Dhirubhai Ambani
Institute of Information and Communication Technology

# Documentation

for

# Dataset - 4

By Group -16

# PROJECT PIPELINE

## Understanding of Data (Problem Statement)

**Data Preprocessing** - checked null values, attribute types, duplicate values etc

## Data Visualization -

1. Count plot for categorical columns
2. Violin plot for numerical columns
3. Time series analysis of numerical columns (month wise, day wise, hour wise, minute wise)
4. Line plot for Daily analysis of numerical columns
5. Boxplot of numerical columns over months

## ML pipeline

- *Feature selection using correlation analysis*
- *Splitting of data*
- *Scaling of data*
- *Model Fitting and Hyperparameter Tuning*
  → Linear Regression
  → Polynomial Regression
  → Ridge Regression
  → Random Forest Regressor
- *Optimisation*
  → SGDRegressor
- *Time Series Prediction using*
  → Linear regression
  → ARIMA (Daily basis, Hourly basis)
- *Model Evaluation*
  → RMSE
  → MAE
  → R2 SCORE
  → NMSE

## Contributions :

| | |
|---|---|
| **Chinmaya (202218054)** | Data Visualization, Feature Selection, Model Fitting |
| **Riya (202218049)** | Data Visualization, Time Series Prediction (ARIMA) |
| **Swayista (202218035)** | Problem Statement, Dataset Description, Time Series Prediction (Linear Regression) |
| **Asish (202218022)** | Optimisation, Model Fitting |
| **Kashish (202001425)** | Data Preprocessing, Model Evaluation |

# 1. Dataset Description :

- The MetroPT-3 Dataset collects data collected from metro trains' Air Production Units (APU), which include pressure, temperature, motor current, and air intake data.
- An insight into the most prevalent predictive maintenance issues in the industry can be gained by examining this dataset.
- In this case, a multivariate time series derived from a multitude of analogue and digital sensors installed on the train's compressor provides a comprehensive overview of its operation.

# 2. A Summary of the Dataset and features

**Data Source** : The data is collected from analogue and digital sensors installed on the APU(Air Processing Unit) of a metro train's compressor. These sensors monitor different aspects of the compressor's operation.

**Sensors**: The dataset includes readings from the following sensors:

- Pressure Sensor:Monitors pressure levels within the APU.
- Temperature Sensor: Measures the temperature of the APU.
- Motor Current Sensor: Records the electrical current consumed by the compressor's motor.
- Air Intake Valve Sensor: Monitors the status or position of the air intake valve.

## What is APU?

- An APU, or Air Processing Unit, in the context of a metro train's compressor, refers to a component that plays a crucial role in providing clean and conditioned air for various systems within the train. The APU is responsible for filtering, cooling, and sometimes heating the air before it is

distributed to different parts of the train, ensuring a comfortable and safe environment for passengers and crew.

## Problem Statement :

On thorough analysis of the dataset, we observed that the most interesting problem was predicting motor current, for which we used two approaches.

**1. Prediction of motor current using regression techniques:**

Following the basic ML pipeline of representation, evaluation and optimization of models like linear regression, polynomial regression, ridge regression and random forest.

**2. Prediction using time series:**

This was done in two ways -

- On an hourly basis as well as daily basis using the ARIMA model.
- Prediction using linear regression wrt time

## Preprocessing:

- Import necessary libraries, including pandas for data handling.
- Mount Google Drive to access the dataset, then load it into a DataFrame.
- Identified attribute types, where only the "timestamp" column is categorical, while the rest are numerical. No encoding is needed.
- Validate for missing values and detect duplicates.
- Eliminate redundant columns.
- Transform the 'timestamp' column into a datetime format.
- Determine the count of unique values for each column.

### Observation after preprocessing :

- Some columns may be nominal/ordinal type since only two unique values for them.Getting categorical columns which are already encodedCategorical columns Numerical columns.
- 'TP2', 'TP3', 'H1', 'DV_pressure', 'Reservoirs', 'Oil_temperature', 'Motor_current' are quantitative variables.
- 'COMP', 'DV_eletric', 'Towers', 'MPG', 'LPS', 'Pressure_switch', 'Oil_level', 'Caudal_impulses' are qualitative variables. Moreover, these variables are binary in nature.

## Description of Attributes:

### Attributes in the dataset:

**1. Unnamed 0:** An unnamed index or identifier for each record in the dataset.

**2. Timestamp:** The timestamp indicating the time at which the readings were recorded.

**3. TP2:** Reading from the Pressure sensor, TP2 measures the pressure on the compressor.

**4. TP3:** Reading from the Pressure sensor, TP3 measures the pressure generated at the pneumatic panel.

**5. H1:** Reading from the Pressure sensor, H1 measures the pressure generated due to pressure drop when the discharge of the cyclonic separator filter occurs.

**6. DV_pressure:** Reading from the Pressure sensor, which measures the pressure drop generated when the towers discharge air dryers, a zero reading indicates that the compressor is operating under load.

**7. Reservoirs:** Reading related to reservoirs which has the measure of the downstream pressure of the reservoirs, which should be close to the pneumatic panel pressure (TP3).

**8. Oil_temperature:** Reading of oil temperature on the compressor.

**9. Motor_current:** Reading of motor current which has the measure of the current of one phase of the three-phase motor it presents values close to

- 0A - when it turns off,
- 4A - when working offloaded,
- 7A - when working under load and
- 9A - when it starts working.

**10. COMP:** Reading related to the electrical signal of the air intake valve on the compressor.

- it is active when there is no air intake, indicating that the compressor is either turned off or operating in an offloaded state.

**11. DV_eletric:** Reading related to electrical signal that controls the compressor outlet valve.

- it is active when the compressor is functioning under load
- inactive when the compressor is either off or operating in an offloaded state.

**12. Towers:** Reading related to the electrical signal that defines the tower responsible for drying the air and the tower responsible for draining the humidity removed from the air.

- when not active, it indicates that tower one is functioning
- when active, it indicates that tower two is in operation.

**13. MPG:** Reading related to MPG (miles per gallon). It measures the electrical signal responsible for starting the compressor under load by activating the intake valve when the pressure in the air production unit (APU) falls below 8.2 bar.
It activates the COMP sensor, which assumes the same behavior as the MPG sensor.

**14. LPS:** Reading of LPS (low pressure system) which measures the electrical signal that detects and activates when the pressure drops below 7 bars.

**15. Pressure_switch:** Reading from the pressure switch which measures the electrical signal that detects the discharge in the air-drying towers.

**16. Oil_level:** It measures the electrical signal that detects the oil level on the compressor. It is active when the oil is below the expected values.

**17. Caudal_impulses:** The electrical signal that counts the pulse outputs generated by the absolute amount of air flowing from the APU to the reservoirs.

## Data Visualization, summarizing insights about the dataset through EDA.

- Refer to Seaborn documentation to define a custom color palette.
- Create count plots for the categorical columns after encoding.
- Plot **histograms**,**box** and **violin plots** for the numerical attributes.
- Configure the style and context settings for all the visualizations.
- Generate a line plot depicting the monthly variations in Motor current with respect to the Compressor state.

### Insights from Violin Plots for Numerical columns :

**TP2 :** TP2 primarily falls within the 0 to 1 range, with occasional spikes between 8 to 11, indicating two distinct operating modes - low throttle positions and intermittent higher values.

**TP3:** TP3 remains stable, with 99% of data between 7 to 10, suggesting a consistent throttle position and specific operational mode.

**H1:** H1 exhibits two modes - one in the low-pressure range and another around 8 to 9. The system frequently operates within the latter range, with occasional lower pressure conditions.

**DV Pressure:** DV Pressure primarily operates at low values (0), with rare occurrences around 2, indicating a predominantly low-pressure system.

**Reservoirs:** Reservoir levels are stable, with around 99% of data between 7 to 10, mirroring the median at 8 to 9.

**Oil Temperature:** Oil temperature varies between 40 to 80, with a median around 65, indicating a typical operational range.

**Motor Current:** Motor Current is right-skewed, concentrated at low levels, with occasional spikes around 4 and 6, suggesting occasional fluctuations.

# Time series analysis :

1. Prepare for time-series analysis by converting the 'timestamp' column to a datetime format.
2. Extract the 'date' component for daily analysis and separate year, month, day, hour, minute, and second.
3. Create line plots to visualize the numeric column's trends over months, hours, and minutes.
4. Generate daily line plots for the numeric column.
5. Group the data by date and calculate the mean for the numeric column.
6. Display a boxplot to illustrate the numeric column's distribution across months.

## Insights from the line plot of numerical columns over each month :

**TP2:** TP2 displays an increase in February, followed by a gradual decrease from March to April. There's a slight increase in April, with a sharp rise in May, reaching its highest value by month-end. From there, it gradually decreases for the remaining months.

**TP3:**
TP3 experiences an increase in February, followed by a slight decrease in March. There's a notable drop in April, reaching its lowest point in May. Starting from June, it shows a continuous monotonic increase.

**H1:**
H1 exhibits a sharp decrease in February, followed by a slight increase in March. April sees a sudden drop, reaching its lowest point at the end of May. From there, it experiences a sharp monotonic increase for the subsequent months.

**DV Pressure Over the Months:**
DV Pressure monotonically increases from February until the end of May. There's a sudden drop in June, reaching its lowest value, and thereafter, it remains nearly constant for the subsequent months.

**Reservoirs Over the Months:**
Reservoir levels increase in February, followed by a slight decrease in March. April experiences a sudden drop, reaching its lowest point in May. Starting from June onwards, reservoir levels display a continuous monotonic increase.

**Oil Temperature Over the Months:**
Oil temperature shows a linear increase during February and then decreases in March. It gradually increases until the end of June, reaching its maximum value. Afterward, it gradually decreases during the following months.

**Motor Current Over the Months:**
Motor Current follows a similar pattern, with a linear increase in February, followed by a decrease in March. It gradually increases until the end of June, reaching its maximum value.It gradually decreases during the subsequent months.

## Insights from the line plot of numerical columns over days of each month :

**TP2 Across Days of the Month:**
TP2 exhibits a cyclic pattern, consistently reaching its maximum peak every 6 days within each month.

**TP3 Across Days of the Month:**
TP3 shows a semi-cyclic pattern, with minimum values occurring approximately every 6 days. However, some fluctuations are present.

**H1 Across Days of the Month:**
H1 displays a cyclic pattern characterized by a sharp drop every 6 days of the month. Similar to TP3, there are also some fluctuations.

**DV Pressure Across Days of the Month:**
DV Pressure demonstrates an interesting pattern. It increases on every 7th day of the first week, experiences sharp fluctuations in the 3rd week, possibly due to malfunction, and tends to reach minimal values in the last week of the month.

**Reservoirs Across Days of the Month:**
Reservoir levels follow a semi-cyclic pattern, reaching their minimum in approx. every 6 days. However, like TP3, there are some accompanying fluctuations.

**Oil Temperature Across Days of the Month:**
Oil temperature exhibits a fluctuating trend, with values increasing in the first week and fluctuating until the end of the month. There's a notable sharp rise towards the end of the month.

**Motor Current Across Days of the Month:**
Motor Current reaches its highest value at the beginning of the month and its lowest value in the last week. There are fluctuations in between, reflecting dynamic behavior over the course of the month.

## Insights from the line plot of numerical columns over hours within a day :

**TP2 Across Hours of the Day:**
TP2 follows a repeating pattern every 5-6 days. At the end of each cycle, the last day hits its highest TP2 value, which is around 2.2.

**TP3 Across Hours of the Day:**
TP3 values constantly change, going up and down throughout the day. The highest values are typically observed during the early hours of the day.

**H1 Across Hours of the Day:**
Similar to TP2, H1 displays a cyclic pattern every 5-6 days. However, the last day of each cycle usually has the lowest H1 value.

**DV Pressure Across Hours of the Day:**
DV Pressure remains relatively constant around -0.01 to 0 for most of the day. A cyclic pattern is noticeable, with the last hour of each cycle experiencing the highest pressure.

**Reservoir Across Hours of the Day:**
Reservoir levels vary significantly throughout the day, with multiple peaks and lows. The column displays a dynamic pattern.

**Oil Temperature Across Hours of the Day:**
Oil temperature steadily rises and falls during the day. The highest oil temperature is typically observed towards the end of the month.

**Motor Current Across Hours of the Day:**
Motor current exhibits a consistent zig-zag pattern, constantly going up and down. The highest current values are typically observed at the beginning and end of each month.

## Insights from the line plot of numerical columns over minutes within an hour :

**TP2 Across Minutes per Hour:**
TP2 reaches its highest value at the beginning of each hour. As the hour progresses, TP2 mostly remains within the range of 1.33 to 1.38.

**TP3 Across Minutes per Hour:**
TP3 initially decreases, then steadily increases, peaking at 7 minutes into the hour, after which it gradually falls.

**H1 Across Minutes per Hour:**
H1 starts with a decrease, then steadily increases, remaining mostly within the range of 7.55 to 7.65.

**DV Pressure Across Minutes per Hour:**
DV Pressure continuously decreases until six minutes into the hour, followed by a zig-zag trend with a slight increase towards the end.

**Reservoir Across Minutes per Hour:**
Initially, reservoir levels decrease, reach a peak at 7 minutes into the hour, and then steadily decline.

**Oil Temperature Across Minutes per Hour:**
Oil temperature decreases initially, peaks at 7 minutes into the hour, and then gradually falls.

**Motor Current Across Minutes per Hour:**
Motor current constantly decreases throughout the hour, exhibiting a downward trend.

## Insights from the line plot for numerical columns over each date:

**TP2 Daily Analysis:**
TP2 remains relatively constant for most of the time but reaches its maximum value once a month.

**TP3 Daily Analysis:**
TP3 shows overall stability, except between April and June when it reaches its minimum value multiple times during those months.

**H1 Daily Analysis:**
Similar to TP3, H1 maintains stability for most of the time but experiences multiple occurrences of reaching its minimum value between April and July.

**DV Pressure Daily Analysis:**
DV Pressure hovers around zero for the majority of the year but occasionally increases for a few days. Between March and June, there are instances of sudden increases, followed by drops the next day.

**Reservoir Daily Analysis:**
Reservoir performance resembles TP3 and H1, with overall stability but several instances of reaching its minimum value between April and July.

**Oil Temperature Daily Analysis:**
Oil temperature exhibits a continuous zig-zag pattern, constantly increasing and decreasing over the observed period.

**Motor Current Daily Analysis:**
Motor current follows a pattern similar to oil temperature, with a complete zig-zag pattern. Most values fall within the range of 1 to 2.

## Insights from the box plot for numerical columns over months:

**TP2 Month-Wise Box Plots:**
The boxplot for TP2 shows a very large range in the month of June and is right-skewed, with no outliers compared to the remaining months, which have a much smaller range and lots of outliers.

**TP3 Month-Wise Box Plots:**
Similar box plots can be observed for all the months, except for February and September, which don't have outliers like the other months. The median is approximately around 9.

**H1 Month-Wise Box Plots:**
All the months exhibit similar types of box plots, except for June, which is left-tailed. However, the median of all the plots is more or less around 9.

**DV Pressure Month-Wise Box Plots:**
The median of DV pressure is around zero for all months. The month of September doesn't have outliers like the rest of the months.

**Reservoir Month-Wise Box Plots:**
All the box plots are similarly placed. The median for all the months is around 9, while February and September don't have outliers.

**Oil Temperature Month-Wise Box Plots**
Box plots for oil temperature are differently placed for each month. The median for all the months ranges from 56 to 65. All months have outliers, except for September.

**Motor Current Month-Wise Box Plots:**
All the box plots are right-tailed. The median value for the box plots is around 3.8 for all months.

# Regression Analysis:

Regression problems involve predicting a continuous numeric value based on input features. Given the attributes in the metro train dataset, here are a few potential regression problems:

**1. Predict Motor Current:** Given sensor readings such as 'TP2', 'TP3', 'H1', and 'Oil_temperature', predict the 'Motor_current' value, which represents the electrical current consumed by the motor. This could be valuable for monitoring motor health and efficiency.

**2. Oil Temperature Prediction:** Use attributes like 'TP2', 'TP3', 'H1', and 'Motor_current' to predict the 'Oil_temperature'. Accurate prediction of oil temperature is crucial for maintaining optimal compressor operation and preventing overheating.

**3. Air Pressure Prediction:** Given sensor readings like 'TP2', 'TP3', 'H1', and 'Oil_temperature', predict the 'DV_pressure' or other pressure-related attributes. Accurate pressure prediction is essential for maintaining safe and efficient compressor operation.

**4. Estimating Reservoir Levels ('Reservoirs'):** Predict the levels of the 'Reservoirs' using other attributes. This could help in maintaining appropriate fluid levels and preventing issues due to under- or overfilling.

# Selecting the Most Interesting Problem:

Choosing the most interesting problem depends on goals, domain expertise, and the potential impact of solving the problem. However, one problem that stands out is predicting the 'Motor_current'. Motor current consumption is a crucial indicator of the motor's health, efficiency, and potential issues. By accurately predicting motor current, you could:

- Identify abnormal motor behavior or impending failures.
- Optimize energy consumption by understanding how motor current changes in different operating conditions.
- Schedule maintenance more effectively, preventing unexpected breakdowns.
- Enhance passenger safety by addressing potential motor-related risks.
- Solving this problem would likely have a direct impact on the overall reliability, efficiency, and safety of the metro train system.

### Reasons for Choosing this Problem:

1. **Maintenance Planning:** The prediction of motor current offers valuable insights for maintenance planning. By identifying abnormal current patterns, maintenance actions can be scheduled proactively, reducing the risk of unexpected breakdowns and minimizing operational disruptions.

2. **Data Availability:** Motor current data is often readily available in such systems, making it a practical choice for predictive modeling.This availability ensures that sufficient data can be collected for accurate predictions.

3. **Real-time Monitoring:** Accurate motor current predictions can support real-time monitoring, allowing for immediate responses to any deviations from expected current levels.

## An end-to-end Machine Learning pipeline :

- Your pipeline should include components for dataset preprocessing, transformation, regression model building hyperparameter tuning, grid search or optimization, and evaluation.
- Report results on the regression models with hyperparameter tuning, and report the best hyperparameter values.
- Report results using at least two relevant evaluation metrics like RMSE,MAE.

- Compare results for different models and give the reasoning for that.

# Dataset Preprocessing :

- prepare the data for modeling

Major steps for preprocessing are :

**1. Handle Missing Values:** Checking for missing values and use appropriate measures to clean them. Already done in Task-01.No missing values are present in the dataset.

**2. Feature Selection:** Decide which features (attributes) to include as input (X) for predicting 'Motor_current'.

**3. Train-Test Split:** Split the data into training and testing sets to evaluate the model's performance.

**4. Scaling/Normalization:** Scale or normalize the features to ensure they are on a similar scale.

## Conducting Feature Selection through Correlation Analysis:

1. Begin by creating a correlation matrix.
2. Visualize the correlation matrix with a heatmap.
3. Examine the distributions of Motor current concerning each encoded binary attribute for insights.
4. Configure the style and context settings for the visualizations.
5. Plot line graphs depicting Motor current against encoded categorical columns.
6. Prioritize Motor current as the target variable.
7. Assess correlations with the 'Motor_current' target variable.
8. Print the correlation values with 'Motor_current.'
9. Select relevant features.
10. Verify the mutual correlations among the chosen features.

**Insights for Violin plots for MOTOR CURRENT wrt Categorical Columns:**

### COMP with Respect to Motor Current:

- For class 0 category: The median motor current is around 6, with the majority of data falling between 4 and 6. There are outliers near 0.
- For class 1 category: The median motor current is near 0, and it closely resembles the distribution of motor current when plotted independently.

### DV ELECTRIC with Respect to Motor Current:

- For class 0 category: The median motor current is near 0, matching the distribution of motor current when plotted independently.
- For class 1 category: The median motor current is around 6, with most data between 4 and 6, and outliers near 0.

### TOWERS with Respect to Motor Current:

- For class 0 category: The median motor current is around 6, and the majority of data falls between 4 and 6, with a few outliers near 0.
- For class 1 category: The median motor current is near 0, and it resembles the distribution of motor current when plotted independently, with very few values around 6.

### MPG with Respect to Motor Current:

- For class 0 category: The median motor current is around 6, with most data between 4 and 6 and outliers near 0.
- For class 1 category: The median motor current is near 0, matching the distribution of motor current in the range of 0 to 4 when plotted independently.

### LPS with Respect to Motor Current:

- For class 0 category: The interquartile range spans from 0 to 5, with a median at 4 and almost equal concentration at the tails.
- For class 1 category: The median motor current is near 0, resembling the distribution of motor current when plotted independently, with very few values around 6.

### PRESSURE SWITCH with Respect to Motor Current:

- For class 0 category: The median motor current is around 0, and it closely matches the distribution of motor current when plotted independently.

- For class 1 category: The interquartile range spans from 0 to 5, with a median near 0, similar to the independent motor current distribution.

**OIL LEVEL with Respect to Motor Current:**

- For class 0 category: The median motor current is near 4, and it matches the distribution of motor current when plotted independently.
- For class 1 category: The median motor current is near 0, closely resembling the distribution of class 0 with more concentration near 0.

**CAUDAL IMPULSES with Respect to Motor Current:**

- For class 0 category: The median motor current is near 4, and it matches the distribution of motor current when plotted independently.
- For class 1 category: The median motor current is near 0, closely resembling the distribution of class 0 with more concentration near 0.

# Model Building and Evaluation :

## Linear Regression :

1. Train the model using the training dataset.
2. Make predictions on the training set.
3. Generate predictions on the validation set.
4. Perform predictions on the test set.
5. Calculate performance metrics for the training dataset.
6. Compute performance metrics for the validation dataset.
7. Determine performance metrics for the test dataset.
8. Display and print the performance metrics obtained.

## Polynomial Regression :

1. Generate Polynomial Features.
2. Transform the input features using polynomial expansion.

3. Fit the model with the polynomial features.
4. Make predictions based on the model.
5. Evaluate the model's performance.

## Ridge Regression:

1. Train the model using Ridge training.
2. Make predictions on the training, validation and test data.
3. Evaluate the model.
4. Create a LassoCV model instance.
5. Train the model on the scaled training data and evaluate the model.

## Random Forest Regression:

1. Create a random forest regression instance.
2. Train the model and make predictions on the training, validation and test data.
3. Evaluate the model.
4. Find the number of trees and the number of features to be considered at every split.
5. Define the maximum number of levels in the tree.
6. Define the minimum number of samples required to split the node and minimum samples required at each leaf node.
7. Define the method of selecting samples for training each tree.
8. Create the random grid.
9. Make predictions on the validation and test data and evaluate the model.

## SGD Regressor with hyperparameter tuning :

1. Construct a parameter grid for hyperparameter optimization.
2. Initialize the SGDRegressor model.

3. Set up a GridSearchCV object.
4. Train the model using GridSearchCV.
5. Retrieve the best hyperparameters from the search.
6. Create a new SGDRegressor with the optimal hyperparameters.
7. Fit the model to the entire training dataset.
8. Generate predictions using the trained model.
9. Evaluate the model's performance.

**Model Performance :**

```
1 model_performance
```

1 to 9 of 9 entries Filter

| Models for Regression | RMSE Train | RMSE Validation | RMSE Test | MAE Train | MAE Validation | MAE Test | R-2 Score Train | R-2 Score Validation | R-2 Score Test |
|---|---|---|---|---|---|---|---|---|---|
| Linear Regression | 1.1885536885409353 | 0.9651519491925404 | 0.73348660736383 | 1.188364562000739 | 0.9657562118929405 | 0.733420968267177 | 1.1796046842160923 | 0.9617893865596869 | 0.7371741314685589 |
| Linear Regression Scaled | 0.5162493512210644 | 0.4192146071348753 | 0.7334866073638298 | 0.5161672039402 | 0.4194770691759731 6 | 0.7334209682671768 | 0.5123623432370614 | 0.4177540750660364 | 0.7371741314685589 |
| Polynomial Regression | 0.9631921339266704 | 0.726676640215803 | 0.8249720993175069 | 0.9610115533087608 | 0.7266164653683 | 0.8256653078557339 | 0.9617580178420833 | 0.7261985408040456 | 0.8252863799187846 |
| Polynomial Regression Scaled | 0.9631921339266906 | 0.7266766341601845 | 0.8249720993174996 | 0.961011553885944 | 0.7266164598243874 | 0.8256653076463232 | 0.9617580177277406 | 0.7261985349950157 | 0.8252863799603277 |
| Ridge Regression | 1.1885536885437922 | 0.9651521386984181 | 0.7334866073625489 | 1.1883645502833051 | 0.9657563904493766 | 0.7334209735241869 | 1.1796046685542354 | 0.9617895679569697 | 0.7371741384477463 |
| Ridge Regression Scaled | 0.4424285474369935 | 0.17672280613213742 | 0.9599432348926844 | 0.594810223055524 | 0.25007289442665326 | 0.935548575885609 | 0.4970446192846836 | 0.20792569549873954 | 0.950584611062906 |
| Random Forest | 0.46185977431442293 | 0.18392330844198954 | 0.9563474248996436 | 0.4934723730057509 | 0.21410776354762473 | 0.9556389969847388 | 0.5210243839057878 | 0.22265822928190948 | 0.9457015338094531 |
| Random Forest Scaled | 0.13496110050734597 | 0.044969332786408946 | 0.981785501349846 | 0.31428658060319786 | 0.11179520156515203 | 0.9011681674939043 | 0.31536429077263267 | 0.11220157319342741 | 0.9004276553539825 |
| SGD Regressor | 1.188553779441252 | 0.9651939210772285 | 0.7334865665980608 | 1.188362394406524 | 0.9657956414382756 | 0.7334219407543388 | 1.179601788133078 | 0.9618291453348738 | 0.7371754220103978 |

The "Random Forest Regressor with Scaled data" performs the best among the models. It has the lowest RMSE and MAE, indicating accurate predictions. Feature scaling enhances its performance.

# Time Series prediction of motor current using linear regression :

Here we use time step features in our model because given dataset is time series. We predict the motor current using Time step features and other highly correlated features.Split the data into three states training, Validation and testing over the period of months. We apply the Lasso Regression model which uses L1 norm to regularized the parameters. Train the model on training data, predict on validation data then evaluate on test data.

# DAILY  BASIS [MOTOR CURRENT]

### ARIMA Model Diagnostic Analysis:

## 1. Standard Residual Plot:

- The standard residual plot displays the residuals over time, comparing them to the horizontal line at zero. In our analysis, we observe a generally horizontal line, which is a positive sign. This indicates that the ARIMA model has successfully captured the underlying trend.

## 2. Histogram plus Estimated Density:

- The estimated density curve exhibits a peak that is slightly higher than a typical normal distribution. This suggests that the residuals might have a slightly heavier tail compared to a perfect normal distribution.

## 3. QQ Plot (Quantile-Quantile Plot):

- The QQ plot is a graphical tool for seeing the normality of the residuals. The QQ plot displays a linear, nearly straight line with minimal deviations from quantiles.

## 4. Correlogram:

- The correlogram, depicting the autocorrelation and partial autocorrelation functions, helps identify any remaining patterns or correlations in the residuals. In our analysis, all data points fall within the shaded region. This indicates that there are no significant autocorrelations left in the residuals.

The ARIMA model's diagnostic analysis indicates that it generally performs well in capturing the underlying time series patterns.

# ARIMA Model PREDICTION

The visual representation of the ARIMA model's predictions shows that the "ARIMA_Pred" curve closely follows the "Test" data, showcasing a good fit to the data, i.e., the ARIMA model's predictions coincide with the actual testing data points, illustrating its ability to capture both short-term fluctuations and long-term trends.

The Root Mean Squared Error (RMSE) of 846.159, obtained from our ARIMA model, states that  a lower RMSE indicates better predictive performance

The NMSE of 0.0499 suggests that our model's predictions are very close to the actual values.The low NMSE indicates that our model reliably forecasts future data points.

# HOURLY BASIS [MOTOR CURRENT]

**ARIMA Model Diagnostic Analysis:**
Standard Residual Plot: The ARIMA model effectively captures the data's trend and seasonality, but a zig-zag pattern hints at unaccounted variability.

Histogram + Estimated Density: Residuals have a slightly heavier tail than a perfect normal distribution, suggesting room for improvement in modeling extreme values.

QQ Plot (Quantile-Quantile): Residuals closely follow a normal distribution with minor deviations at both ends, indicating potential adjustments for extreme values.

Correlogram (ACF/PACF Plot): Most residuals show no significant autocorrelation, with a single outlier. Minor discrepancies at the plot's edges hint at minor refinements for better performance.

**ARIMA Model PREDICTION**
The ARIMA model appears to provide a reasonably good fit to the test data. Although it's difficult to get exact results from very large datasets due to the large amount of data(hourly basis), but we can draw conclusion based RMSE and NMSE as :

1.RMSE (Root Mean Squared Error): The RMSE value of 111.7589 indicates that, on average, the model's predictions deviate from the actual values by approximately 112 units. This suggests that the model's predictive accuracy is relatively good.

2. NMSE (Normalized Mean Squared Error): With an NMSE of 0.1321, the model exhibits relatively low prediction errors, indicating that it explains a significant portion of the variance in the data.