Cool Train

INFOH423 Data Mining Project 2023/24

Students:

- Mohamed Louai Bouzaher - 000585861
- Hieu Nguyen Minh - 000583782
- Jesús Celada García - 000588299
- Eva Groth - 000588492

Professors:

- Mahmoud SAKR
- Raphaël GYORI

Table of Content

Business Understanding

What are the SNCB's business goals?

- Safety
- Increase passenger satisfaction
- Innovation
- Continuous improvement

What are the mains tasks of B-technics SNCB's section?

- Preventive maintenance
- Repairs

Data Understanding

2GB

CSV Data

Aug 22 - Sep 23

12

Features

92

Number of Vehicles

Data Understanding

Fields & Significance

1. **Mapped Vehicle ID**: Unique identifier for each vehicle. Helps in tracking individual vehicles throughout the dataset.
2. **Timestamps (UTC)**: Indicates the time at which each sample was taken. Facilitates time-based analysis, trend identification, and temporal patterns.
3. **Latitude & Longitude**: GPS coordinates of the vehicles. Provides spatial information, enabling mapping and geospatial analysis to track the movement and location of vehicles.
4. **RS_E_InAir Temp_PC1 & RS_E_InAir Temp_PC2**: Temperature readings from redundant engine cooling systems (e.g., primary and secondary cooling systems). Monitoring these temperatures helps identify cooling system efficiency, potential overheating, or anomalies.
5. **RS_E_OilPressure_PC1 & RS_E_OilPressure_PC2**: Pressure readings from engine oil systems (primary and secondary). Crucial for assessing engine health, oil system efficiency, and detecting potential issues like leaks or pressure irregularit ies.
6. **RS_E_RPM_PC1 & RS_E_RPM_PC2**: RPM (Revolutions Per Minute) of the engines. Helps in evaluating engine performance, detecting engine malfunctions, and identifying irregularities in engine speed.
7. **RS_E_WaterTemp_PC1 & RS_E_WaterTemp_PC2**: Water temperature readings from engine systems. Monitoring water temperature is vital for engine health and detecting issues like overheating or cooling system inefficiencies.
8. **RS_T_OilTemp_PC1 & RS_T_OilTemp_PC2**: Oil temperature readings from engine systems. Significant for assessing engine performance, oil viscosity, and identifying issues related to oil temperature such as overheating or inadequate lubrication.

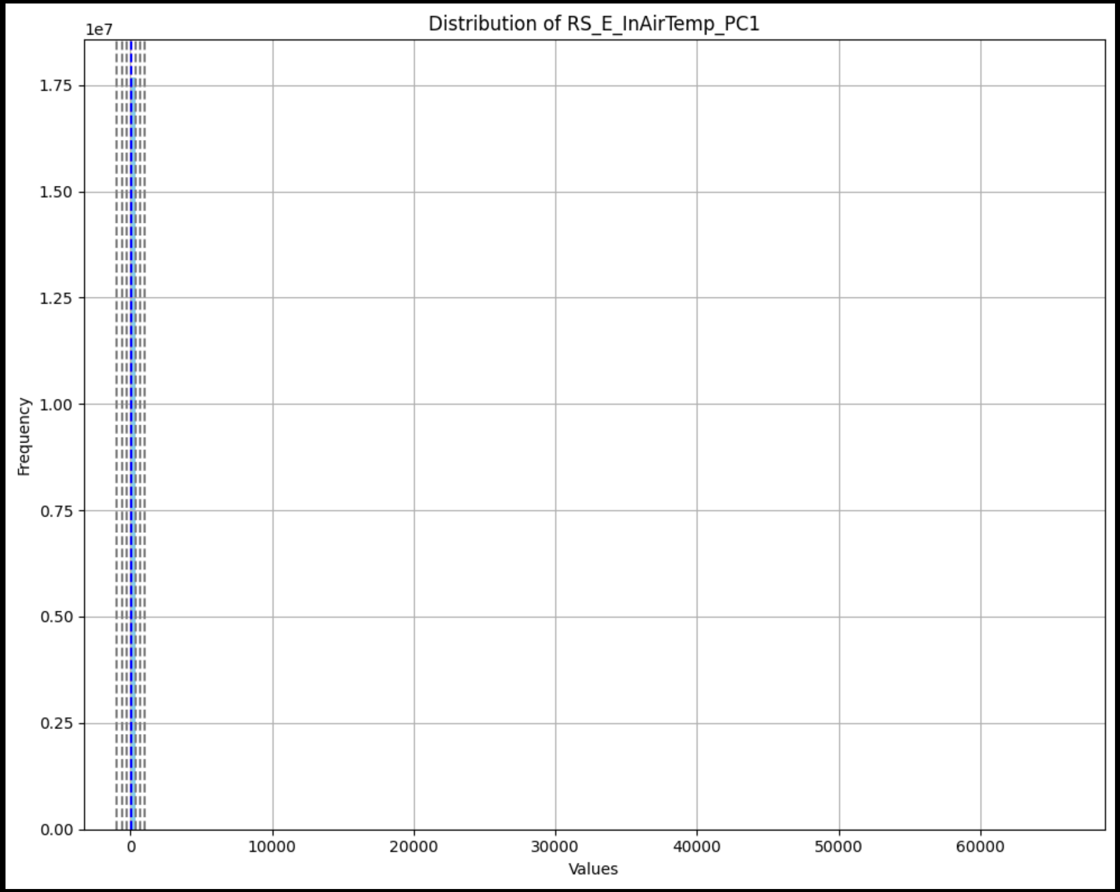
Data Understanding

Data Distribution

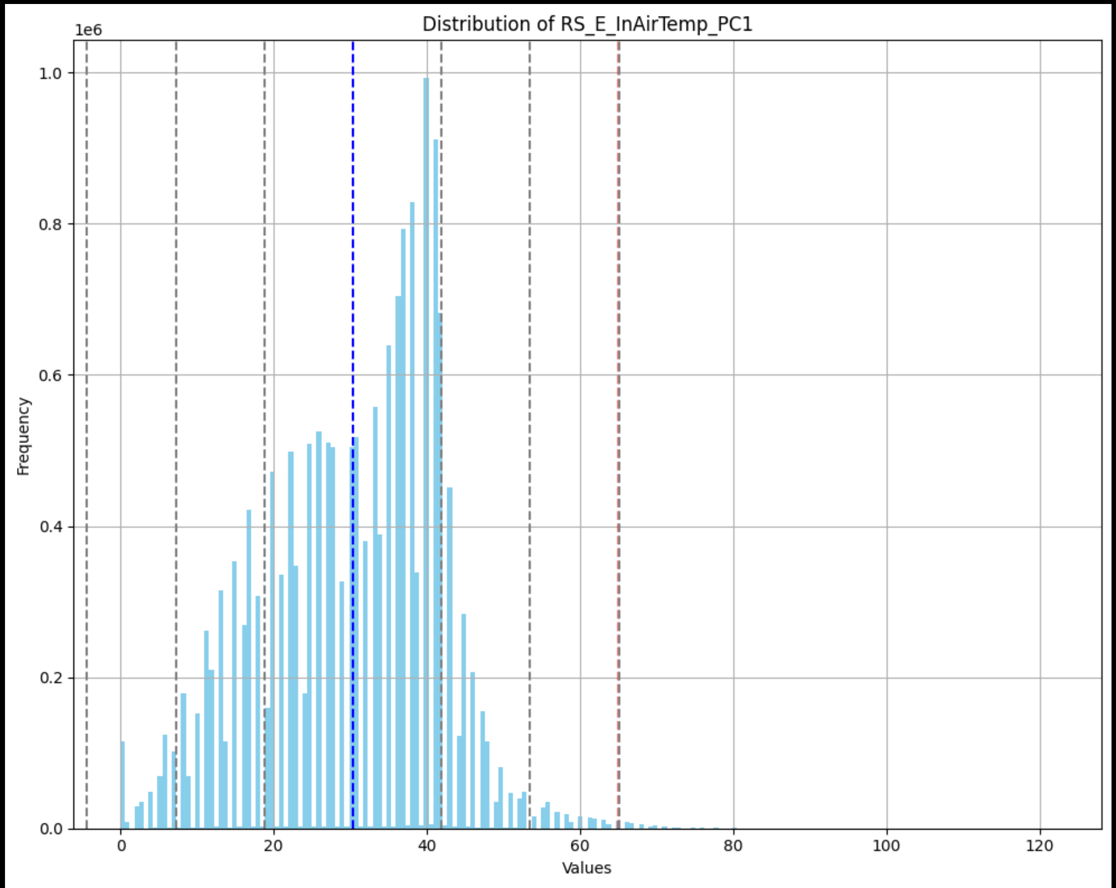
- Initial inspection of the Air Temperature in PC1 data reveils the existence of outliers which affect the data distribution and can be considered as **noise**.

Range of Air Temperature in PC1:

[0, 65535.0]



Data Understanding
Data Distribution

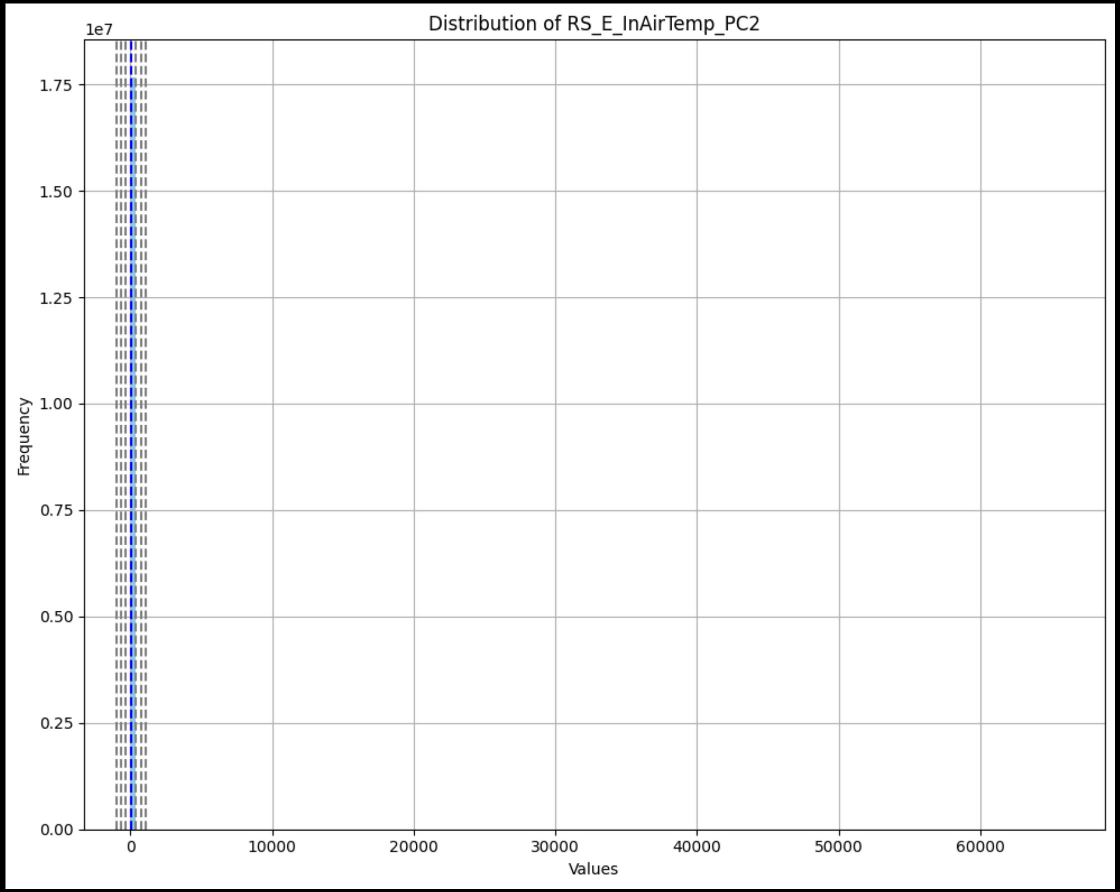


Data Understanding
Data Distribution

- Same goes for Air Temperature in PC2.

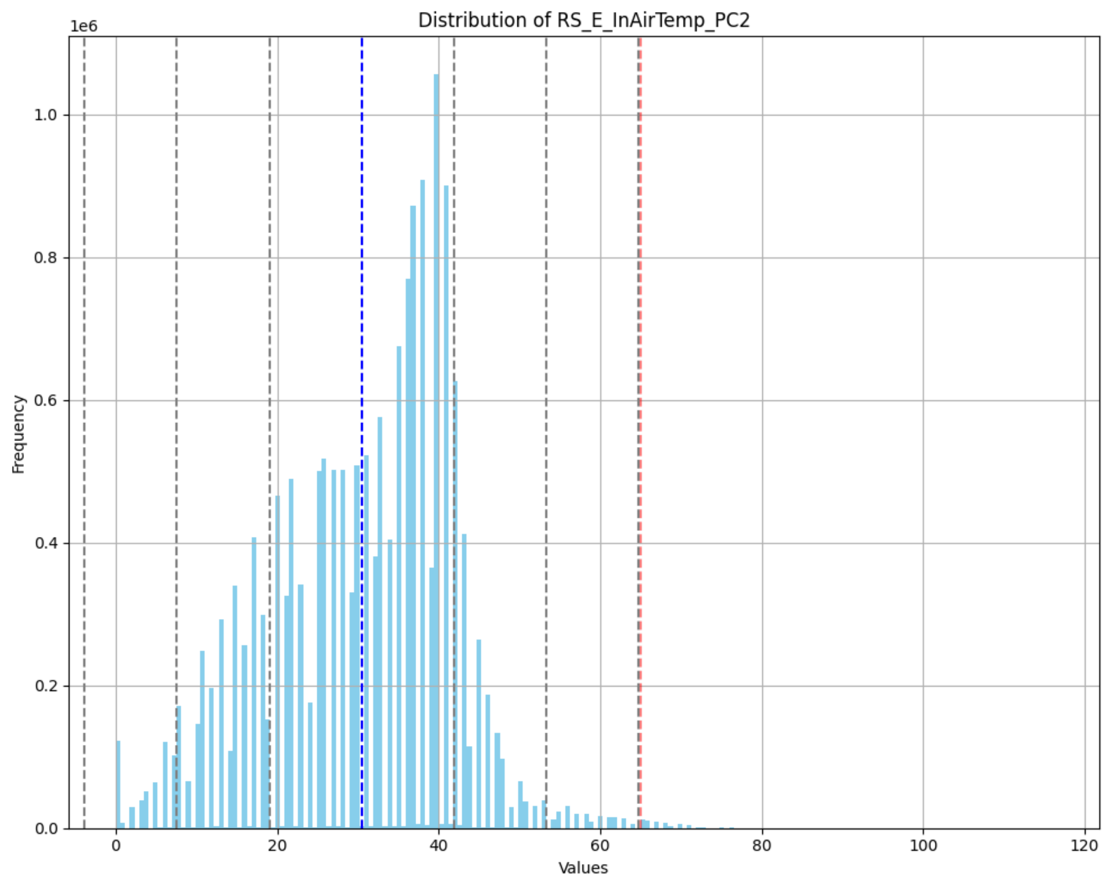
Range of Air Temperature in PC2:

[0, 65535.0]



Data Understanding

Data Distribution

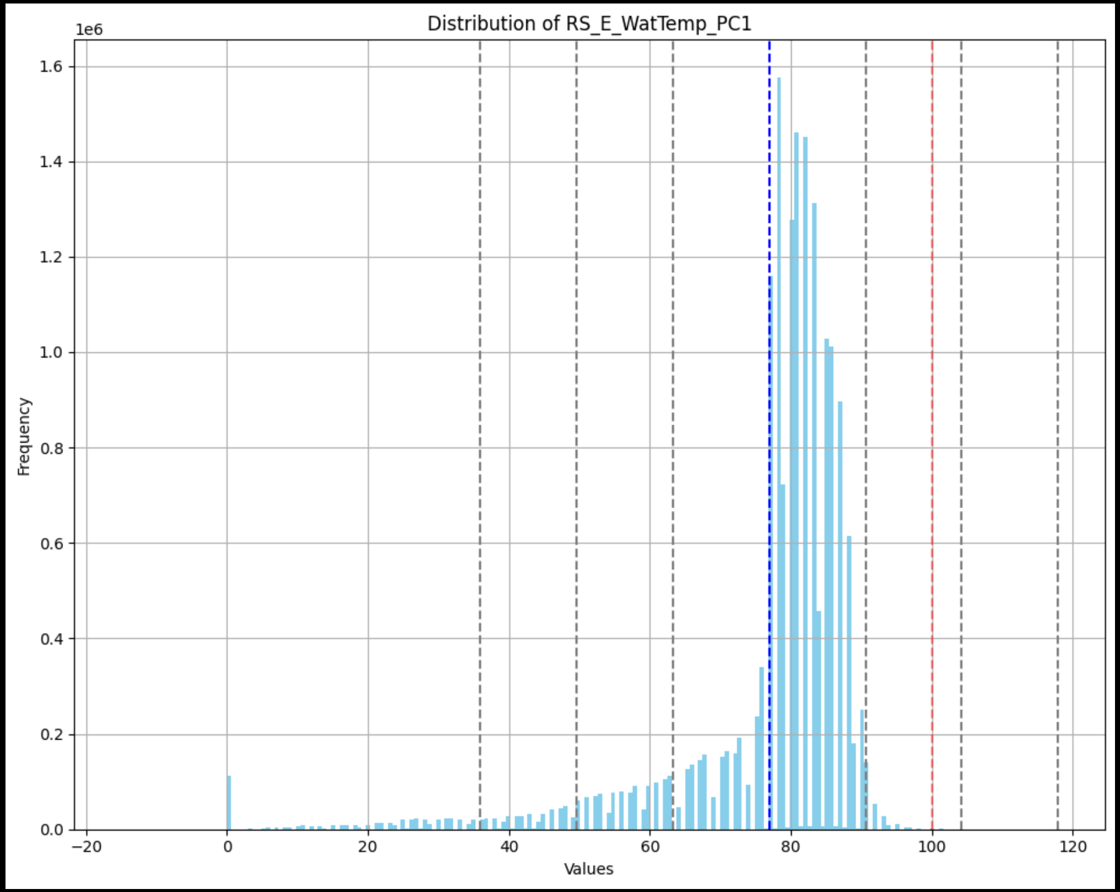


Data Understanding
Data Distribution

- Inspecting Water Temperature in PC1 shows that some values fall outside the permissible maximum value (100°C).

Range of Water Temperature in PC1:

[-15.0, 109.0]



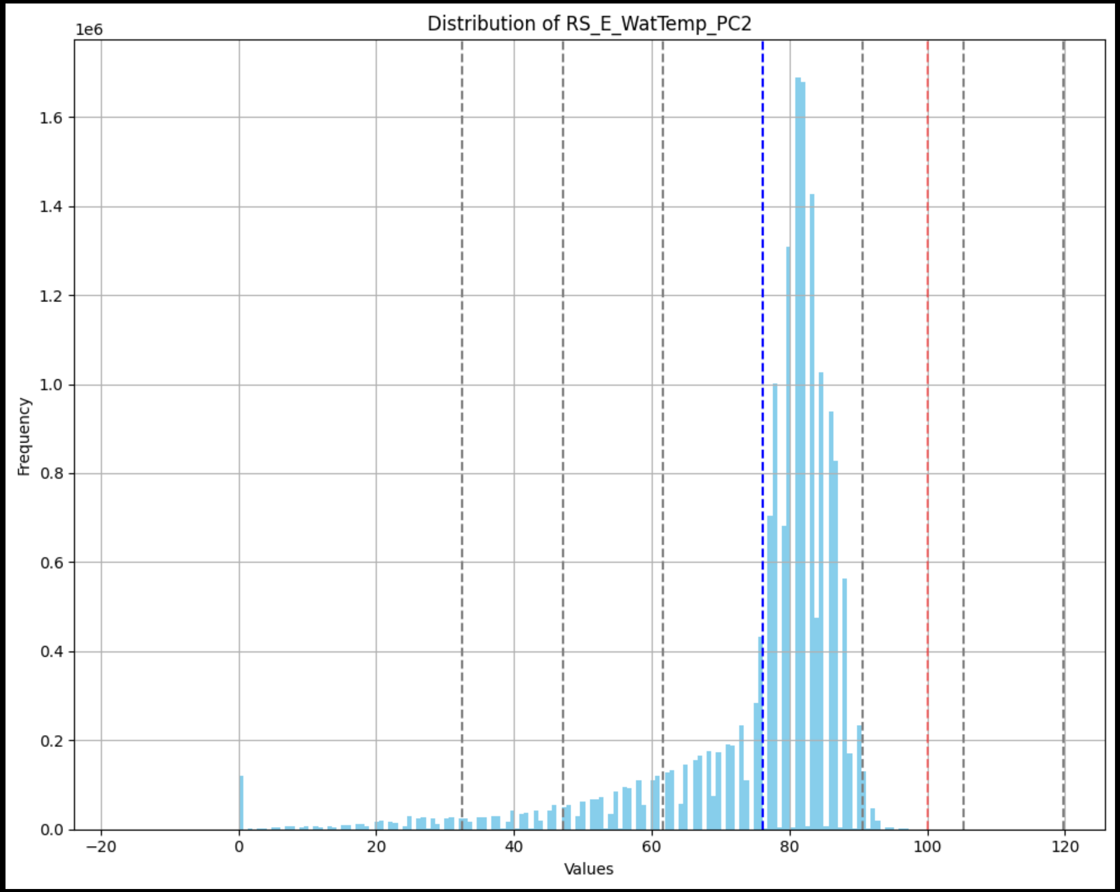
Data Understanding

Data Distribution

- Same applies to Water Temperature in PC2.

Range of Water Temperature in PC2:

[-17.0 , 119.0]



Data Understanding

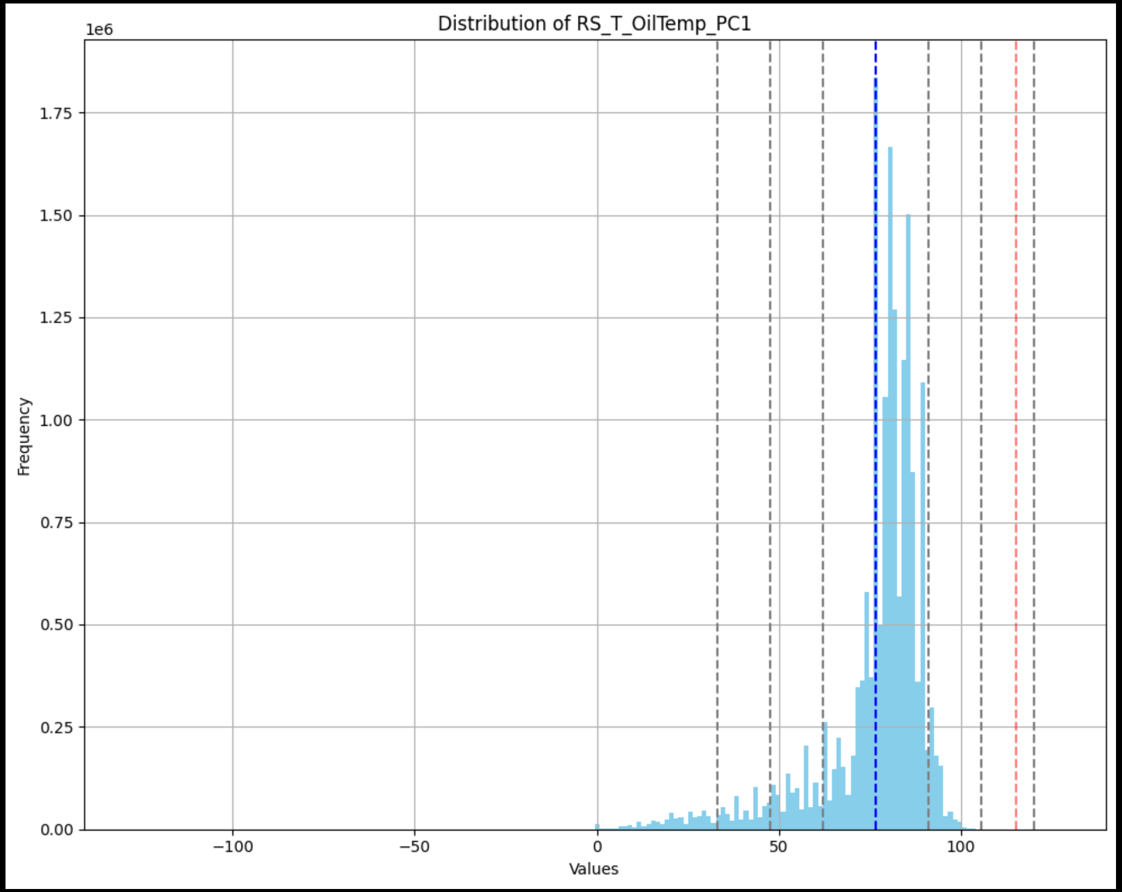
Data Distribution

- Initial inspection of the Oil Temperature in PC1 data reveils the existence of outliers which affect the data distribution.

This can considered as form of **noise.**

Range of Oil Temperature in PC1:

[-128.0 , 127.0]

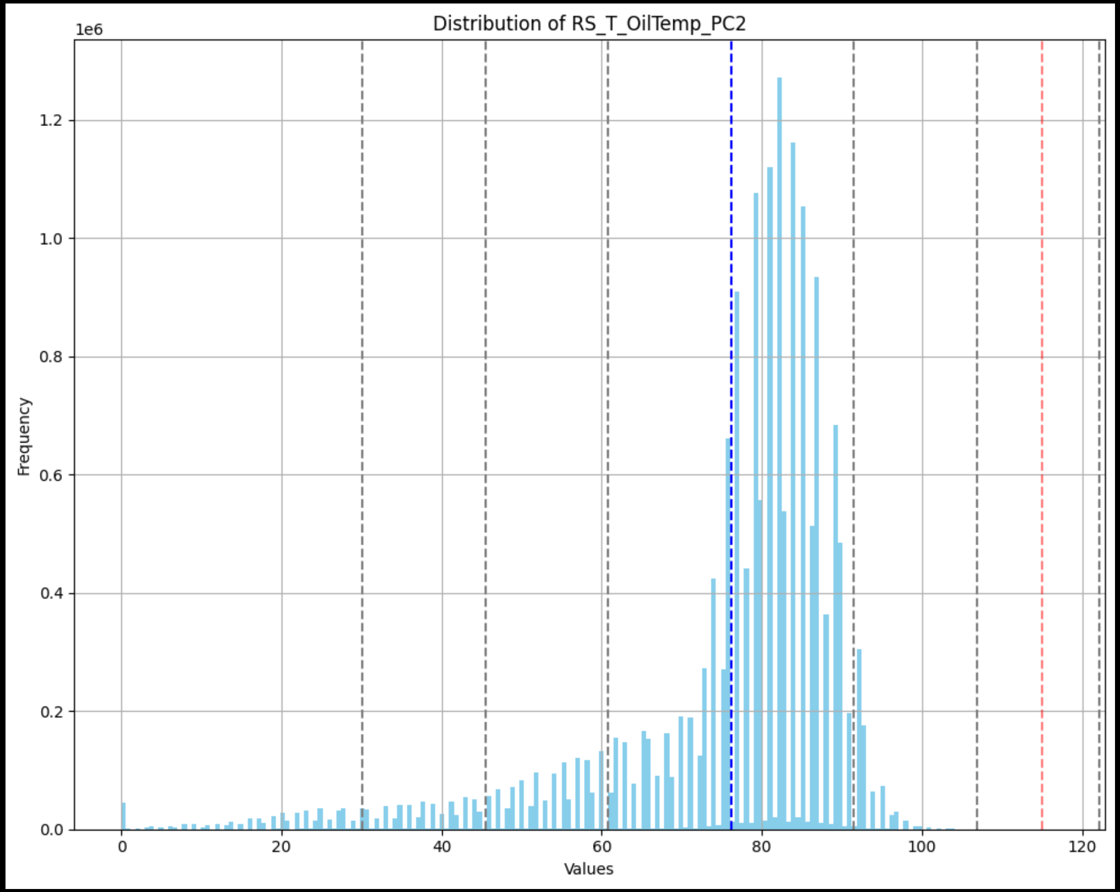


Data Understanding
Data Distribution

- As for the Oil Temperature in PC2, most data point fall within the range, and only few fall outside the range of 3 Standard Deviations.

Range of Oil Temperature in PC2:

[0.0 , 117.0]

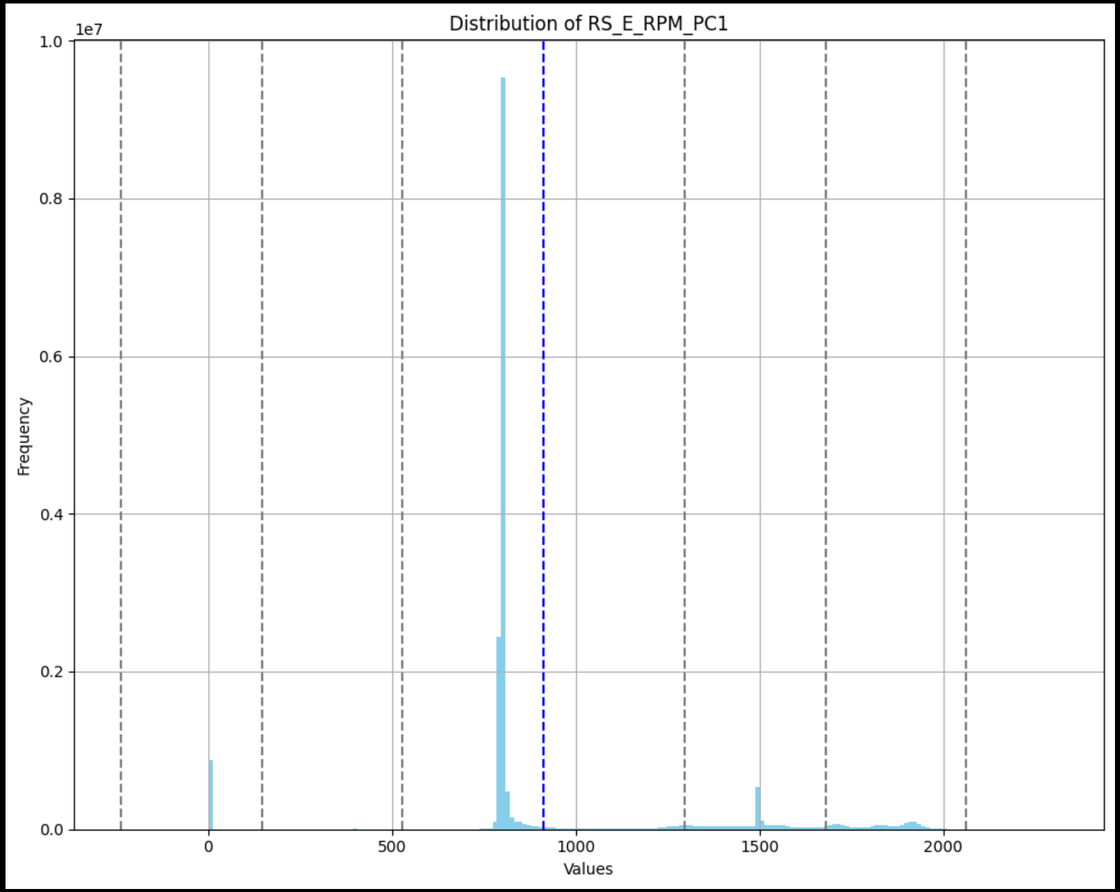


Data Understanding
Data Distribution

- Most data points of RPM in PC1 fall in range of [700, 2000]. However, There are many instances of 0 values.

Range of RPM PC1 Values:

[0.0, 2309.0]

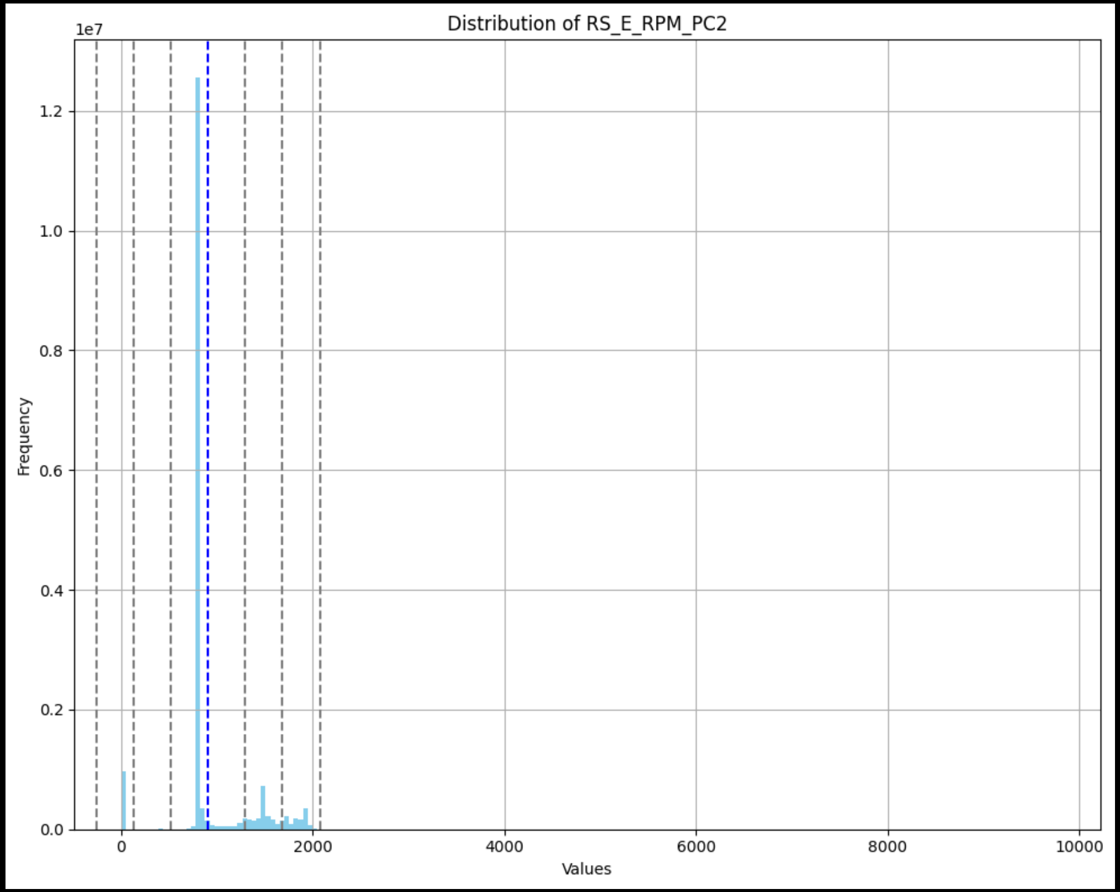


Data Understanding
Data Distribution

- Also, most data points of RPM in PC2 fall in range of [700, 2000]. However, There are many instances of 0 values and outliers that go beyond 2000.

Range of RPM PC2 Values:

[0.0 , 9732.0]



Data Preparation

In order to facilitate data manipulation, we loaded the source data to an Azure PostgreSQL Server,

Where we performed the data cleaning operations.

Data Preparation
Data Cleaning

- After exploring the data distribution, we decided to clean the data by:
  + Drop all NULL values
  + Drop cases where latitude and longitude values are not specified
  + Drop cases where value>mean+6*std (noises)
  + Drop the duplicates

Data Preparation

Weather Data

- After cleaning the data, we integrated weather data from Open Meteo API.
- Link: https://open-meteo.com/ Open-Meteo
- To reduce API calls:
  - Round the latitudes and longitudes (e.g. 50.7698183 --> 50.8)
  - Get the weather data every hour (e.g. 7:00:00, 8:00:00)
- Weather data:
  - Temperature (°C)
  - Humidity (%)
  - Rain (mm)
  - Snow (m)
  - Weather code (for weather description)
  - Cloud cover (%)
  - Evapotranspiration (mm)
  - Wind speed (km/h)

Data Preparation

Weather Data

- API Call

```python
for _, row in short_time_location_data.iterrows():
  params = {
    "latitude": row['lat'],
    "longitude": row['lon'],
    "hourly": ["temperature_2m", "relative_humidity_2m", "rain", "snow_depth", "weather_code", "cloud_cover", "et0_fao_evapotranspiration", "wind_speed_10m"],
    "start_date": row['timestamp'],
    "end_date": row['timestamp']
  }

  responses = openmeteo.weather_api(url, params=params)
```

- Results

mapped_veh_id	timestamps_UTC	lat	lon	RS_E_InAirTemp_P	RS_E_InAirTemp_P	RS_E_OilPress_PC1	RS_E_OilPress_PC2	RS_E_RPM_PC1	RS_E_RPM_PC2	RS_E_WatTemp_PC	RS_E_WatTemp_PC	RS_T_OilTemp_PC	RS_T_OilTemp_PC	temperature	humidity	rain	snow_depth	weather_code	cloud_cover	evapotranspiration	wind_speed
116	2023-06-13 9:01:11	51.016409	3.7729487	34	40	193	169	802	801	87	85	85	82	25.039	39.178665	0	0	0	0	0.46318206	16.071491
146	2023-06-13 9:01:11	51.014318	3.7792497	40	34	220	244	798	801	78	82	79	75	25.039	39.178665	0	0	0	0	0.46318206	16.071491
176	2023-06-13 9:01:12	50.9953063	3.8122252	46	43	355	379	1785	1752	87	87	87	87	25.039	39.178665	0	0	0	0	0.46318206	16.071491
148	2023-06-13 9:01:13	51.0216133	3.7626621	37	32	351	224	799	801	51	75	52	76	25.039	39.178665	0	0	0	0	0.46318206	16.071491
102	2023-06-13 9:01:21	51.0137487	3.779246	38	32	213	207	795	793	80	85	75	80	25.039	39.178665	0	0	0	0	0.46318206	16.071491
146	2023-06-13 9:01:21	51.0141058	3.7798364	40	34	224	244	806	798	78	82	76	76	25.039	39.178665	0	0	0	0	0.46318206	16.071491
102	2023-06-13 9:01:24	51.0139681	3.7786963	38	32	213	207	799	791	80	85	75	80	25.039	39.178665	0	0	0	0	0.46318206	16.071491
194	2023-06-13 9:01:29	51.0210669	3.763724	35	41	220	244	800	805	78	80	73	79	25.039	39.178665	0	0	0	0	0.46318206	16.071491
147	2023-06-13 9:01:37	50.99301	3.8198325	45	42	200	203	796	804	90	86	84	89	25.039	39.178665	0	0	0	0	0.46318206	16.071491
170	2023-06-13 9:01:47	51.016194	3.7735464	32	37	196	251	794	804	78	86	76	82	25.039	39.178665	0	0	0	0	0.46318206	16.071491
119	2023-06-13 9:01:51	51.0130876	3.7808002	33	32	3	10	0	0	31	31	31	27	25.039	39.178665	0	0	0	0	0.46318206	16.071491
102	2023-06-13 9:01:54	51.013974	3.7787042	38	32	210	203	795	799	80	85	76	79	25.039	39.178665	0	0	0	0	0.46318206	16.071491
148	2023-06-13 9:01:57	51.0213395	3.7631515	37	32	351	220	800	799	51	74	52	73	25.039	39.178665	0	0	0	0	0.46318206	16.071491
112	2023-06-13 9:01:57	50.9919067	3.8270181	41	43	251	244	797	804	80	83	79	81	25.039	39.178665	0	0	0	0	0.46318206	16.071491
170	2023-06-13 9:01:57	51.0164015	3.7729755	32	37	200	251	799	810	78	86	77	81	25.039	39.178665	0	0	0	0	0.46318206	16.071491
112	2023-06-13 9:02:00	50.991751	3.8288266	41	43	251	244	795	804	80	83	79	81	25.039	39.178665	0	0	0	0	0.46318206	16.071491

Data Modeling

Upon the completion of data preparation, we started investigating different anomaly detection methods, namely:

1. **Statistical Methods**
2. **Time Series Analysis**
3. **Machine Learning Models**

Data Modeling

Statistical Methods

This was mainly looking for one of the three:

1. **Finding temperatures thresholds: A**bove 65°C (for air), 100°C (for water), or 115°C (for oil) as anomalies (Realistic but calls for stopping the engine)
2. **Detecting sensor malfunction:** Unrealistic temperatures (> 200°C) and/or pressure values; Pressure < mean - 3 SD Or Pressure > mean + 3 SD.
3. **Detecting engine failure:** huge difference between RPM in PC1 and PC2.

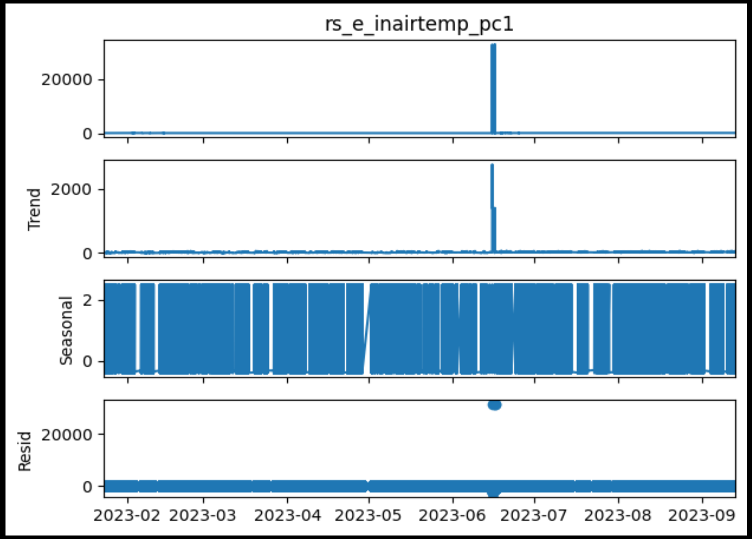
Data Modeling
Statistical Methods - Findings

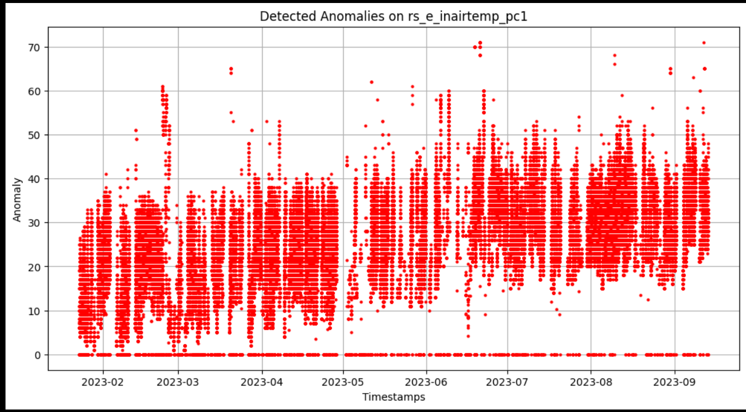
- Air Temperature Anomalies:
  + Number of instances: **78446**
- Water Temperature Anomalies:
  + Number of instances: **1986**
- Oil Temperature Anomalies:
  + Number of instances: **76**
- Engine Failure Anomalies:
  + Number of instances: **864551**

Data Modeling
Time Series Analysis

In order to catch a pattern/trend in the changes happening on the level of one train at a time, we used **Seasonal Decomposition.**

However, this method did not yield useful results as it reported a very high percentage of instances as anomalies.





Data Modeling

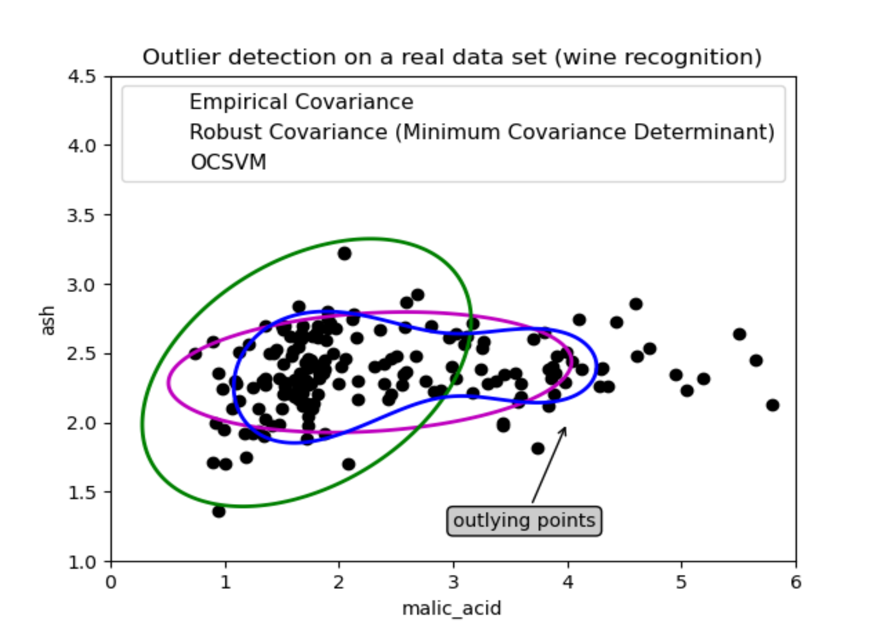
Machine Learning Methods

We experimented with the following methods:

1. Elleptic Envelope
2. Isolation Forest
3. Local Outlier Factor
4. K-Means Clustering

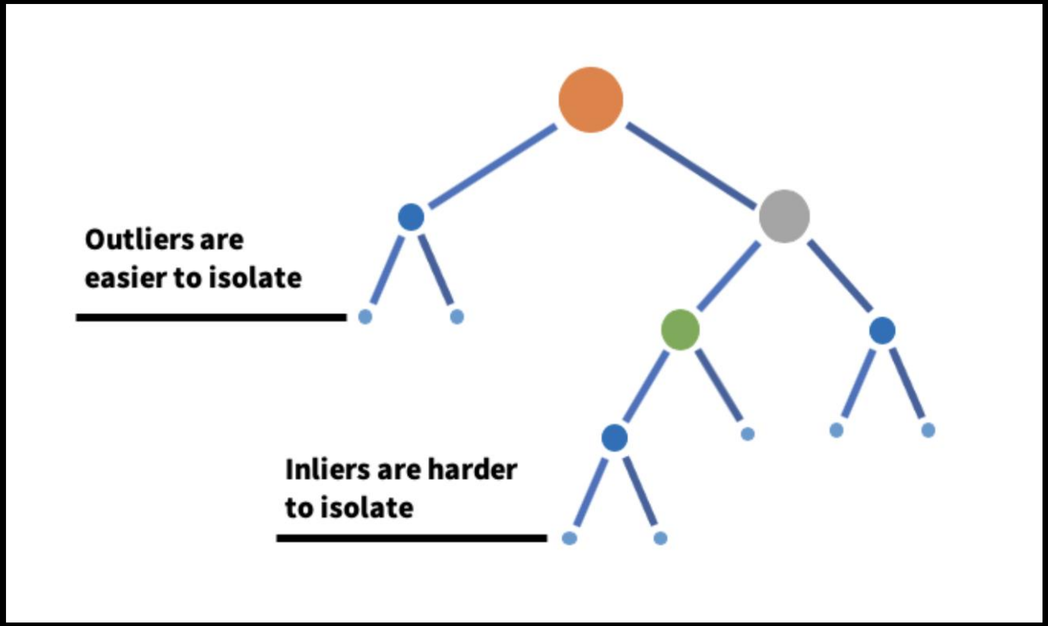
Data Modeling
Elleptic Envelope

The Elliptic Envelope is an algorithm used for outlier detection in machine learning and statistical analysis. Its primary purpose is to identify outliers in a dataset by assuming the inlying data to be Gaussian distributed and fitting an ellipse to the central data points. Data points lying outside this ellipse are considered potential outliers.



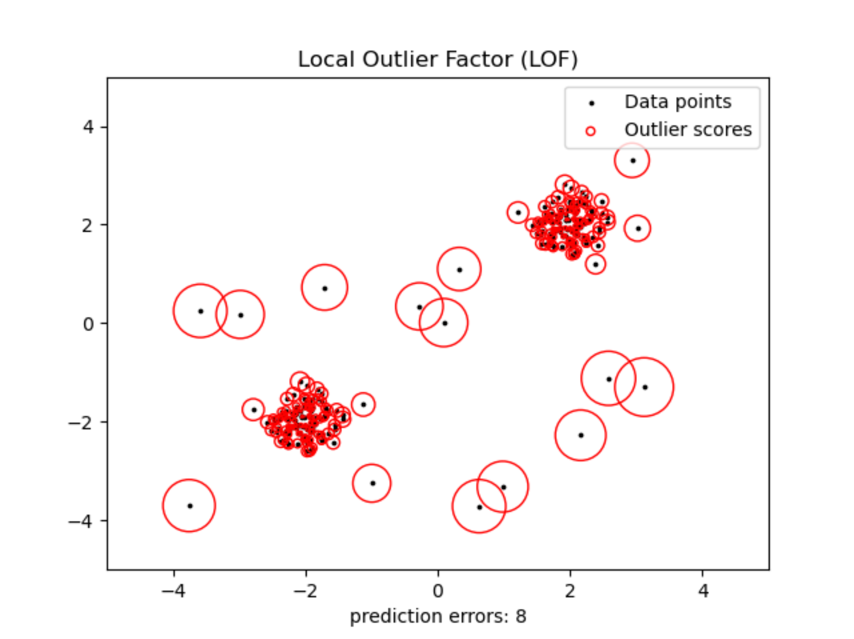
Data Modeling
Isolation Forest

Isolation Forest is an unsupervised machine learning algorithm used for outlier detection. It stands out for its ability to efficiently detect anomalies (outliers) in datasets, especially in large datasets, by leveraging the concept of decision trees.



Data Modeling
Local Outlier Factor

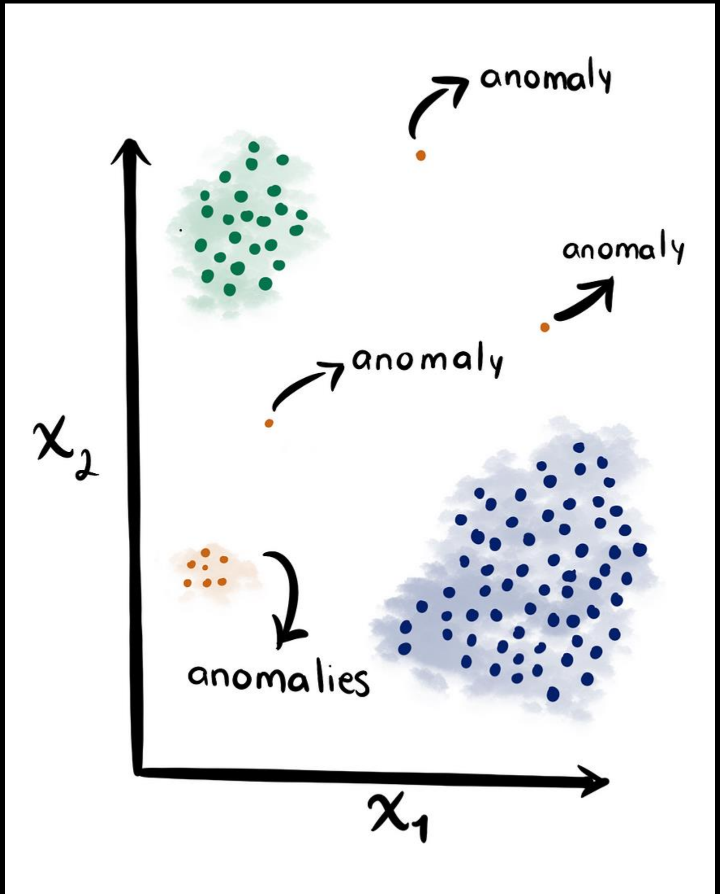
- The Local Outlier Factor (LOF) is an unsupervised machine learning algorithm used for outlier detection. It assesses the local deviation of density for each data point concerning its neighbors, identifying instances that have a significantly different density compared to their local neighborhood.



Data Modeling

K-Means Clustering

- K-means clustering is an unsupervised machine learning algorithm used for partitioning a dataset into K distinct, non-overlapping clusters. The primary objective of this algorithm is to group data points into clusters where each data point belongs to the cluster with the nearest mean (centroid), serving as a prototype of the cluster.



Training Setup

- Divide the data into a training set (80%) and a test set (20%)
- Apply data cleaning procedure to training set only (because test set is supposed to be unknown, or unseen)
- In order to classify types of anomaly:
  - Detect anomalies separately for every data feature (except for RPM)
    'RS_E_InAirTemp_PC1', 'RS_E_InAirTemp_PC2', 'RS_E_OilPress_PC1', 'RS_E_OilPress_PC2', 'RS_E_WatTemp_PC1', 'RS_E_WatTemp_PC2', 'RS_T_OilTemp_PC1', 'RS_T_OilTemp_PC2'
  - Rules for anomaly classification:
    - If anomaly detected in a feature and:
      - RMP==0: Sensor problem if feature #0, normal if feature==0 exactly
      - RPM#0: Engine problem if feature #0, sensor problem if feature==0 exactly

--> We train a separate model with specific parameters for each feature

Label Generation

- Since the data do not have labels, we define the labels for model evaluation.
- Convention: label=1 for anomalies, else label=0
- Rules for label generation:
  - RS_InAirTemp_PC1, RS_E_InAirTemp_PC2: label=1 for value>65 or value==0, else label=0
  - RS_E_OilPress_PC1, RS_E_OilPress_PC2: label=1 for value>mean+3*std or value==0, else label=0
  - RS_E_WatTemp_PC1, RS_E_WatTemp_PC2: label=1 for value>100 or value==0, else label=0
  - RS_T_OilTemp_PC1,RS_T_OilTemp_PC2: label=1 for value>115 or value==0, else label=0
- Note: An anomaly in a feature does not mean an anomaly in the system, e.g. feature value==0 and RMP==0, then it is a normal point because the train is not running.

Label Generation

Feature	Percentage of anomalies in labels
RS_InAirTemp_PC1	0.8647%
RS_InAirTemp_PC2	0.9685%
RS_E_OilPress_PC1	3.8428%
RS_E_OilPress_PC2	3.9149%
RS_E_WatTemp_PC1	0.6333%
RS_E_WatTemp_PC2	0.6779%
RS_T_OilTemp_PC1	0.0787%
RS_T_OilTemp_PC2	0.254%

Metrics

Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

- As the labels is highly unbalanced, it does not make much sense to use only accuracy
- F1 score gives a more reliable metric for model evaluation

Results

Elliptic Envelope	Accuracy	F1
RS_E_InAirTemp_PC1	0.987098213	0.3670369342
RS_E_InAirTemp_PC2	0.98350866	0.3411241901
RS_E_OilPress_PC1	0.9712153678	0.6403171561
RS_E_OilPress_PC2	0.9922834374	0.8815363559
RS_E_WatTemp_PC1	0.9998570745	0.9857694367
RS_E_WatTemp_PC2	0.9999852829	0.9989992687
RS_T_OilTemp_PC1	1	1
RS_T_OilTemp_PC2	0.9999971698	0.9998430141
Average	0.9917431507	0.7768282945

Results

Isolation Forest	Accuracy	F1
RS_E_InAirTemp_PC1	0.9828500706	0.3037343445
RS_E_InAirTemp_PC2	0.9795259911	0.2942961106
RS_E_OilPress_PC1	0.962720215	0.5788687859
RS_E_OilPress_PC2	0.9667843939	0.3664347141
RS_E_WatTemp_PC1	0.9998570745	0.9857694367
RS_E_WatTemp_PC2	0.9999852829	0.9989992687
RS_T_OilTemp_PC1	0.9992449005	0
RS_T_OilTemp_PC2	0.9999971698	0.9998430141
Average	0.9863706373	0.5659932093

Results

Local Outlier Detector	Accuracy	F1
RS_E_InAirTemp_PC1	0.991312393	0.1129349208
RS_E_InAirTemp_PC2	0.9902052156	0.3265877958
RS_E_OilPress_PC1	0.9561711133	0.001264051284
RS_E_OilPress_PC2	0.968389971	0.0001969385015
RS_E_WatTemp_PC1	0.9942368471	0.03652708777
RS_E_WatTemp_PC2	0.9916842824	0.003189035147
RS_T_OilTemp_PC1	0.9985110275	0
RS_T_OilTemp_PC2	0.990265216	0
Average	0.9850970082	0.06008747866

Results

K-Means	Accuracy	F1
RS_E_InAirTemp_PC1	0.9913302233	0
RS_E_InAirTemp_PC2	0.988348033	0
RS_E_OilPress_PC1	0.9584417893	0
RS_E_OilPress_PC2	0.9712883872	0
RS_E_WatTemp_PC1	0.9949067574	0
RS_E_WatTemp_PC2	0.9926408927	0
RS_T_OilTemp_PC1	0.9992449005	0
RS_T_OilTemp_PC2	0.9909843719	0
Average	0.9858981694	0

Comparison

Method	Average Accuracy	Average F1
Elliptic Envelope	0.9917431507	0.7768282945
Isolation Forest	0.9863706373	0.5659932093
Local Outlier Detector	0.9850970082	0.06008747866
K-Means	0.9858981694	0

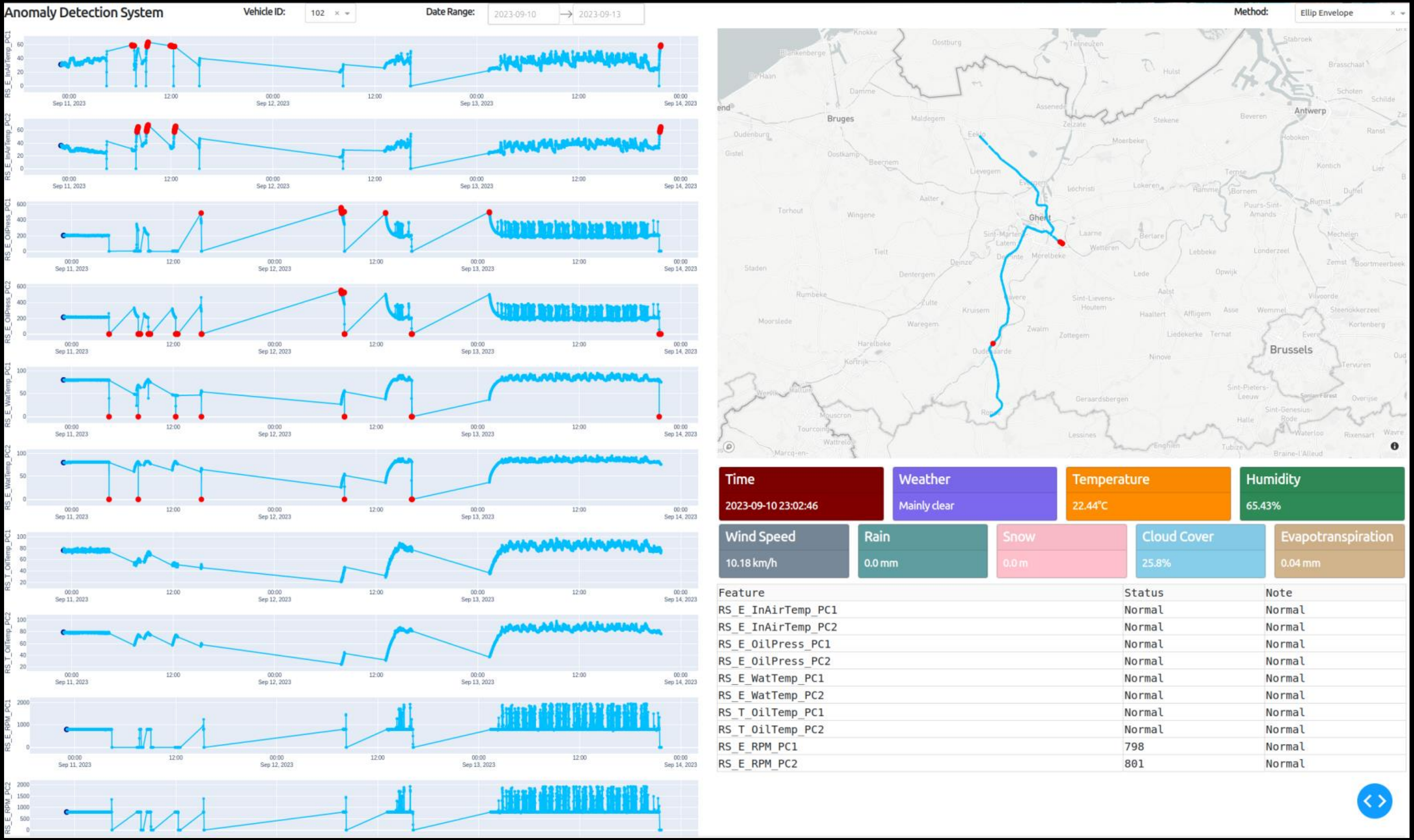
Best method: Elliptic Envelope

Unsuitable methods: Local Outlier Detector, K-Means

Dashboard

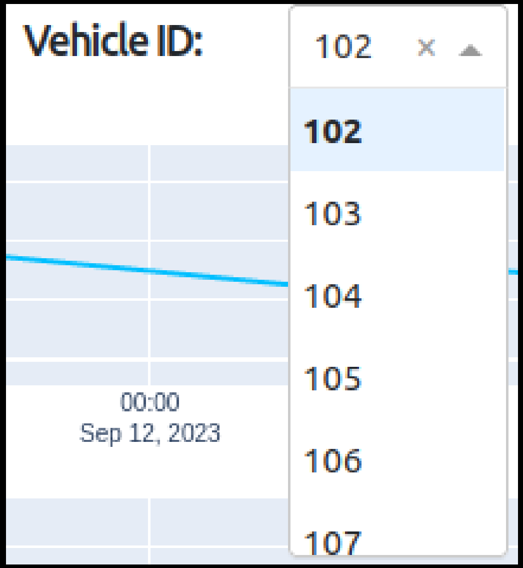
- Tools for building dashboard: **plotly** (https://plotly.com/), **dash** (https://dash.plotly.com/)
- Main dashboard functions:
  - Choose VehicleID for showing data
  - Choose start date and end date for showing data
  - Choose Anomaly Detection Method
  - Visualization of data signal with anomalies
  - A map for showing locations of data points
  - Weather data information card
  - A table for summarizing data status and note
  - Types of data note: normal, high air temperature, high oil temperature high oil pressure, high water temperature, exactly zero, other problem

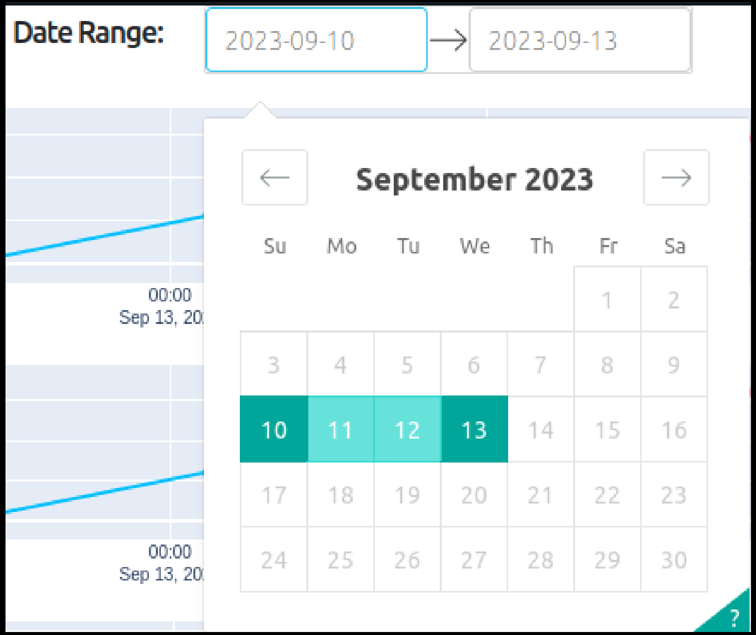
Dashboard

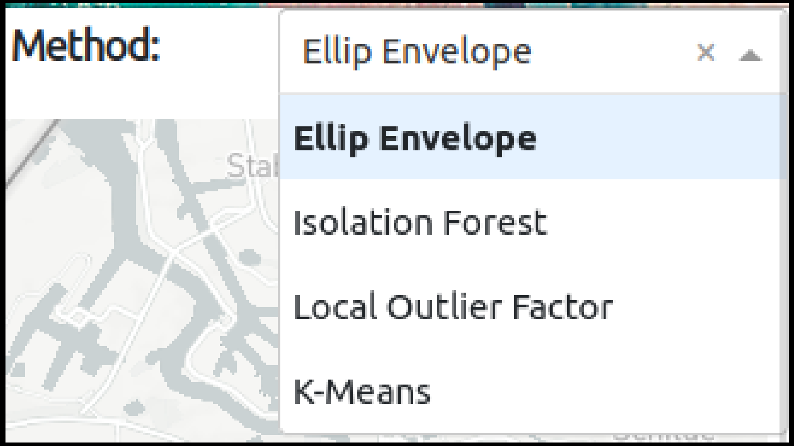


Anomaly Detection System
Vehicle ID:
102
Date Range:
2023-09-10
2023-09-13
Method:
Ellip Envelope
Time
2023-09-10 23:02:46
Weather
Mainly clear
Temperature
22.44°C
Humidity
65.43%
Wind Speed
10.18 km/h
Rain
0.0 mm
Snow
0.0 m
Cloud Cover
25.8%
Evapotranspiration
0.04 mm
Feature | Status | Note
RS_E_InAirTemp_PC1 | Normal | Normal
RS_E_InAirTemp_PC2 | Normal | Normal
RS_E_OilPress_PC1 | Normal | Normal
RS_E_OilPress_PC2 | Normal | Normal
RS_E_WatTemp_PC1 | Normal | Normal
RS_E_WatTemp_PC2 | Normal | Normal
RS_T_OilTemp_PC1 | Normal | Normal
RS_T_OilTemp_PC2 | Normal | Normal
RS_E_RPM_PC1 | 798 | Normal
RS_E_RPM_PC2 | 801 | Normal
Bruges
Ghent
Antwerp
Brussels

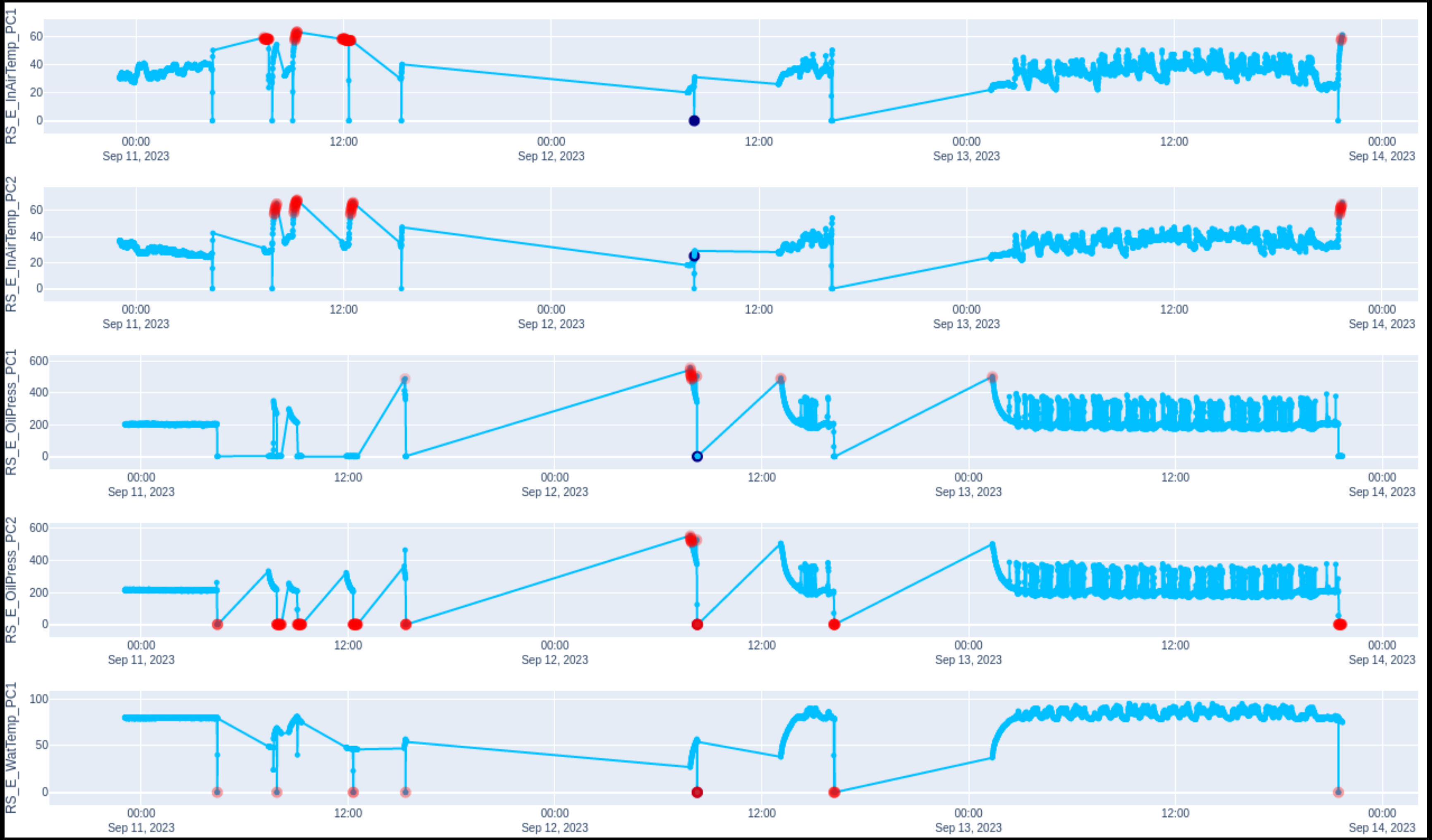
Dashboard


Vehicle ID:
102
102
103
104
105
106
107
00:00
Sep 12, 2023

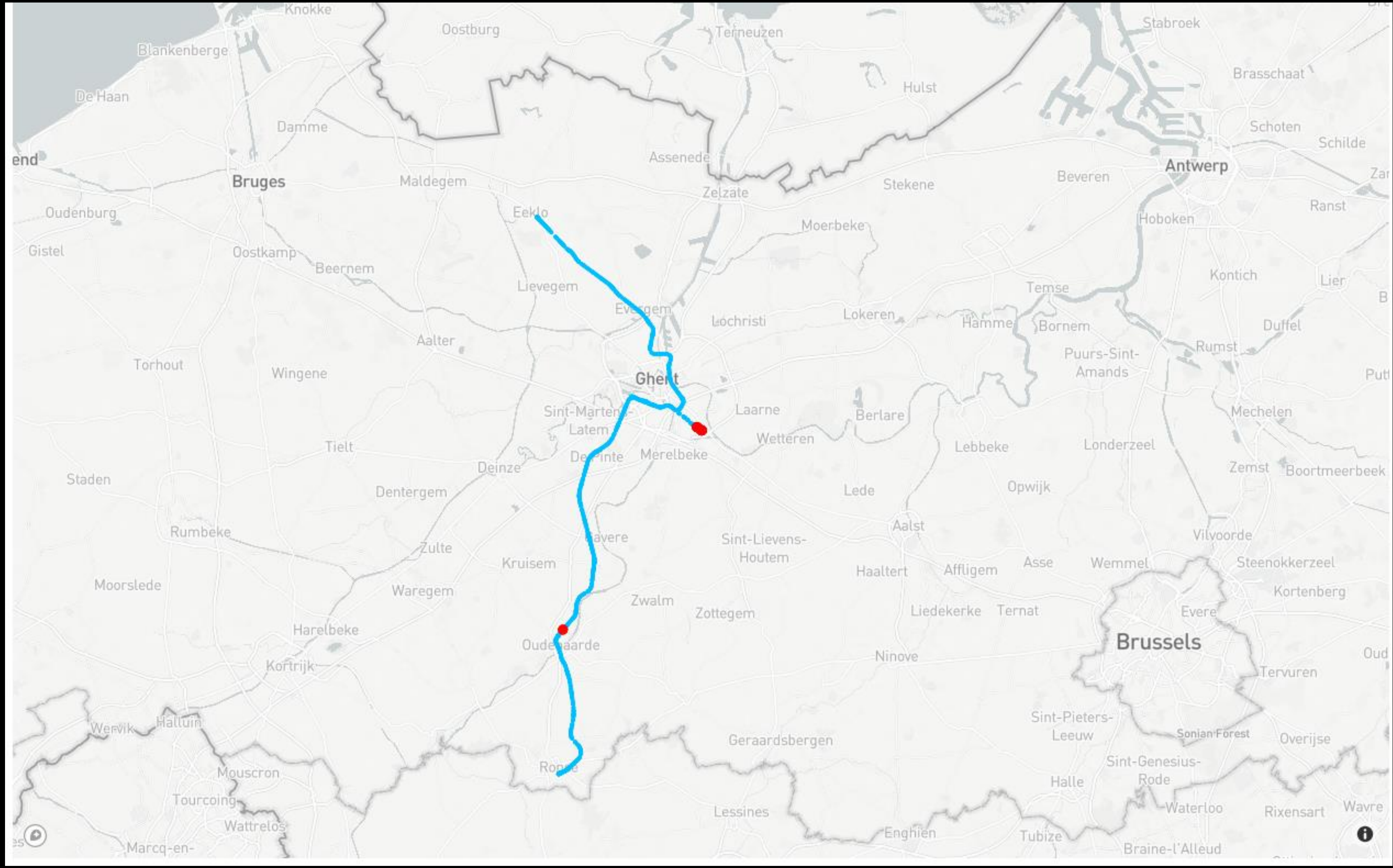

Date Range:
2023-09-10
2023-09-13
September 2023
Su Mo Tu We Th Fr Sa
1 2
3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
00:00


Method:
Ellip Envelope
Ellip Envelope
Isolation Forest
Local Outlier Factor
K-Means

Dashboard


RS_E_InAirTemp_PC1
RS_E_InAirTemp_PC2
RS_E_OilPress_PC1
RS_E_OilPress_PC2
RS_E_WatTemp_PC1
00:00 Sep 11, 2023
12:00
00:00 Sep 12, 2023
12:00
00:00 Sep 13, 2023
12:00
00:00 Sep 14, 2023

Dashboard


Knokke
Oostburg
Terneuzen
Stabroek
Blankenberge
Brasschaat
De Haan
Hulst
Damme
Schoten
Schilde
Assenede
Antwerp
Bruges
Maldegem
Zelzate
Stekene
Beveren
Oudenburg
Eeklo
Moerbeke
Ranst
Hoboken
Gistel
Oostkamp
Beernem
Kontich
Lier
Lievegem
Temse
Evergem
Lochristi
Lokeren
Hamme
Bornem
Duffel
Aalter
Rumst
Puurs-Sint-Amands
Torhout
Wingene
Ghent
Laarne
Berlare
Mechelen
Sint-Martens-Latem
Wetteren
Lebbeke
Londerzeel
Tielt
Merelbeke
Zemst
Boortmeerbeek
Deinze
Staden
Dentergem
Lede
Opwijk
Aalst
Vilvoorde
Rumbeke
Sint-Lievens-Houtem
Zulte
Asse
Wemmel
Steenokkerzeel
Kruisem
Haaltert
Affligem
Moorslede
Waregem
Kortenberg
Zwalm
Zottegem
Liedekerke
Ternat
Evere
Harelbeke
Brussels
Ninove
Kortrijk
Tervuren
Sint-Pieters-Leeuw
Wervik
Halluin
Sonian Forest
Overijse
Geraardsbergen
Sint-Genesius-Rode
Mouscron
Halle
Tourcoing
Lessines
Waterloo
Rixensart
Wavre
Wattrelos
Enghien
Tubize
Marcq-en-
Braine-l'Alleud

Dashboard

Time	Weather	Temperature	Humidity
2023-09-12 07:51:11	Partly cloudy	17.59°C	92.68%

Wind Speed	Rain	Snow	Cloud Cover	Evapotranspiration
11.18 km/h	0.0 mm	0.0 m	60.0%	0.05 mm

Feature	Status	Note
RS_E_InAirTemp_PC1	Normal	Normal
RS_E_InAirTemp_PC2	Normal	Normal
RS_E_OilPress_PC1	Abnormal	High oil pressure
RS_E_OilPress_PC2	Abnormal	High oil pressure
RS_E_WatTemp_PC1	Normal	Normal
RS_E_WatTemp_PC2	Normal	Normal
RS_T_OilTemp_PC1	Normal	Normal
RS_T_OilTemp_PC2	Normal	Normal
RS_E_RPM_PC1	802	Train is running
RS_E_RPM_PC2	797	Train is running

Conclusion

- Lots of noises, NA values in data
- Data cleaning and preprocessing is of importance in data mining
- Anomalies can be detected without pre-defined labels (unsupervised learning)
- Some methods are unsuitable for anomaly detection
- Data augmentation gives context information about data
- Interactive visualization gives a comprehensive view and deep understanding of data

References

- Detecting and preventing abuse on LinkedIn using isolation forests: https://engineering.linkedin.com/blog/2019/isolation-forest
- SK Learn Elliptic Envelope: https://scikit-learn.org/stable/modules/generated/sklearn.covariance.EllipticEnvelope.html
- SK Learn Local Outlier Detection: https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html
- Unsupervised Anomaly detection: K-Means vs Local Outlier Factor: https://towardsdatascience.com/unsupervised-anomaly-detection-on-spotify-data-k-means-vs-local-outlier-factor-f96ae783d7a7