Project

KDD CUP Dataset(Intrusion Detection System)

By-Surjeet Singh Yadav

Introduction-

- My project is on the KDD CUP dataset.
- Project is on the multi classification problem
- I have applied the multilayer perceptron model for the multi classification problem.
- I have split the data in seventy v/s thirty ratio for training and testing the model.

Dataset Description[1]

- The used dataset is KDD-CUP 99. This dataset is extracted from the network.
- This dataset contain forty one features.
- Main classification done on the dataset into five classes but training dataset contain twenty-four classification category.
- This dataset contain both attack and non attack data.

- In attack data there are four category-
  - Denial of service attack (DoS)
  - User to root attack(U2R)
  - Remote to local attack(R2L)
  - Probing attack
- In non-Attack data-
  - Normal

Reduction of features[2]

- In KDD CUP dataset there are forty one features but all are not important, it may be the case that some of them are irrelevant features.
- For reduction of the features so many method has been used like mutual information, Chi-square etc.
- The reduction of features has been done from forty-one to in the range of ten to twenty two.

Data Preprocessing:--

- I have applied correlation coefficient for reduction of features. Correlation coefficient-

$$\frac{\sum (X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum (X_i - \overline{X})^2 \sum (Y_i - \overline{Y})^2}}$$

i=1,2,.....n;

- I have taken if r has value more than 0.5 then those feature are related so we can take any of them.
- On the basis of above information, I have reduced the features from forty one to twenty six.

Normalization-

- Neural network work in efficient way if the value are in the specific set(range).
- To convert the value in normalized form we are using the following formula-

  $(x\text{-}min)(r_1 - r_2)/(max\text{-}min)+r_2$.

- So by this the value are fall into the range of o to 1.

Result-1

- I have applied the Multilayer perceptron model for the classification problem.
- I have applied the model in two way, in the first way, I have taken one hidden layer and in the second way I have taken two hidden layer.
- In the first method I have taken the twenty six input neuron, twenty four hidden neuron and twenty two output neuron, the learning rate set to 0.3 and momentum set to 0.2.
- I have split the dataset into the ratio of seventy and thirty for training and testing the model.

- Correctly Classified Instances 89519 95.93%
- Incorrectly Classified Instances 3790 4.06%

Accuracy by Class-

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.993	0.042	0.851	0.993	0.916	0.995	normal
0.999	0.001	1.000	0.999	1.000	1.000	dos
0.000	0.000	0.000	0.000	0.000	0.765	u2r
0.984	0.001	0.958	0.984	0.971	0.998	r2l
0.483	0.000	0.908	0.483	0.631	0.941	probe
0.000	0.000	0.250	0.000	0.001	0.983	snmpgetattack
0.000	0.000	0.000	0.000	0.000	0.692	7d
0.000	0.000	0.000	0.000	0.000	0.678	xlock
0.000	0.000	0.000	0.000	0.000	?	xsnoop
0.000	0.000	0.000	0.000	0.000	0.696	sendmail
0.812	0.003	0.397	0.812	0.533	0.873	saint
1.000	0.000	0.971	1.000	0.985	1.000	apache
0.000	0.000	0.000	0.000	0.000	0.923	3storm
0.000	0.000	0.000	0.000	0.000	0.671	xterm
0.987	0.002	0.622	0.987	0.763	1.000	mscan
0.987	0.000	0.996	0.987	0.991	0.999	processtable
0.000	0.000	0.000	0.000	0.000	0.744	ps
0.000	0.000	0.000	0.000	0.000	0.809	3tunnel
0.000	0.000	0.000	0.000	0.000	0.783	worm
0.999	0.000	0.981	0.999	0.990	1.000	mailbomb
0.000	0.000	0.000	0.000	0.000	?	sqlattack
0.000	0.000	0.000	0.000	0.000	0.995	snmpguess

Confusion Matrix-

- a b c d e f g h i j k l m n o p q r s t u v <-- classified as
- 18073 12 0 46 23 3 0 0 0 0 0 2 0 0 17 1 0 0 0 26 0 0 | a = normal
- 8 66870 0 1 2 0 0 0 0 0 0 3 0 0 21 0 0 0 0 2 0 2 | b = dos
- 9 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | c = u2r
- 29 0 0 1810 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | d = r2l
- 11 1 0 0 347 0 0 0 0 0 263 1 0 0 95 0 0 0 0 0 0 0 | e = probe
- 2327 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | f = snmpgetattack
- 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | g = 7d
- 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | h = xlock
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | i = xsnoop
- 4 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | j = sendmail
- 8 0 0 0 3 0 0 0 0 0 173 0 0 0 29 0 0 0 0 0 0 0 | k = saint
- 0 0 0 0 0 0 0 0 0 0 0 231 0 0 0 0 0 0 0 0 0 0 | l = apache
- 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | m = 3storm
- 3 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | n = xterm
- 2 0 0 2 0 0 0 0 0 0 0 0 0 0 304 0 0 0 0 0 0 0 | o = mscan
- 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 230 0 0 0 0 0 0 | p = processtable
- 2 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | q = ps
- 9 0 0 18 7 0 0 0 0 0 0 0 0 0 22 0 0 0 0 0 0 0 | r = 3tunnel
- 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | s = worm
- 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1480 0 0 | t = mailbomb
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | u = sqlattack
- 753 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | v = snmpguess

Result-2

- In this I have applied two hidden layer, in the first there are twenty four input neuron and in the second I have used ten input neuron.
- All other parameters remain the same.

- Correctly Classified Instances 86525 92.72%
- Incorrectly Classified Instances 6784 7.27 %

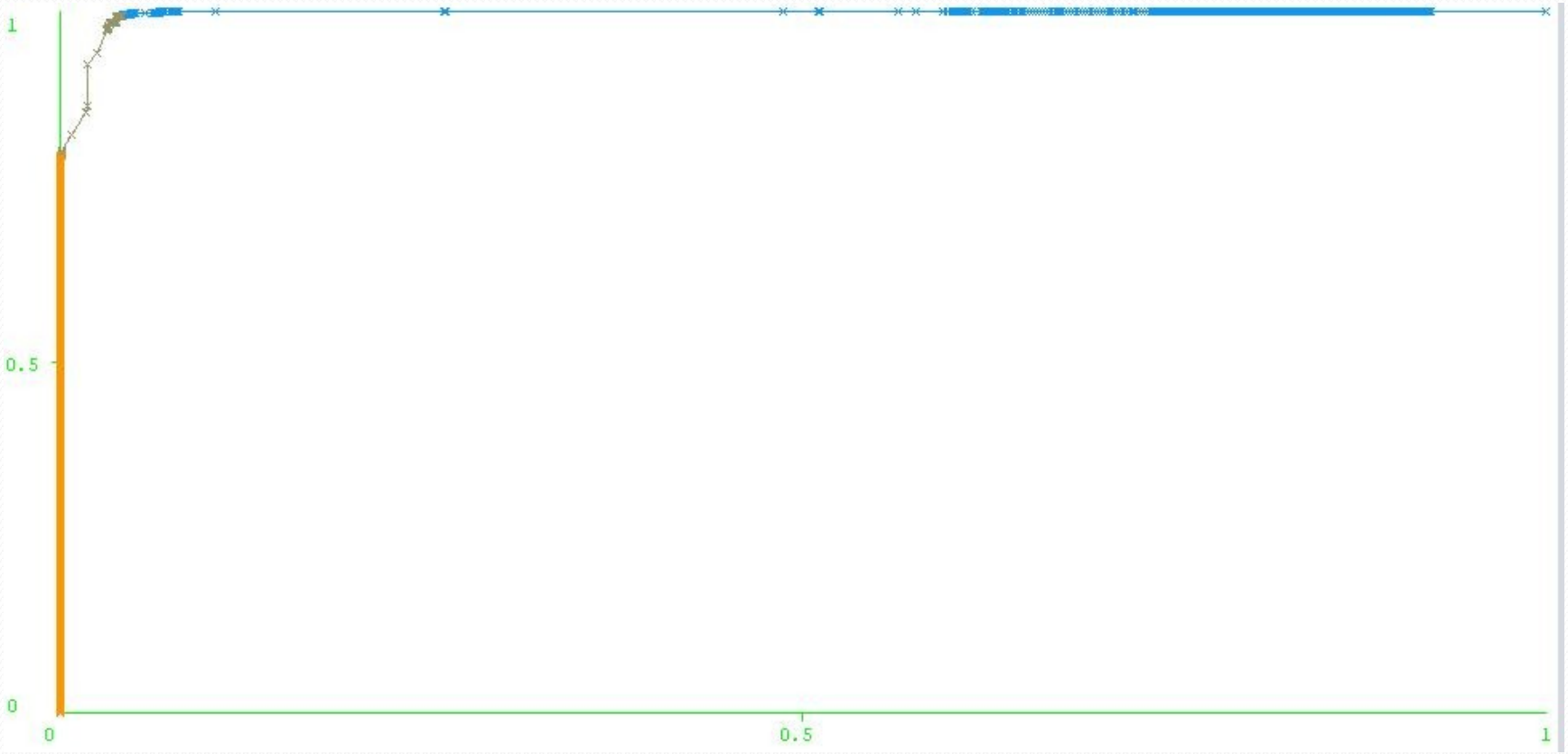
Accuracy By class-

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.990	0.083	0.742	0.990	0.848	0.987	normal
0.995	0.005	0.998	0.995	0.996	0.999	dos
0.000	0.000	0.000	0.000	0.000	0.820	u2r
0.973	0.001	0.930	0.973	0.951	0.996	r2l
0.000	0.000	0.000	0.000	0.000	0.826	probe
0.000	0.000	0.000	0.000	0.000	0.974	snmpgetattack
0.000	0.000	0.000	0.000	0.000	0.718	7d
0.000	0.000	0.000	0.000	0.000	0.715	xlock
0.000	0.000	0.000	0.000	0.000	?	xsnoop
0.000	0.000	0.000	0.000	0.000	0.792	sendmail
0.000	0.000	0.000	0.000	0.000	0.735	saint
0.000	0.000	0.000	0.000	0.000	0.738	apache
0.000	0.000	0.000	0.000	0.000	0.933	3storm
0.000	0.000	0.000	0.000	0.000	0.833	xterm
0.000	0.000	0.000	0.000	0.000	0.730	mscan
0.000	0.000	0.000	0.000	0.000	0.760	processtable
0.000	0.000	0.000	0.000	0.000	0.826	ps
0.000	0.000	0.000	0.000	0.000	0.726	3tunnel
0.000	0.000	0.000	0.000	0.000	0.726	worm
0.117	0.003	0.393	0.117	0.181	0.987	mailbomb
0.000	0.000	0.000	0.000	0.000	?	sqlattack
0.000	0.000	0.000	0.000	0.000	0.942	snmpguess

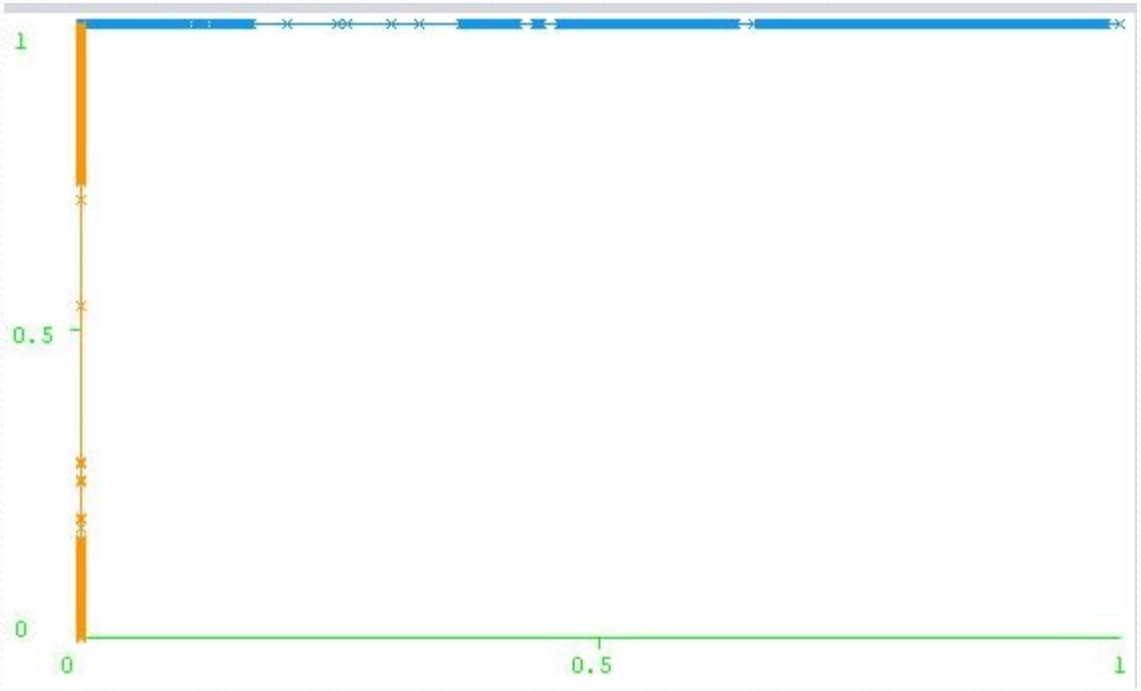
Confusion Matrix:-

- a b c d e f g h i j k l m n o p q r s t u v <-- classified as
- 18020 16 0 114 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 53 0 0 | a = normal
- 350 66542 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 17 0 0 | b = dos
- 8 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | c = u2r
- 48 0 0 1789 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 | d = r2l
- 613 98 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 | e = probe
- 2328 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | f = snmpgetattack
- 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | g = 7d
- 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | h = xlock
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | i = xsnoop
- 3 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | j = sendmail
- 209 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4 0 0 | k = saint
- 92 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 139 0 0 | l = apache
- 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | m = 3storm
- 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | n = xterm
- 257 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 47 0 0 | o = mscan
- 231 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | p = processtable
- 2 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | q = ps
- 53 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | r = 3tunnel
- 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | s = worm
- 1305 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 174 0 0 | t = mailbomb
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | u = sqlattack
- 754 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | v = snmpguess

ROC-Curve for Normal-Class-



ROC Curve for DoS Class-



Conclusion-

- By increasing the number of hidden layer the accuracy of the system has decreased.
- It has taken more time than single hidden layer.

References-

1. Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani, “A Detailed Analysis of the KDD CUP 99 Data Set” 978-1-4244-3764-1/09/$25.00 2009 IEEE.
2. Bhavin Shah, Bhushan H Trivedi, “Reducing Features of KDD CUP 1999 Dataset For Anomaly DetectionUsing Back Propagation Neural Network”, 2327-0659/15 $31.00 © 2015 IEEE.
3. Xiujuan Wang, Chenxi Zhang, Kangfeng Zheng, “Intrusion Detection Algorithm Based on Density, Cluster Centre, and Nearest Neighbors” Network Coding and Algorithm China Communications.

Thanks….