Intrusion Detection System Using Ensemble learning

MENTOR - Prof. O.P. Vyas

Mantek Singh (IIT2016007)
Gagan Ganapathy (IIT2016038)
Saurabh Mishra (IIT2016045)
Ridam Arora (IIT2016134)
Niharika Shrivastava (IIT2016501)

The Task

- Build an ensemble model of different classifiers to distinguish between good and bad network connections.
- Classify bad network connections into 22 different types of attacks.

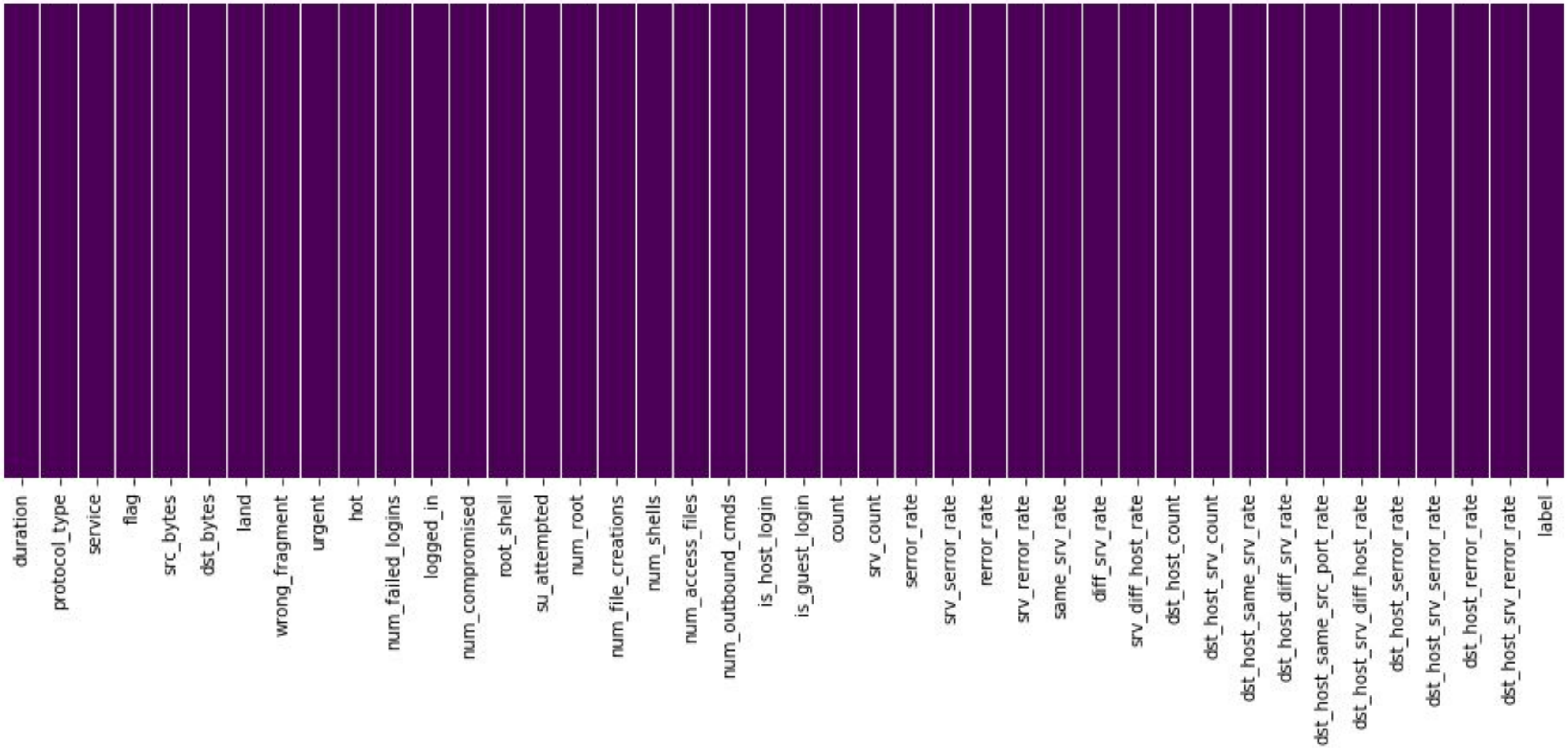
Dataset

- **Training attack types**: 22 types of attacks- DoS, Multihop, Smurf, .. etc. and a class indicating normal connection.
- **41 features**: protocol_type, service, flag, src_bytes, dst_bytes, land, … etc.
- The **test data** is not from the same probability distribution as the training data, and it **includes 14 specific attack types not in the training data.**
- **Training Data :-** About 5 million entries.

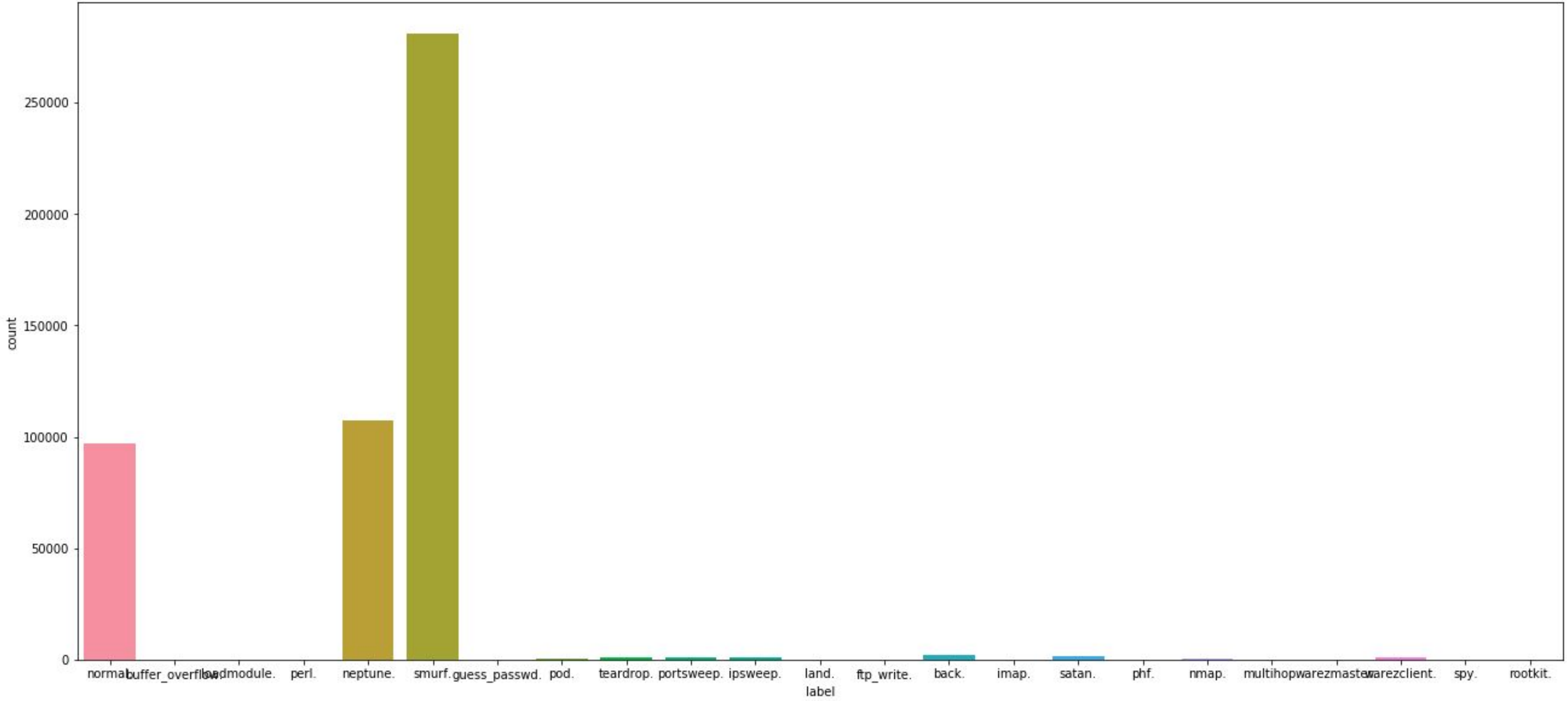
Dataset Preprocessing

- **Removing Redundancies:** To prevent bias towards more frequent attacks.
- **Normalization :** Adjusting values measured in a different scale to a notionally common scale.
- **Converting Categorical data:** Example : handling string values → TCP -0, UDP -1.
- **Removing null values:** Filled using the average of all given values.
- **Handling Dummy Variable trap**: Avoiding meaningless calculations. Example :- Feature which takes values from A, B, C must have value C if not A and B.

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd712a505c0>
```


duration
protocol_type
service
flag
src_bytes
dst_bytes
land
wrong_fragment
urgent
hot
num_failed_logins
logged_in
num_compromised
root_shell
su_attempted
num_root
num_file_creations
num_shells
num_access_files
num_outbound_cmds
is_host_login
is_guest_login
count
srv_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
label

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd71332a2b0>
```


count
250000
200000
150000
100000
50000
0
normal.
buffer_overflow.
loadmodule.
perl.
neptune.
smurf.
guess_passwd.
pod.
teardrop.
portsweep.
ipsweep.
land.
ftp_write.
back.
imap.
satan.
phf.
nmap.
multihop.
warezmaster.
warezclient.
spy.
rootkit.
label

Basic Classifiers

Naive Bayes

1. All features are assumed to be independent of each other.
2. Useful for large datasets.

Accuracy: 76.7%

Logistic Regression

1. Multiclass classification through the one-vs-all scheme.
2. Changing the loss function to cross-entropy loss.

Accuracy: 80.6%

Decision Tree

1. Can handle both categorical and numeric data.
2. Computationally efficient.
3. Easily handles multiple classes.

Accuracy : 80.4%

Multi Layer Perceptron

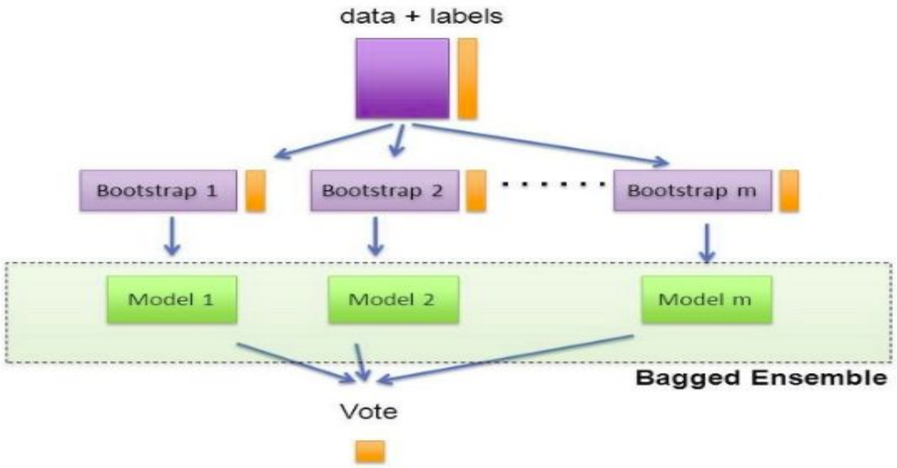
1. Capable of approximating non-linear functions.
2. MLPs with one hidden layer can approximate any continuous function.
3. Softmax used on output layer for multiclass classification.

Accuracy: 88.8%

Ensemble Methods

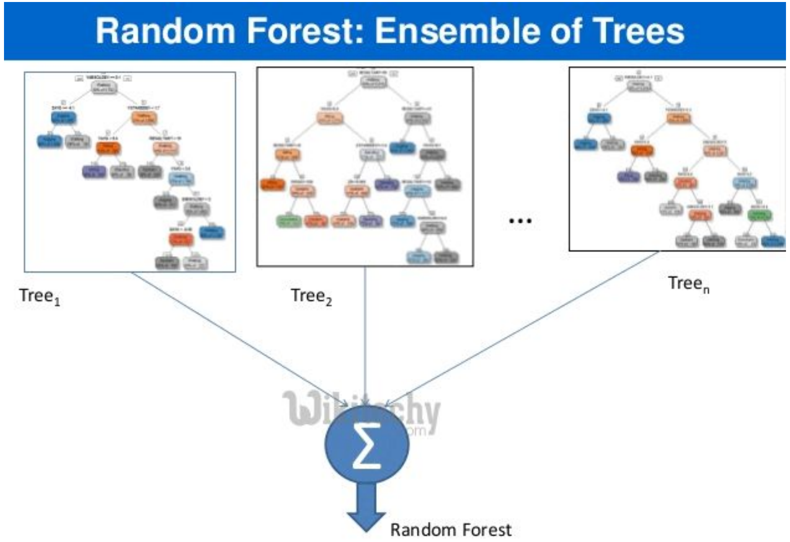
Bagging

- **Bagging** is used when our goal is to reduce the variance of a decision tree.
- First, we create several subsets of data. Then we train various classifiers on these samples in parallel.
- Finally, using an ensemble of these models,an average of the predictions of these models is used to predict the final output, thereby reducing overfitting by reducing the dependence on a single model.



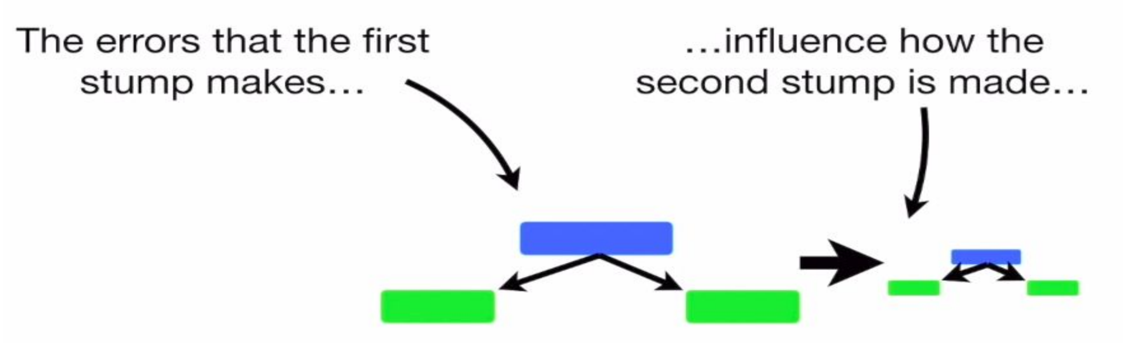
Random Forest

- ***Random Forest*** is an extension over bagging. It takes one extra step where in addition to taking the random subset of data, it also takes the random selection of features rather than using all features to grow trees.

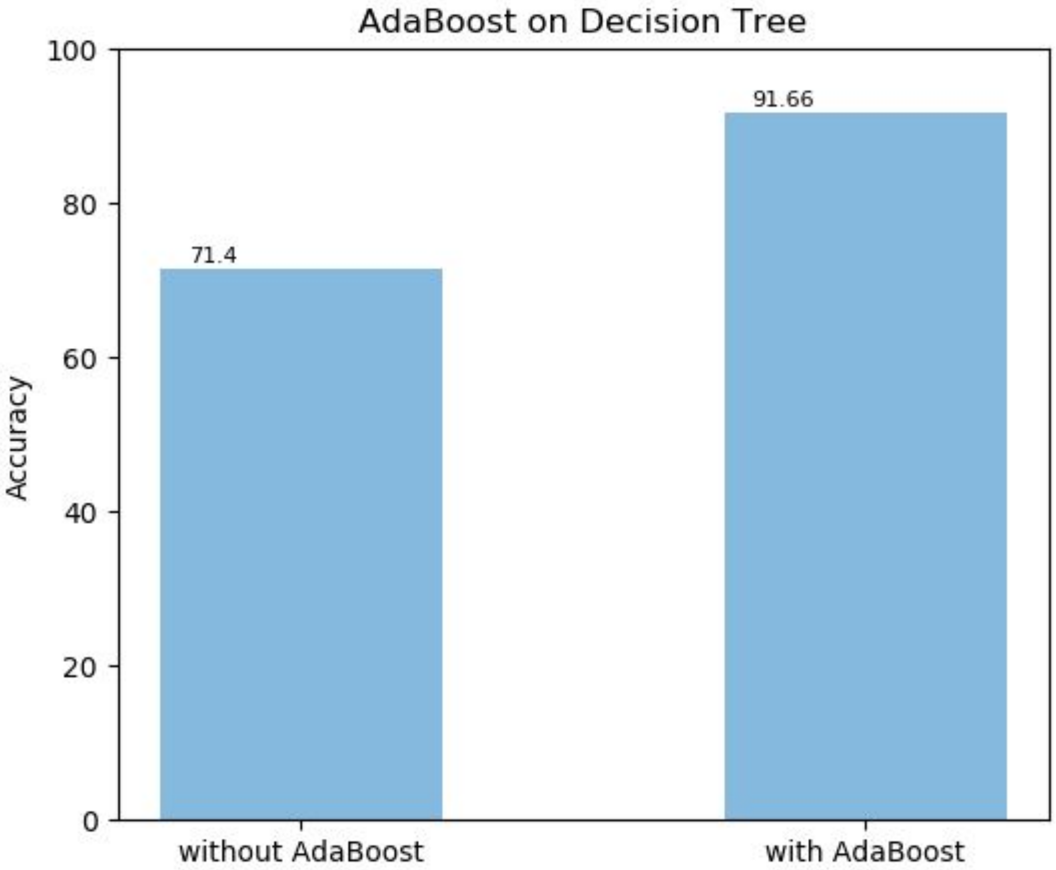


AdaBoost

- The main idea of boosting is to add additional models to the overall ensemble model sequentially.

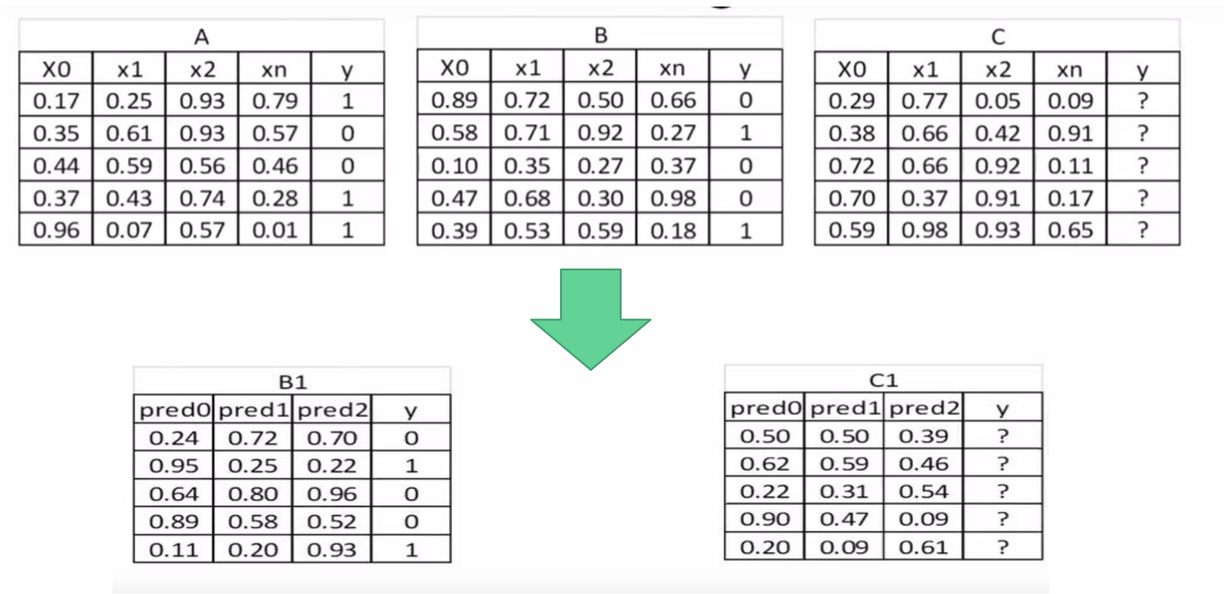

The errors that the first stump makes...
...influence how the second stump is made...

- This helps convert the weak learners to strong learners via an iterative process.


AdaBoost on Decision Tree
Accuracy
100
80
60
40
20
0
71.4
91.66
without AdaBoost
with AdaBoost

Stacking

- This technique helps combine the predictions of several trained models into one, thereby building one ensembled meta model that can be used to accurately classify the input.



A

X0	x1	x2	xn	y
0.17	0.25	0.93	0.79	1
0.35	0.61	0.93	0.57	0
0.44	0.59	0.56	0.46	0
0.37	0.43	0.74	0.28	1
0.96	0.07	0.57	0.01	1

B

X0	x1	x2	xn	y
0.89	0.72	0.50	0.66	0
0.58	0.71	0.92	0.27	1
0.10	0.35	0.27	0.37	0
0.47	0.68	0.30	0.98	0
0.39	0.53	0.59	0.18	1

C

X0	x1	x2	xn	y
0.29	0.77	0.05	0.09	?
0.38	0.66	0.42	0.91	?
0.72	0.66	0.92	0.11	?
0.70	0.37	0.91	0.17	?
0.59	0.98	0.93	0.65	?

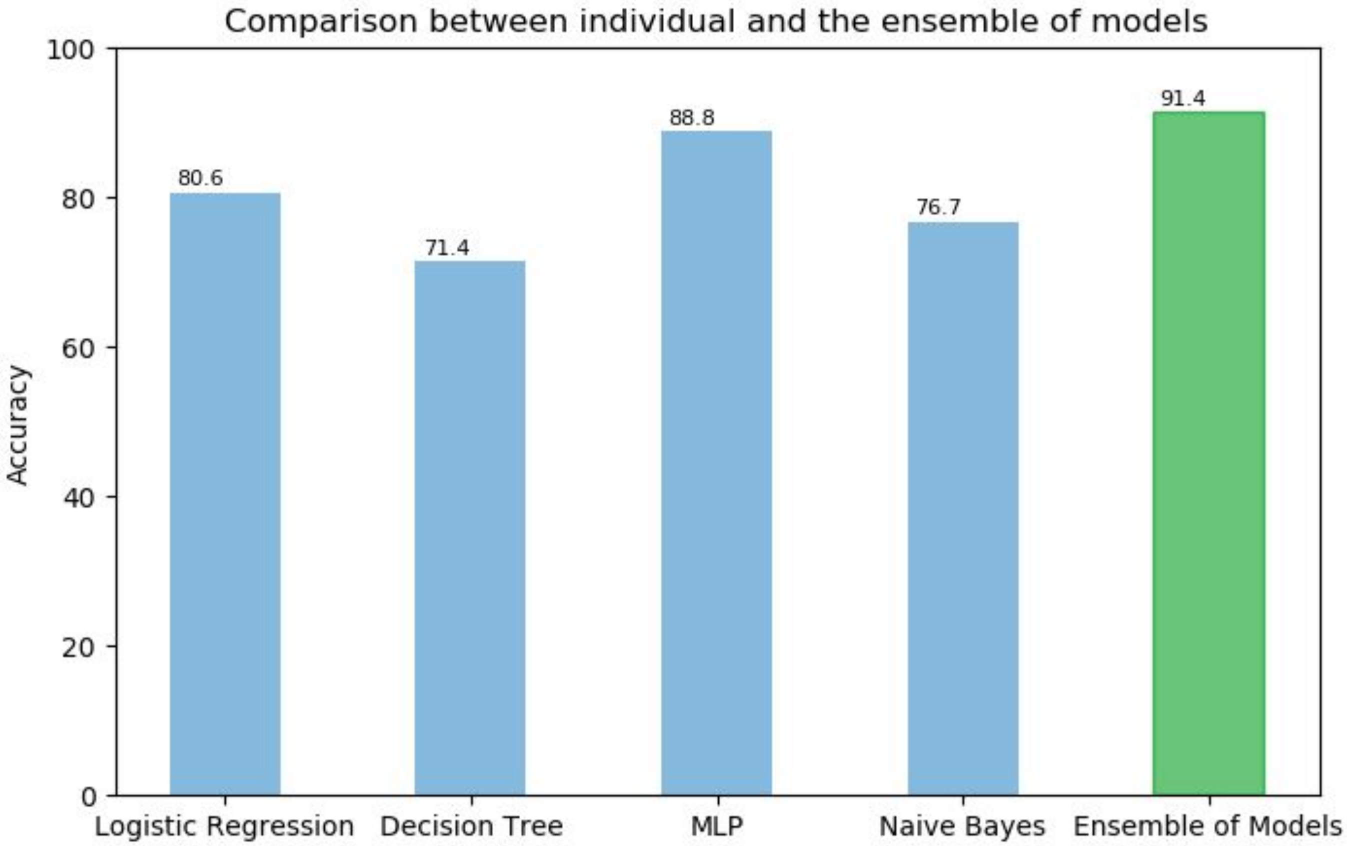
B1

pred0	pred1	pred2	y
0.24	0.72	0.70	0
0.95	0.25	0.22	1
0.64	0.80	0.96	0
0.89	0.58	0.52	0
0.11	0.20	0.93	1

C1

pred0	pred1	pred2	y
0.50	0.50	0.39	?
0.62	0.59	0.46	?
0.22	0.31	0.54	?
0.90	0.47	0.09	?
0.20	0.09	0.61	?

Summary and Results



Future Scope of the project :

- To the make the project commercially viable we shall implement the model into a network analysis software such as wireshark.
- Doing this will enable us to classify network packets in real-time.

References

- Evaluation of Machine Learning Algorithms for Intrusion Detection System
- Research on Intrusion Detection Model Using Ensemble learning Methods
- An Ensemble Approach for Intrusion Detection System Using Machine Learning Algorithms
- MCS '00 Proceedings of the First International Workshop on Multiple Classifier Systems
- A Survey on Ensemble Learning for Data Stream Classification
- Ensemble learning Methods
- Ensemble Methods in Machine Learning
- Ensemble learning methodologies

THANK YOU!