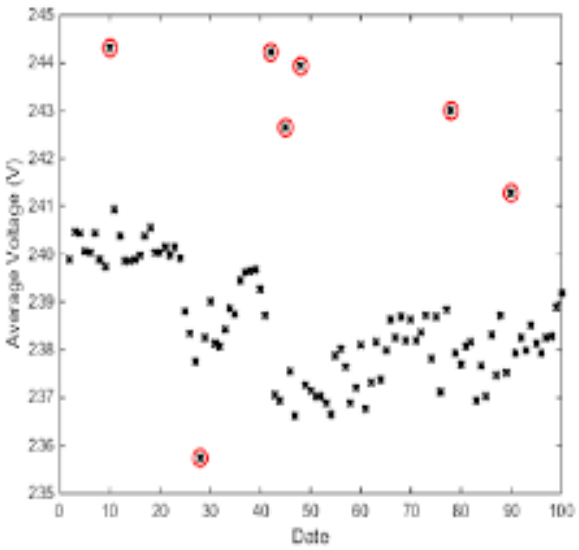
Case Study: Network Intrusion Detection by an Anomaly Detection Platform

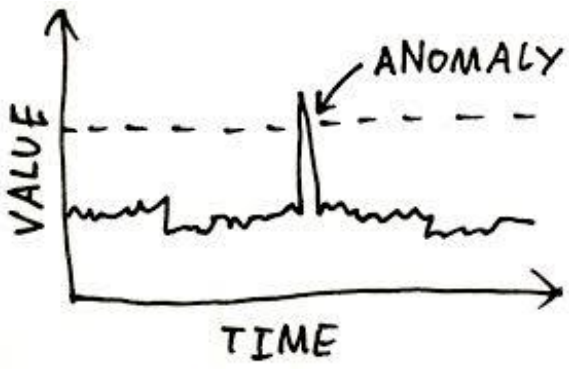
Francesco Pugliese, PhD

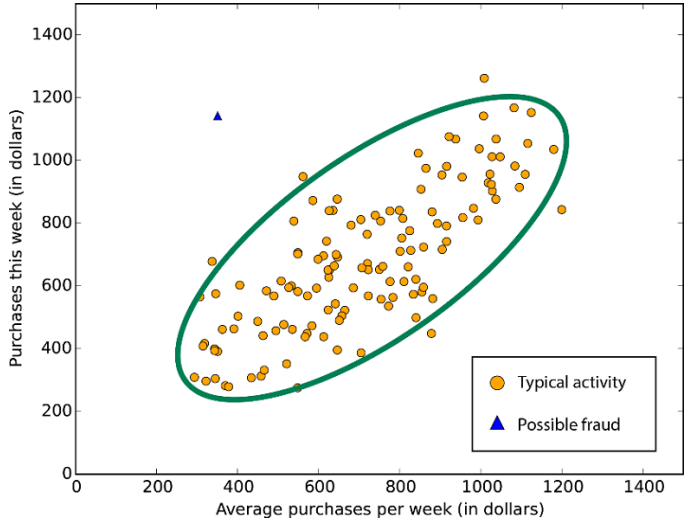
neural1977@gmail.com

What is Anomaly Detection ?

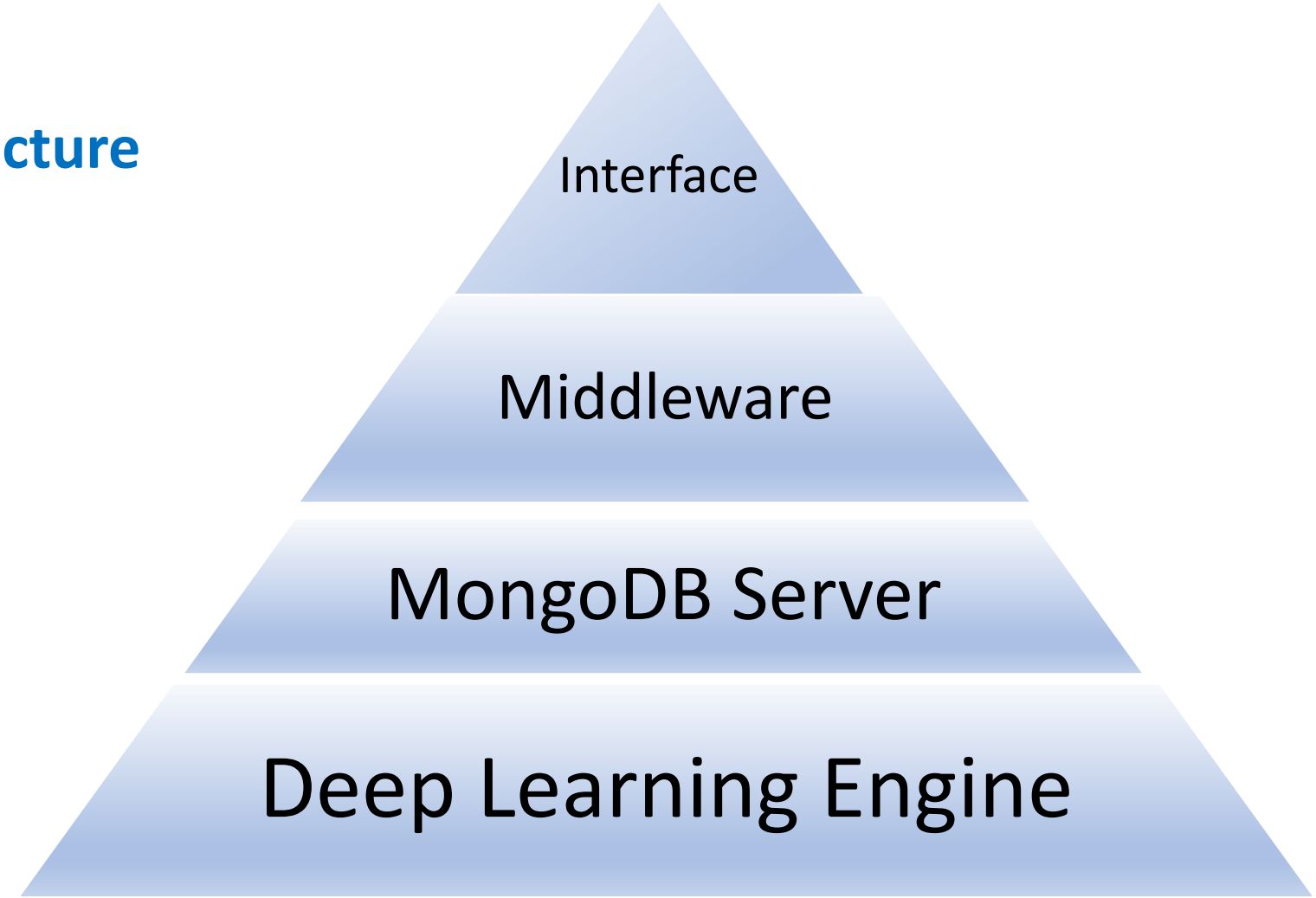
- **Anomaly Detection** (also known as **Outlier Detection**) is a discipline dealing with the detection of rare terms (events or observations) within the data, which turn out as suspicious since they appear significantly different from most of data. Tipical problems in anomaly detection are: bank frauds, manifacturing fauls, medical mistakes and intrusions in a network (intrusion detection).







Application Architecture



Principal Functions of Anomaly Detection Interface

Dataset List

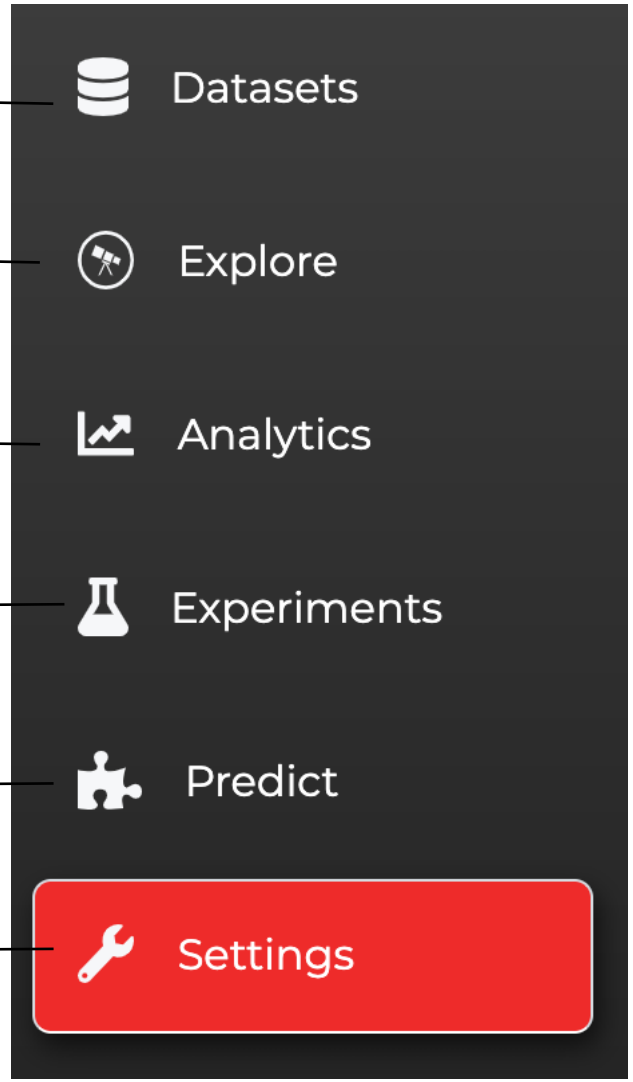
Explore and recap Datasets

Statistical Aanalysis of Data

Advanced Analysis of Datasets based on Deep Learning

Adoption of Deep Learning models trained for making predictions on real data.

Hyper-parameters configuration of Deep Learning Models.



Dataset List

- ✓ **Datasets summarization**
- ✓ It is possible to load a new dataset from the local machine using the button on the top right of the screen.
- ✓ Once, the dataset is loaded, it will be uploaded on the **MongoDB** server by means of the Middleware.
- ✓ The adoption of the **MongoDB** technology to store dataset and the system configuration files provides a huge flexibility to the platform in terms of quick analysis and scaling capability on very big datasets in order to execute heavy computations also exploiting **MapReduce** in **Pre** and **Post** processing.

Dataset

#	Name	Size	Columns	Rows
1	Id_Testset	19.47 MB	41	22544
2	nations	1.27 MB	11	5275
3	titanic	198.7 KB	15	891
4	Id_Trainset	22.2 MB	42	25192

Explore the Dataset

- ✓ In this section it is possible to get more details about the Dataset structure, like the field type (Integer, Float, Categorical), by grouping each field according to its type.

Properties

*Structure of current dataset

Name	Type	NA count
duration	Integer	0
protocol_type	Categorical	0
service	Categorical	0
flag	Categorical	0
src_bytes	Integer	0
dst_bytes	Integer	0
land	Integer	0
wrong_fragment	Integer	0

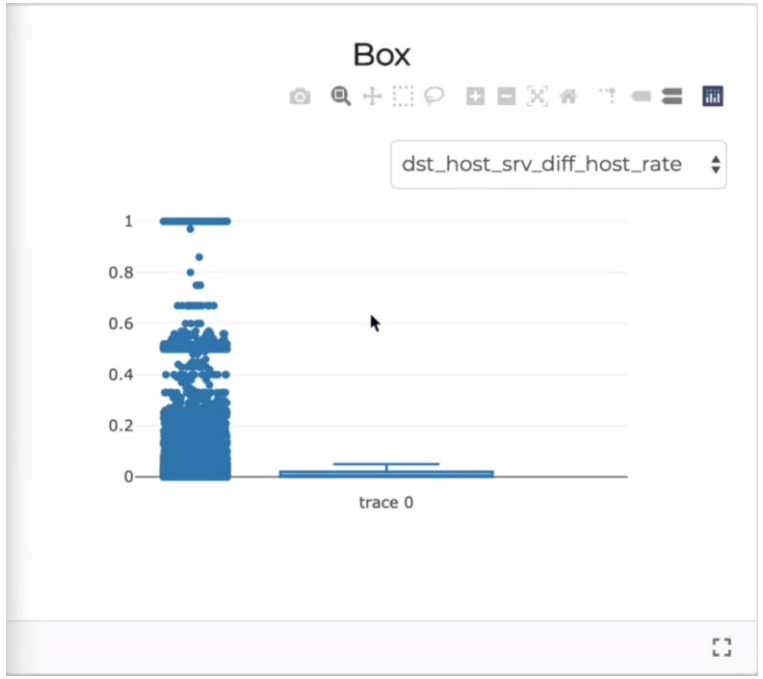
Numeric

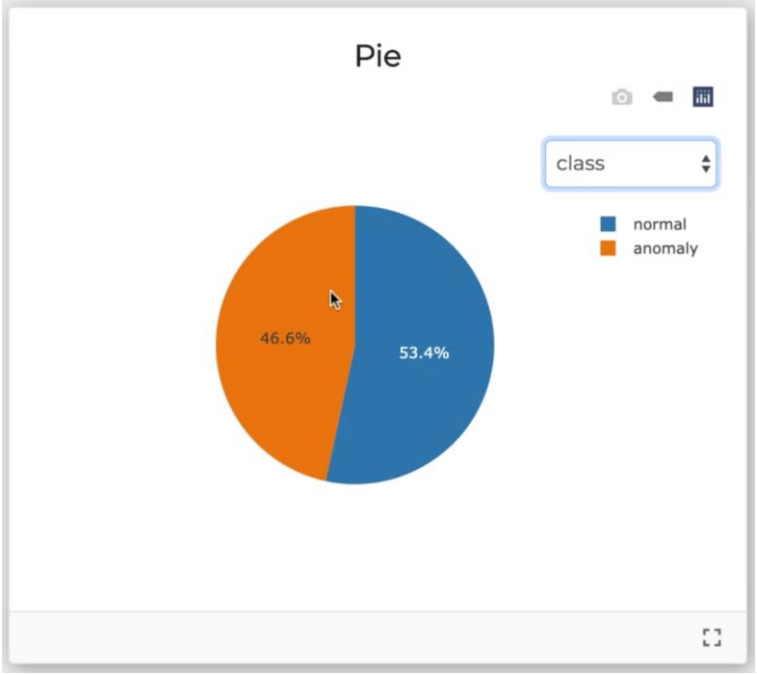
*Summary statistics of all the numeric fields of the data set

	count	mean	std	min	25%	50%	75%	max
duration	25192.0	305.1	2686.6	0.0	0.0	0.0	0.0	42862.0
src_bytes	25192.0	24330.6	2410805.4	0.0	0.0	44.0	279.0	381709090.0
dst_bytes	25192.0	3491.8	88830.7	0.0	0.0	0.0	530.2	5151385.0
land	25192.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
wrong_fragment	25192.0	0.0	0.3	0.0	0.0	0.0	0.0	3.0
urgent	25192.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
hot	25192.0	0.2	2.2	0.0	0.0	0.0	0.0	77.0
num_failed_logins	25192.0	0.0	0.0	0.0	0.0	0.0	0.0	4.0

Analysis – Dataset understanding by a base visualization

- ✓ **Detailed description** of the Dataset structure and **outlier** identification (by box plot), provides a first glance to high correlated variables (by means of heat map), categorical and numerical variables distribution (by means of Piecharts and Histograms)





Launch of Experiments by means of the best Machine Learning and Deep Learning algorithms

- ✓ Comprehension of more important **features** within the dataset and **training** of deep learning models, everything only by pressing a couple of buttons.

Metrics

*Metric values sorted by accuracy

Algoritm	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1 Score	Test F1 Score	Train ROC	Test ROC
Random Forest	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Decision tree	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Gradient Boosting	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
KNN	0.994	0.991	0.994	0.991	0.994	0.991	0.994	0.991	0.994	0.991
Logistic regression	0.955	0.956	0.955	0.956	0.955	0.956	0.955	0.956	0.954	0.956
SGD	0.914	0.934	0.921	0.936	0.914	0.934	0.914	0.934	0.918	0.936
Gaussian Naive Bayes	0.895	0.890	0.896	0.890	0.895	0.890	0.895	0.890	0.896	0.889

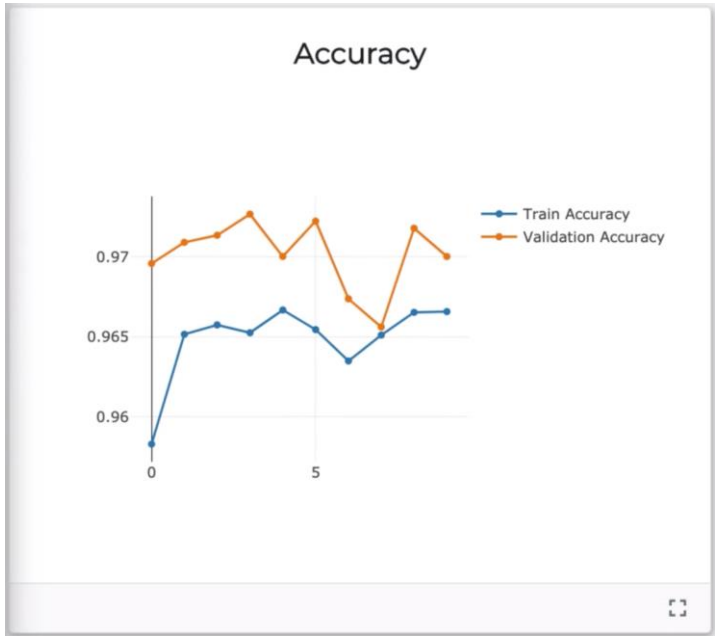
Importance

*Features sorted by their importance

Feature	Trainset score	Testset score
src_bytes	0.2	0.3
flag	0.1	0.2
dst_host_same_srv_rate	0.1	0.1
count	0.0	0.1
logged_in	0.0	0.1
dst_bytes	0.1	0.0
dst_host_same_src_port_rate	0.0	0.0
protocol_type	0.0	0.0

Experiments with the best algorithms for Machine Learning

Comprehend how Machine Learning and Deep Learning models are **performing** on the dataset, by means of the visualization of **training and test curves** and the visualization of related **metrics**.



History

*Results of model training

epoch	acc	loss	val_acc	val_loss
0	0.958	0.110	0.970	0.068
1	0.965	0.082	0.971	0.072
2	0.966	0.080	0.971	0.063
3	0.965	0.076	0.973	0.063
4	0.967	0.071	0.970	0.058
5	0.965	0.068	0.972	0.057
6	0.963	0.069	0.967	0.058
7	0.965	0.066	0.966	0.059

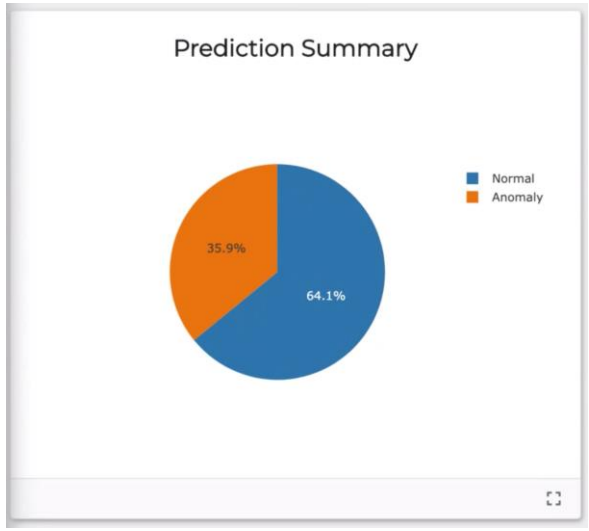
Dl_metrics

*Metrics of the model

Metric Name	Train set	Validation set
Accuracy	0.970	0.970
Precision	0.971	0.969
Recall	0.970	0.967
F1 score	0.970	0.967
ROC score	0.971	0.969

Prediction: Infer on new real data in a simple and quick way using trained models which arise as the best during the training stage.

This performs predictions over the data with only one click, by harnessing deep learning models.



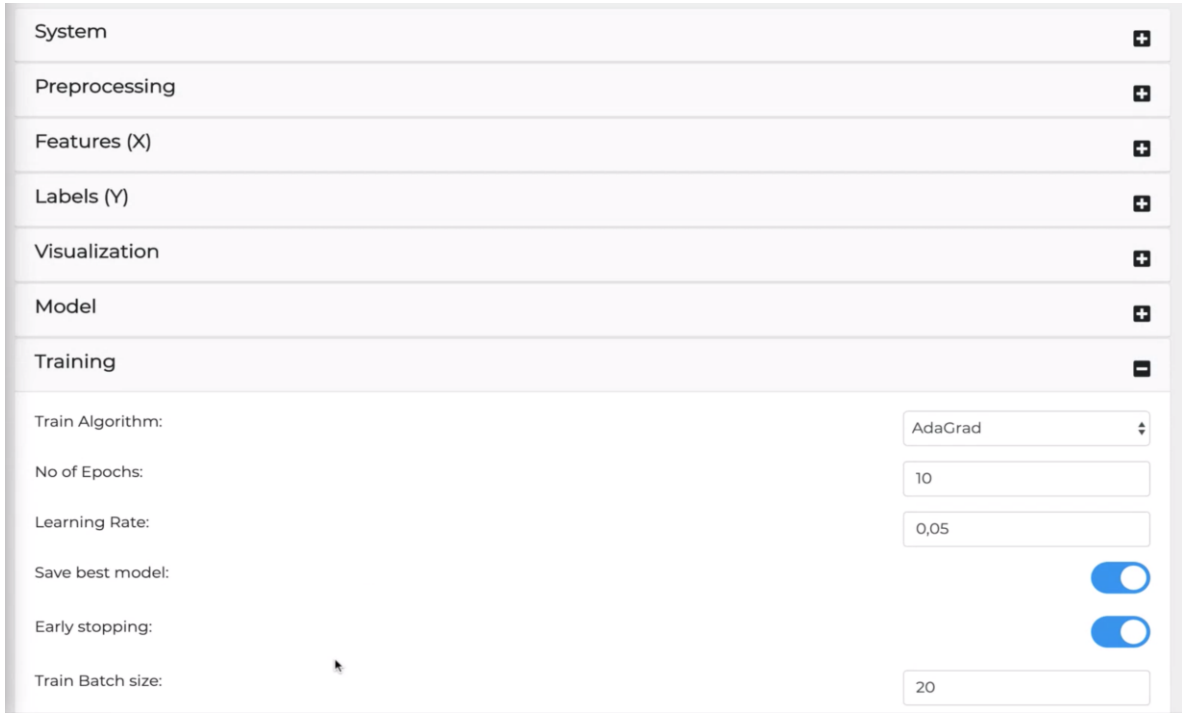
Prediction

*Predictions of DL model on new data

duration	protocol_type	service	flag	src_bytes	dst_bytes	prediction
0	tcp	private	REJ	0	0	Anomaly
0	tcp	private	REJ	0	0	Anomaly
2	tcp	ftp_data	SF	12983	0	Normal
0	icmp	eco_i	SF	20	0	Anomaly
1	tcp	telnet	RSTO	0	15	Normal
0	tcp	http	SF	267	14515	Normal
0	tcp	smtp	SF	1022	387	Normal
0	tcp	telnet	SF	129	174	Normal

Quick and Simple Machine Learning Configuration

- ✓ Configure hyperparameters of the machine learning model which is simple like filling a form.



Main characteristics of the Platform

- ✓ Use of the simple **interface** which is accessible via web from any device, the application adapts to different screen resolutions such as PC, tablets e smartphones.
- ✓ It is possible to have a quick look at pre-loaded datasets with all the analysis settings.
- ✓ Dedicated platform for the **anomaly detection**. This allows to get improvements in the field of **decision making** since the platform is continuously updated with new datasets and models.
- ✓ There are not technological **barriers** in the use of the platform since, within the design stage, we gave a special regard to **Usability**, more than something else.
- ✓ This provides all the **power** of deep learning with some clicks on the interface.
- ✓ Explorations and visualizations which quickly **interpretable**.

Network Intrusion Detection

- **Dataset Description:** The Dataset is made of a wide variety of **simulated intrusions** within a military network. The **U.S. Air Force** created an environment to acquire raw TCP/IP data simulating them in a **LAN (Local Atea Network)**. The **LAN** was configured as an environment very similar to a **real** network, which is flooded by **intrusion attacks**.
- Every single datum is a **connection**, namely a sequence of **TCP** packets which have a begin and an end, and so they have a **specific** duration. During this connection, packets are transmitted from a source **IP (Internet Protocolo Address)** to a destination IP by means of a specific **communication protocol**.
- Each connection is labeled as normal or anomalous, meaning that there was an **intrusion cyber-attack** and so it speciefies the type of attack.
- Each **TCP/IP** connection is made of **100 bytes** of **transmission,** and for each of them there are **41 quantitative** and **categorical variables**.
- Classifcation variables have two only possible values: **Normal and Anomaly**.

Francesco **Pugliese**