Soutenance finale

Théo Ripoll – Tom Genlis – Arnaud Baradat – Quentin Fisch

01 Dataset Presentation

Chosen dataset

02 EDA

Exploratory data analysis

03 Analysis - Network

Methods and processes

04 Analysis - Physical

Methods and processes

05 Demo

Key results & adversarial attack

06 Conclusion

Recap of results and takeaways

01

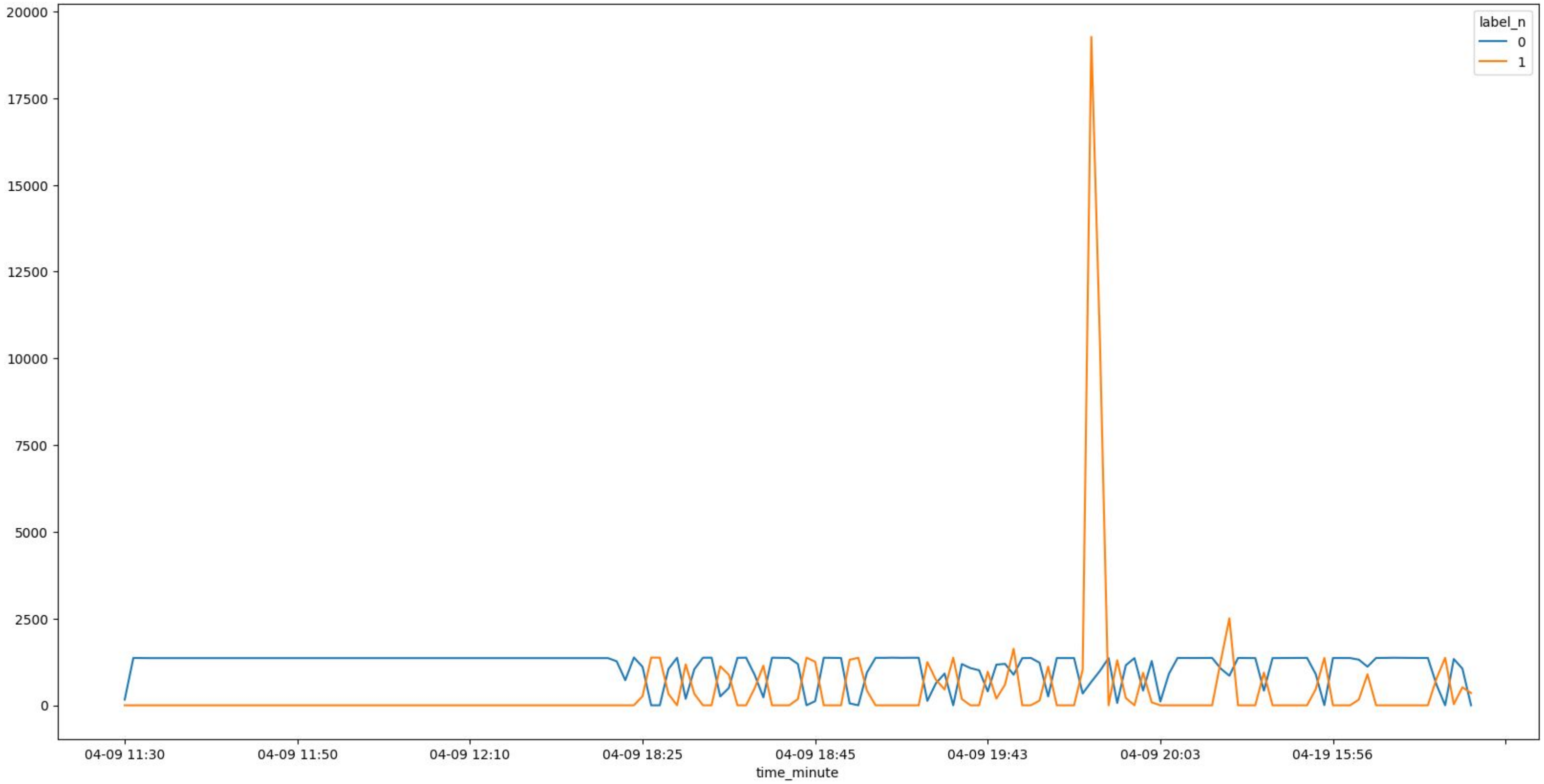
Dataset Presentation

Chosen dataset

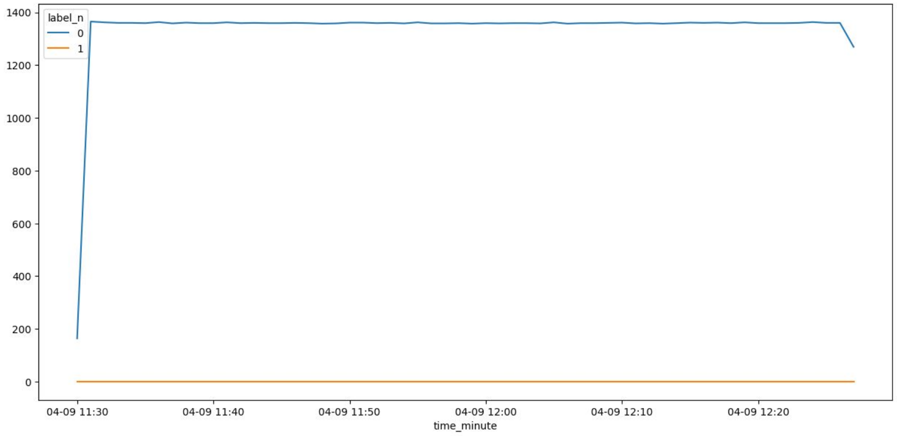
Hardware In The Loop

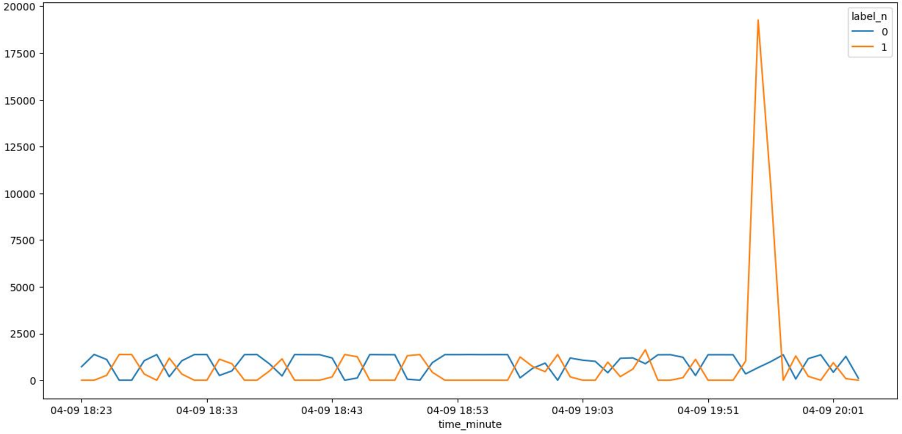
Physical and network data from a Water Distribution Testbed, simulating water flow with real and virtual components to analyze effects of cyber and physical attacks in a 2-hour period.

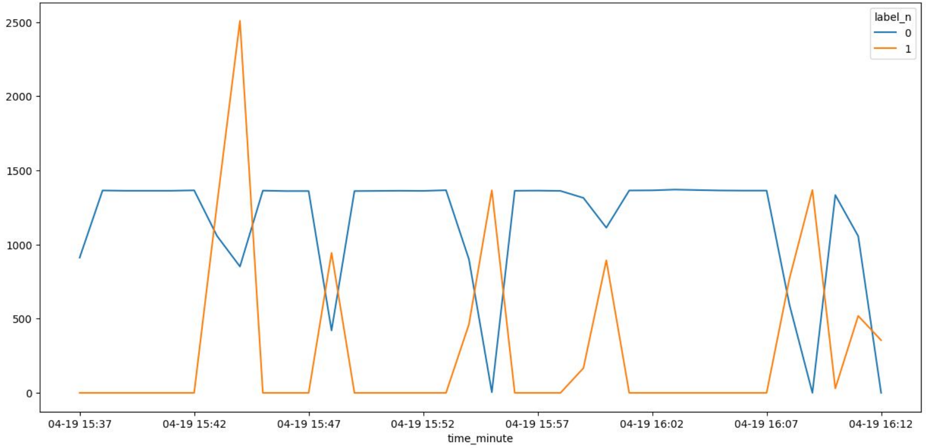
Timeline of attacks



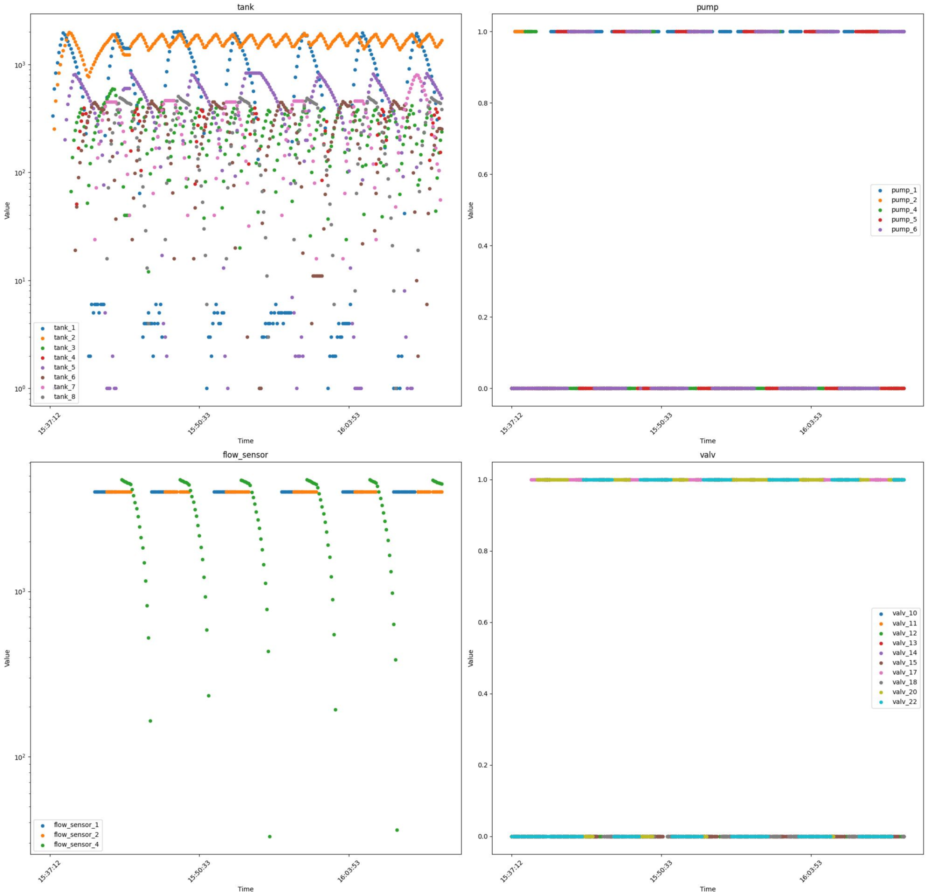
Timeline of attacks (2)







Physical sensors behavior



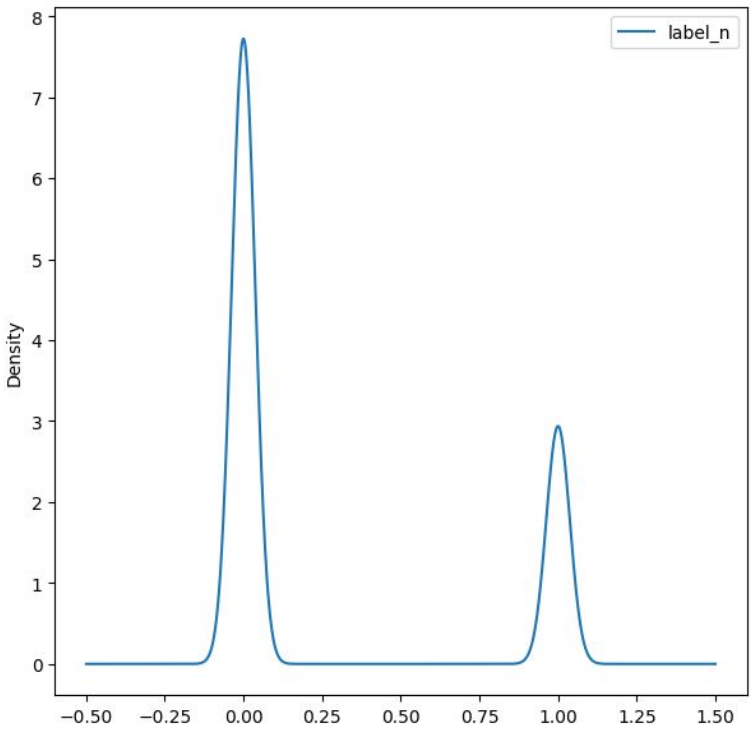
Scatter chart of each sensor type

02

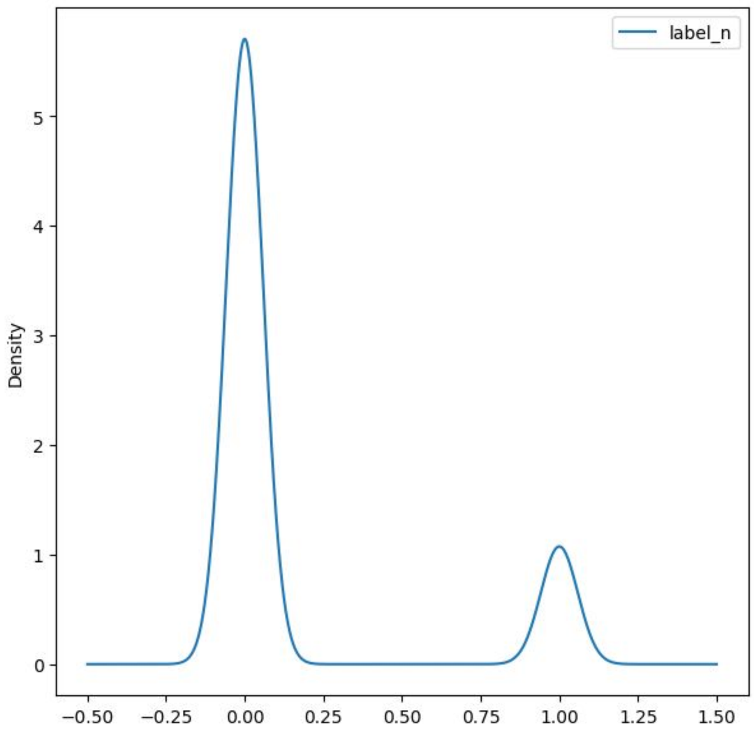
EDA

Exploratory Data Analysis

Labels - Binary class

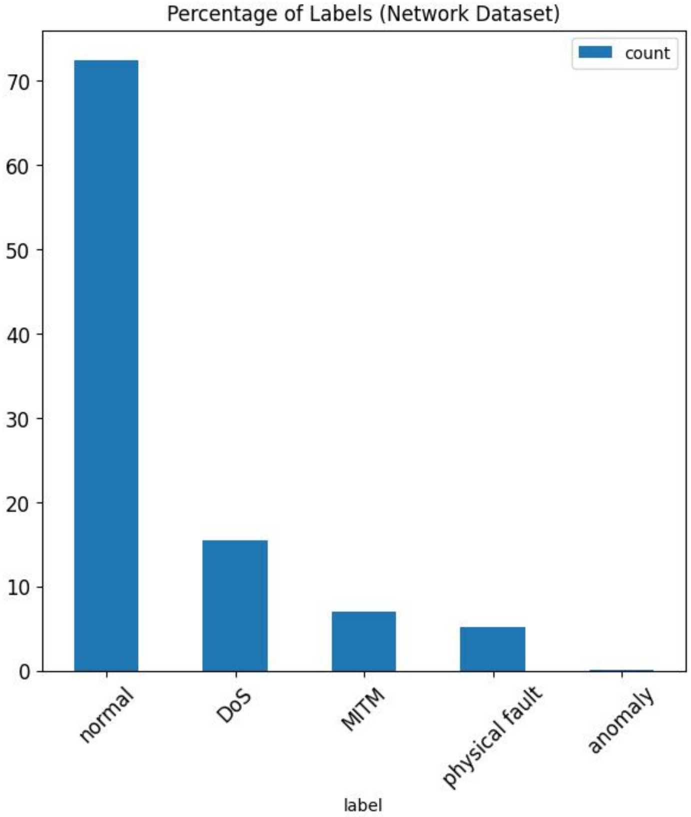


Density Plot of Network Dataset Labels

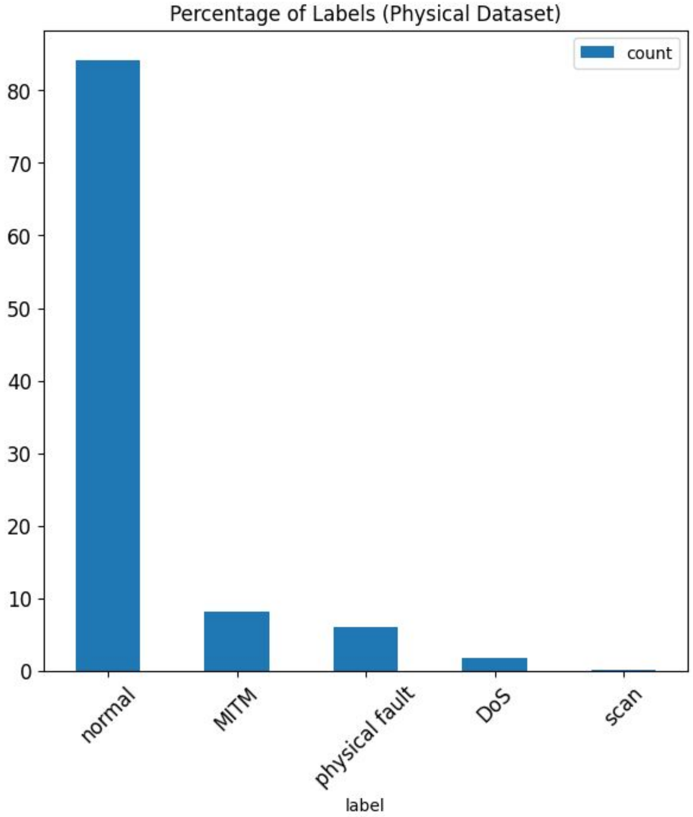


Density Plot of Physical Dataset Labels

Labels - Multi-class

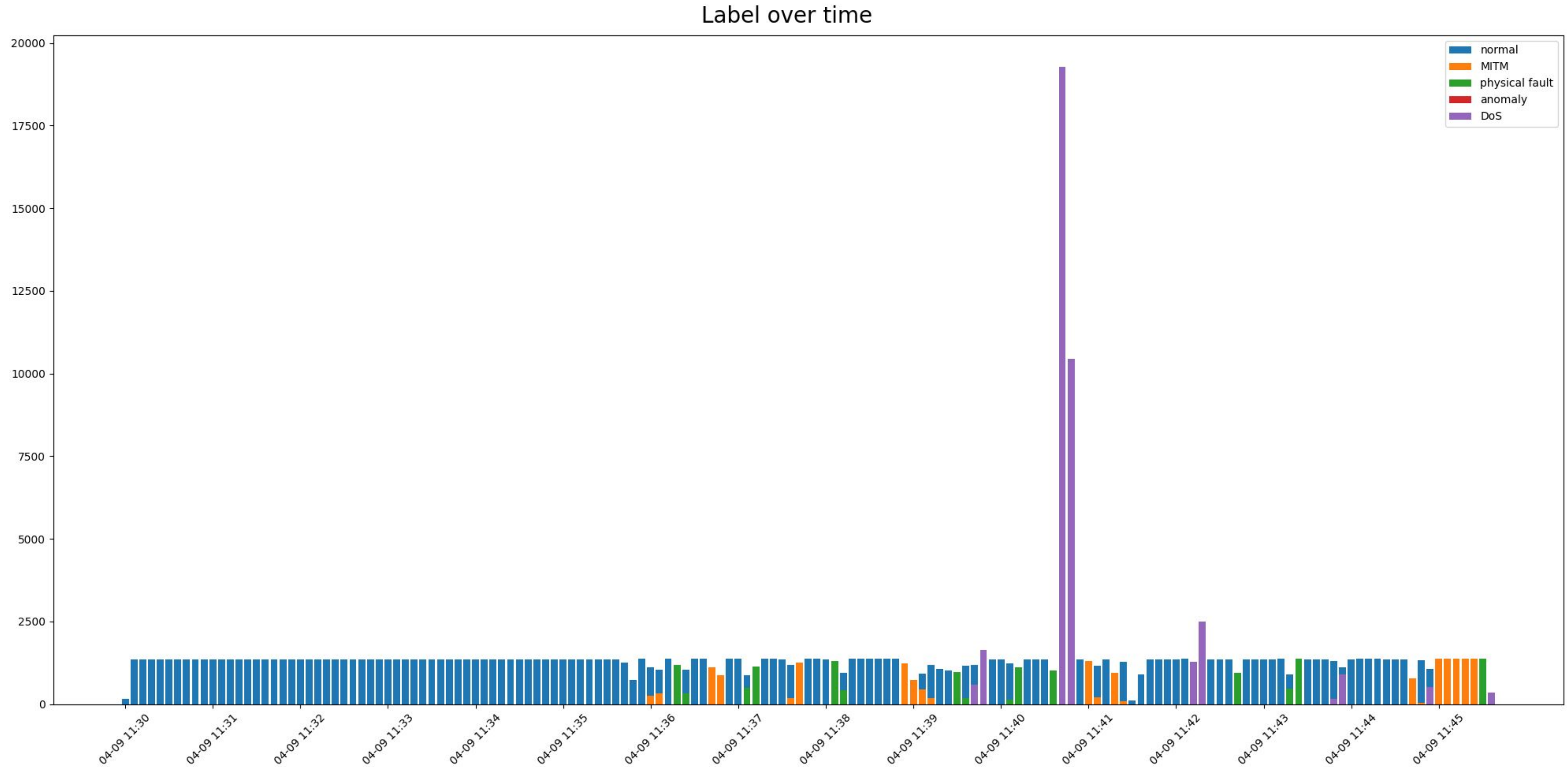


Percentage of Labels (Network Dataset)



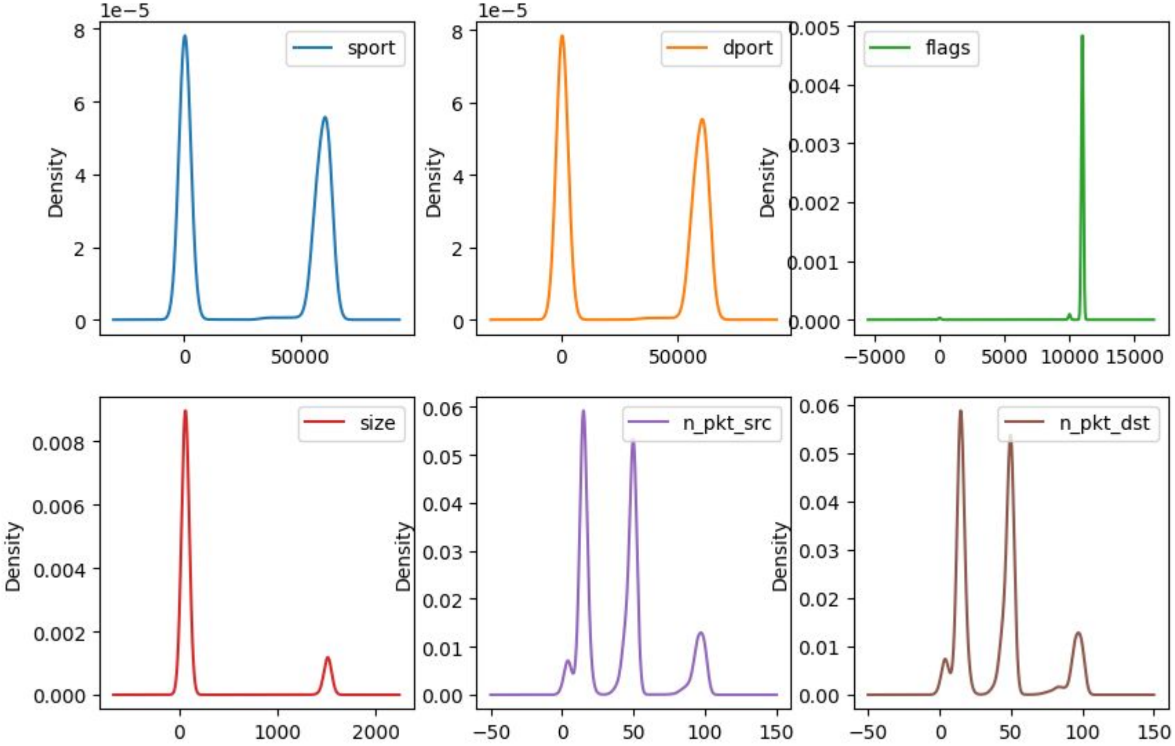
Percentage of Labels (Physical Dataset)

Quick rewind to our timeline

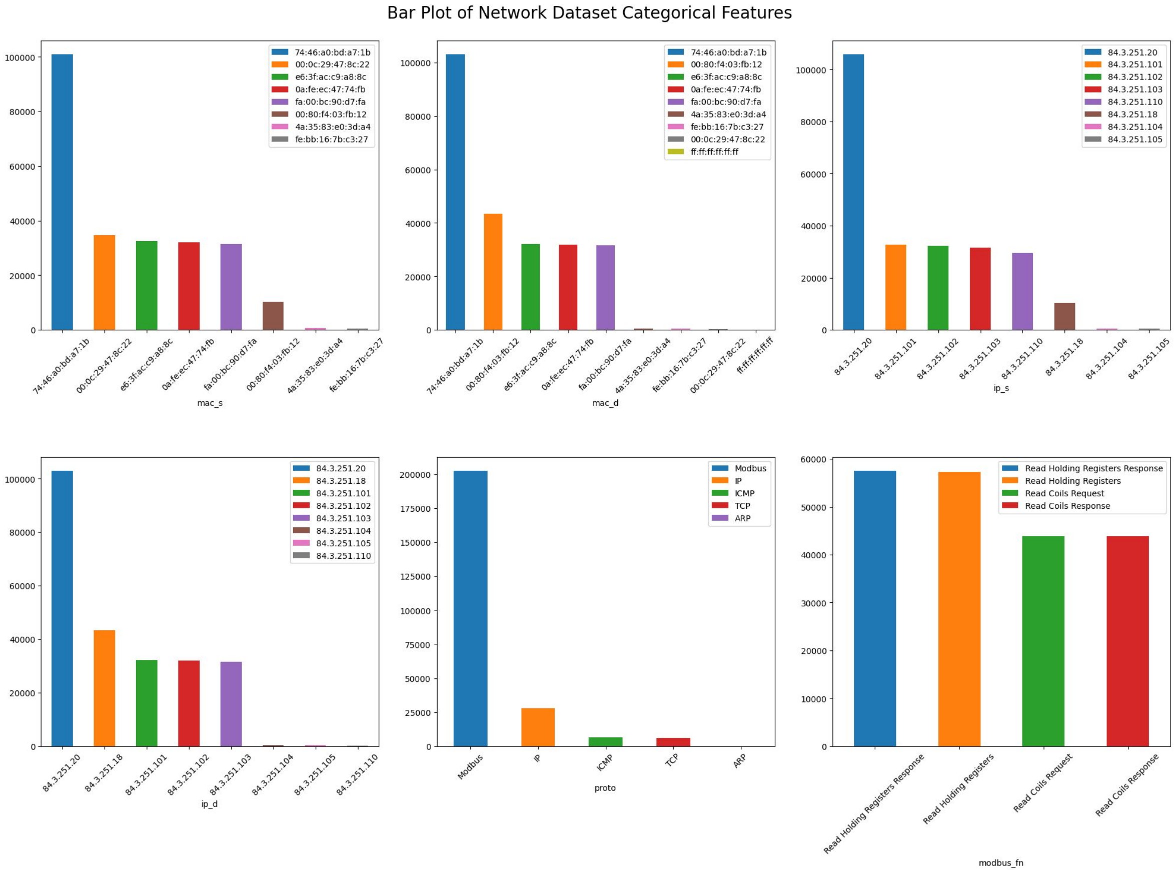


Network features

Density Plot of Network Dataset Numerical Features

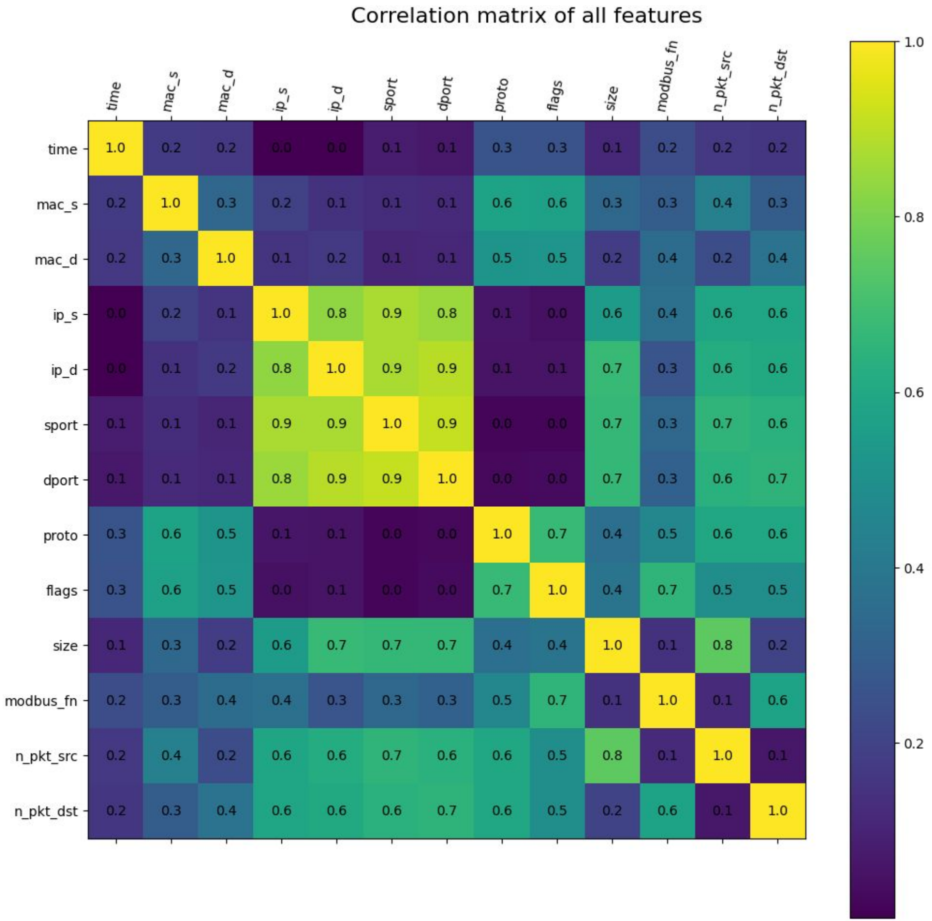


Network features (2)



Bar Plot of Network Dataset Categorical Features

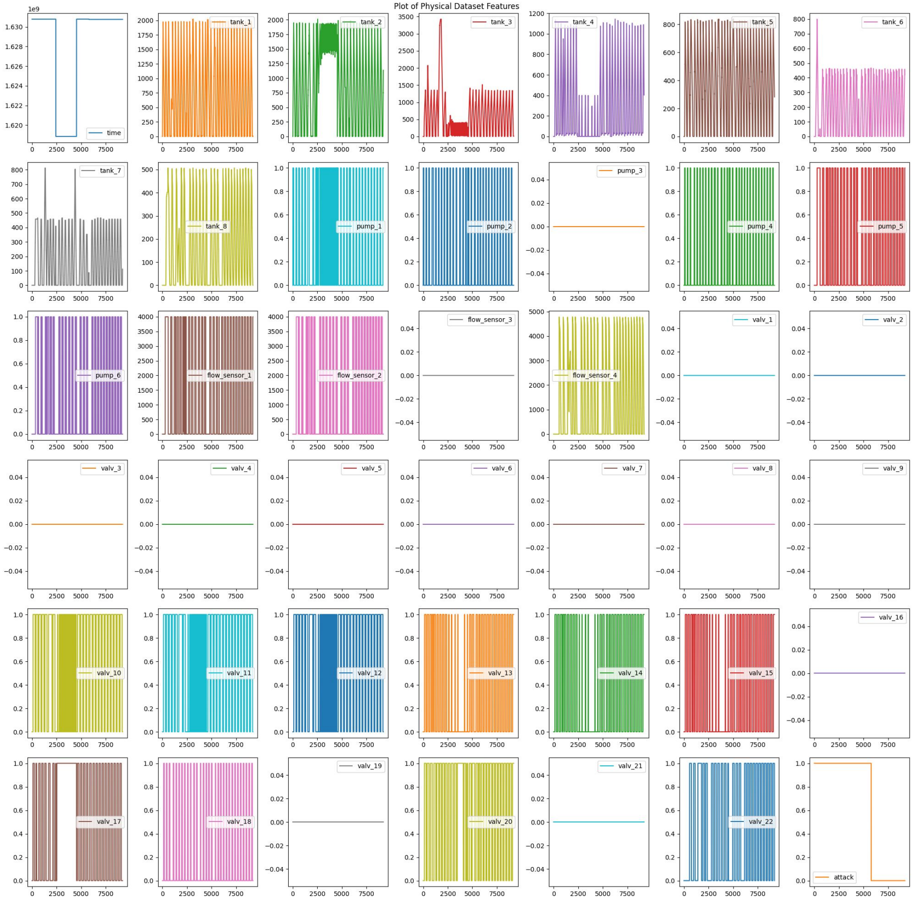
Network features (3)



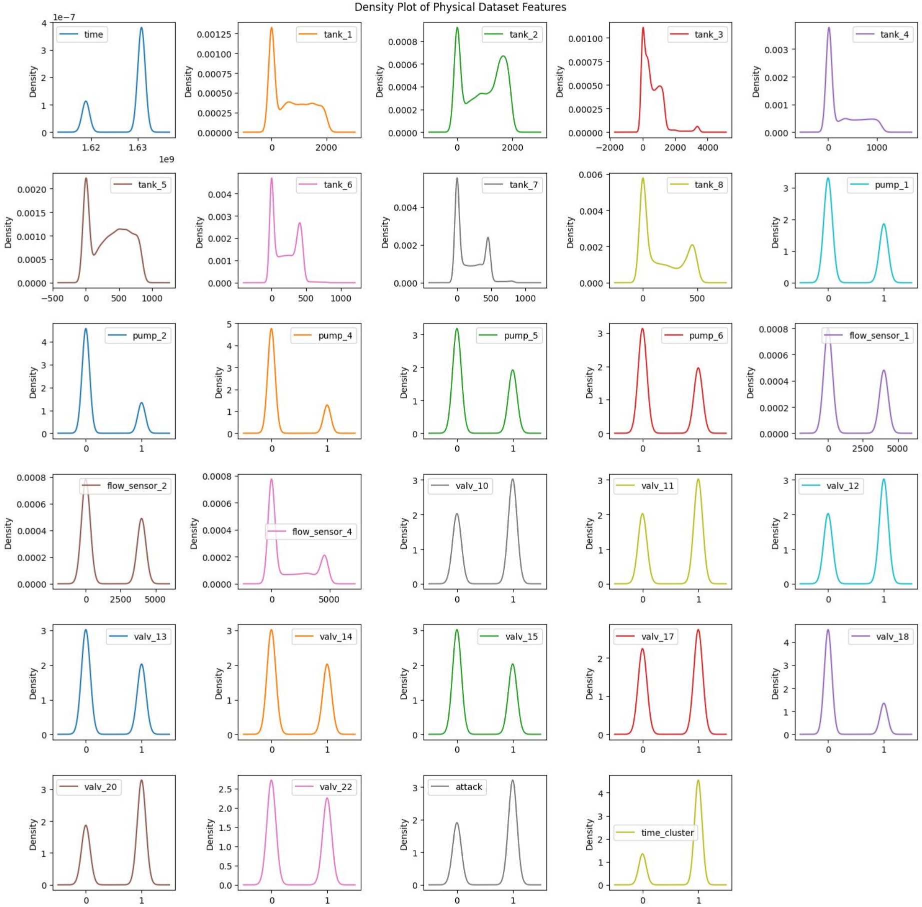
Correlation matrix of all features

	time	mac_s	mac_d	ip_s	ip_d	sport	dport	proto	flags	size	modbus_fn	n_pkt_src	n_pkt_dst
time	1.0	0.2	0.2	0.0	0.0	0.1	0.1	0.3	0.3	0.1	0.2	0.2	0.2
mac_s	0.2	1.0	0.3	0.2	0.1	0.1	0.1	0.6	0.6	0.3	0.3	0.4	0.3
mac_d	0.2	0.3	1.0	0.1	0.2	0.1	0.1	0.5	0.5	0.2	0.4	0.2	0.4
ip_s	0.0	0.2	0.1	1.0	0.8	0.9	0.8	0.1	0.0	0.6	0.4	0.6	0.6
ip_d	0.0	0.1	0.2	0.8	1.0	0.9	0.9	0.1	0.1	0.7	0.3	0.6	0.6
sport	0.1	0.1	0.1	0.9	0.9	1.0	0.9	0.0	0.0	0.7	0.3	0.7	0.6
dport	0.1	0.1	0.1	0.8	0.9	0.9	1.0	0.0	0.0	0.7	0.3	0.6	0.7
proto	0.3	0.6	0.5	0.1	0.1	0.0	0.0	1.0	0.7	0.4	0.5	0.6	0.6
flags	0.3	0.6	0.5	0.0	0.1	0.0	0.0	0.7	1.0	0.4	0.7	0.5	0.5
size	0.1	0.3	0.2	0.6	0.7	0.7	0.7	0.4	0.4	1.0	0.1	0.8	0.2
modbus_fn	0.2	0.3	0.4	0.4	0.3	0.3	0.3	0.5	0.7	0.1	1.0	0.1	0.6
n_pkt_src	0.2	0.4	0.2	0.6	0.6	0.7	0.6	0.6	0.5	0.8	0.1	1.0	0.1
n_pkt_dst	0.2	0.3	0.4	0.6	0.6	0.6	0.7	0.6	0.5	0.2	0.6	0.1	1.0

Physical Features



Physical Features (2)



03

Analysis - Network

Methods and processes

Process

- Focus on multi-class classification
- Use a StandardScaler
- Separate analysis:
  - Using full contextual information
  - Without contextual information
- Use the following metrics:
  - Accuracy
  - Recall
  - F1
  - MCC
  - Balanced accuracy
- Plot and analyse feature importance
- Same for network and physical datasets

Non-supervised Algorithms

Isolation Forest

- Without a fixed contamination rate:
  - **12505** outliers detected (5%)
  - **7695** are real anomalies
  - **61.5%** of precision
- Contamination rate at 27.5%:
  - 66k anomalies to find
  - **37940** real anomalies detected
  - **57.7%** of precision

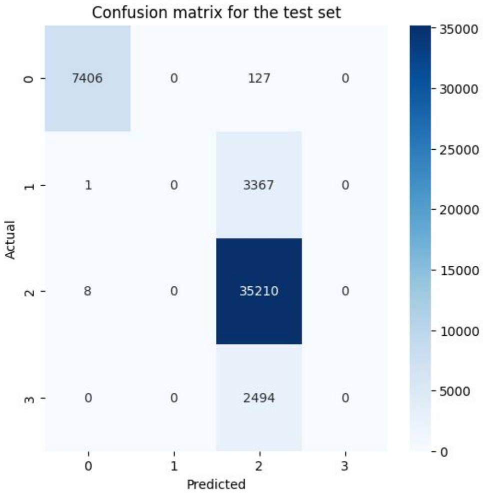
Local Outlier Factor

- Too slow for any production use
- Contamination rate at 27.5%:
  - **30919** outliers detected (12%)
  - Only **3599** are real anomalies

Deep Learning

Neural network

- Tried multiple architectures
- Final one has:
  - 3 hidden layers (1024, 256, 64)

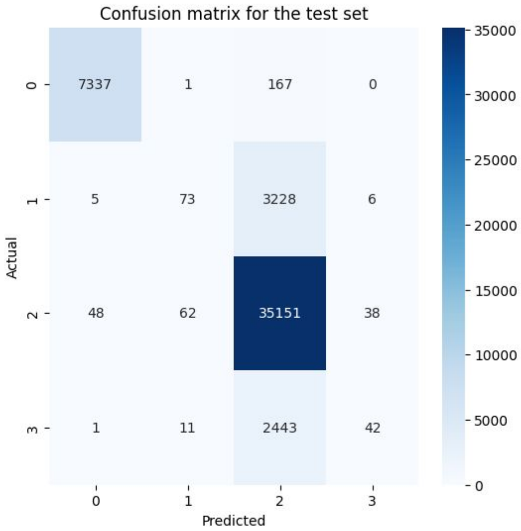


Confusion matrix for the test set

Actual \ Predicted	0	1	2	3
0	7406	0	127	0
1	1	0	3367	0
2	8	0	35210	0
3	0	0	2494	0

LSTM

- Similar results to the neural network

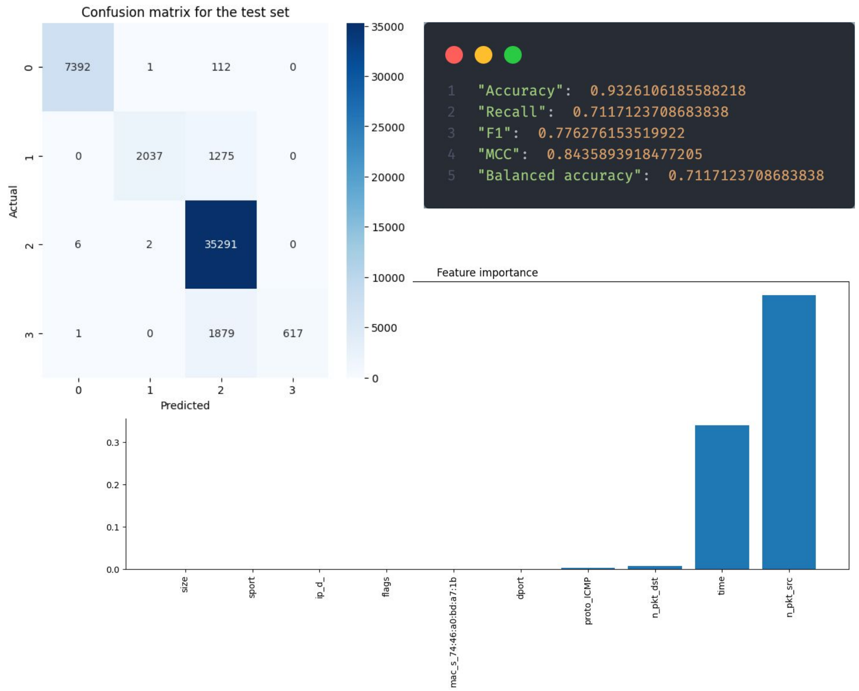


Confusion matrix for the test set

Actual \ Predicted	0	1	2	3
0	7337	1	167	0
1	5	73	3228	6
2	48	62	35151	38
3	1	11	2443	42

Classifiers - Decision Tree

With contextual information



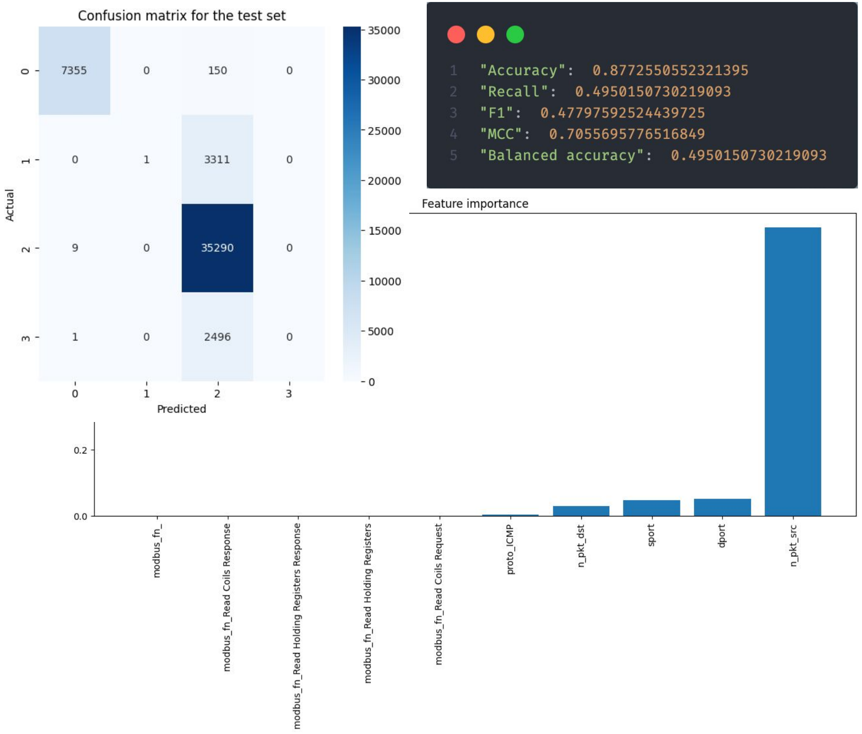
Confusion matrix for the test set

Actual \ Predicted	0	1	2	3
0	7392	1	112	0
1	0	2037	1275	0
2	6	2	35291	0
3	1	0	1879	617

```
"Accuracy":  0.9326106185588218
"Recall":  0.7117123708683838
"F1":  0.776276153519922
"MCC":  0.8435893918477205
"Balanced accuracy":  0.7117123708683838
```

Feature importance

No contextual information



Confusion matrix for the test set

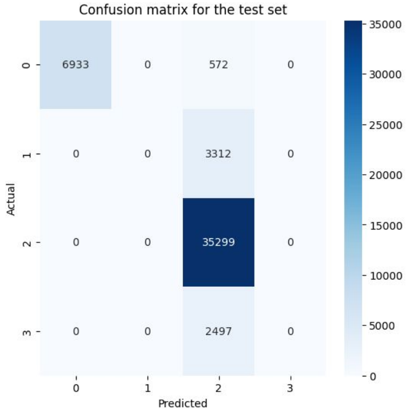
Actual \ Predicted	0	1	2	3
0	7355	0	150	0
1	0	1	3311	0
2	9	0	35290	0
3	1	0	2496	0

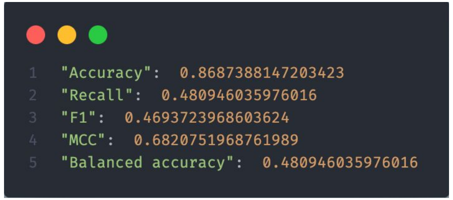
```
"Accuracy":  0.8772550552321395
"Recall":  0.4950150730219093
"F1":  0.47797592524439725
"MCC":  0.7055695776516849
"Balanced accuracy":  0.4950150730219093
```

Feature importance

Classifiers - Random Forest

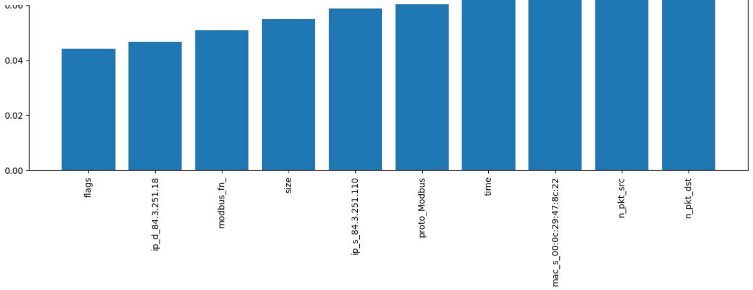
With contextual information



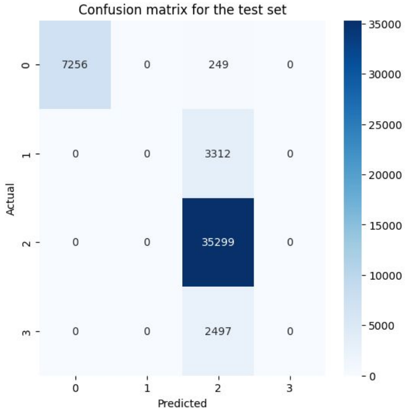


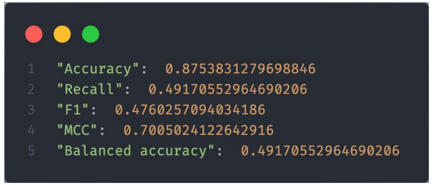
```
"Accuracy":  0.8687388147203423
"Recall":  0.480946035976016
"F1":  0.4693723968603624
"MCC":  0.6820751968761989
"Balanced accuracy":  0.480946035976016
```

Feature importance



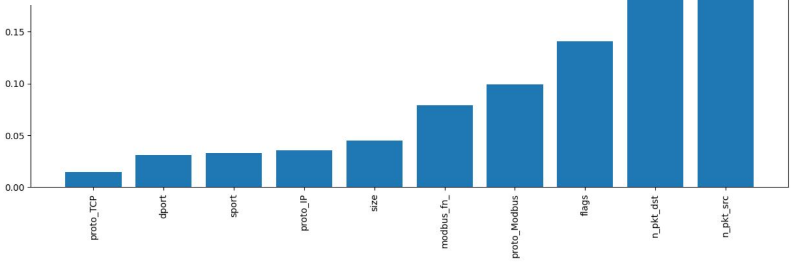
No contextual information

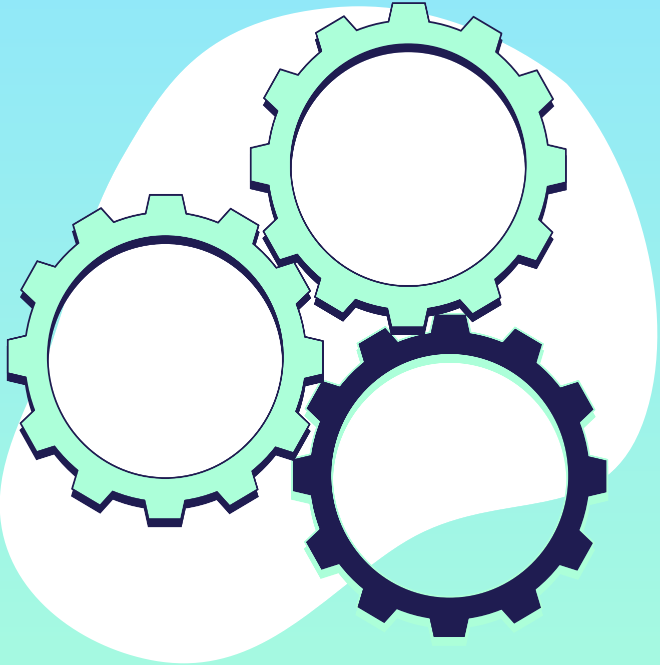




```
"Accuracy":  0.8753831279698846
"Recall":  0.49170552964690206
"F1":  0.4760257094034186
"MCC":  0.7005024122642916
"Balanced accuracy":  0.49170552964690206
```

Feature importance





04

Analysis - Physical

Methods and processes

Non-supervised Algorithms

Isolation Forest

- Without a fixed contamination rate:
  - **6177** outliers detected
  - **1180** are real anomalies
  - **19%** of precision
- Contamination rate at 16%:
  - 66k anomalies to find
  - **37940** real anomalies detected
  - **57.7%** of precision

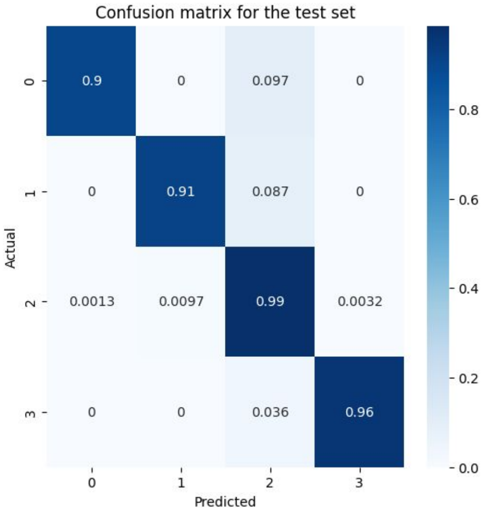
Local Outlier Factor

- Too slow for any production use
- **71** outliers detected (12%)
- Only **24** are real anomalies

Deep Learning

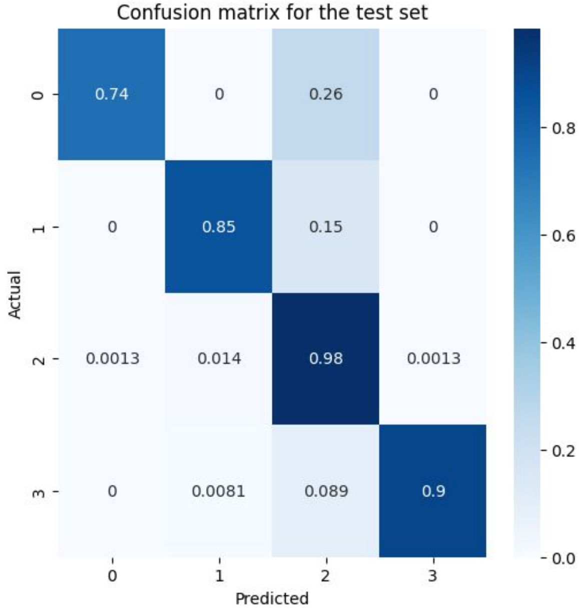
Neural network

- Tried multiple architectures
- Final one has:
  - 3 hidden layers (1024, 256, 64)



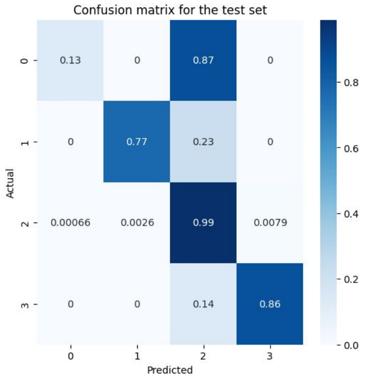
LSTM

- Similar results to the neural network

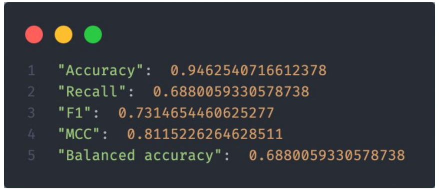


Classifiers - Decision Tree

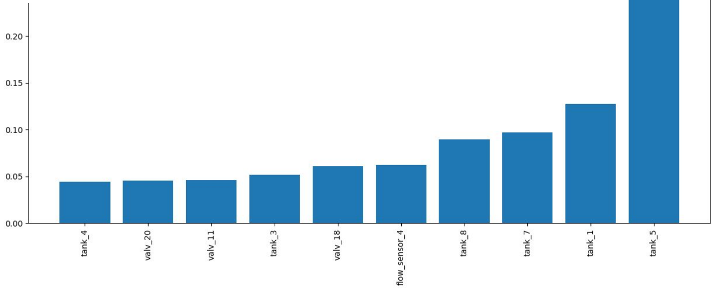
With contextual information


Confusion matrix for the test set

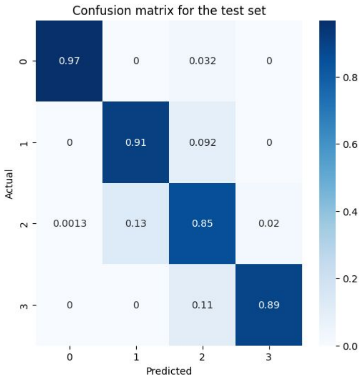
Actual \ Predicted	0	1	2	3
0	0.13	0	0.87	0
1	0	0.77	0.23	0
2	0.00066	0.0026	0.99	0.0079
3	0	0	0.14	0.86



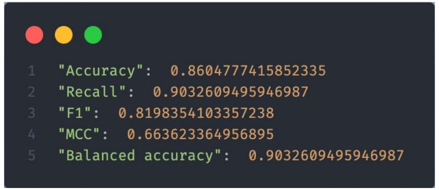
```
"Accuracy":  0.9462540716612378
"Recall":  0.6880059330578738
"F1":  0.7314654460625277
"MCC":  0.8115226264628511
"Balanced accuracy":  0.6880059330578738
```


Feature importance

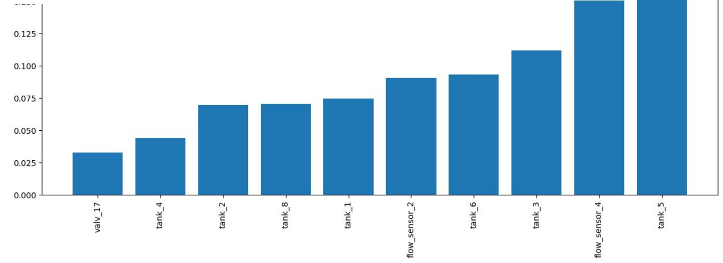
No contextual information and with balanced classes


Confusion matrix for the test set

Actual \ Predicted	0	1	2	3
0	0.97	0	0.032	0
1	0	0.91	0.092	0
2	0.0013	0.13	0.85	0.02
3	0	0	0.11	0.89

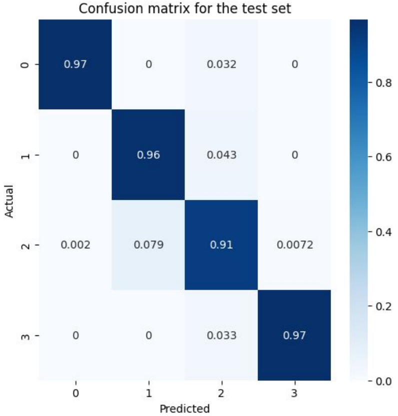


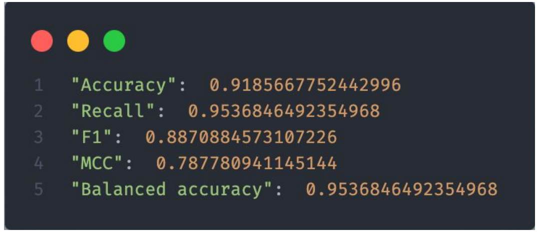
```
"Accuracy":  0.8604777415852335
"Recall":  0.9032609495946987
"F1":  0.8198354103357238
"MCC":  0.663623364956895
"Balanced accuracy":  0.9032609495946987
```


Feature importance

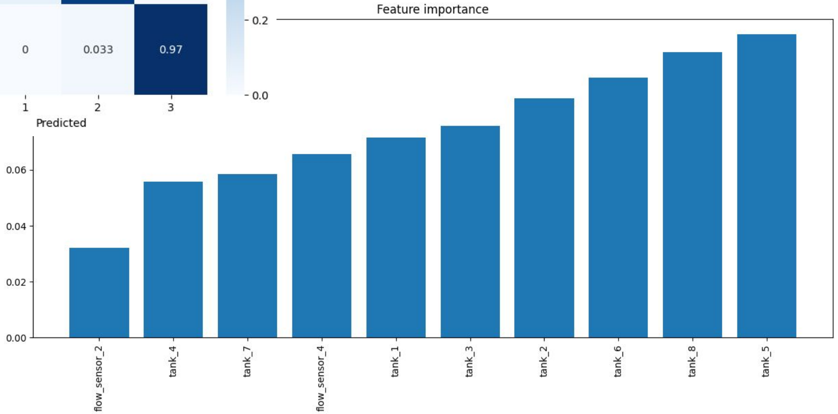
Classifiers - Random Forest

With contextual information and balanced classes


Confusion matrix for the test set

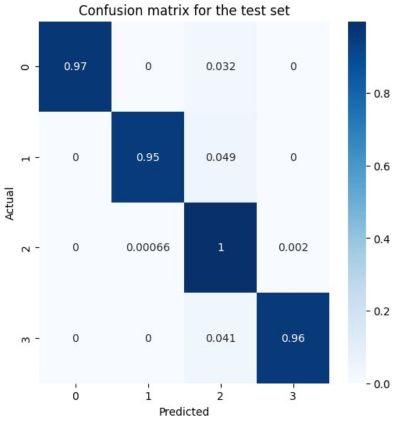


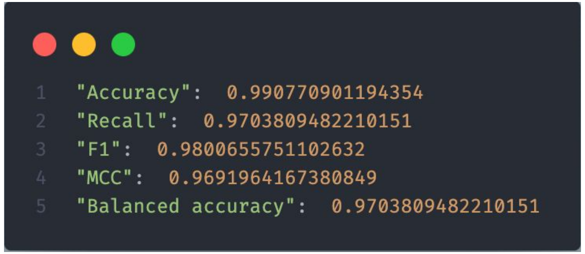
```
"Accuracy":  0.9185667752442996
"Recall":  0.9536846492354968
"F1":  0.8870884573107226
"MCC":  0.787780941145144
"Balanced accuracy":  0.9536846492354968
```


Feature importance

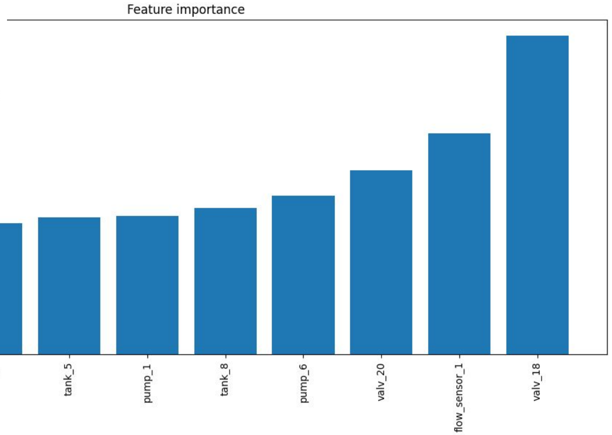
Classifiers - XGBoost

With contextual information and balanced classes


Confusion matrix for the test set



```
"Accuracy":  0.990770901194354
"Recall":  0.9703809482210151
"F1":  0.9800655751102632
"MCC":  0.9691964167380849
"Balanced accuracy":  0.9703809482210151
```


Feature importance

05

Demo

Key results & adversarial attack

06

Conclusion

Recap of results and takeaways

Conclusion

XGBoost is the best

92.3% accuracy on the network test set

Security breach

The network models would still let attacks go through if used in production, so they need to be used carefully

Adversarial Attack

We showed how easy it can be to change the data (and not the model !) and drastically confuse a model => security breach