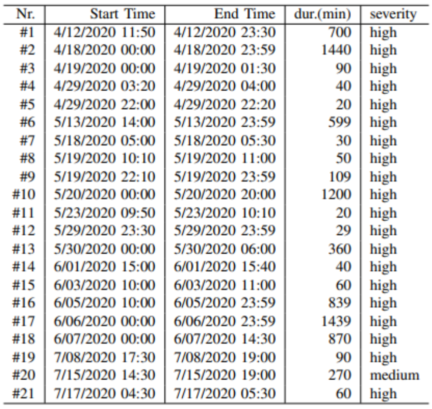
Subject: DADS 6005 Data Streaming and Real Time Analytics
By Thunchanok Surasunsanee 6520422007, Budsadee Sareerasart 6520422009

Title: Predictive Anomaly Detection of the Air Production Unit (APU) failure

Context: The dataset includes 15 features that are measurements and electric signals, obtained from several analogue and digital sensors installed on the compressor of a train from February to August 2020. We demonstrate the data streaming by using this dataset as time series data coming in every 10 seconds without status of anomaly detection. However, with the APU failure report below, medium to high anomalies are irregularly detected during April and July 2020.



Nr.	Start Time	End Time	dur.(min)	severity
#1	4/12/2020 11:50	4/12/2020 23:30	700	high
#2	4/18/2020 00:00	4/18/2020 23:59	1440	high
#3	4/19/2020 00:00	4/19/2020 01:30	90	high
#4	4/29/2020 03:20	4/29/2020 04:00	40	high
#5	4/29/2020 22:00	4/29/2020 22:20	20	high
#6	5/13/2020 14:00	5/13/2020 23:59	599	high
#7	5/18/2020 05:00	5/18/2020 05:30	30	high
#8	5/19/2020 10:10	5/19/2020 11:00	50	high
#9	5/19/2020 22:10	5/19/2020 23:59	109	high
#10	5/20/2020 00:00	5/20/2020 20:00	1200	high
#11	5/23/2020 09:50	5/23/2020 10:10	20	high
#12	5/29/2020 23:30	5/29/2020 23:59	29	high
#13	5/30/2020 00:00	5/30/2020 06:00	360	high
#14	6/01/2020 15:00	6/01/2020 15:40	40	high
#15	6/03/2020 10:00	6/03/2020 11:00	60	high
#16	6/05/2020 10:00	6/05/2020 23:59	839	high
#17	6/06/2020 00:00	6/06/2020 23:59	1439	high
#18	6/07/2020 00:00	6/07/2020 14:30	870	high
#19	7/08/2020 17:30	7/08/2020 19:00	90	high
#20	7/15/2020 14:30	7/15/2020 19:00	270	medium
#21	7/17/2020 04:30	7/17/2020 05:30	60	high

Figure 1: APU failure reported by the expert from April to July 2020

Objective:

- To simulate real time data streaming of signals, design a data pipeline and proceed streaming integration with Kafka connect.
- To predict anomaly detection using traditional machine learning methods (Offline) and Online machine learning.

Solution Ideas:

- **Flowchart**

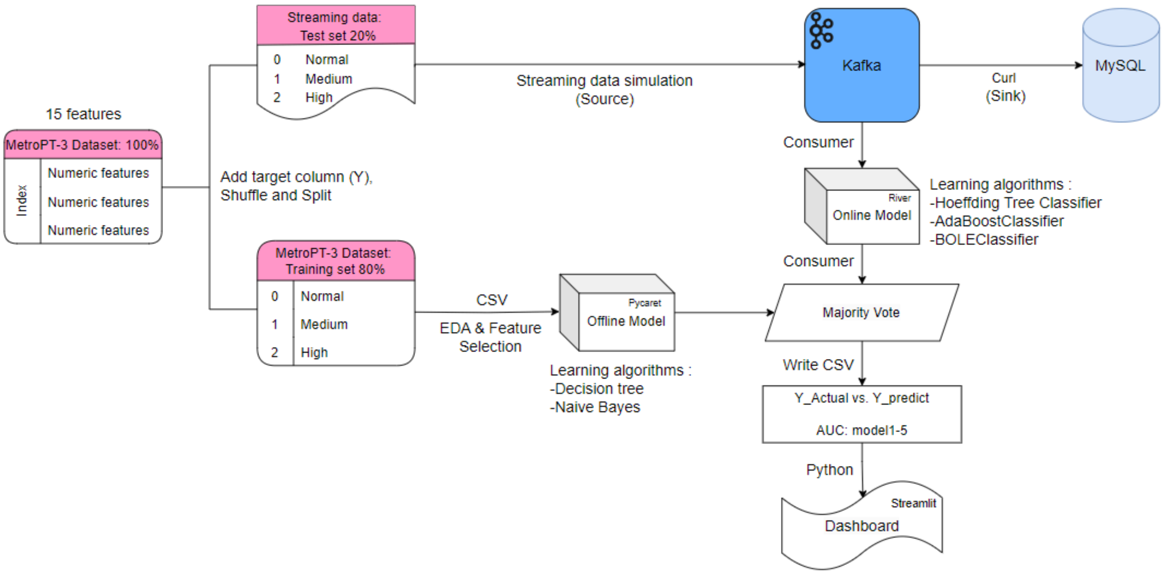


Figure 2: Pipeline

- **Algorithms**
  - **Offline -** Use Pycaret to obtain 2 offline models for further prediction.
    - Decision Tree
    - Naive Bayes
  - **Online -** Use 3 models from River in order to lower the high chance of concept drift in streaming data. The following 3 selected models are most similar in terms of capability in handling streaming data and being suitable for real time anomaly detection due to its ability to adapt to changes in the data distribution.
    - Hoeffding Tree Classifier
    - AdaBoostClassifier
    - BOLEClassifier

Experiment:

- **EDA**

The correlation matrix below shows less correlation from 4 features (Caudal Impulses, Oil level, Pressure Switch and LPS) which are all electrical signals.

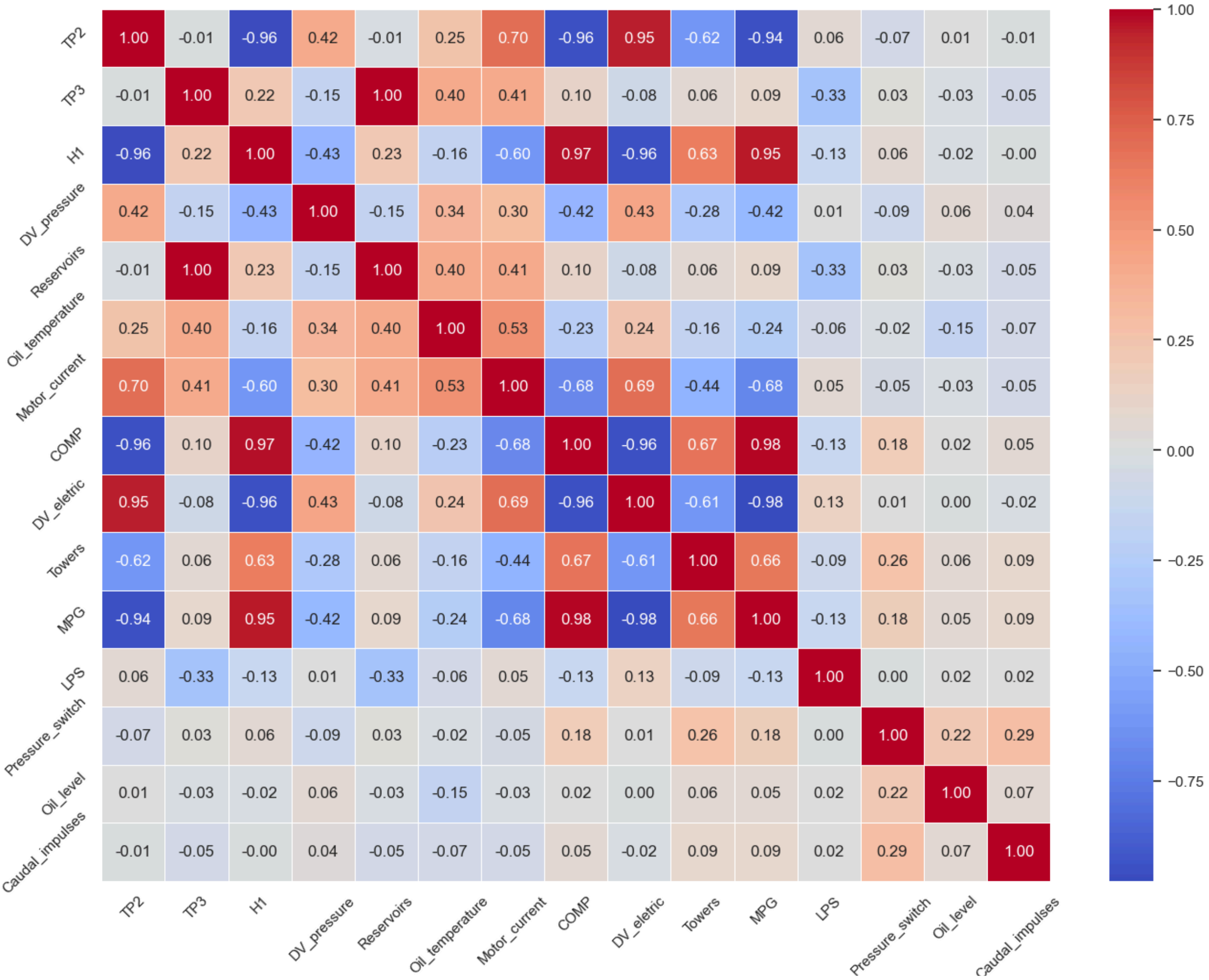


Figure 3: Correlation Matrix of numerical features

DV electric, COMP, MPG and Towers are shown least correlated with the target column (Y), so a total of 8 features are omitted from building offline ML models.

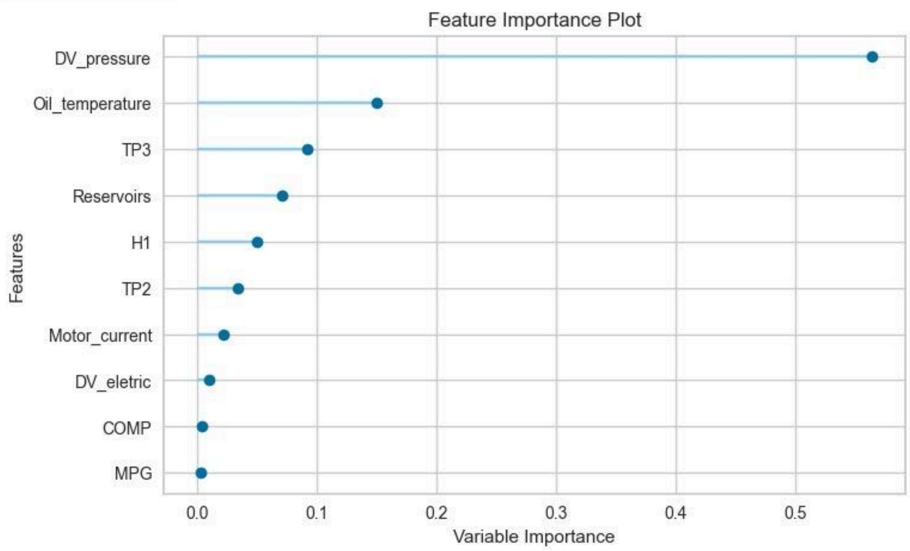


Figure 4: Feature Importance Plot from PyCaret

As such, only a total of 7 features including TP2, TP3, H1, DV_pressure, Reservoirs, Oil_temperature and Motor_current are used to build models.

- **Train and test process**
  - Data splitting

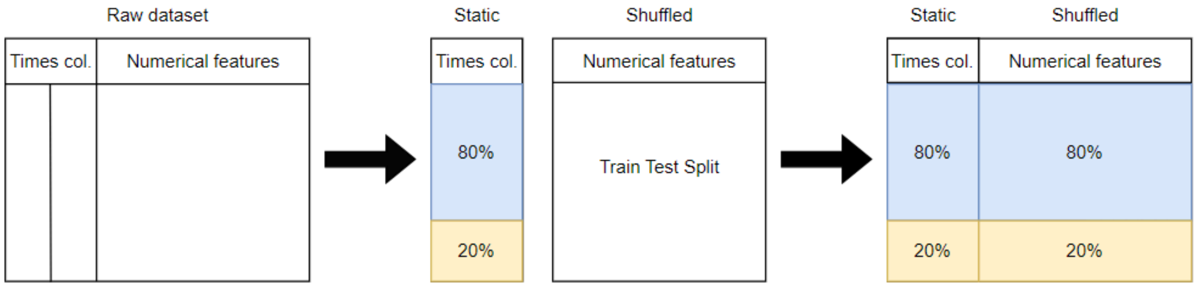


Figure 5: Data Splitting process

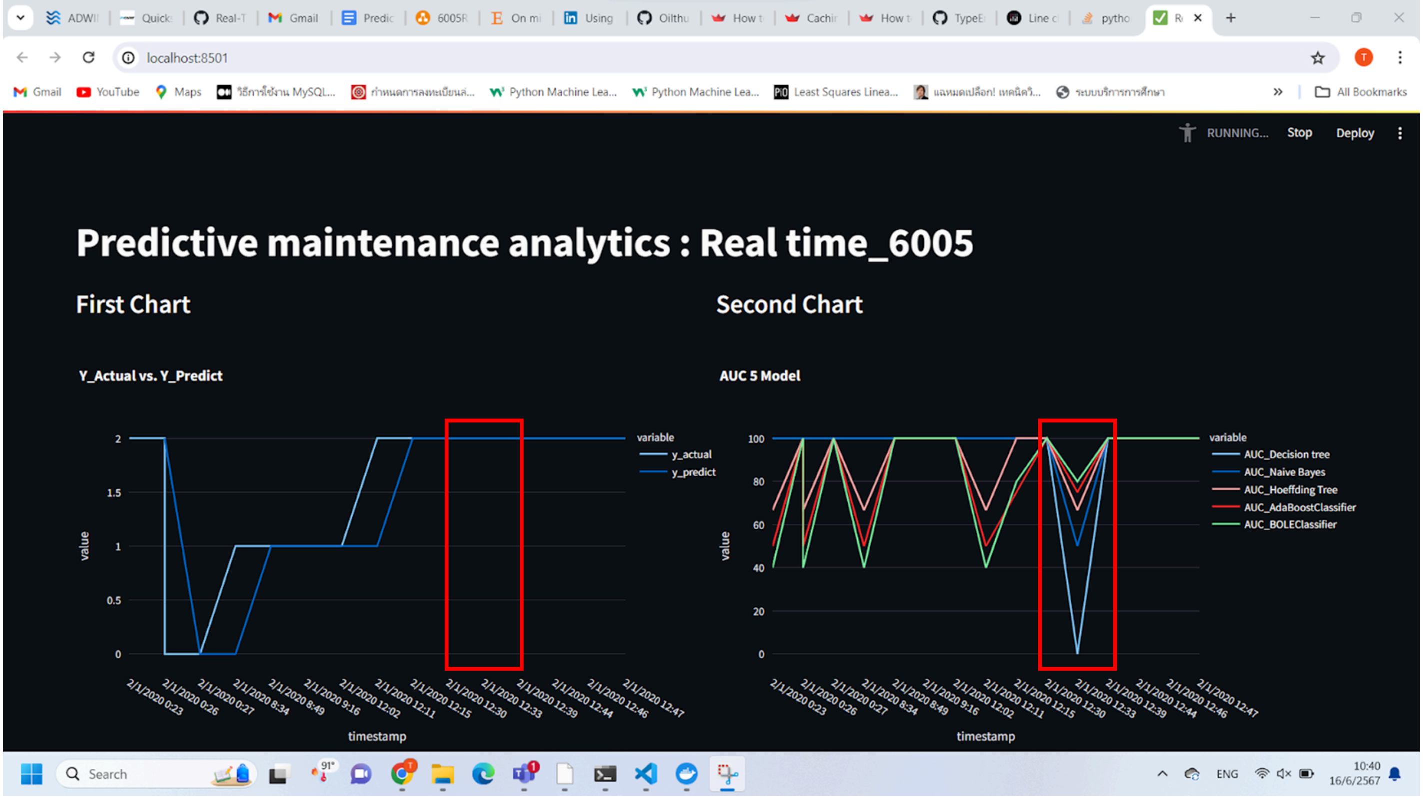
This process allows only numerical features to be shuffled to handle the imbalance dataset.

  - Train and test process

Once both dataframes are concatenated, 80% of the shuffled dataset is used to train models in PyCaret. We obtain 2 offline models (Decision Tree and Naive Bayes) to work with other 3 selected online models (Hoeffding Tree Classifier, AdaBoostClassifier and BOLEClassifier) on 20% of the separated datasets consumed from Kafka Cluster.

Both offline models and online models are preliminarily working independently to acquire individual y_predicted. Then we use the majority vote function to select only the most common y_predicted value from 5 individual y_predicted values.

Results:

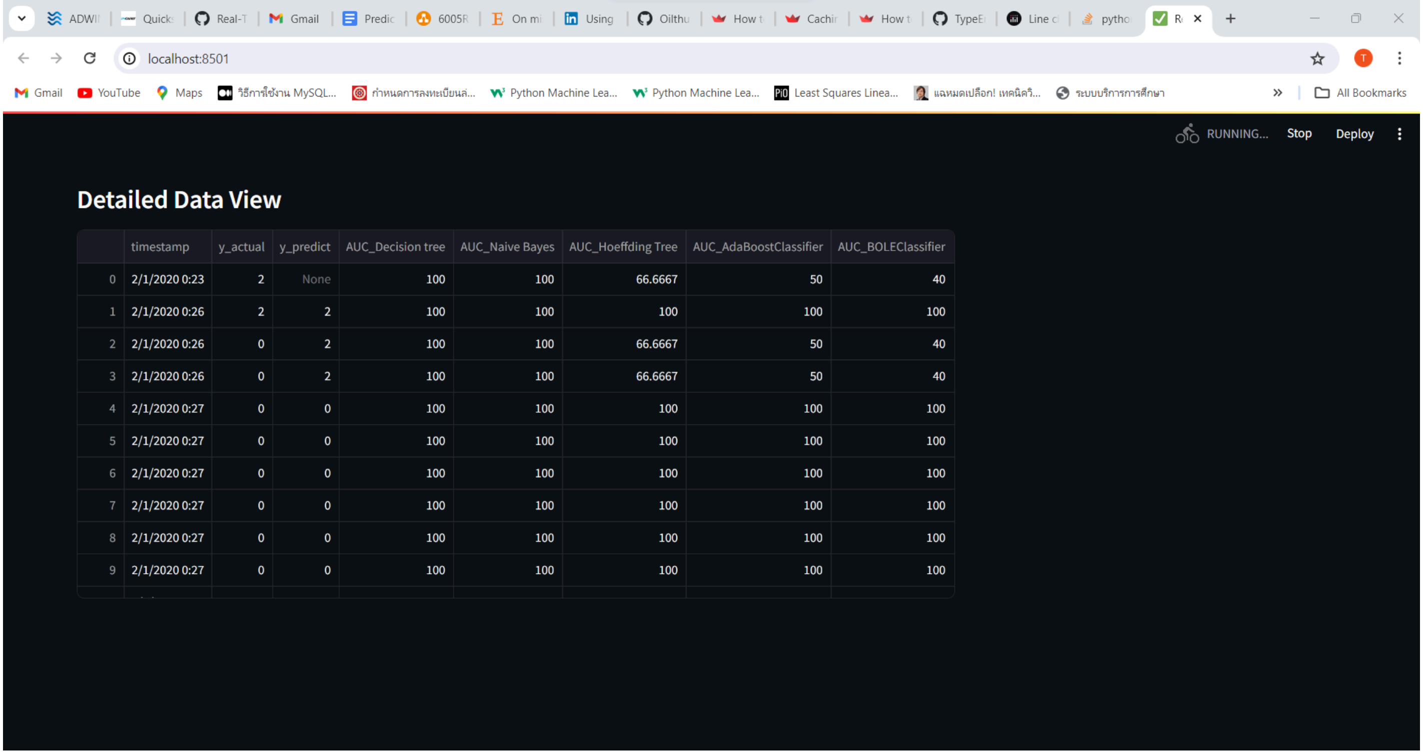


Predictive maintenance analytics : Real time_6005

First Chart

Second Chart

Figure 6: Graphs of actual Y and the predicted Y from Majority Vote and AUC of each offline and online models



Detailed Data View

	timestamp	y_actual	y_predict	AUC_Decision tree	AUC_Naive Bayes	AUC_Hoeffding Tree	AUC_AdaBoostClassifier	AUC_BOLEClassifier
0	2/1/2020 0:23	2	None	100	100	66.6667	50	40
1	2/1/2020 0:26	2	2	100	100	100	100	100
2	2/1/2020 0:26	0	2	100	100	66.6667	50	40
3	2/1/2020 0:26	0	2	100	100	66.6667	50	40
4	2/1/2020 0:27	0	0	100	100	100	100	100
5	2/1/2020 0:27	0	0	100	100	100	100	100
6	2/1/2020 0:27	0	0	100	100	100	100	100
7	2/1/2020 0:27	0	0	100	100	100	100	100
8	2/1/2020 0:27	0	0	100	100	100	100	100
9	2/1/2020 0:27	0	0	100	100	100	100	100

Figure 7: Dataframe of the obtained Y values and AUC from each models

Reference: https://archive.ics.uci.edu/dataset/791/metropt+3+dataset