SAPIENZA
UNIVERSITÀ DI ROMA

ANOMA: AUTO ML PLATFORM FOR ANOMALY DETECTION

Name: Gunturi Vamsi Krishna Varma
Metricola ID: 1794653
Advisor: Prof. Filomena Maggino
Co-advisor: Dr. Francesco Pugliese

Outline

Anomaly detection

Automated Machine Learning

ANOMA Platform

Datasets

Results

Summary

Future work

Anomaly detection

What is Anomaly Detection?

- Anomaly detection [3] refers to the problem of finding **patterns** in data that do not conform to expected behavior.
- Anomalies could often **influence** the data analysis, which could be the cause of drawing wrong conclusions which can influence important business decisions.
- Thus, in every data analysis, it is of pristine importance to accurately define anomalous **behaviour** pertaining to a certain domain and there by apply appropriate anomaly detection model.

Use-cases of Anomaly Detection

- Fraud detection
- Intrusion detection
- Predictive analytics
- Anomaly detection in Social Networks
- Log Anomaly Detection
- Anomaly Detection in Time Series
- Video Surveillance
- Face recognition
- Satellite imagery

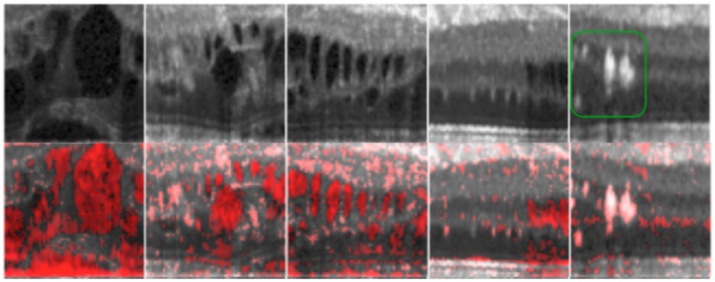
Deep learning for Anomaly Detection

In recent years, deep learning-based anomaly detection algorithms have become increasingly popular and have been applied for a **diverse** set of tasks as illustrated below

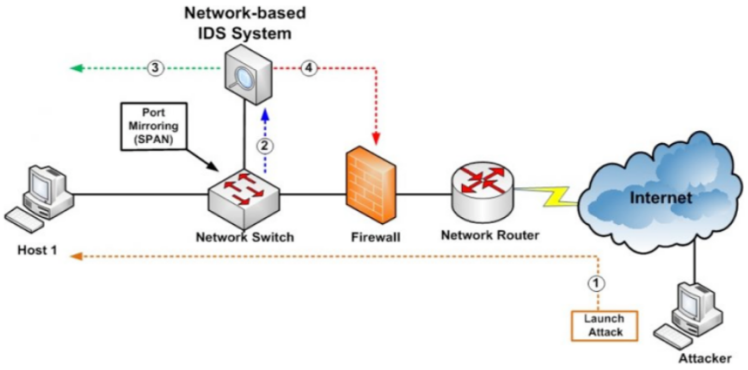
Deep learning for Anomaly Detection

In recent years, deep learning-based anomaly detection algorithms have become increasingly popular and have been applied for a **diverse** set of tasks as illustrated below

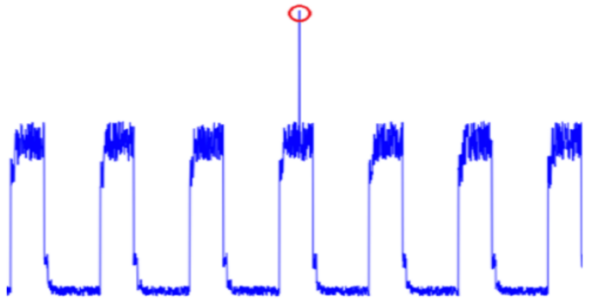
(a) Illegal Traffic Flow detection



(b) Detecting Retinal Damage



(c) Cyber-Network Intrusion detection



(d) Internet Of Things (IoT) Big-Data
Anomaly detection

Why use deep learning for Anomaly detection?

Below are some of the motivations for Deep anomaly detection (DAD) techniques [2],

Why use deep learning for Anomaly detection?

Below are some of the motivations for Deep anomaly detection (DAD) techniques [2],

- Application of Anomaly detection for **image** (e.g. medical images) and **sequence** datasets.

Why use deep learning for Anomaly detection?

Below are some of the motivations for Deep anomaly detection (DAD) techniques [2],

- Application of Anomaly detection for **image** (e.g. medical images) and **sequence** datasets.
- Need for **large-scale** anomaly detection.

Why use deep learning for Anomaly detection?

Below are some of the motivations for Deep anomaly detection (DAD) techniques [2],

- Application of Anomaly detection for **image** (e.g. medical images) and **sequence** datasets.
- Need for **large-scale** anomaly detection.
- **Automatic** feature learning capability eliminates the need of developing manual features by **domain** experts.

Why use deep learning for Anomaly detection?

Below are some of the motivations for Deep anomaly detection (DAD) techniques [2],

- Application of Anomaly detection for **image** (e.g. medical images) and **sequence** datasets.
- Need for **large-scale** anomaly detection.
- **Automatic** feature learning capability eliminates the need of developing manual features by **domain** experts.
- Defining well defined **boundary** between normal and anomalous behavior.

Automated Machine Learning

What is Auto ML and Why it is important ?

What is Auto ML and Why it is important ?

- Automated Machine Learning (AutoML) [4] is the process of automating **end-to-end** process of applying Machine Learning (ML)

What is Auto ML and Why it is important ?

- Automated Machine Learning (AutoML) [4] is the process of automating **end-to-end** process of applying Machine Learning (ML)
- AutoML covers the complete **pipeline** from the raw dataset to the deployable Machine Learning model

What is Auto ML and Why it is important ?

- Automated Machine Learning (AutoML) [4] is the process of automating **end-to-end** process of applying Machine Learning (ML)
- AutoML covers the complete **pipeline** from the raw dataset to the deployable Machine Learning model
- Allows **non-experts** to make use of Machine Learning models and techniques without requiring to become an expert in this field first

What is Auto ML and Why it is important ?

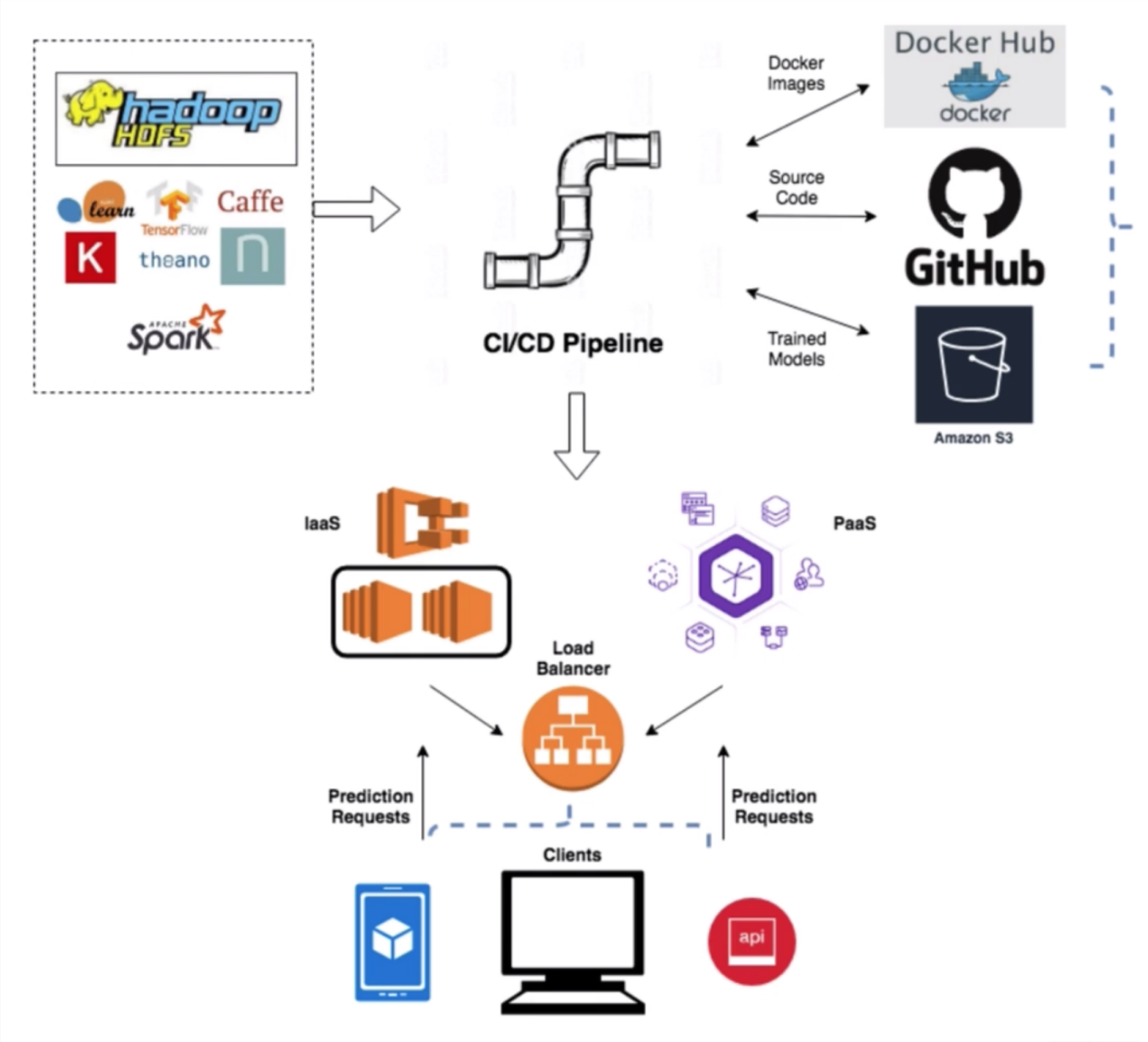
- Automated Machine Learning (AutoML) [4] is the process of automating **end-to-end** process of applying Machine Learning (ML)
- AutoML covers the complete **pipeline** from the raw dataset to the deployable Machine Learning model
- Allows **non-experts** to make use of Machine Learning models and techniques without requiring to become an expert in this field first
- As the amount of data and complexity of business problems increase, more Machine Learning systems need to be deployed and hence AutoML is applied to increase the **productivity** of data science teams.

Auto ML tools

Following are some of the popular Auto ML tools and packages,

- Auto-Sklearn
- TPOT
- H2O AutoML
- Auto Keras
- Uber Ludwig
- Rapid Miner
- Weka

Auto ML components



ANOMA Platform

Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset

Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset
- Simple and easy to use interface which is **responsive** to any screen resolution like PC, tablet and Mobile

Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset
- Simple and easy to use interface which is **responsive** to any screen resolution like PC, tablet and Mobile
- **Automate** the process of data exploration, exploratory data analysis

Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset
- Simple and easy to use interface which is **responsive** to any screen resolution like PC, tablet and Mobile
- **Automate** the process of data exploration, exploratory data analysis
- **Adaptable** platform to a broader audience with zero to limited technological barriers

Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset
- Simple and easy to use interface which is **responsive** to any screen resolution like PC, tablet and Mobile
- **Automate** the process of data exploration, exploratory data analysis
- **Adaptable** platform to a broader audience with zero to limited technological barriers
- **Full** fledged machine learning workflow and data pipeline

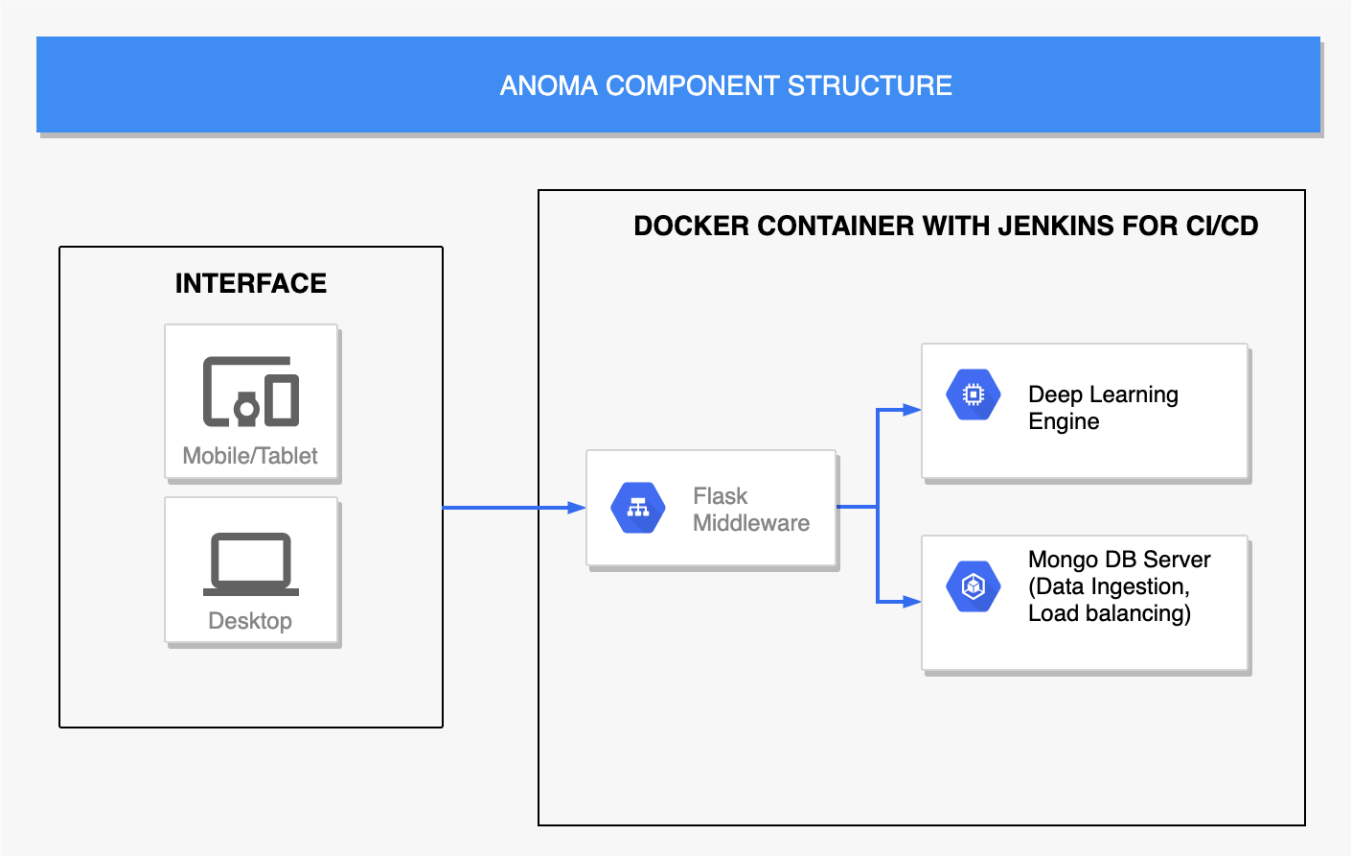
Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset
- Simple and easy to use interface which is **responsive** to any screen resolution like PC, tablet and Mobile
- **Automate** the process of data exploration, exploratory data analysis
- **Adaptable** platform to a broader audience with zero to limited technological barriers
- **Full** fledged machine learning workflow and data pipeline
- Power of **deep learning** with just a few button clicks

Main goals of ANOMA

- Build a **general-purpose** automated machine learning (Auto ML) engine for anomaly detection for a given dataset
- Simple and easy to use interface which is **responsive** to any screen resolution like PC, tablet and Mobile
- **Automate** the process of data exploration, exploratory data analysis
- **Adaptable** platform to a broader audience with zero to limited technological barriers
- **Full** fledged machine learning workflow and data pipeline
- Power of **deep learning** with just a few button clicks
- **Scalable** and loosely coupled architecture able to tackle similar usecases in future

ANOMA Components



ANOMA Technical stack

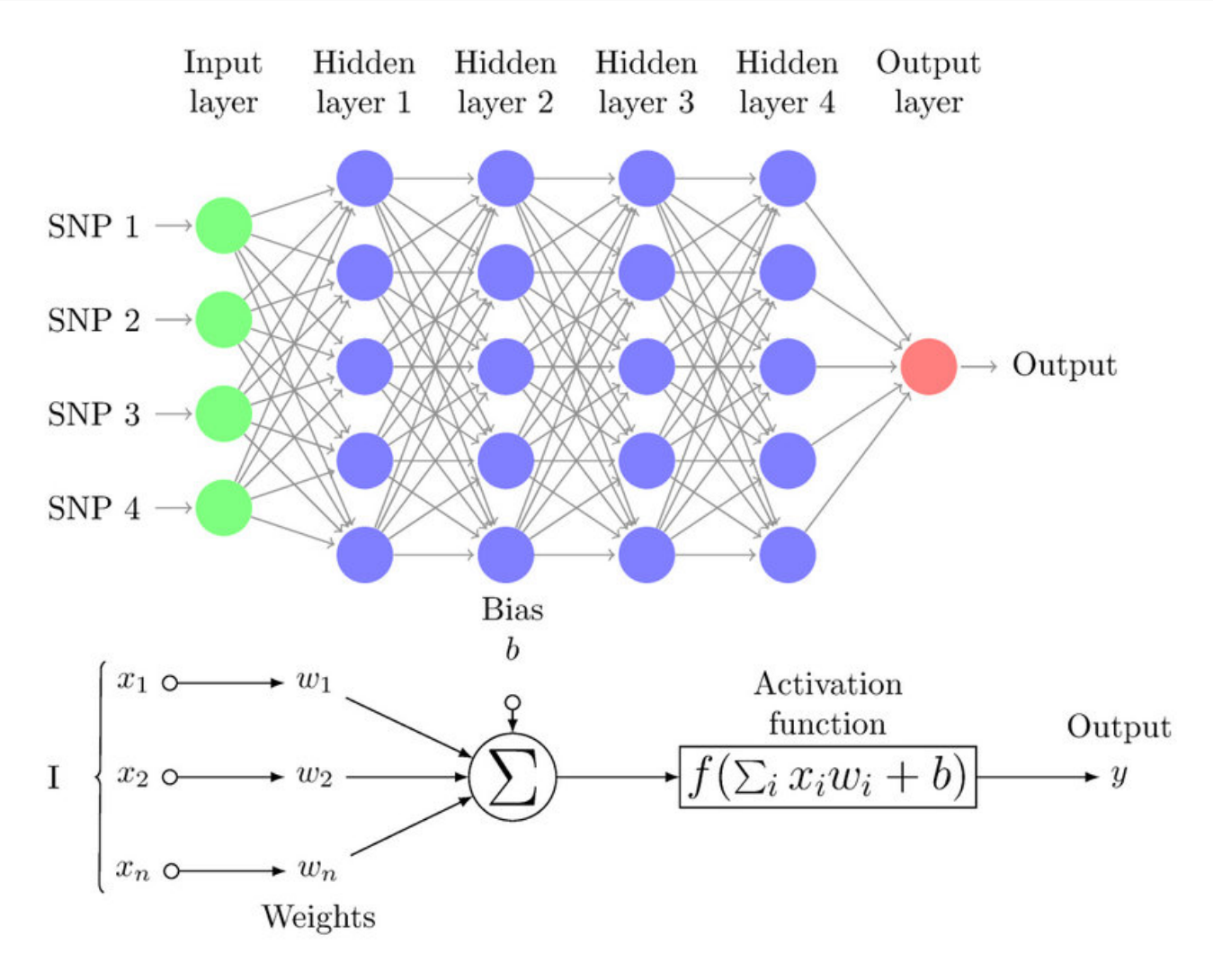
Machine Learning Step	Tool
Data pre-processing	Numpy, Pandas, Dask
Feature Engineering	Pandas, Dask, Scikit-learn
Exploratory data analysis	Matplotlib, Seaborn, Plotly
Machine learning	Scikit-learn
Deep learning	Tensorflow, Keras
Middleware	Flask
Database	MongoDB
CI/CD	Git, Jenkins
Deployment	Docker, Azure
Code minification	Gulp
Front-end	Boostrap, D3.js, Vue.js

ANOMA DL Models

Following are Deep Learning (DL) architecture models we used for our platform:

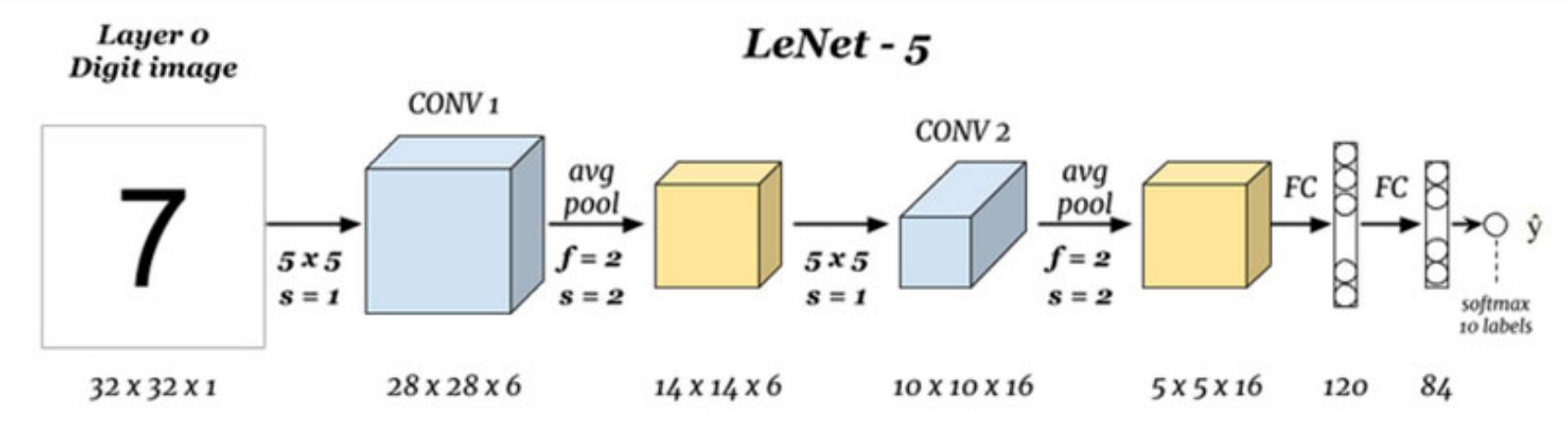
- Complex Multi-layer perceptron (MLP)
- Lenet5
- Alexnet
- VGG 16

Complex MLP



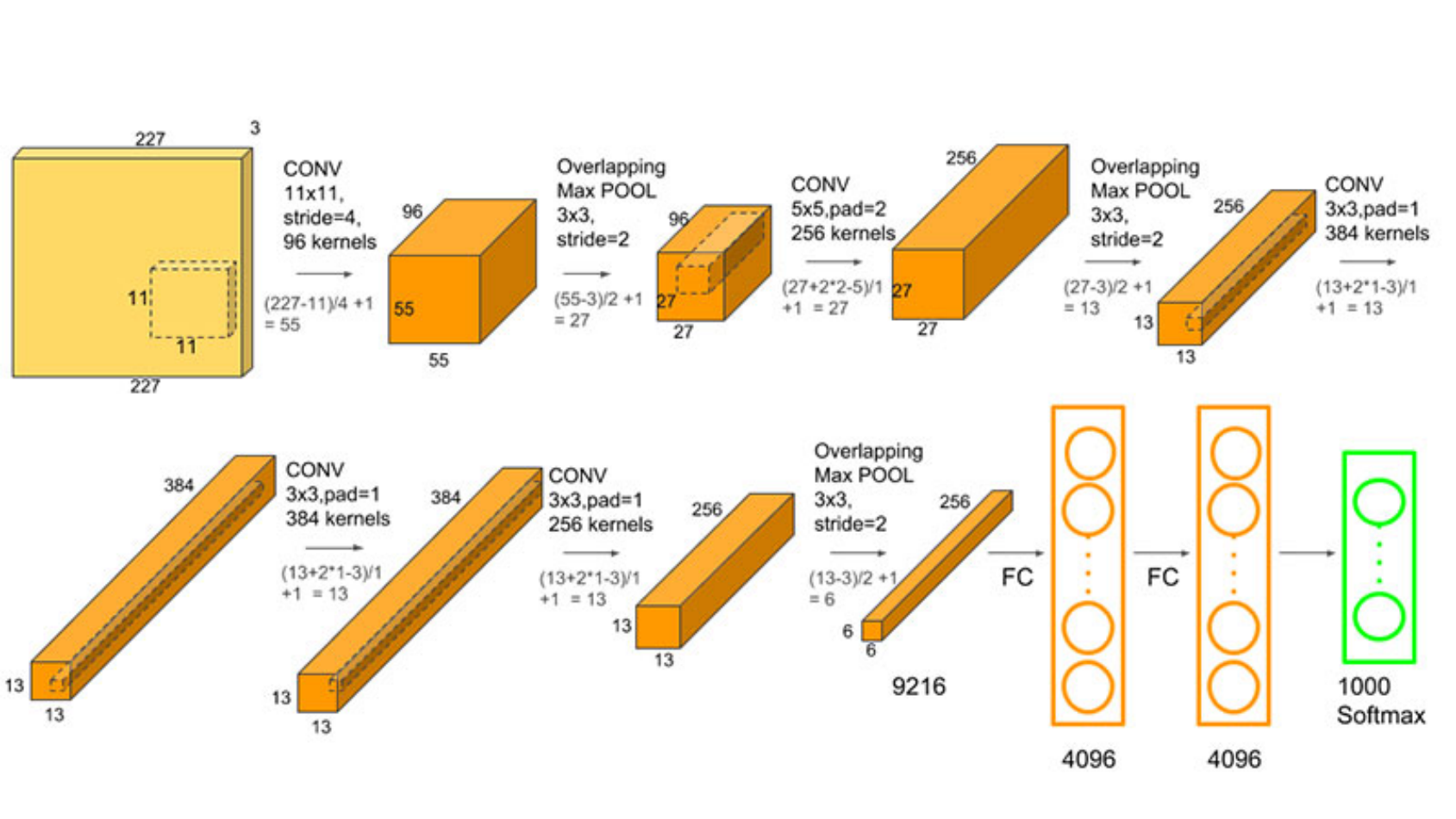
MLP architecture

Lenet 5



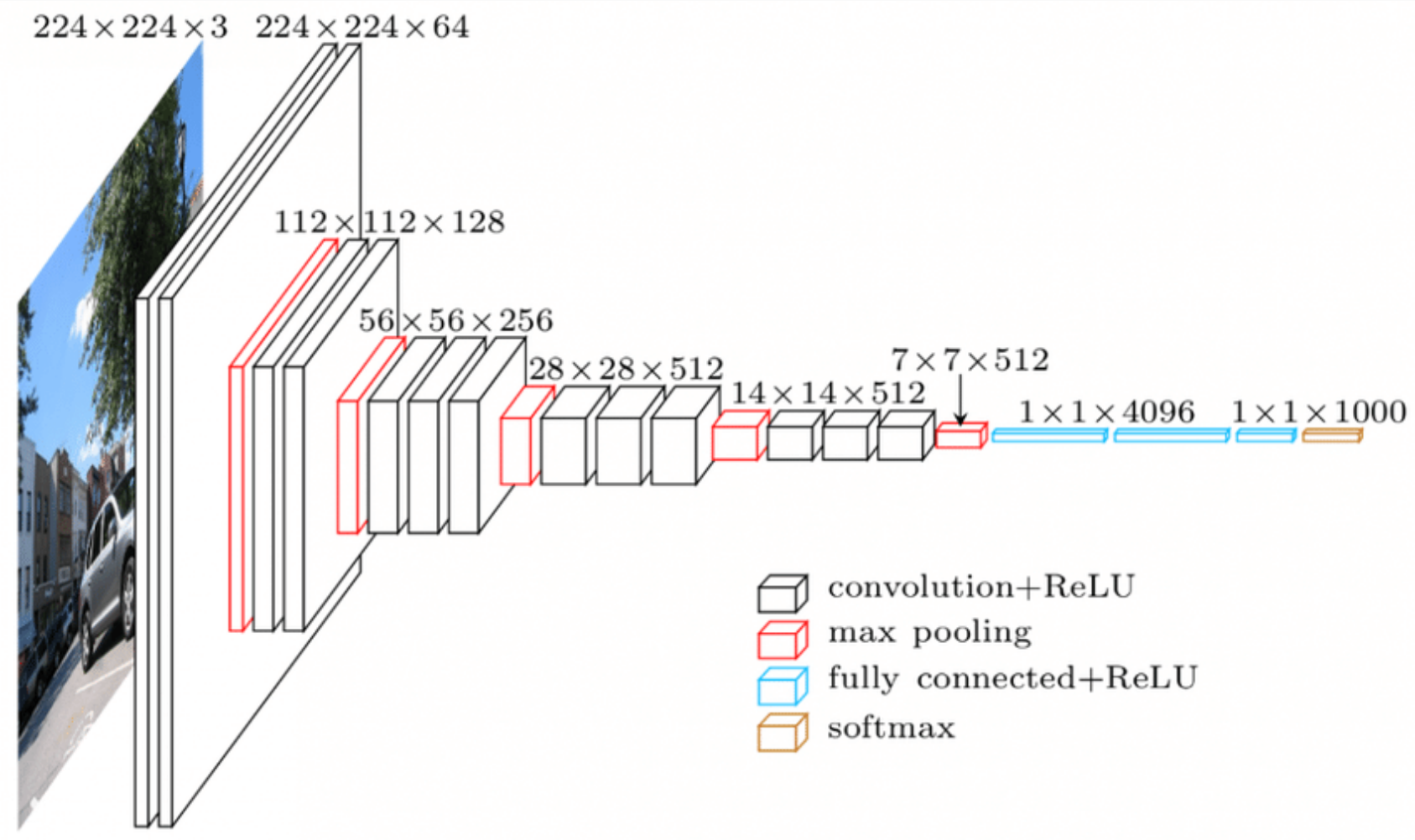
LeNet-5 architecture

Alexnet (under research)



AlexNet architecture

VGG 16 (under research)



VGG 16 architecture

Datasets

Network intrusion detection dataset

- The dataset [1] consists of a wide variety of intrusions simulated in a military network environment of a typical US Air Force LAN.
- The LAN was focused like a real environment and blasted with multiple attacks.
- For each connection, 41 quantitative and qualitative features are obtained from normal and attack data (3 qualitative and 38 quantitative features).
- Each connection is labelled as either normal(0) or as an attack(1). So, this is a **binary classification dataset**

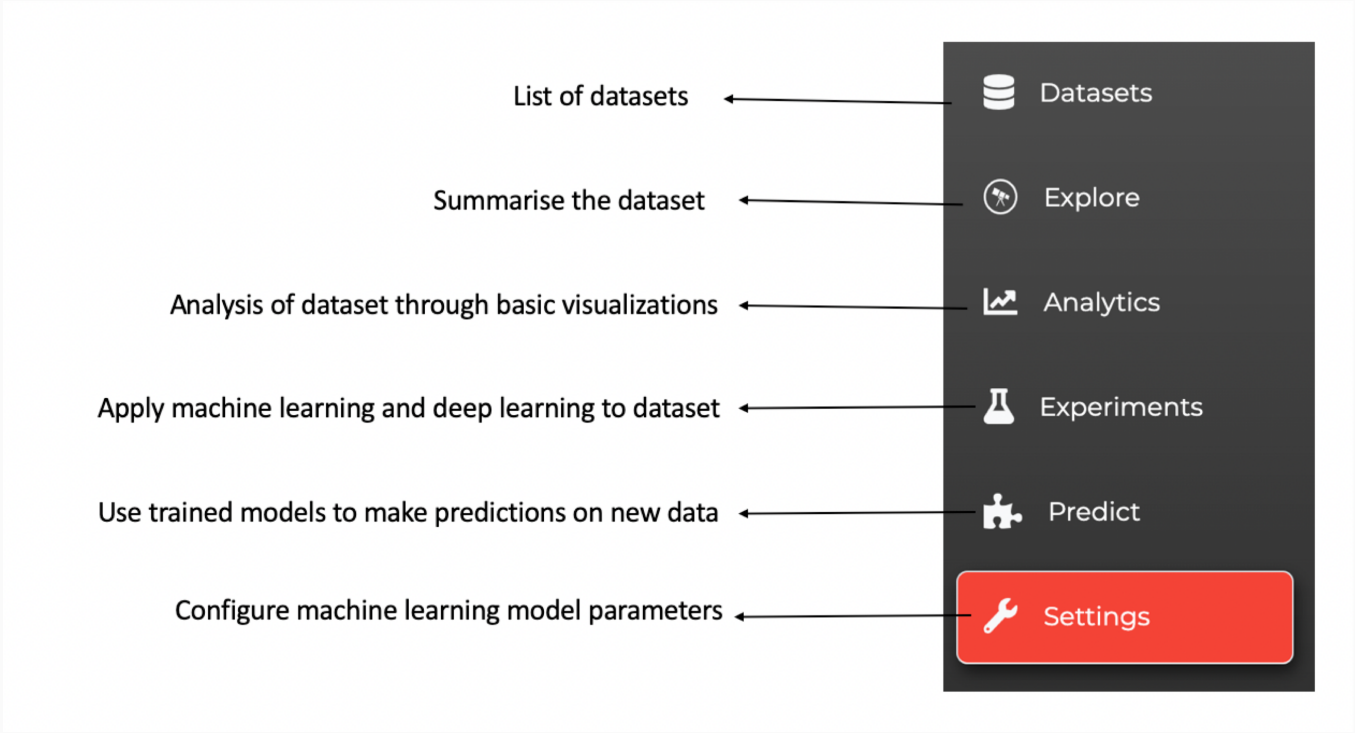
KDD Cup 1999 dataset

- KDD dataset [5] is an elaborated version of Network intrusion dataset.
- The dataset contains a total of 24 training attack types. So, this is a **multi-class classification dataset**.
- Attacks fall into four main categories:
  - **DOS**: denial-of-service, e.g. syn flood
  - **R2L**: unauthorized access from a remote machine, e.g. guessing password
  - **U2R**: unauthorized access to local superuser (root) privileges, e.g. various 'buffer overflow' attacks
  - **Probing**: surveillance and other probing, e.g. port scanning.

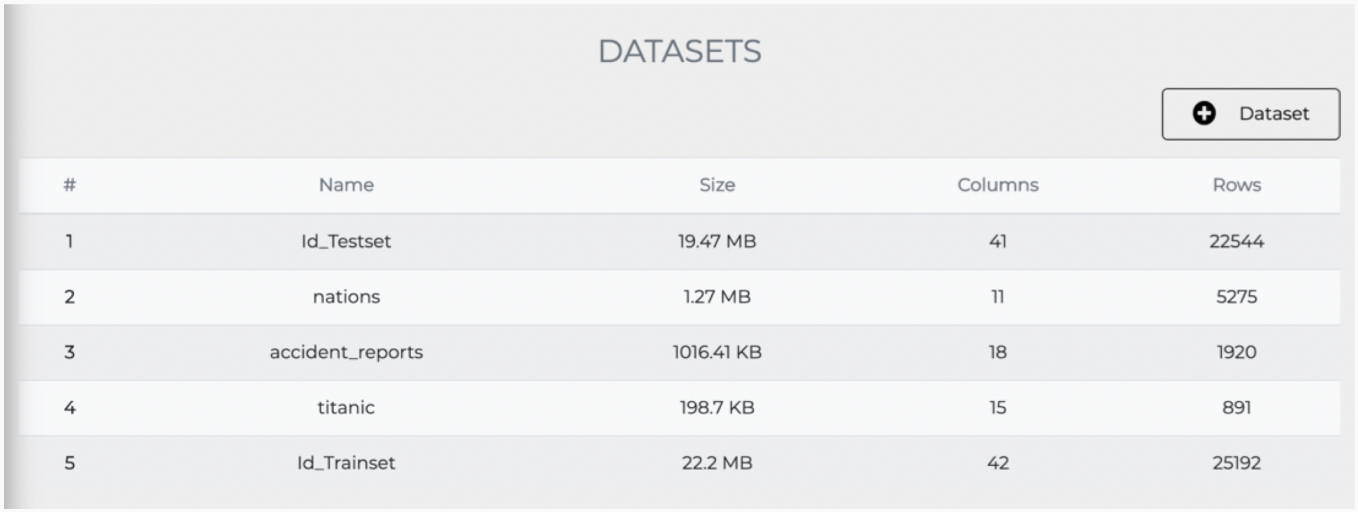
Results

Sections

Interface has following sections:



List of data sets



Data Exploration

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	5000.0	330.5	2828.7	0.0	0.0	0.0	0.0	41802.0
src_bytes	5000.0	86005.6	5400370.1	0.0	0.0	44.0	280.0	381709090.0
dst_bytes	5000.0	1968.6	24702.8	0.0	0.0	0.0	538.2	1639484.0
land	5000.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
wrong_fragment	5000.0	0.0	0.3	0.0	0.0	0.0	0.0	3.0
urgent	5000.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
hot	5000.0	0.2	2.0	0.0	0.0	0.0	0.0	30.0
num_failed_logins	5000.0	0.0	0.0	0.0	0.0	0.0	0.0	2.0

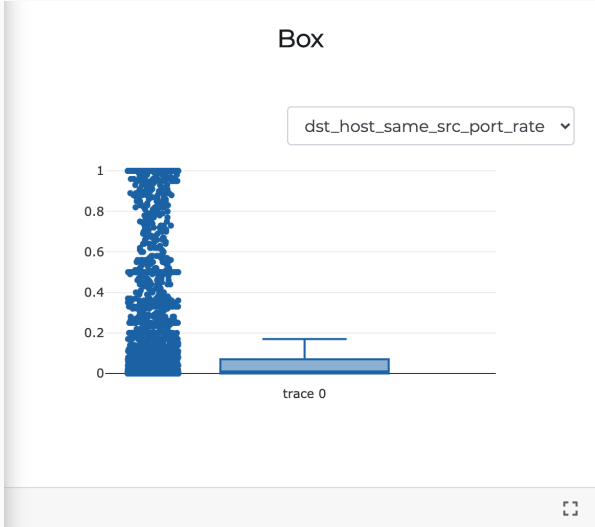
Categorical

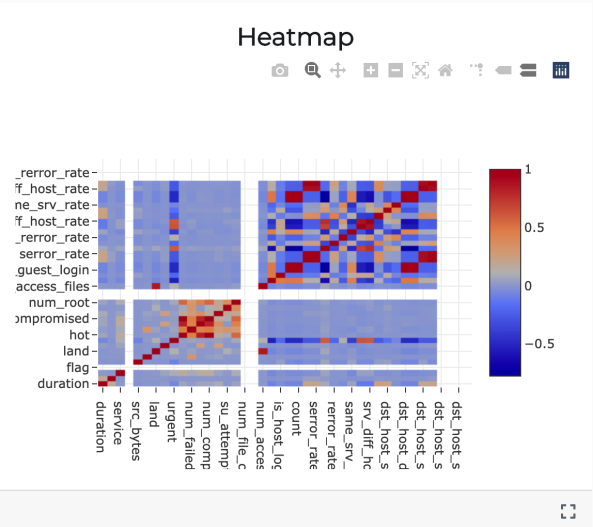
*Summary statistics of all the categorical fields of the data set

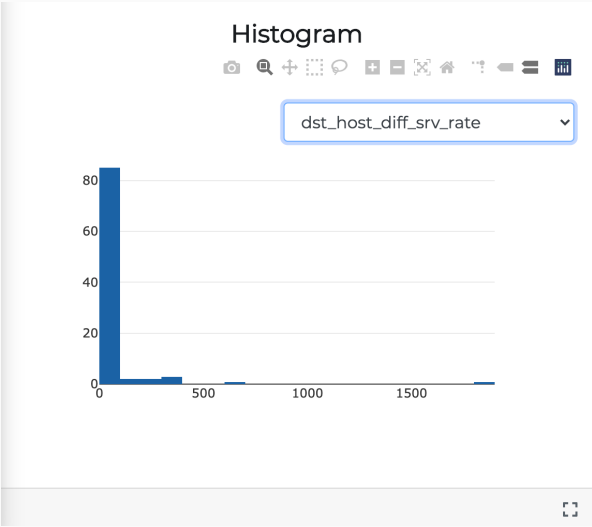
	count	unique	top	freq
protocol_type	5000	3	tcp	4080
service	5000	64	http	1564
flag	5000	11	SF	2964
class	5000	2	normal	2648

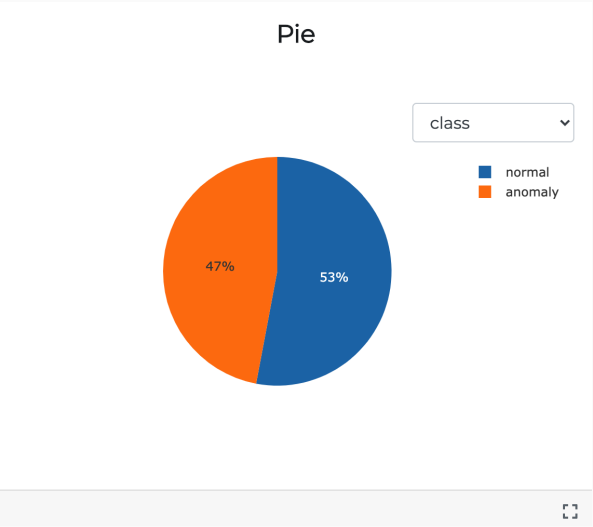
Exploring data set

Analytics









Analytics of data set

Experiment - Machine Learning

Importance

*Features sorted by their importance

Feature	Trainset score	Testset score
src_bytes	0.239	0.275
dst_host_same_srv_rate	0.018	0.116
dst_bytes	0.036	0.089
dst_host_srv_count	0.062	0.078
flag	0.067	0.070
diff_srv_rate	0.056	0.066
same_srv_rate	0.127	0.064
protocol_type	0.030	0.048

Feature importance sorted

Experiment - Machine Learning

Importance

*Features sorted by their importance

Feature	Trainset score	Testset score
src_bytes	0.239	0.275
dst_host_same_srv_rate	0.018	0.116
dst_bytes	0.036	0.089
dst_host_srv_count	0.062	0.078
flag	0.067	0.070
diff_srv_rate	0.056	0.066
same_srv_rate	0.127	0.064
protocol_type	0.030	0.048

Feature importance sorted

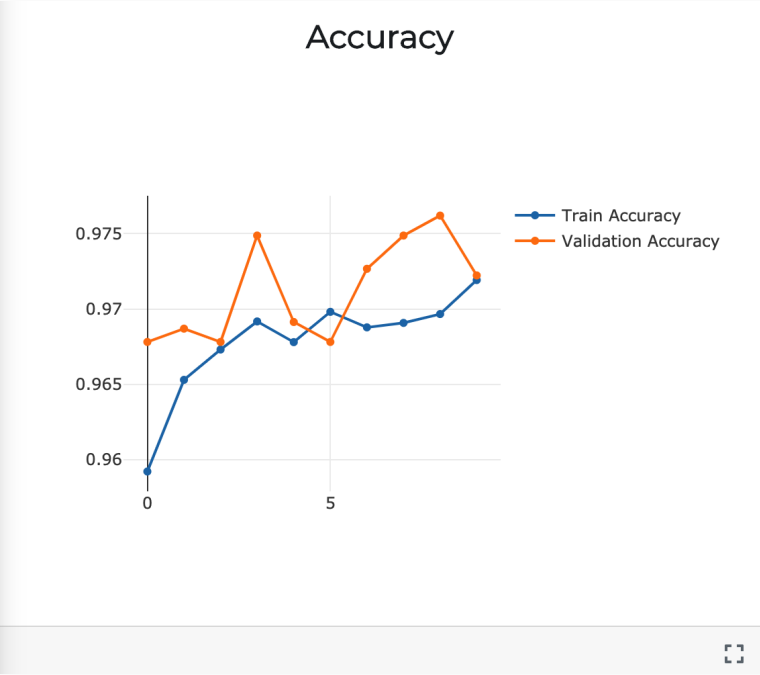
Metrics

*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	0.990	0.981	0.990	0.981	0.990	0.981	0.990	0.981	0.990	0.980
Logistic regression	0.961	0.958	0.961	0.958	0.961	0.958	0.961	0.958	0.960	0.958
SGD	0.921	0.804	0.925	0.853	0.921	0.804	0.921	0.799	0.924	0.811
Gaussian Naive Bayes	0.890	0.901	0.890	0.903	0.890	0.901	0.890	0.901	0.889	0.902

Metrics for training and validation sets

Experiment - Deep Learning



Visualization of Accuracy and Loss

Experiment - Deep Learning

History

*Results of model training

epoch	acc	loss	val_acc	val_loss
0	0.959	0.108	0.968	0.069
1	0.965	0.081	0.969	0.068
2	0.967	0.075	0.968	0.063
3	0.969	0.073	0.975	0.067
4	0.968	0.071	0.969	0.077
5	0.970	0.069	0.968	0.080
6	0.969	0.067	0.973	0.059
7	0.969	0.064	0.975	0.055

Training history

Experiment - Deep Learning

History

*Results of model training

epoch	acc	loss	val_acc	val_loss
0	0.959	0.108	0.968	0.069
1	0.965	0.081	0.969	0.068
2	0.967	0.075	0.968	0.063
3	0.969	0.073	0.975	0.067
4	0.968	0.071	0.969	0.077
5	0.970	0.069	0.968	0.080
6	0.969	0.067	0.973	0.059
7	0.969	0.064	0.975	0.055

Training history

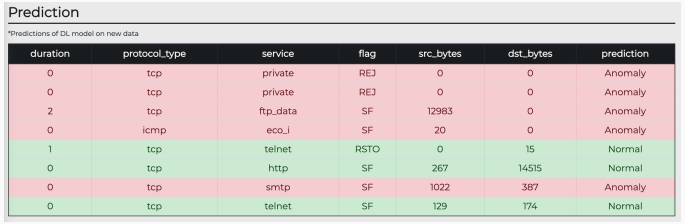
Dl_metrics

*Metrics of the model

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

Metrics

Predict



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	Anomaly
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	Anomaly
0	tcp	telnet	SF	129	174	Normal

Prediction on new data

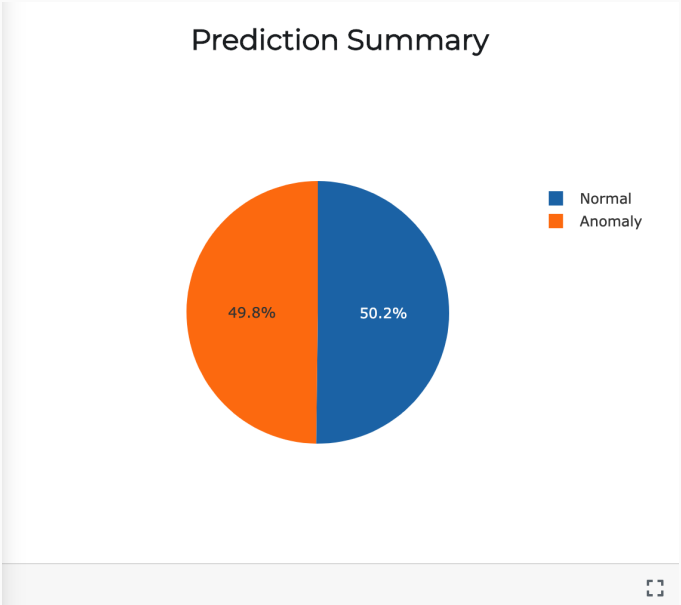
Predict

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	Anomaly
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	Anomaly
0	tcp	telnet	SF	129	174	Normal

Prediction on new data

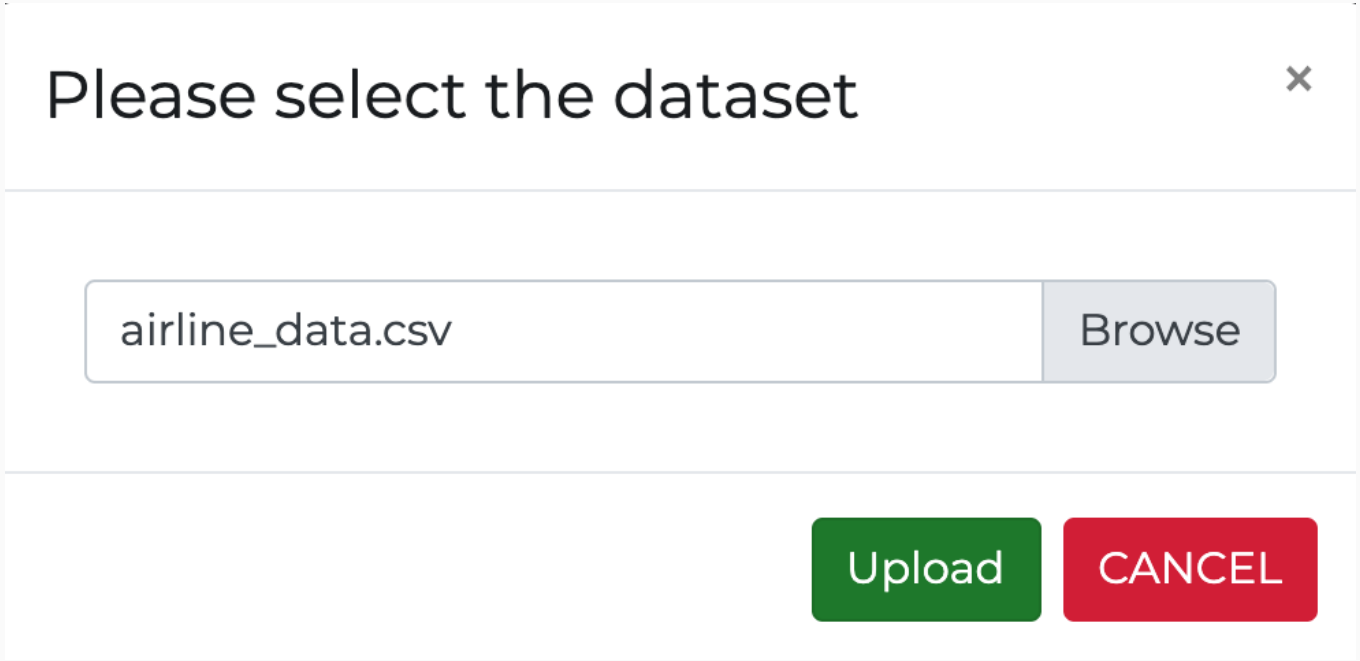


Prediction summary

Settings


System
Preprocessing
Features (X)
Labels (Y)
Visualization
Model
Training
Testing
Save Settings

Adding new data sets



ANOMA Demo

- ANOMA is deployed on Microsoft Azure and can be accessed by clicking on ▸ Link
- **Note**: I am using a student trail version of Azure, so it can be slow.

Summary

Summary of our research

Summary of our research

- **Dedicated** platform for the area of anomaly detection

Summary of our research

- **Dedicated** platform for the area of anomaly detection
- Simple and easy to use interface **accessible** from any device.

Summary of our research

- **Dedicated** platform for the area of anomaly detection
- Simple and easy to use interface **accessible** from any device.
- Ability to quickly get a **glance** of your data sets with dedicated sections for analysis and data exploration

Summary of our research

- **Dedicated** platform for the area of anomaly detection
- Simple and easy to use interface **accessible** from any device.
- Ability to quickly get a **glance** of your data sets with dedicated sections for analysis and data exploration
- Less technological **barrier** in using the platform

Summary of our research

- **Dedicated** platform for the area of anomaly detection
- Simple and easy to use interface **accessible** from any device.
- Ability to quickly get a **glance** of your data sets with dedicated sections for analysis and data exploration
- Less technological **barrier** in using the platform
- Power of deep learning with just a **few** button clicks

Summary of our research

- **Dedicated** platform for the area of anomaly detection
- Simple and easy to use interface **accessible** from any device.
- Ability to quickly get a **glance** of your data sets with dedicated sections for analysis and data exploration
- Less technological **barrier** in using the platform
- Power of deep learning with just a **few** button clicks
- Easily **interpretable** summaries and visualizations

Summary of our research

- **Dedicated** platform for the area of anomaly detection
- Simple and easy to use interface **accessible** from any device.
- Ability to quickly get a **glance** of your data sets with dedicated sections for analysis and data exploration
- Less technological **barrier** in using the platform
- Power of deep learning with just a **few** button clicks
- Easily **interpretable** summaries and visualizations
- **Scalable** and loosely coupled architecture able to tackle similar usecases in future

Future work

Future work

Future work

- Support for multiple data **connectors** like EC2, SQL, HDFS, JDBC etc.., **Note**: As of now only XLS, CSV and JSON file formats are supported.

Future work

- Support for multiple data **connectors** like EC2, SQL, HDFS, JDBC etc.., **Note**: As of now only XLS, CSV and JSON file formats are supported.
- Support for analyzing anomalies in **real-time**

Future work

- Support for multiple data **connectors** like EC2, SQL, HDFS, JDBC etc.., **Note**: As of now only XLS, CSV and JSON file formats are supported.
- Support for analyzing anomalies in **real-time**
- Adapt the platform to **similar** use cases like,
  - Predictive analytics
  - Face recognition
  - Video surveillance

Future work

- Support for multiple data **connectors** like EC2, SQL, HDFS, JDBC etc.., **Note**: As of now only XLS, CSV and JSON file formats are supported.
- Support for analyzing anomalies in **real-time**
- Adapt the platform to **similar** use cases like,
  - Predictive analytics
  - Face recognition
  - Video surveillance
- More research on **interpretability** and **explainability** of Deep learning model results using packages such as LIME and SNAP.

Future work

- Support for multiple data **connectors** like EC2, SQL, HDFS, JDBC etc.., **Note**: As of now only XLS, CSV and JSON file formats are supported.
- Support for analyzing anomalies in **real-time**
- Adapt the platform to **similar** use cases like,
  - Predictive analytics
  - Face recognition
  - Video surveillance
- More research on **interpretability** and **explainability** of Deep learning model results using packages such as LIME and SNAP.
- Integrate more **complex** deep learning models like VGG-Net, Inception, ResNet for computer vision use-cases like Face recognition and Video surveillance.

References i

Sampada Bhosale.
Network Intrusion Detection.
https://www.kaggle.com/sampadab17/network-intrusion-detection/, 2018.

Raghavendra Chalapathy and Sanjay Chawla.
Deep learning for anomaly detection: A survey.
arXiv preprint arXiv:1901.03407, 2019.

Varun Chandola, Arindam Banerjee, and Vipin Kumar.
Anomaly detection: A survey.
ACM computing surveys (CSUR), 41(3):1–58, 2009.

References ii

Xin He, Kaiyong Zhao, and Xiaowen Chu.
Automl: A survey of the state-of-the-art.
arXiv preprint arXiv:1908.00709, 2019.

KDD.
KDD Cup 1999 Data.
`http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html/`, 1999.

Thank you for your attention !