# Task 4.2

## Intrusion Detection using Supervised Learning Techniques

SIT719

Sanket Thakur

The intrusion detection system plays a major role in the defence of the network. The model for intrusion detection is a predictive model used for predicting the data traffic on the network as natural or interference. Machine Learning algorithms are used to build specific clustering models, prediction and classification. Classification and predictive models for intrusion detection are constructed in this paper by using computer Classification algorithms for learning, namely, Functinos.Logistic, Trees.j48, Trees.DecisionStump, rules.ZeroR, rules.Jrip, and Functions.SMO. These algorithms are tested with NSL-KDD data set

For this assignment we would use WEKA as a tool for assessment of algorithms. The Weka machine learning workbench is a modern platform for applied machine learning. Weka is an acronym which stands for Waikato Environment for Knowledge Analysis. It has a user interface with a graphical (GUI). This enables you to without code, complete the machine learning tasks

Summary of both the data sets derived by using WEKA

Test Data set

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       18200               80.731  %
Incorrectly Classified Instances      4344               19.269  %
Kappa statistic                          0.623
Mean absolute error                      0.1924
Root mean squared error                  0.4371
Relative absolute error                 39.2297 %
Root relative squared error             88.2712 %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.950    0.301    0.705      0.950    0.809      0.652    0.958     0.949     normal
                 0.699    0.050    0.949      0.699    0.805      0.652    0.949     0.944     anomaly
Weighted Avg.    0.807    0.158    0.844      0.807    0.807      0.652    0.953     0.946

=== Confusion Matrix ===

    a    b   <-- classified as
 9225  486 |    a = normal
 3858 8975 |    b = anomaly
```

Training Data Set

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      113858               90.3829 %
Incorrectly Classified Instances     12115                9.6171 %
Kappa statistic                          0.806
Mean absolute error                      0.0965
Root mean squared error                  0.3058
Relative absolute error                 19.3947 %
Root relative squared error             61.3067 %
Total Number of Instances           125973

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.936    0.134    0.890      0.936    0.912      0.807    0.967     0.964     normal
                 0.866    0.064    0.922      0.866    0.893      0.807    0.965     0.949     anomaly
Weighted Avg.    0.904    0.101    0.905      0.904    0.904      0.807    0.966     0.957

=== Confusion Matrix ===

     a     b   <-- classified as
 63060  4283 |     a = normal
  7832 50798 |     b = anomaly
```

Confusion Matrix Comparison :
A confusion matrix is a way to summarise the efficiency of a classification algorithm

| | Event | No-event |
|---|---|---|
| Event | True positive | False positive |
| No-event | False negative | True negative |

| Test data | Train Data |
|---|---|
| `=== Confusion Matrix ===`<br><br>`a b <-- classified as`<br>`9225 486 \| a = normal`<br>`3858 8975 \| b = anomaly` | `=== Confusion Matrix ===`<br><br>`a b <-- classified as`<br>`63060 4283 \| a = normal`<br>`7832 50798 \| b = anomaly` |

From the matrices we learn that

- The classifier made total of 22544 and 125973 predictions in the test data and train data sets respectively
- The number of true positive and True negative in both the sets is high

- False positives in both the sets are very low, that means the type 1 errors in the data sets are low
- False positives in both the sets are low as well, that means the type 2 errors in the sets are low
- The accuracy of the classifier in test data is 81% and train data is 90%
- The misclassification or the error rate of the classifier for the test data is 19% and train data is 10%
- The true positive rate or sensitivity for the test data is 95% and for the train data is 94%
- The False positive rate for test data is 5% and for the train data is 6%
- The true negative rate or specificity for the test data is 70% and for the train data is 87%

Detailed Performance comparison of the Training data set compared to the supplied data set in different algorithms

| Algorithms | TP Rate | FP Rate | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|---|
| Functinos.Logistic | 0.926 | 0.372 | 0.653 | 0.926 | 0.766 | 0.777 |
| Trees.j48 | 0.973 | 0.304 | 0.708 | 0.973 | 0.819 | 0.840 |
| Trees.DecisionStump | 0.955 | 0.317 | 0.695 | 0.955 | 0.804 | 0.819 |
| rules.ZeroR | 1.000 | 1.000 | 0.431 | 1.000 | 0.602 | 0.500 |
| rules.Jrip | 0.972 | 0.375 | 0.662 | 0.972 | 0.788 | 0.800 |
| Functions.SMO | 0.987 | 0.04 | 0.962 | 0.987 | 0.975 | 0.971 |
| | | | | | | |

Comparing the algorithms

- rules.ZeroR is the best performing algorithm in the given data set. As the TP rate ,FP rate and Recall is 1.00, which is the highest amongst the score
- The F measure, precision and ROC are is the highest in the Trees.j48 algorithm
- Functions.SMO is an average preforming algorithm
- Rules.Jrip and Trees.DecisionStump have no significant outcome in the given data set

For the Last section of this assignment we have resampled the data size to 20% and used the SVM based SMO algorithm and compared two different kernels that be used as a part of the algorithm. We have used PolyKernel and RBFKernel respectively in the same algorithm and run two separate tests

| Algorithms | TP Rate | FP Rate | Precision | Recall | F-Measure | ROC Area |
|---|---|---|---|---|---|---|
| Functions.SMO - PolyKernel | 0.924 | 0.422 | 0.624 | 0.924 | 0.745 | 0.609 |
| Functions.SMO -RBFKernel | 0.924 | 0.408 | 0.631 | 0.924 | 0.750 | 0.758 |
| | | | | | | |

We can see that both the algorithms have very close results, However the TP rate and FP rate of the Function.SMO-RBFKernel are better than that of Functions.SMO- PolyKernel

Function SMO RKB Kernel

```
=== Summary ===

Correctly Classified Instances       16566               73.483  %
Incorrectly Classified Instances      5978               26.517  %
Kappa statistic                          0.4881
Mean absolute error                      0.2652
Root mean squared error                  0.5149
Relative absolute error                 52.5318 %
Root relative squared error            101.7807 %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.924    0.408    0.631      0.924    0.750      0.529    0.758     0.616     normal
                 0.592    0.076    0.911      0.592    0.718      0.529    0.758     0.772     anomaly
Weighted Avg.    0.735    0.219    0.791      0.735    0.732      0.529    0.758     0.705

=== Confusion Matrix ===

    a    b   <-- classified as
 8972  739 |    a = normal
 5239 7594 |    b = anomaly
```

Function SMO PolyKernel

```
=== Summary ===

Correctly Classified Instances       16395               72.7244 %
Incorrectly Classified Instances      6149               27.2756 %
Kappa statistic                          0.4746
Mean absolute error                      0.2728
Root mean squared error                  0.5223
Relative absolute error                 54.0344 %
Root relative squared error            103.2262 %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.924    0.422    0.624      0.924    0.745      0.518    0.751     0.609     normal
                 0.578    0.076    0.910      0.578    0.707      0.518    0.751     0.766     anomaly
Weighted Avg.    0.727    0.225    0.786      0.727    0.723      0.518    0.751     0.698

=== Confusion Matrix ===

    a    b   <-- classified as
 8973  738 |    a = normal
 5411 7422 |    b = anomaly
```

# JRIP Rules

```
=== Summary ===

Correctly Classified Instances       18245               80.9306 %
Incorrectly Classified Instances      4299               19.0694 %
Kappa statistic                          0.6284
Mean absolute error                      0.1957
Root mean squared error                  0.4349
Relative absolute error                 38.7642 %
Root relative squared error             85.9519 %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.969    0.312    0.702      0.969    0.814      0.663    0.831     0.696     normal
               0.688    0.031    0.967      0.688    0.804      0.663    0.831     0.847     anomaly
Weighted Avg.  0.809    0.152    0.853      0.809    0.808      0.663    0.831     0.782

=== Confusion Matrix ===

    a    b   <-- classified as
 9413  298 |    a = normal
 4001 8832 |    b = anomaly
```

# ZeroR Rules

```
=== Summary ===

Correctly Classified Instances        9711               43.0758 %
Incorrectly Classified Instances     12833               56.9242 %
Kappa statistic                          0
Mean absolute error                      0.5048
Root mean squared error                  0.5059
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               1.000    1.000    0.431      1.000    0.602      ?        0.500     0.431     normal
               0.000    0.000    ?          0.000    ?          ?        0.500     0.569     anomaly
Weighted Avg.  0.431    0.431    ?          0.431    ?          ?        0.500     0.510

=== Confusion Matrix ===

     a     b   <-- classified as
  9711     0 |     a = normal
 12833     0 |     b = anomaly
```

# Functions.Logistics

```
=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.14 seconds

=== Summary ===

Correctly Classified Instances       17464               77.4663 %
Incorrectly Classified Instances      5080               22.5337 %
Kappa statistic                          0.5605
Mean absolute error                      0.2317
Root mean squared error                  0.4639
Relative absolute error                 45.9029 %
Root relative squared error             91.6837 %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.925    0.339    0.674      0.925    0.780      0.590    0.804     0.609     normal
               0.661    0.075    0.921      0.661    0.770      0.590    0.804     0.877     anomaly
Weighted Avg.  0.775    0.189    0.814      0.775    0.774      0.590    0.804     0.761

=== Confusion Matrix ===

    a    b   <-- classified as
 8983  728 |    a = normal
 4352 8481 |    b = anomaly
```

## Trees.DecisionStump

```
=== Summary ===

Correctly Classified Instances       18032               79.9858 %
Incorrectly Classified Instances      4512               20.0142 %
Kappa statistic                          0.6096
Mean absolute error                      0.2472
Root mean squared error                  0.4209
Relative absolute error                 48.9758 %
Root relative squared error             83.188  %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.955    0.317    0.695      0.955    0.804      0.642    0.819     0.683     normal
                 0.683    0.045    0.952      0.683    0.795      0.642    0.819     0.831     anomaly
Weighted Avg.    0.800    0.162    0.841      0.800    0.799      0.642    0.819     0.767

=== Confusion Matrix ===

    a    b   <-- classified as
 9271  440 |    a = normal
 4072 8761 |    b = anomaly
```

## Trees.J48

```
=== Summary ===

Correctly Classified Instances       17713               78.5708 %
Incorrectly Classified Instances      4831               21.4292 %
Kappa statistic                          0.5851
Mean absolute error                      0.2172
Root mean squared error                  0.4604
Relative absolute error                 43.0361 %
Root relative squared error             90.9987 %
Total Number of Instances            22544

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.970    0.354    0.675      0.970    0.796      0.629    0.791     0.661     normal
                 0.646    0.030    0.966      0.646    0.774      0.629    0.791     0.854     anomaly
Weighted Avg.    0.786    0.169    0.841      0.786    0.784      0.629    0.791     0.771

=== Confusion Matrix ===

    a    b   <-- classified as
 9421  290 |    a = normal
 4541 8292 |    b = anomaly
```

## Functions.SMO

```
=== Summary ===

Correctly Classified Instances        4899               97.241  %
Incorrectly Classified Instances       139                2.759  %
Kappa statistic                          0.9444
Mean absolute error                      0.0276
Root mean squared error                  0.1661
Relative absolute error                  5.5445 %
Root relative squared error             33.3002 %
Total Number of Instances             5038

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.987    0.045    0.962      0.987    0.975      0.945    0.971     0.957     normal
                 0.955    0.013    0.985      0.955    0.970      0.945    0.971     0.962     anomaly
Weighted Avg.    0.972    0.030    0.973      0.972    0.972      0.945    0.971     0.959

=== Confusion Matrix ===

    a    b   <-- classified as
 2659   34 |    a = normal
  105 2240 |    b = anomaly
```