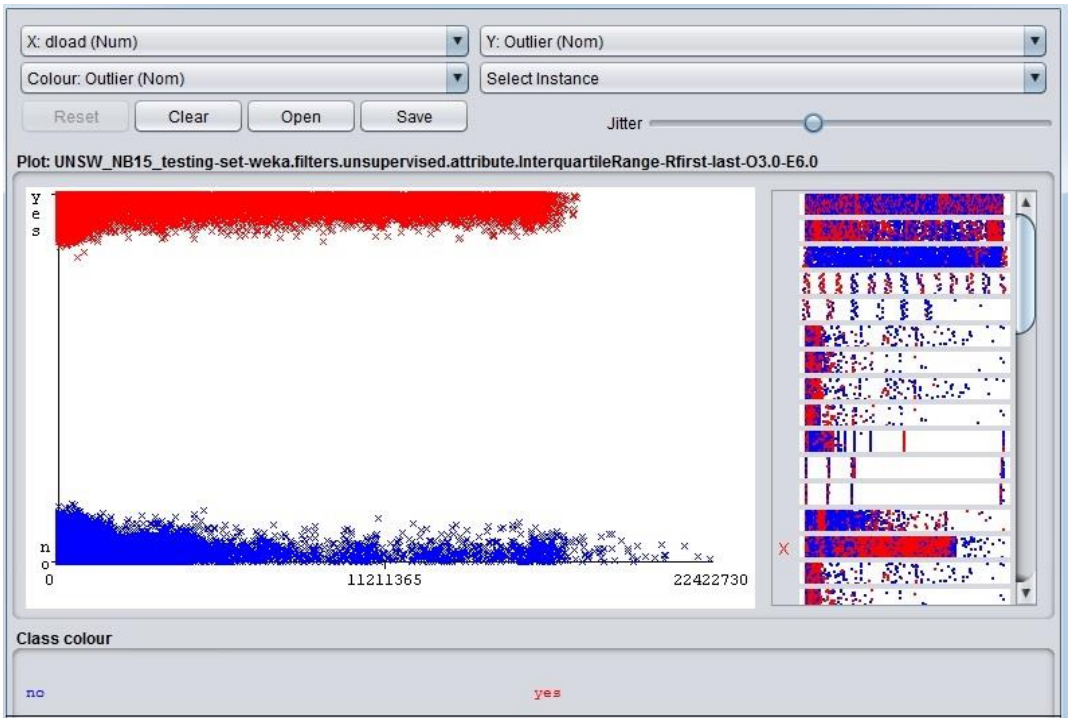


Figure 9: Represents the Outlier Values for dload attribute of UNSW-NB 15 dataset



Then the extreme values and outliers were removed with *RemoveWithValues* Filter option in WEKA (Figure 10 and Figure 11). Figure 11 shows the dataset after removal of these values.

Figure 10: Represents the removal of Extreme and Outlier values of UNSW-NB 15 dataset

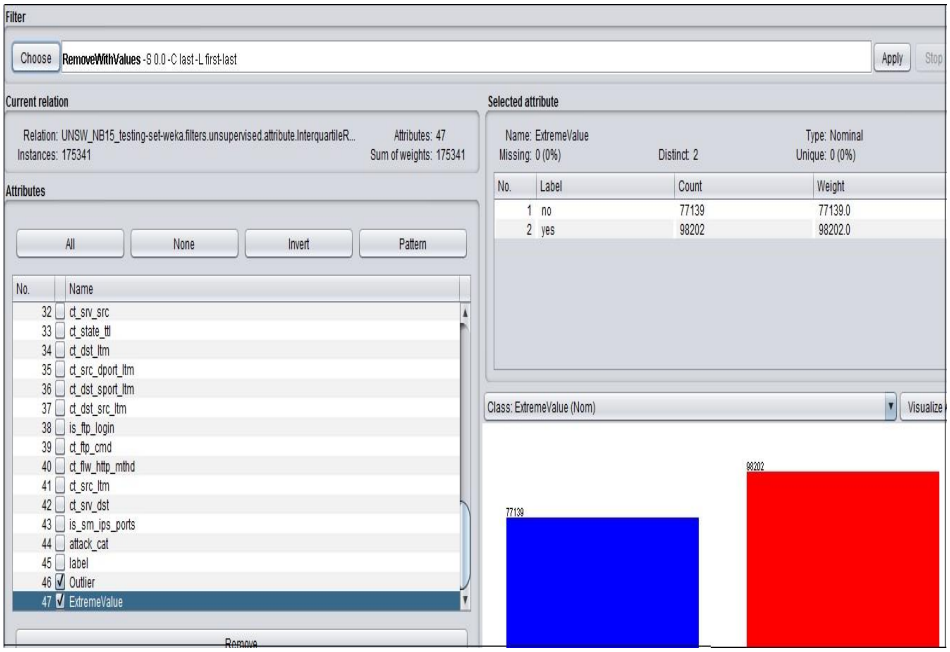
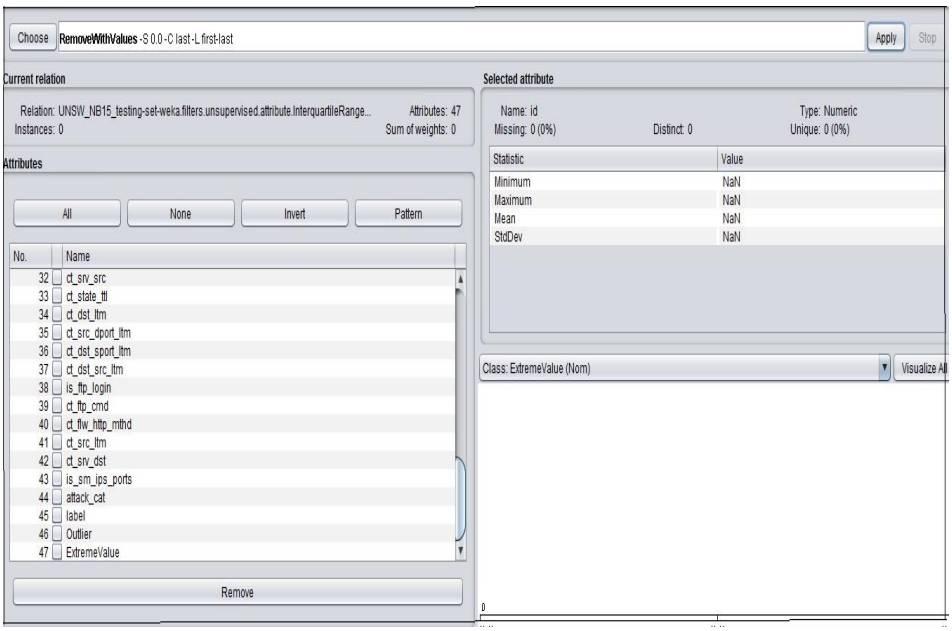


Figure 11: Represents after the removal of Extreme and Outlier values of UNSW-NB 15 dataset



4.4 Data Cleaning

For the removal of invalid records in the standard KDD dataset, some records were removed from the original KDD data set in the reduction phase of the data cleaning. The KDD set has "125,973 training" and "22,544 testing". The main challenge while processing the data, is to consider how much discretization is required without much loss of the important information. In order to discretize properly, the WEKA tool has been used to discretize the real-valued data features. In KDD99 dataset, there are 34 features that are real-valued, continuous. For discretizing them, the *Discretize* option is used in the Filter of WEKA. The number of bins that are used for each features are 10. Following Figure 12 shows the discretization of the feature "hot" with bin width as 3. Figure 13 shows another feature "num_failed_logins" where the number of bins

considered is 10 and bin size is 0.5. For binary and nominal features, discretization is not needed as the data can be easily mirrored to the binary equivalent of the feature values.

Figure 12: Represents the discretization of the feature "hot" with bin width 3

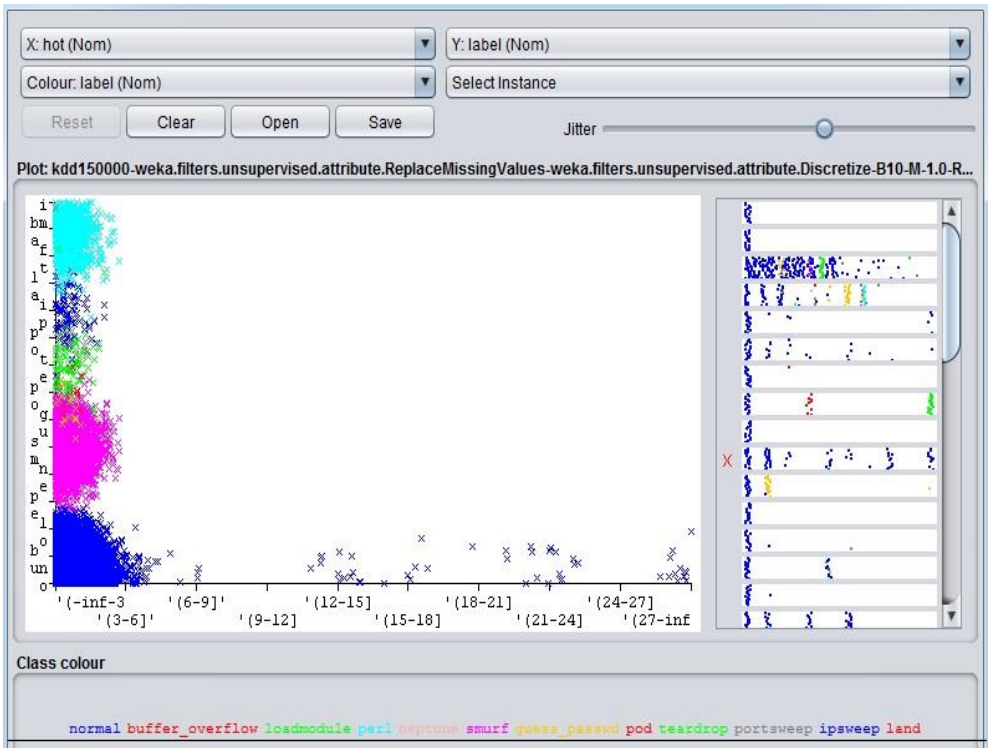
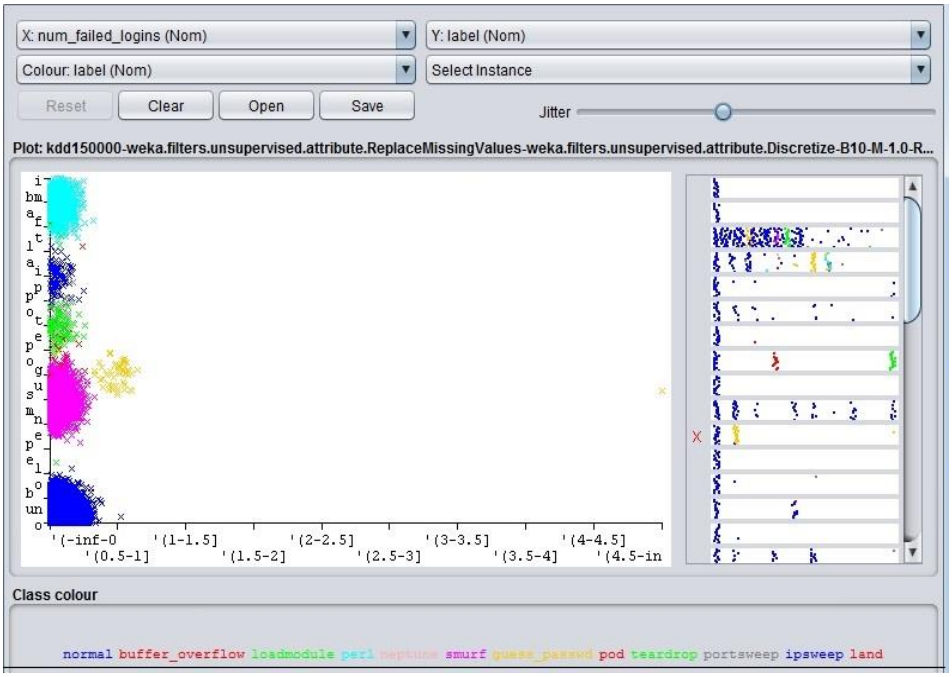


Figure 13: Represents the discretization of the feature "num_failed_logins" bin size is 0.5



For discretization of UNSW-NB 15 dataset, the WEKA tool has been used. Figures 14, 15, 16 and 17 show the discretization of feature "dur", "spkts", "dpkts" and "rate" with number of bins 10 and equal-sized bin width.

Figure 14: Represents the discretization of the feature "dur" with equal bin size

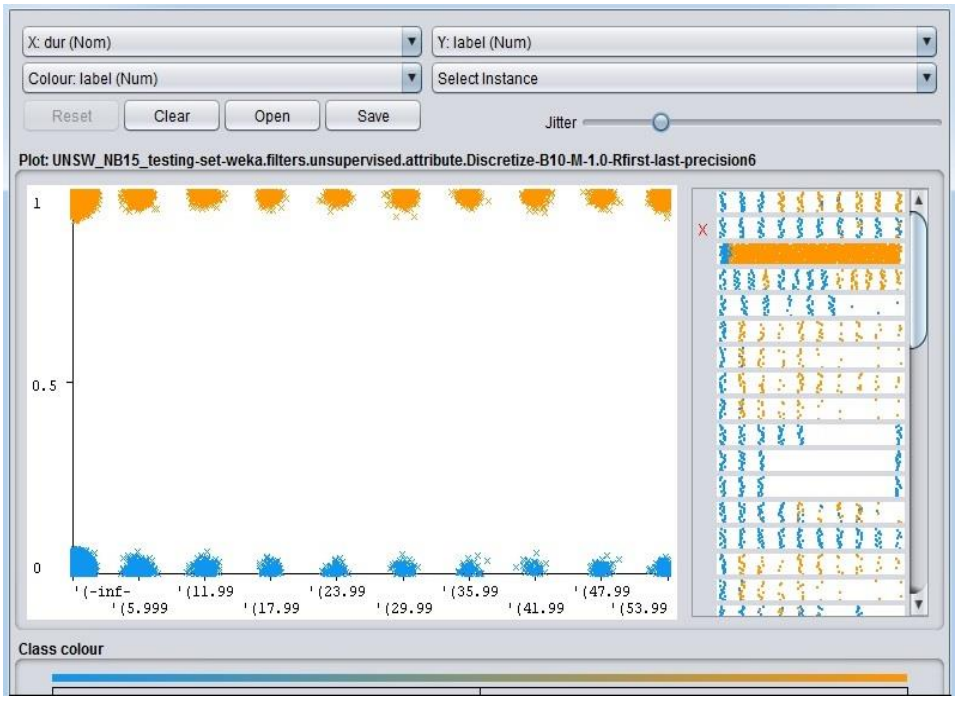


Figure 15: Represents the discretization of the feature "spkts" with equal bin size

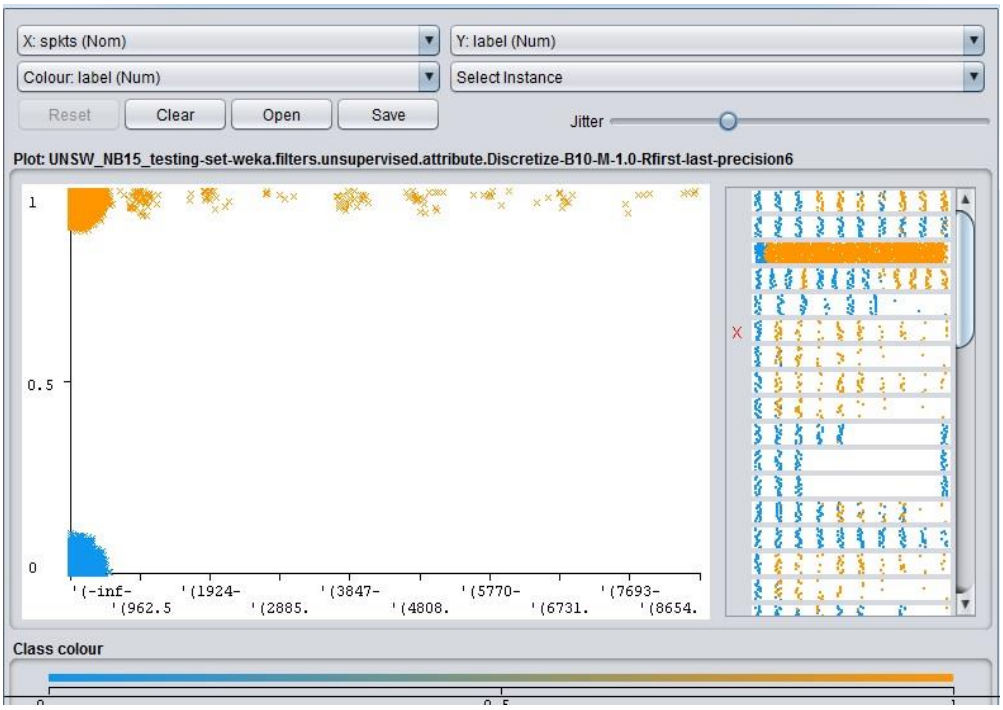


Figure 16: Represents the discretization of the feature "dpkts" with equal bin size

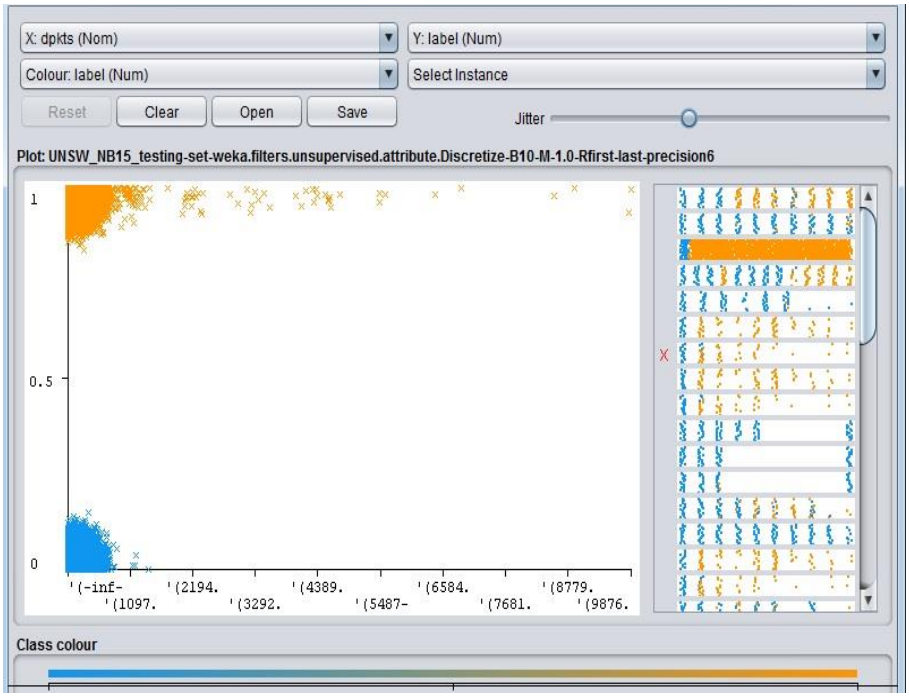
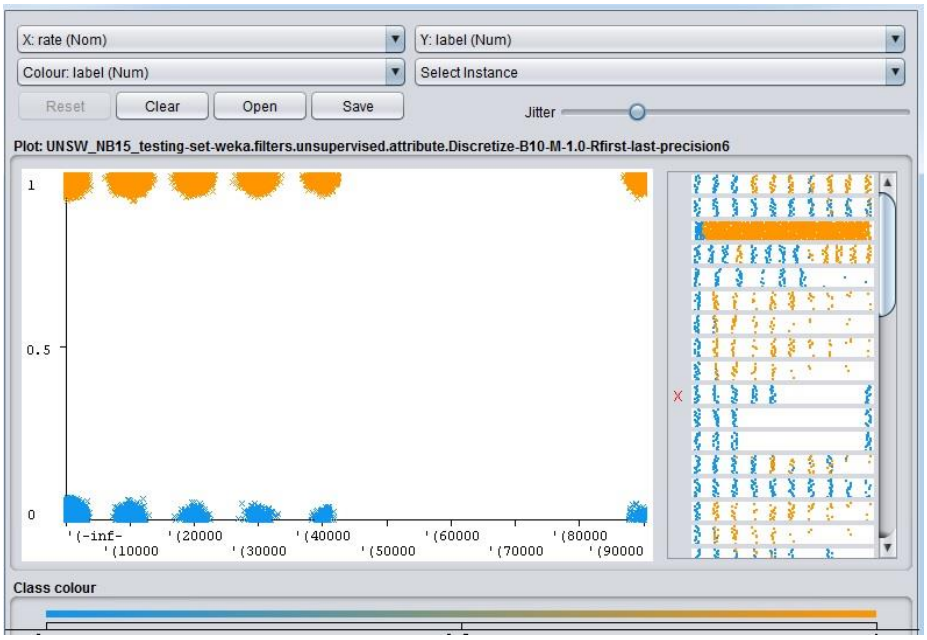


Figure 17: Represents the discretization of the feature "rate" with equal bin size



5. Conclusion

In this paper, an analysis on two well-known datasets KDD'99 and UNSW–NB15 has been conducted. In addition to this, a new dataset has been presented using network traffic of one week. This dataset has been prepared considering the resource consumption based features, as it is found that these kind of features play an important role in identifying intrusions. It is also found that pre-processing plays a crucial role in any prediction task which is also applicable to intrusion detection systems. Pre-processing of KDD'99 and UNSW–NB15 datasets has been demonstrated in this paper using WEKA data miner. Detail analysis on the newly developed dataset has not been considered in this paper and will be presented in future works.

References

1. Hofmeyr S. A., Forrest S. (2000). Architecture for an Artificial Immune System, Evolutionary Computation, 8(4), 443-473.
2. Wang K. & Stolfo S.J. (2004). Anomalous payload-based network intrusion detection. In: Jonsson E., Valdes A., Almgren M. (eds.) RAID 2004. Lecture Notes in Computer Science, Springer, 3224, 203-222.
3. Dutt I., Borah S., Maitra I.K., (2020). Immune System Based Intrusion Detection System (IS-IDS): A Proposed. IEEE Access 8 (2020), 34929--34941
4. Borah S., Panigrahi R., Chakraborty A.(2018), An enhanced intrusion detection system based on clustering. In: Saeed, K., Chaki, N., Pati, B., Bakshi, S., Mohapatra, D. (eds.) Progress in Advanced Computing and Intelligent Engineering. Advances in Intelligent Systems and Computing, vol. 564. Springer, Singapore (2018)
5. Dutt I., Borah S., (2015) Some studies in intrusion detection using data mining techniques. Int J Innov Res Sci Eng Technol 4(7):5500–5511
6. Panigrahi R., Borah S. (2019), Dual-stage intrusion detection for class imbalance scenarios, Comput. Fraud Secur., 2019 (2019), pp. 12-19
7. Dutt I., Borah S., Maitra I.K., Bhowmik K., Maity A., Das S. (2018), Real-time hybrid intrusion detection system using machine learning techniques Advances in Communication, Devices and Networking, Springer (2018), pp. 885-894
8. Borah, S., Chakravorty, D., Chawhan, C., Saha, A. (2011): Advanced Clustering based Intrusion Detection (ACID) Algorithm, Advances in Computing and Communications, Springer CCIS series, Vol. 192, Part 1, ISSN: 1865:0929, pp. 35–43, (2011) http://dx.doi.org/10.1007/978-3-642-22720-2_4
9. Stolfo S., Lee W., Prodromidis A. & Chan P. (1999). Cost-Based modeling and evaluation for data mining with application to fraud and intrusion detection: Results from the JAM project.
10. Tavallaee M., E. Bagheri, W. Lu and A. A. Ghorbani (2009), "A detailed analysis of the KDD CUP 99 data set," 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009, pp. 1-6, doi: 10.1109/CISDA.2009.5356528.
11. Gunes Kayacık A., Nur Zincir-Heywood A., Nur Zincir-Heywood, Malcolm I. Heywood (2005), Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99, Proc. of Third Annual Conference on Privacy, Security and Trust, October 12-14, 2005, The Fairmont Algonquin, St. Andrews, New Brunswick, Canada
12. Moustafa N. Slay and J. (2015), "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," 2015 Military Communications and Information Systems Conference (MilCIS), 2015, pp. 1-6, doi: 10.1109/MilCIS.2015.7348942.
13. Moustafa N., (2019), "UNSW_NB15 dataset", IEEE Dataport, doi: https://dx.doi.org/10.21227/8vf7-s525.
14. IXIA PerfectStorm tool, URL: https://www.keysight.com/in/en/products/network-test/network-test-hardware/perfectstorm.html
15. Lippmann R., Haines J.W., Fried D.J., Korba J., Das K. (2000) Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation. In: Debar H., Mé L., Wu S.F. (eds) Recent Advances in Intrusion Detection. RAID 2000. Lecture Notes in Computer Science, vol 1907. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-39945-3_11.