AI for cybersecurity

Simone Conti Mat. 675682

Dataset

The dataset I had was BCCC-cPacket-Cloud-DDoS-2024[1] which was made for Cloud-base DDoS Attack Classification. It consists of over 300 features and has a size of around 1,21 GB.

Project Goal

Identify most reliable model that can distinguish between normal traffic and DDOS attacks

Identify the model which minimise the false negatives

1: https://www.kaggle.com/datasets/bcccdatasets/bccc-cpacket-cloud-ddos-2024

First look at the dataset

	flow_id	timestamp	src_ip	src_port	dst_ip	dst_port	protocol	duration	packets_count	fwd_packets_count	...
0	35.203.211.133_54573_10.0.4.57_25094_TCP_2023-...	2023-12-14 09:01:03.508091	35.203.211.133	54573	10.0.4.57	25094	TCP	0.000063	3	2	...
1	10.0.4.57_25094_35.203.211.133_54573_TCP_2023-...	2023-12-14 09:01:03.508156	10.0.4.57	25094	35.203.211.133	54573	TCP	0.0	1	0	...
2	35.203.211.133_54573_10.0.4.57_25094_TCP_2023-...	2023-12-14 09:01:03.508431	35.203.211.133	54573	10.0.4.57	25094	TCP	0.000028	3	1	...
3	162.142.125.181_9147_10.0.4.57_18060_TCP_2023-...	2023-12-14 09:01:06.696817	162.142.125.181	9147	10.0.4.57	18060	TCP	0.000055	3	2	...
4	10.0.4.57_18060_162.142.125.181_9147_TCP_2023-...	2023-12-14 09:01:06.696874	10.0.4.57	18060	162.142.125.181	9147	TCP	0.0	1	0	...

5 rows × 324 columns

01 We have 324 columns

02 We have a mix of categorical and numerical features

03 Can we remove some of them?

The flow_id is a composite of all the other columns.
I did not want my model to limit itself to a specific timestamp period.
The column protocol only had tcp as a value, not useful.

What about the ip addresses?

	flow_id	timestamp	src_ip	src_port	dst_ip	dst_port	protocol	duration	packets_count	fwd_packets_count	...
0	35.203.211.133_54573_10.0.4.57_25094_TCP_2023-...	2023-12-14 09:01:03.508091	35.203.211.133	54573	10.0.4.57	25094	TCP	0.000063	3	2	...
1	10.0.4.57_25094_35.203.211.133_54573_TCP_2023-...	2023-12-14 09:01:03.508156	10.0.4.57	25094	35.203.211.133	54573	TCP	0.0	1	0	...
2	35.203.211.133_54573_10.0.4.57_25094_TCP_2023-...	2023-12-14 09:01:03.508431	35.203.211.133	54573	10.0.4.57	25094	TCP	0.000028	3	1	...
3	162.142.125.181_9147_10.0.4.57_18060_TCP_2023-...	2023-12-14 09:01:06.696817	162.142.125.181	9147	10.0.4.57	18060	TCP	0.000055	3	2	...
4	10.0.4.57_18060_162.142.125.181_9147_TCP_2023-...	2023-12-14 09:01:06.696874	10.0.4.57	18060	162.142.125.181	9147	TCP	0.0	1	0	...

5 rows × 324 columns

01 There are over 300+ uniques ips for both src and dst ips.

02 One Hot encoder unfeasible due to amount of ips

03 I don’t want my model to use the ips for classification

Because there might be cases in which i will not be able to have the ips information during an attack i don’t want my model to learn that a specific address is always malign. I want a more generalized model, i will remove the columns with the ports.

timestamp	2
payload_bytes_skewness	540271
payload_bytes_cov	540111
fwd_payload_bytes_skewness	476027
fwd_payload_bytes_cov	475285
bwd_payload_bytes_skewness	196913
bwd_payload_bytes_cov	196675
skewness_header_bytes	456723
fwd_skewness_header_bytes	477282
bwd_skewness_header_bytes	175973
active_skewness	362
idle_skewness	359
packets_IAT_skewness	449922
packets_IAT_cov	242
fwd_packets_IAT_skewness	496055
fwd_packets_IAT_cov	65307
bwd_packets_IAT_skewness	540398
bwd_packets_IAT_cov	345715
skewness_packets_delta_time	119741
cov_packets_delta_time	243
skewness_bwd_packets_delta_time	51388
cov_bwd_packets_delta_time	129
skewness_fwd_packets_delta_time	35066
cov_fwd_packets_delta_time	195
skewness_packets_delta_len	161362
cov_packets_delta_len	49033
skewness_bwd_packets_delta_len	52604
cov_bwd_packets_delta_len	1445
skewness_fwd_packets_delta_len	79765
cov_fwd_packets_delta_len	78916
skewness_header_bytes_delta_len	180376
cov_header_bytes_delta_len	126468
skewness_bwd_header_bytes_delta_len	53949
cov_bwd_header_bytes_delta_len	32549
skewness_fwd_header_bytes_delta_len	81862
cov_fwd_header_bytes_delta_len	81405
skewness_payload_bytes_delta_len	210316
cov_payload_bytes_delta_len	210016
skewness_bwd_payload_bytes_delta_len	54125
cov_bwd_payload_bytes_delta_len	53489
skewness_fwd_payload_bytes_delta_len	80738
cov_fwd_payload_bytes_delta_len	80150

dtype: int64

NaN values and labels

Many NaN Values in different columns.
Removed them because i already have a massive dataset, and the remaining amount of instances are more than enough

label	
Benign	60243
Attack	57654
Suspicious	37903
label	2

Wrong label “label” with only 2 instances,.
Will be removed completely.

Duplicated values and inf values

35	False
72	False
73	False
74	False
75	False
...	...
700767	False
700768	False
700770	False
700771	False
700772	False

155800 rows × 1 columns

dtype: bool

There were no duplicated values, nothing to be done.

```
Number of infinite values in column 'cov_packets_delta_len': 134
Number of infinite values in column 'cov_bwd_packets_delta_len': 245
Number of infinite values in column 'cov_fwd_packets_delta_len': 808
Number of infinite values in column 'cov_header_bytes_delta_len': 141
Number of infinite values in column 'cov_bwd_header_bytes_delta_len': 60
Number of infinite values in column 'cov_fwd_header_bytes_delta_len': 812
Number of infinite values in column 'cov_payload_bytes_delta_len': 153858
Number of infinite values in column 'cov_bwd_payload_bytes_delta_len': 100465
Number of infinite values in column 'cov_fwd_payload_bytes_delta_len': 147932
```

Some columns have most of the values inf. The 3 columns 'cov_payload_bytes_delta_len', 'cov_bwd_payload_bytes_delta_len', 'cov_fwd_payload_bytes_delta_len' have mostly inf values, so i decided to drop them completely, while all other inf values are treated as NaN and dropped.

Fix columns type

timestamp	object
src_ip	object
src_port	object
dst_ip	object
dst_port	object
...	...
median_fwd_payload_bytes_delta_len	object
skewness_fwd_payload_bytes_delta_len	object
cov_fwd_payload_bytes_delta_len	object
label	object
activity	object

323 rows × 1 columns

The data is misrepresented in the dataset. We have instances where values like , e.g., 5 is present both as a number and as a string such as ‘5’. I decided to represent all of them as a number.

The columns timestamp, ips and ports are still present but they are removed after this phase.

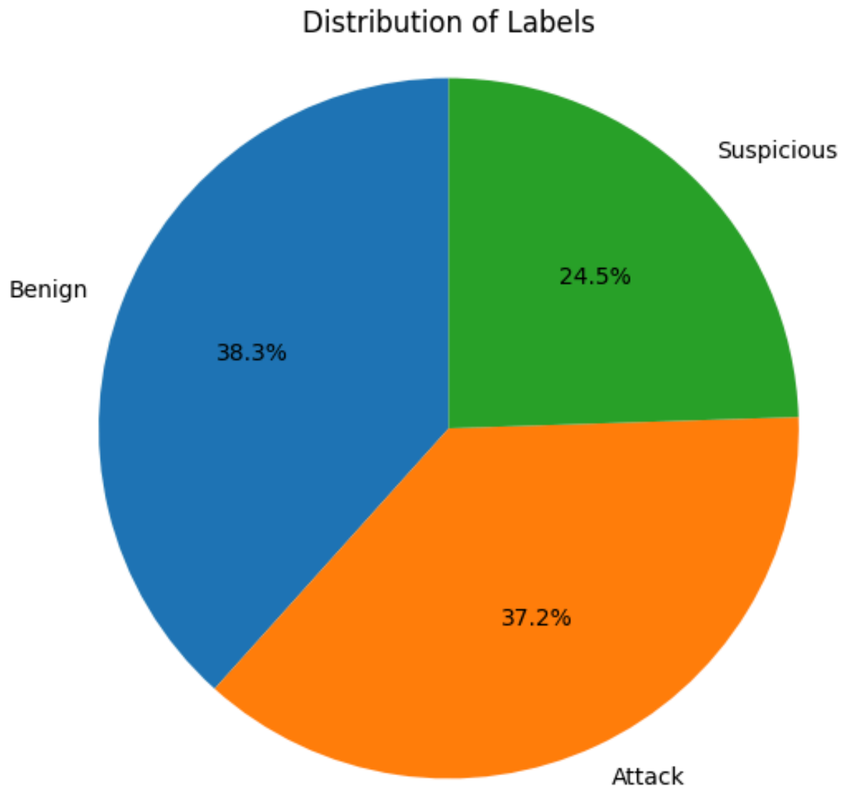
duration	float64
packets_count	int64
fwd_packets_count	int64
bwd_packets_count	int64
total_payload_bytes	int64
fwd_total_payload_bytes	int64
bwd_total_payload_bytes	int64
payload_bytes_max	int64
payload_bytes_min	int64
payload_bytes_mean	float64
payload_bytes_std	float64
payload_bytes_variance	float64
payload_bytes_median	float64
payload_bytes_skewness	float64
payload_bytes_cov	float64
payload_bytes_mode	float64
fwd_payload_bytes_max	int64
fwd_payload_bytes_min	int64
fwd_payload_bytes_mean	float64
fwd_payload_bytes_std	float64
fwd_payload_bytes_variance	float64
fwd_payload_bytes_median	float64

Final representation of the data (truncated for length)

Class distribution

label	
Benign	59304
Attack	57505
Suspicious	37873

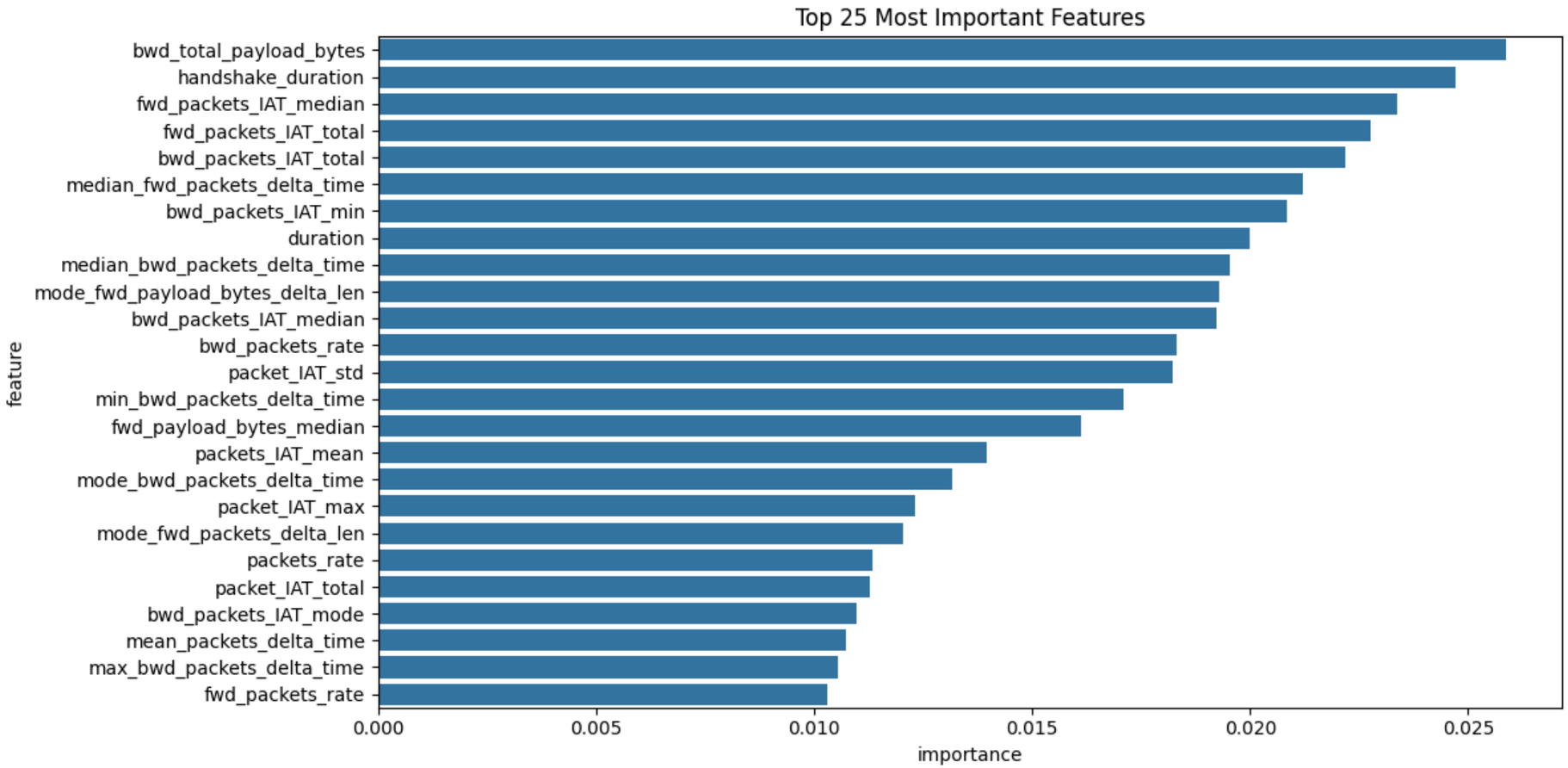
The class distribution is basically the same as before



The classes are slightly unbalanced, we have less instances of suspicious classes

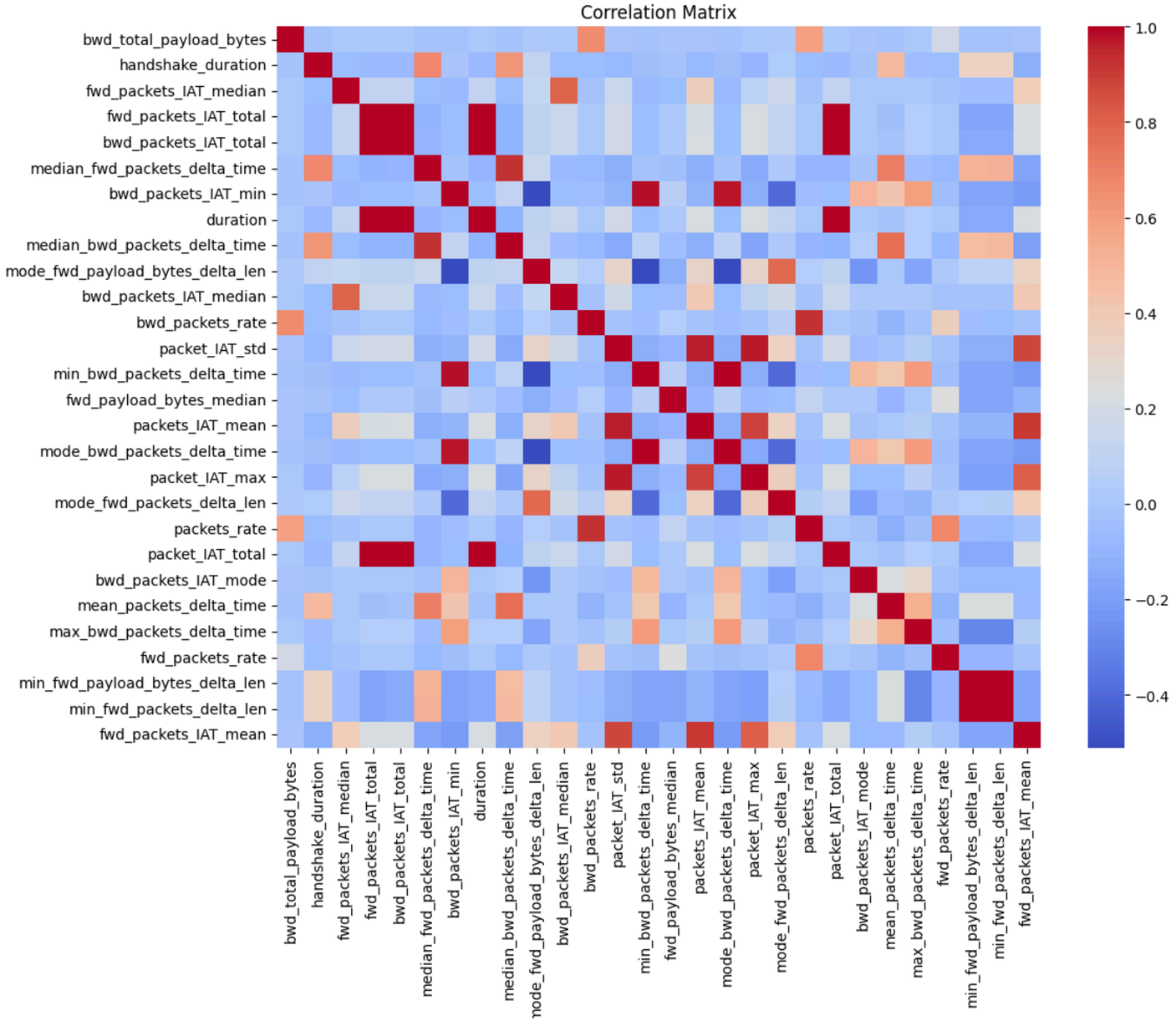
Feature Selection

I divided the dataset as training and test set as 80/20 distribuition

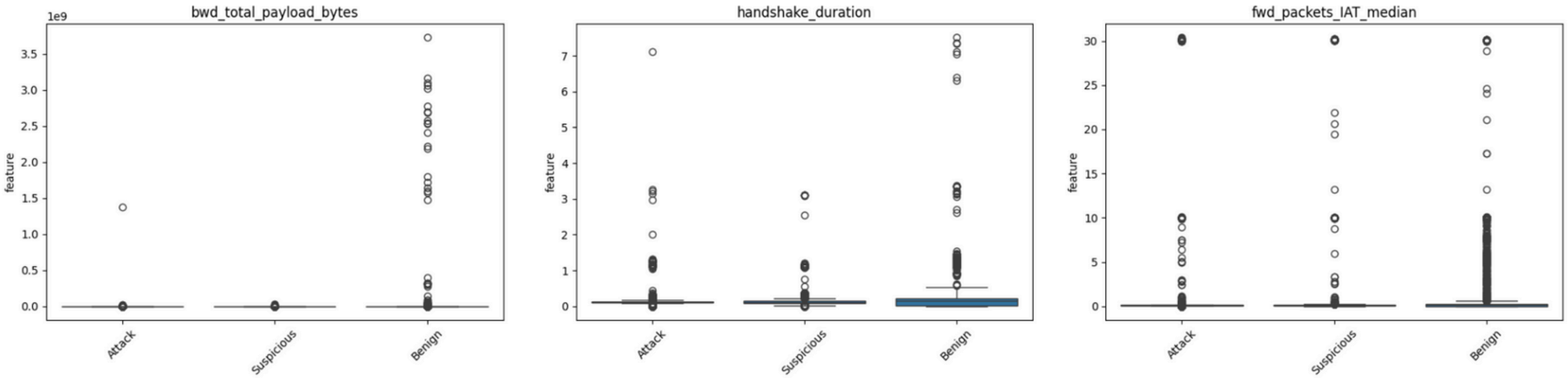


With threshold of 0.010 i kept 28 columns

Correlation Matrix

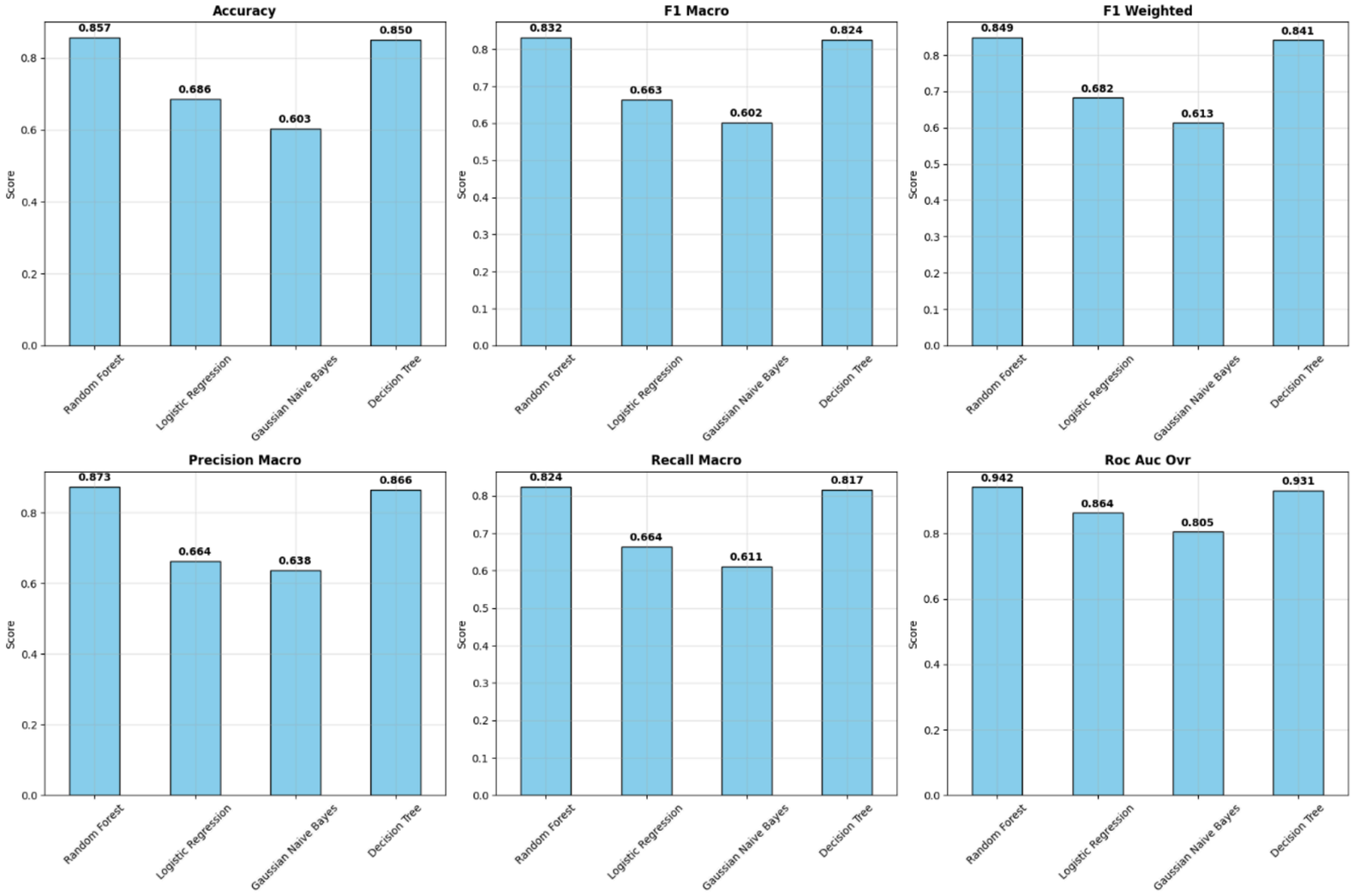


Outlier Detection



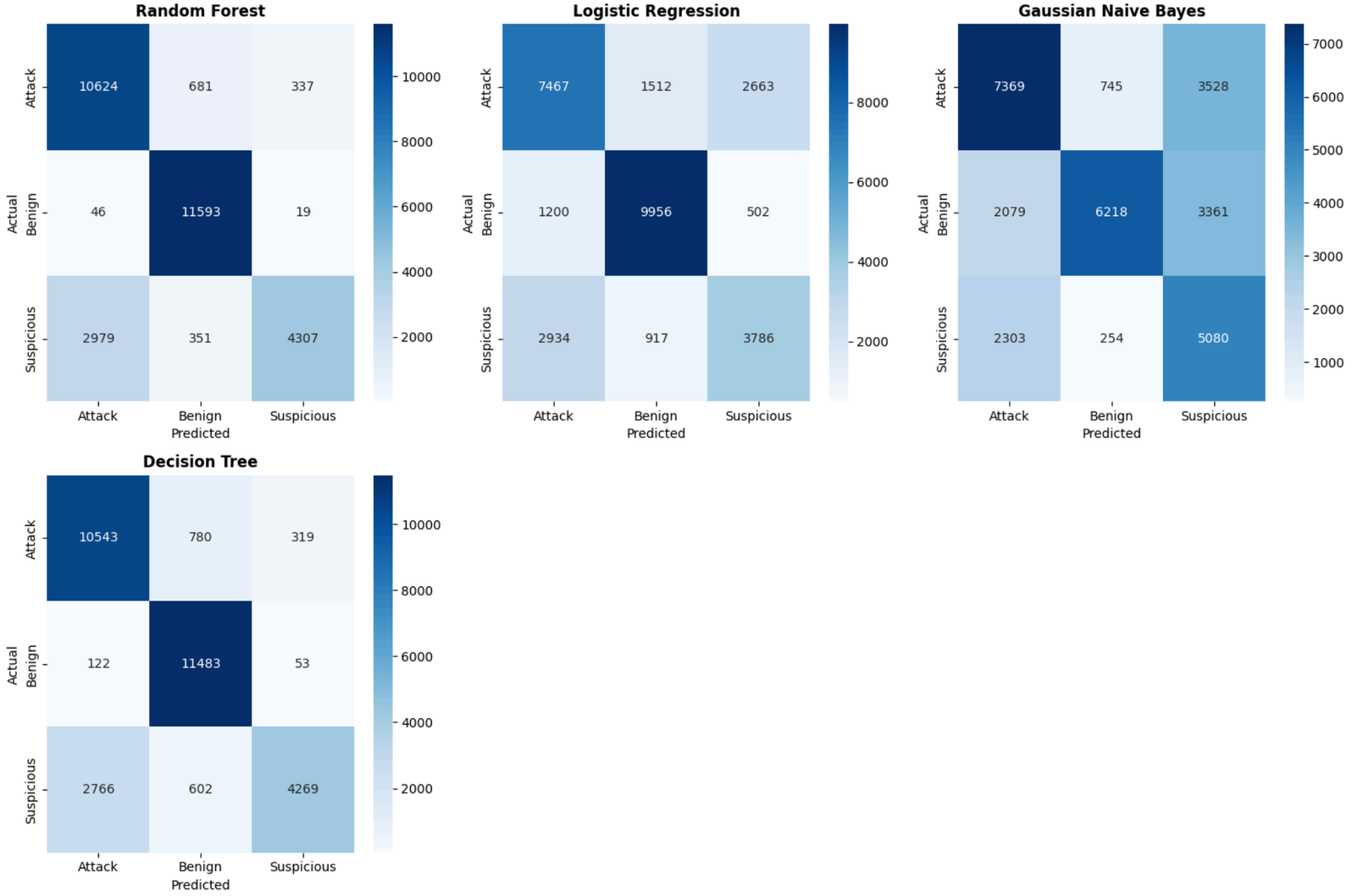
Some of the outliers found.
Most of them are part of the benign class, if i removed them i might “kill” the classification of that class, i decided to keep them

Comparison with other models

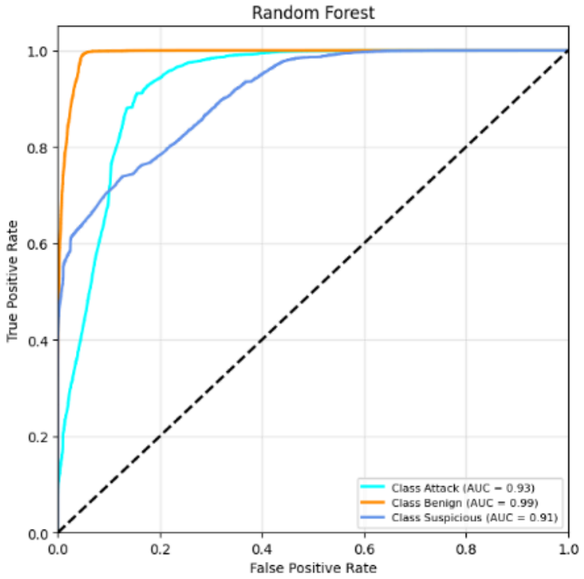


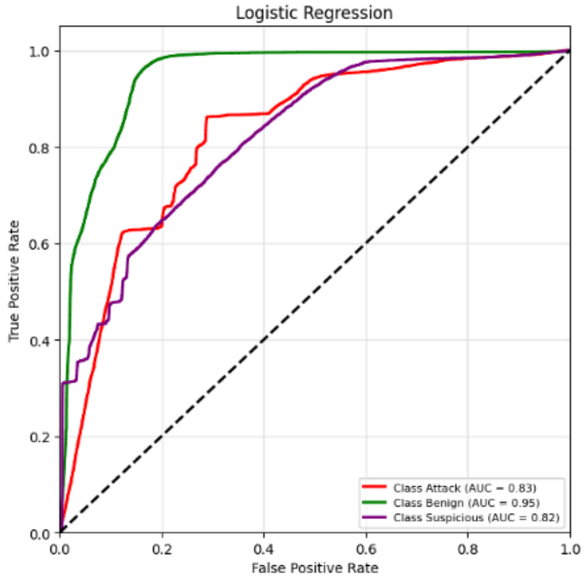
Cross validation was used for the training of all models

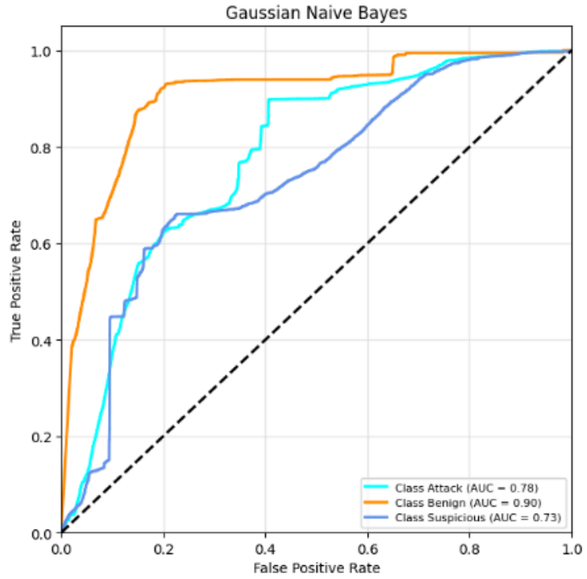
Comparison with other models

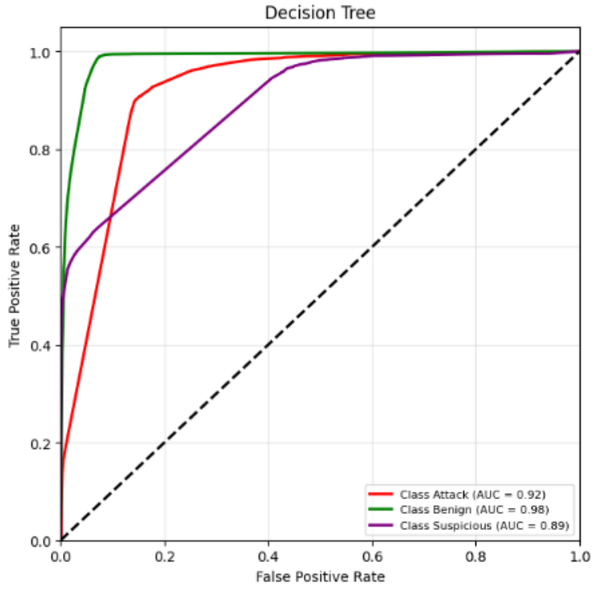


Comparison with other models









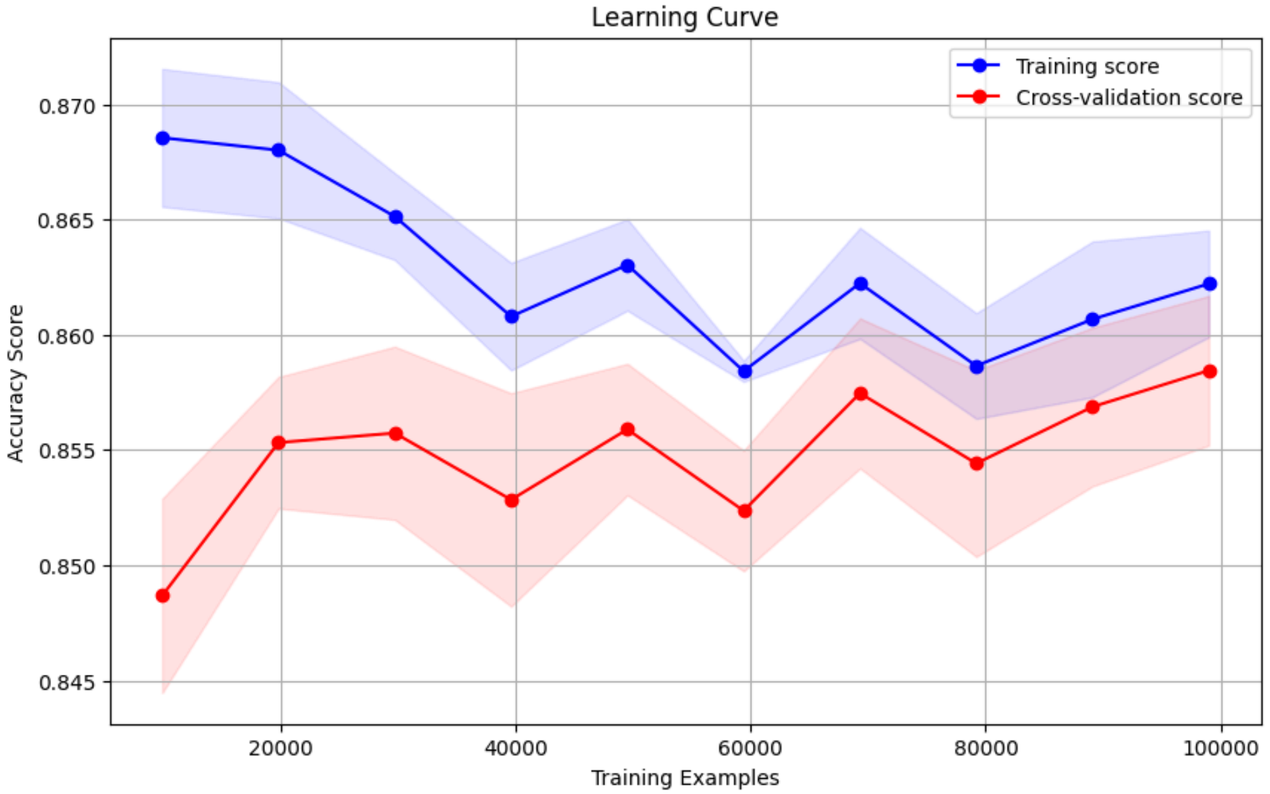
Conclusion

```
MODEL RANKING (by average score):
----------------------------------------
1. Random Forest: 0.8470
4. Decision Tree: 0.8395
2. Logistic Regression: 0.6717
3. Gaussian Naive Bayes: 0.6134

5. Final Results:
                 Model  Accuracy  F1-Score (Macro)  F1-Score (Weighted)
0        Random Forest    0.8574            0.8318               0.8488
1  Logistic Regression    0.6856            0.6633               0.6820
2  Gaussian Naive Bayes    0.6034            0.6024               0.6133
3        Decision Tree    0.8500            0.8243               0.8410

   Precision (Macro)  Recall (Macro)  ROC-AUC (OvR)
0             0.8734          0.8236         0.9418
1             0.6641          0.6637         0.8637
2             0.6377          0.6105         0.8048
3             0.8658          0.8165         0.9314
                 Model  Average Score
0        Random Forest       0.847008
3        Decision Tree       0.839522
```

As we can see from the finall results and all the previous graphs, Decision Tree and Random Forest behave the best in terms of recall and f1 score.

For my goal the best model will be the Random Forest, followed by Decision Tree. The other 2 are not suitable for this classification.

Conclusion



Looking at the learning curve for the Random Forest model we can also confirm that we do not have overfitting in our model.

What’s next

- Try to balance the dataset to see if the results change for the attacks classifications
- Try other models which are not tree based

THANKS FOR THE ATTENTION

Simone Conti Mat. 675682