International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 5083 - 5093

Data Normalization Techniques on Intrusion Detection for Dataset Applications

[1] Dharamvir,[2] Arul Kumar V

[1] Research Scholar,School of CSA, REVA University, Bangalore, India.

[2] Asst. Professor, School of CSA, REVA University, Bangalore, India.

[1] dhiruniit@gmail.com

Abstract

Intrusion Detection System (IDS) is an important security tool for safeguarding the network from both internal and external threats. Conventional IDSs employ signature-based methods or anomaly-based methods which rely on dataset for training and testing the system. KDD CUP 99 is one such widely used dataset. Artificial Neural Networks (ANN), Machine learning, Data mining, Evolutionary computing, Statistical methods, Computational Intelligence, etc., algorithms make use of this KDD CUP 99 dataset for testing. The dataset consists of symbolic, binary, numeric, and continuous features scattered in different range of values. In statistical methods such as Euclidean distance, the larger value dominates the distance measurement. In clustering algorithms, the larger values shift the cluster center. Such disadvantages could be overcome by ensuring uniformity to the dataset while retaining the exactness of the features mapped which could be achieved by a process known as Normalization. Data normalization is a data preprocessing stage which maps data from different ranges on to a common scale. In this paper, a detailed analysis of the existing various data normalization techniques that can be applied on KDD CUP 99 dataset is presented along with the illustration. From the analysis, it was found that different normalization techniques are suitable for different subsets of KDD CUP 99 dataset. The problem under investigation is to prove that the new dataset generated on application of various normalization techniques exhibits the same characteristics as that of the original KDD CUP 99 dataset. Also, the effect of data normalization techniques, viz., of Min-max, Z-Score, Log, and Sigmoid on the neural Network algorithm in terms of detection rate and false alarms were compared and it was experimentally found that the log and sigmoid data normalization techniques result in better detection rate.

Keywords - *IDS, Data Normalization, KDD CUP 99*

1. Introduction

Intrusion is an act by which a person enters another person‘s/organization‘s computer network without rights or permission. Over a decade, intrusion detection system (IDS) has got considerable importance in the field of network security. One important attribute to the growth of IDS is the paradigm shift of the mentality of attackers from script kiddies to sophisticated spy network agents who are politically and monetarily motivated. This has led to strengthen the security premises of network using firewall, IDS [1], Intrusion Prevention System (IPS), etc. IDS can be broadly classified into two methods, viz., signature-based or misuse detection and anomaly-based detection. In misuse detection techniques, signature or pattern of previously seen attack is calculated and configured in to the IDS which in turn alerts the administrator in case of similar attacks. On the other hand, anomaly detection techniques [2], calculate the normalcy of network behavior/activity and fix the activity as an attack on finding any deviating activity from the normal behavior. Misuse detection has high detection rate where as anomaly detection aides in detecting zero day attacks which are never seen before. Anomaly detection and misuse detection are orthogonal to each other.

1.1 KDD CUP 1999 dataset

KDD CUP 99 dataset is the most popular IDS dataset [3]. It consists of a large labeled training data and testing data. Testing data consists of attacks which are not present in the training dataset. The algorithms developed for anomaly detection systems can be tested using this dataset [9]. The dataset contains the four different types of attack data, viz., DoS, Probe, Remote-to-Local (R2L), and User-to-Superuser (U2R) attacks. The dataset is a U*A matrix where U is the set of data instances and A,

the set of features. There are 41 features which are extracted from network packets and classified into four different categories. In this, the first 9 features 1-9 are extracted from the header of network packets. Features 10-22 represent the content area/payload portion of network packets. The next two categories are time window (2 seconds) based features (23-31) and connection-window (100 connections) based features (32-41). The dataset contains 4 lakhs training instances and 2 lakhs testing instances. The training dataset contains 24 different attack vectors and the testing dataset contains 14 extra attack vectors.

2. Data Normalization

Data preprocessing is an important step in knowledge discovery process. It is considered as the fundamental block of data mining. Feature Extraction, Transformation, and Loading (ETL) are the three steps performed before loading the dataset to the learning algorithm. Data normalization technique, a sub division of data analysis, is a process where the attribute data or features are scaled so as to fall within a specific range such as -1.0 to 1.0, or 0.0 to 1.0, has the following advantages:

- enables data mining algorithms to be applied easily
- improves the effectiveness and the performance of mining algorithms
- makes data suitable for a specific analysis to be performed
- improves the normality of the variables/features
- reduces Type I (overestimation/false positives) and Type II (underestimation/false negatives) errors

In many practical applications, the dataset has features which lie in different range of values. Features are not uniform throughout the dataset. It contains both nominal [7] and numeric features with different range values. This results in the feature having larger values dominating the cost function than the features with smaller values, leading to the deterioration in the performance of the algorithm. Therefore the dataset could not be used in non-parametric models, where data does not belong to any particular distribution, such as neural networks, support vector machines, classification algorithms, clustering algorithms, etc., without data

Normalization

The study on intrusion detection is widely spread on the application of Artificial Neural Networks (ANN), data mining algorithms, statistical methods, etc. In statistical methods, Euclidean distance method has been used for grouping the inputs into clusters. In Euclidean distance calculation, the squared distance between two data instances are calculated, on non normalized data instances. A large deviation between instances which are statistically in the same category could be observed. This large deviation, in terms of distance, is due to the large range feature in the data set. Therefore, in order to minimize the large deviation, large range values of the instances must be normalized necessitating the need for normalization techniques.

In neural network algorithms, the need for data normalization arises because the output is normally represented by -1, 0, and 1. So the input to the neural network should be in the range of [-1, 1] or [0, 1]. Hence, data normalization is a prerequisite for neural network algorithms.

Data Normalization [8] is a technique by which data values in different ranges are mapped on to a common scale, preferably in the range [0-1]. The process of normalization [4] should maintain two main properties, viz., robustness and efficiency [6]. Robustness is the property of ensuring that the outliers, data instances which behave in an unexpected way or have abnormal properties, not affect the normalization process. Efficiency of normalization determines the quality by which the exact nature/property of features are retained and not lost in the new range.

Figure 1 shows the taxonomy of data normalization techniques. The data normalization schemes are classified into two schemes, viz., Feature-based schemes and Input vector-based schemes [5].

In feature-based schemes, the normalization is done feature wise. This helps to map each feature value in the dataset to the corresponding value in the new range. Feature-based schemes are further

divided into two methods, viz., linear methods and non-linear methods. The distribution of data around the mean

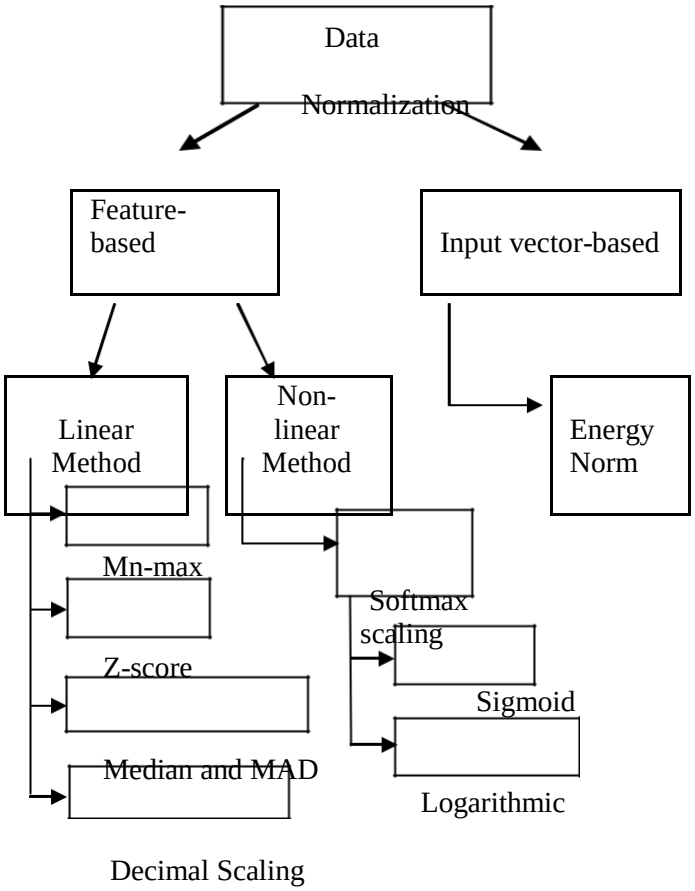

Figure 1. Taxonomy of Data Normalization Techniques

differentiates between the two methods.

2.1 Linear Methods

Linear methods are those where the data lie equally distributed around the mean. The following are the linear method feature-based data normalization schemes:

- Min-max normalization
- Z-score normalization
- Median and MAD normalization
- Decimal scaling normalization
-

A. Min-max Normalization

Min-max Normalization performs a linear interpolation using (1). The underlying distribution of the corresponding feature with the new range of values is sustained.

$$x' = \frac{x - \min f}{\max f - \min f}(n\_\max f - n\_\min f) + n\_\min f \quad (1)$$

x - Feature value to be normalized
x‘ - Normalized feature value of x.
minf - Actual minimum value for feature _f‘ in the given dataset
maxf - Actual maximum value for feature _f‘ in the given dataset
n_minf - Lower value in the new range
n_maxf - Upper value in the new range

B. Z-score Normalization

Z-score normalization is the most commonly used method in normalization. The values are mapped using (2) and (3) to a scale where the mean, the local estimator, is zero and the standard deviation, the scatter estimator, is one. It obeys the standard normal distribution principle. An important aspect of Z-score normalization is the ability to reduce the effect of outliers in the dataset.

$$x' = \frac{x - \overline{x}}{\sigma} \quad (2)$$

$$Mean, \ \overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad S.D. \ \sigma = \sqrt{\frac{\sum_{i=1}^{N}(x - \overline{x})^2}{N-1}} \quad (3)$$

x - Feature value to be normalized

- x‘ - Normalized feature value of x.
- μ - Mean value for feature _f‘
- σ - Standard Deviation (S.D.) for feature _f‘
- N - Number of data instances

C. Median and Median Absolute Deviation (MAD)

Median and Median Absolute Deviation is a rehashed form of Z-score normalization done using (4). MAD is suitable if the feature values are continuous in the given range.

$$x' = \frac{x - median}{MAD} \quad \text{where, MAD} = \text{median}(|x_f - \text{median}|) \quad (4)$$

D. Decimal Scaling Normalization

Decimal Scaling Normalization transforms the data into [0, 1] range by moving the decimal point of values of the attribute, using (5). The capability of this normalization lies in the mapping of very large values in the range [0, 10000] to the range [0, 1].

$$x' = \frac{x}{10^s} \quad \text{where, } s = \log_{10}(\max_f) \qquad (5)$$

x - Feature value to be normalized
x‘ - Normalized feature value of x.

2.2 Non-linear Methods

In non-linear methods the data are not evenly distributed around the mean and the deviations are very large. In such cases, transformations are based upon non-linear functions such as sigmoid or logarithmic to map the data within the specified interval of [0, 1]. Such transformations are known as Softmax scaling. The functions sigmoid and logarithmic are known as squashing function which is used to limit the data in the range of [0, 1].

A. Logarithmic Normalization

In many cases the values of features are exponentially distributed. The logarithmic process of normalization using (6) enables to get more resolution to the lower feature values. If

TABLE I. KDD CUP 99 FEATURES

Sl. No	Features	Type	Range
1	Duration	Discrete	0-100000
2	Protocol_type	Symbolic	NA
3	Service	Symbolic	NA
4	Flag	Symbolic	NA
5	Src_bytes	Discrete	0-10000000
6	Dst_bytes	Discrete	0-10000000
7	Land	Binary	0/1
8	Wrong_fragments	Discrete	0-25
9	Urgent	Discrete	0-30
10	Hot	Discrete	0-35
11	Num_failed_logins	Discrete	0-35
12	Logged_in	Binary	0/1
13	Num_compromised	Discrete	0-1000
14	Root_shell	Binary	0/1
15	Su_attempted	Binary	0/1
16	Num_root	Discrete	0-1000
17	Num_file_creations	Discrete	0-28
18	Num_shells	Discrete	0-40
19	Num_access_files	Discrete	0-60
20	Num_outbound_cmds	Discrete	0-60
21	Is_host_login	Binary	0/1
22	Is_guest_login	Binary	0/1
23	Count	Discrete	0-1000
24	Srv_count	Discrete	0-1000
25	Serror_rate	Continuous	0-1

26	Srv_serroro_rate	Continuous	0-1
27	Rerror_rate	Continuous	0-1
28	Srv_rerror_rate	Continuous	0-1
29	Same_srv_rate	Continuous	0-1
30	Diff_srv_rate	Continuous	0-1
31	Srv_diff_host_rate	Continuous	0-1
32	Dst_host_count	Discrete	0-255
33	Dst_host_srv_count	Discrete	0-255
34	Dst_host_same_srv_rate	Continuous	0-1
35	Dst_host_diff_srv_rate	Continuous	0-1
36	Dst_host_same_src_port_rate	Continuous	0-1
37	Dst_host_srv_diff_host_rate	Continuous	0-1
38	Dst_host_serror_rate	Continuous	0-1
39	Dst_host_srv_serror_rate	Continuous	0-1
40	Dst_host_rerror_rate	Continuous	0-1
41	Dst_host_srv_rerror_rate	Continuous	0-1

the minimum values are known a priori, it might be a good idea to use it as in (6) for initialization during the normalization.

$x' = \log(x - m+1)$ where $m = \min(x_f)$ (6)

x - Feature value to be normalized

B. Sigmoid/Logistic Normalization

Sigmoid normalization as shown in (7) not only normalizes the current dataset values in the range [0, 1] but also ensures

Table II. Subset-1: Attribute Values For Decimal Scaling Norm

Decimal Scaling

Features	***Value of 's'***	***x'***
1	4.766	0.087
5	6.710	1.000
6	6.712	0.000
13	2.946	0.000
16	2.997	0.000
23	2.708	0.002
24	2.708	0.002
32	2.407	0.000
33	2.407	0.000

Table III. Subset-2: Attribute Values For Z-Score Norm

Z-Score

Features	***Mean***	***Standard Deviation***	***x'***
8	0.035	0.782	0.044
9	0.000	0.006	0.000
10	0.000	0.006	0.000
11	0.000	0.016	0.000
17	0.001	0.096	0.028
18	0.001	0.096	0.028
19	0.001	0.036	0.028
20	0.001	0.036	0.028

that any larger data value than the present set maps to [0, 1]. The transformation is more-or-less linear in the middle range around mean value, and has a smooth nonlinearity at the end which ensures that all values are within the range. Values away from the mean are squashed exponentially.

$$x' \quad \frac{1}{1 \; e^{y}} \quad \text{where, } y \quad \frac{x}{r} \qquad (7)$$

and _r‘ is a user defined value .

C. Input vector-based: Energy Normalization

Energy normalization is an input-vector based normalization scheme where normalization is performed on each data instance independently using (8).

$$xi' \quad \frac{x_i}{M_n(x)} \quad \text{where, } M_n(x) \quad \sqrt[n]{\sum_{i\,1}^{N} \left| x_i \right|^n} \qquad (8)$$

x_i - Each feature in a data instance

x_i‘ - Normalized data instance

- $M_n(x)$ - Minkowski norm, n=1, 2
- N - Number of features

It is based on Minkowski Norm. In (8), if the value of n=1, then it is L1 or Taxicab norm. If n=2, then it is L2 or Euclidean norm.

3. Normalization On Kdd Cup 99 Dataset

Table I shows the feature type and the range of values the features possess in the KDD CUP 99 dataset. This facilitates to choose the appropriate data normalization technique for a particular feature. From the Table I it can be noticed that 3 features are symbolic and 6 features take binary values (either 0 or 1). Out of the 32 features which are numeric, 15 features are continuous in the desired range of [0-1]. So the process of data normalization should be applied on the remaining 17 features, which are discrete, whose value lie in different ranges of the order of [0-1000] and [1-10000000]. These features if not normalized will not give good support to the modeling algorithms. One of the

data instances considered for normalization with 41 features along with the data class, r2l from the training dataset of KDD CUP 99 is as follows:

Data Instance before normalization: 5059,1,14,1,5133876,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00, 0.00,0.00,0.00,1.00,0.00,0.00,0,0,1.00,0.00,1.00,0.17,0.00,0.0 0,0.00,0.00,r2l

3.1 Illustration for Min-Max Normalization – Subset-1

Eqn. (1) is applied for the features 1,5,6,13,16,23,24,32, and 33 of Table I. The resultant values obtained are -0.000981, -0.99.0,0,0,0,-0.00000094, -0.00000094,0, and 0.

3.2 Illustration for Z-Score Normalization – Subset-1

Eqn. (2) is applied for the features 1,5,6,13,16,23,24,32, and 33 of Table I. The resultant values obtained are 7.08,7253.73, -0.07, -0.00576, -0.0057,-0.07,-0.07,-3.58, and -1.77.

3.3 Illustration for Median and MAD Normalization – Subset-1

Eqn. (4) is applied for the features 1,5,6,13,16,23,24,32, and 33 of Table I. The resultant value for features 1,6, and 13 cannot be determined as the median and MAD equals to zero and hence the graph is not plotted for this technique.

3.4 Illustration for Decimal Scaling Normalization – Subset-1

The value of ‗s‘ is calculated from the expression in Eqn.

(5). Decimal Scaling normalization is applied on the features 1, 5, 6, 13, 16, 23, 24, 32, and 33. The corresponding value of ‗s‘ is substituted in Eqn. (5) and the new values obtained are 0.087, 1.000, 0.000, 0.000, 0.000, 0.002, 0.002, 0.000, and 0.000.
Table II shows the attributes of Subset-1 as a result of application of Decimal Scaling Normalization. The value of ‗s‘ gives the log values which is used for normalizing the feature values. x‘ gives the new value in the range [0-1].

3.5 Illustration for Z-Score Normalization – Subset-2

The Mean and Standard Deviation for the features 8, 9, 10, 11, 17, 18, 19, and 20 are obtained using Eqn. (3). Then the corresponding value of mean and standard deviation are substituted in Eq. (2). The modulus of the negative values are taken to map it into [0, 1] range. The new values are 0.044, 0.000, 0.000, 0.000, 0.028, 0.028, 0.028, and 0.028.
The results of application of the Mean and Standard Deviation on these features of Subset-2 are tabulated in Table
III for Z - Score Normalization technique. x‘ gives the new value in the range [0-1].

3.6 Resultant Instance of the dataset after Normalization

The data instance, Subset-1 and Subset-2, after applying the Decimal scaling and Z-score normalization techniques is as follows:
Data Instance after normalization:
0.087,1,0.218,1,1,0,0,0.044,0,0,0,1,0,0,0,0,0.028,0.028,0.028,
0.028,0,0,0.002,0.002,0.00,0.00,0.00,0.00,1.00,0.00,0.00,0,0,1
.00,0.00,1.00,0.17,0.00,0.00,0.00,0.00,r2l

4. Comparison Of Distribution

This section compares the distribution of the KDD CUP 99 dataset before and after normalization. SOM toolbox, a software library for MATLAB, is used to perform the data normalization techniques on the KDD CUP 99 training dataset. The dataset consists of 4,94,021 data instances. X-axis represents the data instances and Y-axis represents the feature values in Figures 2 to 5.

4.1 Feature-6 of KDD CUP dataset

Feature 6, dst_bytes in KDD CUP 99, denotes the number of bytes transferred from destination address-port pair to source address-port pair. Figure 2 shows the plot of the distribution for feature 6, dst_bytes, and its feature values. The resultant distribution of dst_bytes on application of Min-max, Z-score, and Decimal scaling normalization techniques are plotted in Figures 3, 4, and 5 respectively.

It can be seen from Figure 3 that the application of Min-max data normalization technique on feature 6 maps the data instances into the range [-1,0] without disturbing the underlying data distribution. The range from -1 to 0 could be mapped into 0 to 1 by taking the modulus values.

Figure 4 shows the distribution of Z-Score normalization on Feature 6. It can be seen from Figure 4 that the range is still high from -20 to 160.

Figure 5 shows the distribution of Decimal scaling normalization on Feature 6.

Therefore it could be concluded that both Min-max and Decimal scaling normalization techniques preserve the distribution. Further it indicates that either Min-max or Decimal scaling normalization technique is suitable as the range is 0 to 1 in both cases.

5. Experiment And Results

Under Matlab 7.6, Network Pattern Recognition Tool, npr tool, was used to conduct the experiments. KDD CUP 99 training dataset was used. The dataset was split in the ratio 0.70, 0.15, and 0.15 for training, validation, and testing purposes. The parameters chosen to conduct the experiment, common for all the four data normalization methods, are
as follows:

- Training function: Scaled conjugate gradient, trainscg

Error function: Mean Squared Error, mse
Validation: 6 rounds
The experiment was conducted as shown in Figure 8. The measured performance metrics, viz., detection rate, sensitivity, specificity, and error are tabulated in Table IV.

Detection rate	:	It is the combined ratio of true positives and true negatives to the total number of instances
Sensitivity	:	Proportion of actual positives which are correctly classified
Specificity	:	Proportion of actual negatives which are correctly classified
Error	:	Low error attained during training iteration
Epochs	:	Number of iterations performed
Time	:	Time taken, in minutes, to complete the training process

From the Table IV it can be inferred that non-linear methods such as log and sigmoid normalization are the best normalization methods for KDD CUP99 Intrusion Detection dataset, which result in a detection rate of 99.9%.

TABLE IV. COMPARISON OF NORMALIZATION TECHNIQUES

Norm Method	**Detection Rate**	**Sensitivity**	**Specificity**	**Error**	**Epoch**	**Time**
Min-max	99.8	99.6	99.9	0.00017	181	25
Z-Score	99.6	99.3	99.7	0.00280	295	90
Log	99.9	99.9	99.9	0.00039	200	29
Sigmoid	99.9	99.8	99.9	0.00051	170	28

6. Summary

In this paper a detailed survey on the various data normalization schemes that can be applied on KDD CUP 99 dataset where the features lie in different range of values is presented. This non uniformity in data which when used in pattern recognition algorithms leads to biased output depending only on a subset of the feature space. These data normalization techniques even out the dataset and ensure a fair representation of all features with the actuality preserved. The application of Min-max, Z-Score, and Decimal Scaling techniques on the KDD CUP 99 dataset has been illustrated. The results are seen to be in the desired range. Further, it has also been shown that the resultant dataset on application of normalization techniques retains the original property as that of the original dataset. Moreover, it is seen that non-linear methods such as log and sigmoid data normalization techniques result in better detection rate.

References

[1] Animesh Patcha and Jung-Min Park, ―An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends‖, Computer Networks Vol. No. 51, 2017, pp. 3448-3470.
[3] KDD CUP 1999. Available on: http://kdd.ics.uci.edu/ databases/ kddcup99/kddcup99.html.
[4] S. Theodoridis and K. Koutroumbas,‖Pattern Recognition‖, Second Edition, Elsevier Academic Press, 2018
[5] Kevin L. Priddy, Paul E. Keller, ―Artificial Neural Networks: An Introduction‖, First Edition, SPIE – The International Society for Optical Engineering, 2019.
[6] J. Song, H. Takakura, Y. Okabe, and Y. Kwon, ―A Robust Feature Normalization Scheme and an Optimized Clustering Method for Anomaly-Based Intrusion Detection System‖, In Proceedings of the 12th International Conference on Database Systems for Advanced Applications, 2017, pp. 140-151.
[7] Mei-Ling Shyu, Kanoksri Sarinnapakorn, Indika Kuruppu-Appuhamilage, Shu-Ching Chen, LiWu Chang, and Thomas Goldring, ―Handling Nominal Features in Anomaly Intrusion Detection Problems‖, In Proceedings of the 15th International Workshop on Research Issues in Data Engineering: Stream Data Mining and Applications, 2015, pp. 55-62.

[8] Long-zheng Cai, Jian Chen, Yun Ke, Tao Chen, and Zhi-gang Li, ―A new data normalization method for unsupervised anomaly detection‖, Journal of Zhejiang University SCIENCE-C (Computers and Electronics), Vol. No. 11, 2018, pp. 778-784.
[9] Manbod Tavallaee, Ebrahim Bagheri, Wei, Lu, and Ali A. Ghorbani, ―A detailed Analysis of the KDD CUP 99 Dataset‖, In Proceedings of the IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA), 2019, pp. 1-6.