International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 2 Volume No.7 March - April – 2023
DOI:10.46647/ijetms.2023.v07i02.010 ISSN: 2581-4621

A Novel Intrusion Detection System Using Multiple Linear Regression

Koushik Paul[1], Sayandeep Paik[2], Siddhartha Kuri[3], Soumyadip Majumder[4], Dr. Avijit Kumar Chaudhuri[5]

[1, 2, 3, 4] B. Tech, Department of CSE, TEC Banipur, West Bengal, India

[5]Assistant Professor, Department of CSE, TEC Banipur, West Bengal, India

Corresponding Author Orcid ID: **(0000-0003-4117-7067)[1] (0000-0002-7114-2378)[2] (0000-0003-3443-9639)[3] (0000-0002-7596-6833)[4] (0000-0002-5310-3180)[5]**

Abstract

The internet is no doubt the biggest and the most important tool of modern civilisation. But along with its numerous benefits, it also comes with its own set of risks, the most important of them being breaches in security and privacy.

An anomaly-based Intrusion Detection System (IDS) is a type of security system that is used to detect and alert on unusual or abnormal behaviour that may indicate an attack or intrusion. Unlike signature-based IDS, which rely on known patterns of attack, anomaly-based IDS is designed to detect previously unseen or unknown attacks by identifying deviations from normal patterns of behaviour.

Multiple linear regression is a statistical technique used to analyse the relationship between a dependent variable and multiple independent variables. In this technique, a linear equation is established between the dependent variable and multiple independent variables, with the aim of predicting the value of the dependent attribute for a given set of values of the independent attribute.

In this paper, we collected a data set of 125974 entries and 42 attributes from Kaggle, pre-processed the data and used logistic regression to predict the dependent variable (called xAttack) using 25 independent variables, as we found a high correlation between the aforementioned variables

The results are simulated using 10-fold cross validation, using various train test splits of the data set. The data has been split into 80-20,50-50, and 66-34. After testing the given data set in different train test splits, an accuracy of 92.73 was achieved.

Keywords: *Intrusion Detection System (IDS), Machine Learning, Multiple Linear regression, security breach.*

1. Introduction

The internet has become an essential tool in modern society. A huge amount of essential and confidential data is present on the internet. This data might be extremely important for the security of the Host. But data on the internet is always at a risk of infringement.

As a result of the recent pandemic COVID-19, a lot of employees were encouraged to work from home. This has led to a massive surge in the transmission of sensitive data online, requiring the employers to provide a safe working environment. Therefore, we need a means of security that protects us against possible cyber-attacks.

An Intrusion Detection System (IDS) is a security technology designed to detect and prevent unauthorized access or malicious activity on a computer network or system. Its primary purpose is to identify and respond to potential security breaches and attacks, alerting security personnel or automated response systems to take action.

This action is performed using Multiple Linear Regression (MLR). Multilinear regression is a statistical method used to analyze the relationship between multiple independent variables and a dependent variable. It is a sort of linear regression where the dependent variable is a linear combination of multiple independent variables.

2. Relevant Literature

Today, security has become a critical concern for individuals, businesses, and governments alike. With the increasing reliance on technology and the internet, the risk of cyber-attacks, data breaches, and other forms of digital threats has also risen.

Intrusion Detection System (IDS) [6] solves majority of the problem. The main purpose of Intrusion Detection System is to identify potential security breaches as early as possible so that appropriate action can be taken to prevent or mitigate the damage. The IDS alerts the system and security administrators of the malicious or anomalous activity such as attempt to access restricted resources, modifications to system files or unusual network traffic pattern.

Ali H. Mirza (2018) [1] used logistic regression, neural networks, and decision trees for intrusion detection and reduced dataset dimensions using PCA. Ensemble learning was implemented by assigning weights to each classifier, and the results were used to determine if a sample was anomalous or not.

T.Saranyaa et al. (2020) [2] also proposed a paper which focuses on exploring different machine learning algorithms used for intrusion detection systems in various environments. The study shows that the detection rate, false positive rate, and accuracy of the algorithms used in IDS not only depend on the algorithm but also on the specific application area. In future, the sought to conduct an extensive study of ML algorithms to provide better solution for the IDS by taking real-time dataset.

However, *Partha Ghosh et al.* (2015) [3] took a different approach using modified genetic algorithm (GA) procedure with probabilistic selection, selective mutation, and a fitness score based on mutual correlation, to reduce storage and processing time without compromising accuracy. The proposed GA-BFSS method produced a better feature set than the ordinary GA method, and with the reduced set of features, an OVA classifier by LR was designed for multiclass classification.

Christiana Ioannou et al. (2017) [4] implemented a model in detecting Selective Forward and Blackhole attacks in various network topologies was evaluated using the Contiki O/S. The results showed a promising accuracy of 91% in detecting both types of attacks simultaneously using the Binary Logistic Regression (BLR) model.

In their study, *Anil Lamba et al.* (2015) [5] compared four supervised machine learning classifiers for intrusion detection using the NSL-KDD dataset. The classifiers tested were Support Vector Machine, Random Forest, Logistic Regression, and Gaussian Naive Bayes. Random Forest was found to be the most effective with an accuracy of 99%. In future they sought to add further work that would could consider multiclass classification and focus on important attributes for intrusion detection.

G. Vandewiele et al. (2020) [7] projected the proper use of oversampling and undersampling of a highly imbalanced dataset in order to generate optimistic results. The method has been followed so that the majority class will not be overly represented. If overly represented, poor generalization of minority class would occur. It was also done so that the training sets would have similar number of samples.

3. Methodology

Data:

In this paper, the data we have been working on has been taken from the kaggle website. The database is based on the networking information of an unknown university of the years 2013 and 2014 whose reference can be found though the mentioned link.

https://www.kaggle.com/datasets/what0919/intrusion-detection

this database contains information on various details like protocol type, number of services and type of attack on the client.

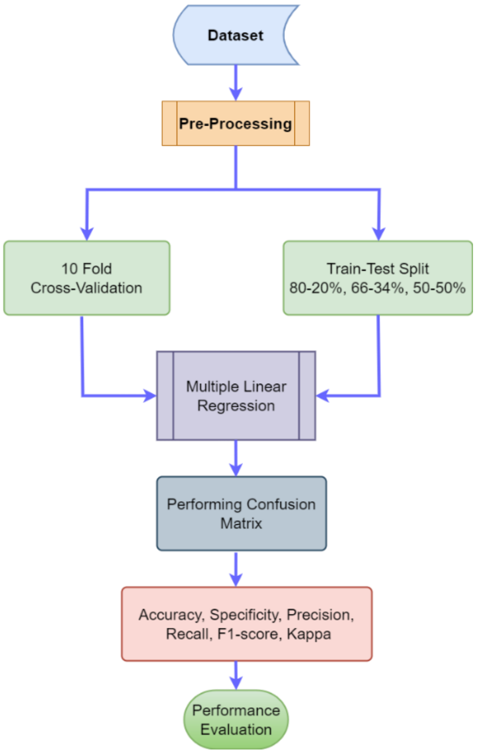
Table 1: Dataset Overview

ATTRIBUES	DEFINITION	MEAN
service	The number of services being run at that particular time	34.292
flag	Denotes the status of the operation	2.44

hot	a system or network component that is currently experiencing a high level of activity or usage	2.00
count	Counts the number of domain visits	60.91
serror_rate	The rate at which a network device or application generates "source errors"	0.194
srv_serror_rate	The percentage of TCP connections that were unsuccessful due to a "syn" packet error or other TCP-related errors at the server end	0.185
rerror_rate	The rate at which a remote system or network returns error messages in response to connection attempts	0.132
srv_rerror_rate	The rate at which a remote service returns error messages in response to requests made to that service	0.136
same_srv_rate	Proportion of requests made to the same service on a target host.	0.738
diff_srv_rate	The percentage of packets that were received on a server with a different service than the one expected	0.0791
srv_diff_host_rate	Percentage of packets that were received on a server from a different host than the one expected	0.0824
dst_host_count	Total number of connections that have been made to a particular destination host within a given period of time	139.48
dst_host_srv_count	A metric that measures the number of distinct services that are available on a destination host	62.71
dst_host_same_srv_rate	Percentage of connections made to a destination host that use the same service as the previous connection to that host.	0.554
dst_host_serror_rate	The rate at which a destination host returns "destination unreachable" errors in response to connection attempts from a source host	0.1899
dst_host_rerror_rate	percentage of connections that have received an error response from a particular destination host, out of all connection attempts made to that host	0.131
dst_host_diff_srv_rate	Proportion of connections made to the same service on a destination host	0.105
dst_host_srv_serror_rate	the rate at which a destination host returns "destination unreachable" errors in response to connection attempts to a specific service on that host	0.186
dst_host_srv_rerror_rate	percentage of connections that have received an error from a particular destination host, out	0.1346

	of all connection attempts made to a specific service on that host.	
dst_host_same_src_port_rate	percentage of connections that have received an error response from a particular destination host, out of all connection attempts made to a specific service on that host	0.376
dst_host_srv_diff_host_rate	the percentage of connections made to a specific service on a destination host that come from a different host than the previous connection to that service	0.071
protocol_type	classification of network protocol used to transmit data between devices on a network	N/A
logged_in	status of a user on a network	0.499
root_shell	used by system administrators or advanced users to perform maintenance and troubleshooting tasks	0.096
is_guest_login	Checks if the client is logged in	0.066
xAttack	The type of threat on the client, if any	N/A

Architectural Design:


Dataset
Pre-Processing
10 Fold Cross-Validation
Train-Test Split 80-20%, 66-34%, 50-50%
Multiple Linear Regression
Performing Confusion Matrix
Accuracy, Specificity, Precision, Recall, F1-score, Kappa
Performance Evaluation

Research Method:

As mentioned earlier, we have used Multiple Linear Regression (MLR), a statistical technique for regression analysis. First, we prioritize predicting the independent variables that influence the dependent variable using their correlation against each other. Now, as we have found the independent variables, namely- ‘service’, ‘flag’, ‘hot’, ‘count’, ‘serror_rate’, ‘srv_serror_rate’, ‘rerror_rate’, ‘srv_rerror_rate’, ‘same_srv_rate’, ‘diff_srv_rate’, ‘srv_diff_host_rate’, ‘dst_host_count’, ‘dst_host_srv_count’, ‘dst_host_same_srv_rate’, ‘dst_host_diff_srv_rate’, ‘dst_host_same_src_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’, ‘protocol_type’, ‘logged_in’, ‘root_shell’, ‘is_guest_login’ and a dependent variable ‘xAttack’.

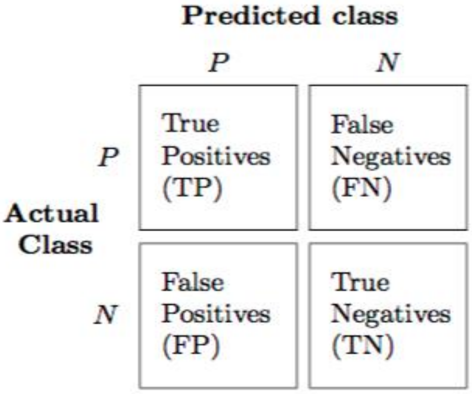
We performed the following steps in MLR to produce our final model

STEP 1: *Understanding the data:*

The first step of predicting the model is to find the dependent and independent variables. After that we try to develop a logistic relation between the dependent & independent variables. We then split the data into three parts as 4/5, 2/3, 1/2 defined as training data and the rest as testing data.

Cross validation also known as sample testing is a method where various parts of the data are used to train the mode and calculate its accuracy in practice. Here we divided the dataset into 10 paths, each time we select a part out of the 10 as the testing data and the remaining a part as training parts.

Confusion matrix, also known as error matrix, shows the performance of the algorithm in the form of a table. A confusion matrix shows a set of test data for which the values are true.



		Predicted class	
		P	*N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Now let’s take,
TP= TRUE POSITIVE
TN= TRUE NEGATIVE
FP= FALSE POSITIVE
FN= FALSE NEGATIVE
Now,

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

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}}$$

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

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

$$\text{F1\_Score} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall + Precision}}$$

$$\text{Kappa(k)} = \frac{accuracy - random\ accuracy}{1 - random\ accuracy}$$

So, to find random accuracy [11],

We know from the confusion matrix that a randomly drawn label from the dataset would be positive with probability P_1 and negative with probability $(1 — P_1)$

Where,

We also know that our classifier produces a positive label with probability P_2 and a negative label with probability $(1 —P_2)$
Where,

Random accuracy is just the probability that the labels produce by these two processes coincide by chance (assuming independence):
random accuracy $= P_1P_2 +(1- P_1) (1- P_2)$
Accuracy means that how precisely or how close the measured value reflects the originals. It must be calculated.
Specificity refers to the test accuracy at identifying the probability of a negative test, provided the condition is absent. It is to be calculated after Accuracy.
Precision study refers to on how precisely or accurately, the model is measured. We develop precision investigations to check if we are getting the correct results.
Recall (also known as sensitivity or true positive rate) is a performance metric that measures the proportion of actual positive instances that were correctly identified by a model out of all positive instances.
F1 score is a measure of a classification model's accuracy. It is the sympathetic mean of precision and recall, two metrics that are commonly used to evaluate the performance of a classifier.
Cohen Kappa score [10] is a numerical measure used to evaluate the performance of a machine learning classification model, particularly when the classes are imbalanced. It ranges from -1 to 1, with 1 representing perfect arrangement and 0 representing arrangement no better than chance. Negative values specify less arrangement than expected by chance.
Oversampling is a technique used in statistical analysis and machine learning to handle imbalanced datasets where the number of observations in one class is significantly lower than the other. In this technique, the inferior class is artificially increased by adding copies of its observations until it reaches a similar or proportional size to the superior class.
Undersampling is a technique used in statistical analysis and machine learning to handle imbalanced datasets where the number of observations in one class is significantly higher than the other. In this technique, the superior class is reduced by randomly removing observations until it reaches a similar or proportional size to the inferior class.

STEP 2: Selecting the suitable method:
The model is made using MLR (Multiple Linear Regression). For this method we first search for correlation between the dependent and independent variables, then we split the data into different fractions such as 80 – 20, 66 – 34, 50 – 50, followed by calculating the confusion matrix.

STEP 3: Developing equation of MLR and Confusion Matrix:
a. The logit (logistic) regression model
The multinomial logistic regression [8] is fairly a generalization of a binary model. In general, logistic regression model is used to find the probability of an existing class such as yes or no based on the observation of a dataset.
It can be defined as a classification problem, where the output or target variable (y) is dependent on the given values or inputs (X) in a dataset.
For a response variable Y with two measurement levels (dichotomous) and explanatory variable X,

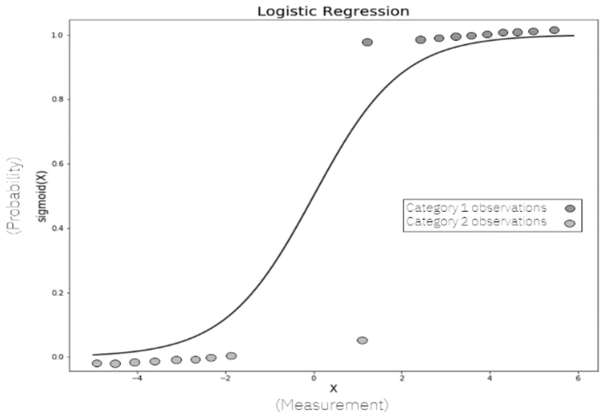
let: *$\pi(x) = p$ (Y = 1 | X = x) = 1 – p (Y = 0 | X = x)*, the logistic regression model has logistic form for logit of this probability:

, where the odds =

The odds = $\exp(\alpha + \beta x)$, and the logarithm of the odds is called logit, so

The logit is the distinct logarithm of the likelihoods. The S – curve shaped for π (x) controls the constraint β with its rate of gain or cut. If (β > 0) the curve ascends and descends for (β < 0).

Figure 3: S – Curve of Logistic Regression



b. Multiple Linear regression:

Multilinear regression [12] is a statistical modeling technique used to examine the linear relationship between a dependent variable and multiple independent variables. In other words, it is a method of estimating the values of a dependent variable based on the values of multiple independent variables. The equation for multilinear regression can be expressed as:

$Y = b_0 + b_1X_1 + b_2X_2 + ... + b_nX_n + e$

Given, Y is referred as the dependent variable, X_1, X_2, X_n as the independent variables, b_0 is the intercept or constant, b_1, b_2, b_n are the regression coefficients, and e is the residual error. The goal of multilinear regression is to determine the coefficients b_0, b_1, $b_{2,}$ ...b_n, that best fit the data to the model, such that the sum of the squared errors is minimized. This is typically done using an optimization algorithm such as ordinary least squares.

Considering an example [9] where a person can know whether he/she is expected to have a heart attack or not depending upon his/her body blood pressure, weight and age. The outcome is a binomial nominal variable i.e., heart attack vs no heart attack. The basic goal of Multiple Logistic Regression is to comprehend the functional relationship between the dependent and independent variables on what effects the probability of the outcome to change.

The logistic regression can be extending to models with multiple explanatory variables. Let k denotes number of predictors for a binary response Y by x_1, $x_{2,}$ ……. $x_{k,}$ the model for log odds is

And the alternative formula, directly specifying π(x), is

Here β refers to the impact of X_i in the odds for Y=1, controlled by other $X_{j.}$

If one has *n* independent observations with p – explanatory variables then to construct the logic, one of the categories must be considered as the base and the rest relative to it. Due to lack of ordering, any category may be used as k. Let π_j denote the multiple probability of an opinion dropping in the j^{th} class, to find the connection amongst this probability and the p illustrative variables, X_1, X_2, ……..., $X_{p,}$ the Multiple logistic regression model then is

Where j= 1, 2, …, (k-1), i = 1, 2, …, n. Since all the π's adds to unity, this reduces to

For j = 1, 2, …, (k-1), the model parameters are estimated by the method of ML.

Practically, we use statistical software to do this fitting.

In this model, the hypothesis that is used:

H_0: None of the controlled variable X1, X2 and X3 is significantly related to Y

H_a: At least one of the controlled variables X1, X2 and X3 is significantly related to Y

The model of Multiple Logistic Regression can be represented as:

$$y = a + b_1x_1 + b_2x_2 + \cdots\cdots + b_nx_n$$

Where i = 1,2,3……n

Where,

y = xAttack = Shows the type of threat on the client, if any

a = Constant variable

b_1 = Coefficient of first controlled variable

b_2 = Coefficient of second controlled variable

b_3 = Coefficient of third controlled variable

b_4 = Coefficient of fourth controlled variable and so on

x_1 = service

x_2 = flag

x_3 = count

x_4 = serror_rate

x_5 = same _srv _rate

x_6 = dst_host_srv_count

x_7 = dst_host_same_srv_rate

x_8 = dst_host_serror_rate

x_9 = dst_host_srv_serror_rate

x_{10} = protocol_type

x_{11} = logged_in

x_{12} = hot

x_{13} = srv_serror_rate
x_{14} = rerror_rate
x_{15} = srv_rerror_rate
x_{16} = diff_srv_rate
x_{17} = srv_diff_host_rate
x_{18} = dst_host_count
x_{19} = dst_host_diff_srv_rate
x_{20} = dst_host_same_src_port_rate
x_{21} = dst_host_srv_diff_host_rate
x_{22} = dst_host_rerror_rate
x_{23} = dst_host_srv_rerror_rate
x_{24} = root_shell
x_{25} = is_guest_login

In the case of b_1, $\bar{x}$ is the mean of service. In the case of b_2, $\bar{x}$ is the mean of flag. In the case of b_3, $\bar{x}$ is the mean of count. In the case of b_4, $\bar{x}$ is the mean of serror_rate.

In every case of b, y¯ is the mean of xAttack.

4. Results and Discussion

After analysing this model, we get the results that are given below. Where all the value ranges are in percentage except for kappa which ranges between 0 and 1.

For 80 - 20% train-test split:

Confusion Matrix:

Predicted Class

Actual Class

	0	1	2	3	4
0	4436	91	177	244	157
1	170	5539	15	46	18
2	94	7	4736	3	65
3	122	49	12	4167	21
4	99	0	402	0	4050

Calculated Results:

	Precision	Recall	F1-score	Specificity	Kappa	Accuracy
0	90	87	88	97.5		
1	97	96	97	99.2		
2	89	97	92	96.9		
3	93	95	94	98.5		
4	94	89	91	98.7		
Macro avg.	93	93	93	98.16	0.90	92.7
Weighted avg.	93	93	93	98		

For 66 - 34% train-test split:

Confusion Matrix:

Predicted Class

Actual Class

		0	1	2	3	4
0		7533	128	310	424	245
1		301	9415	27	78	37
2		174	9	7966	7	122
3		194	61	16	7047	34
4		156	0	889	0	6850

Calculated Results:

	Precision	Recall	F1-score	Specificity	Kappa	Accuracy
0	90	87	89	97.5		
1	98	96	97	99.3		
2	87	96	91	96.3		
3	93	96	95	98.5		
4	94	87	90	98.7		
Macro avg.	92	92	92	98.06	0.90	92.4
Weighted avg.	93	92	92	98		

For 50 - 50% train-test split:

Confusion Matrix:

Predicted Class

Actual Class

	0	1	2	3	4
0	10996	205	447	630	373
1	439	13843	42	114	53
2	276	13	11825	10	182
3	280	86	25	10289	52
4	234	0	1323	0	10061

Calculated Results:

	Precision	Recall	F1-score	Specificity	Kappa	Accuracy
0	90	87	88	97.4		
1	98	96	97	99.3		
2	87	96	91	96.2		
3	93	96	95	98.5		
4	94	87	90	98.6		
Macro avg.	92	92	92	98	0.90	92.3
Weighted avg.	92	92	92	98		

For 10-fold cross-validation:

Test Cases	Accuracy
01	92.45
02	92.66
03	92.55
04	92.51
05	92.88
06	92.77
07	92.87
08	92.75
09	92.55
10	92.63

5. Conclusion

The present study employs the Multiple Linear Regression (MLR) statistical technique to construct an Intrusion Detection System (IDS). To achieve this, the entirety of the data has been segregated into three distinct paths, which are commonly known as the train-test-split approach. Subsequently, a 10-fold cross-validation methodology has been implemented, and the confusion matrix has been developed to evaluate the efficacy of the predictor.

The recorded accuracy for the 4/5, 2/3, and 1/2 train-test-split are 92.7, 92.4 and 92.3 respectively.

Our work encompasses the application of the IDS which can be beneficial in various industries such as finance, healthcare, and government sectors where data privacy and security are of utmost importance. By using this system, organizations can enhance their security measures, reduce the risk of data breaches, and ultimately protect their reputation and customer trust. Moreover, the IDS can also aid in the forensic investigation of cyber-attacks by providing real-time alerts and precise information about the intrusion attempts. Overall, the IDS based on logistic regression has a promising potential to improve the cybersecurity landscape and prevent threats before they cause any significant harm.

6. References

[1] Ali H. Mirza, "Computer Network Intrusion Detection using various Classifiers and Ensemble Learning". 978-1-5386-1501-0/18/$31.00 c 2018 IEEE.

[2] T.Saranya, S.Sridevi, C. Deisy, Tran Duc Chung and M. K. A. Ahamed Khan, "Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review". Third International Conference on Computing and Network Communications (CoCoNet'19), Procedia Computer Science 171 (2020) 1251–1260

[3] Partha Ghosh and Rajarshee Mitra, "Proposed GA-BFSS and Logistic Regression based Intrusion Detection System". 978-1-4799-4445-3/15/$31.00 ©2015 IEEE

[4] Christiana Ioannou, Vasos Vassiliou and Charalampos Sergiou. "An Intrusion Detection System for Wireless Sensor Networks". 978-1-5386-0643-8/17/$31.00 ©2017 IEEE

[5] Anil Lamba, Satinderjeet Singh, Sachin Bhardwaj, Natasha Dutta and Sivakumar Sai Rela Muni, "USES OF ARTIFICIAL INTELLIGENT TECHNIQUES TO BUILD ACCURATE MODELS FOR INTRUSION DETECTION SYSTEM". International Journal For Technological Research In Engineering, Volume 2, Issue 12, August-2015, ISSN (Online): 2347 – 4718

[6] Intrusion Detection System (IDS), **Wikipedia** https://en.wikipedia.org/wiki/Intrusion_detection_system

[7] Gilles Vandewiele, Isabelle Dehaene, Gyorgy Kovacs, Lucas Sterckx, Olivier Janssens, Femke Ongenae, Femke De Backere, Filip De Turck, Kristien Roelens, Johan Decruyenaere, Sofie Van Hoecke and Thomas Demeester, "Overly optimistic prediction results on imbalanced data: a case

study of flaws and benefits when applying over-sampling”. Artificial Intelligence In Medicine 111 (2021) 101987. https://doi.org/10.1016/j.artmed.2020.101987

[8] Koushik Paul, Saheb Karan, Siddhartha Kuri, Sulekha Das and Avijit Kumar Chaudhuri, “Placement Prediction Using Multiple Logistic Regression Method”. *International Journal of Advanced Research in Computer and Communication Engineering*, Impact Factor 7.39|□Vol. 11, Issue 3, March 2022. ISSN (O) 2278-1021, ISSN (P) 2319-5940. DOI: 10.17148/IJARCCE.2022.11337

[9] “Handbook of Biological Statistics~ John H. McDonald”, Multiple Logistic Regression, https://www.biostathandbook.com/multiplelogistic.html

[10] Mary L. McHugh, “Interrater reliability: the kappa statistic”. Biochem Med (Zagreb) . 2012;22(3):276-82. Published online 2012 Oct 15.

[11] Random Accuracy formula to find the Kappa value, **tutorialspoint**, https://www.tutorialspoint.com/statistics/cohen_kappa_coefficient.htm

[12] Lynn E. Eberly, “Multiple Linear Regression”. Methods in Molecular Biology, vol. 404: Topics in Biostatistics. DOI:10.1007/978-1-59745-530-5_9