Data Collection and Preprocessing Phase

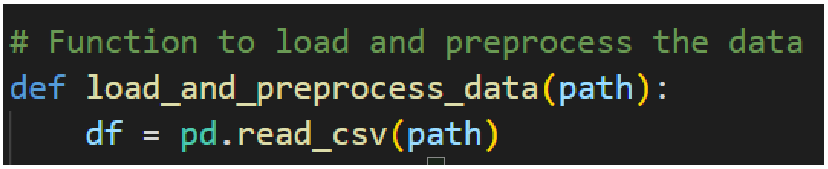
Date	9th July 2024
Team ID	SWTID1720449665
Project Title	Predicting The Energy Output Of Wind Turbine Based On Weather Condition
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

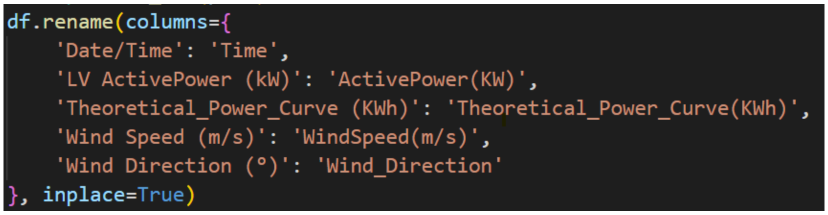
Section	Description
Data Overview	We have a data of LV Active power, Theoritcal wind curve, Wind direction and wind speed from 1st Jan 2018 to 31st Dec 2018 for every 10 mins.
Univariate Analysis	In my wind turbine energy prediction project, univariate analysis involves examining a single variable, such as wind speed, using histograms for visualizing frequency distribution, and calculating summary statistics like mean and standard deviation. Box plots help identify outliers, providing insights crucial for building an effective predictive model.
Bivariate Analysis	In my wind turbine energy prediction project, bivariate analysis examines the relationship between two variables, like wind speed and active power output. This helps identify correlations and patterns using scatter plots and correlation coefficients, providing insights into how changes in one variable affect the other, which is crucial for accurate predictions.
Multivariate Analysis	In my wind turbine energy prediction project, multivariate analysis explores how combinations of variables like wind speed, direction, and temperature collectively influence energy output. It uses techniques such as multivariate regression or PCA to uncover complex relationships and enhance predictive accuracy by understanding interconnected factors affecting

<table>
<tr><td></td><td>turbine performance.</td></tr>
<tr><td>Outliers and Anomalies</td><td>In my wind turbine energy prediction project, outliers and anomalies are significant deviations from the dataset. They can skew analyses and predictions if not addressed properly. Detecting and handling these data points is crucial for accurate modeling and reliable predictions for turbine performance based on weather conditions.</td></tr>
<tr><td colspan="2"><b>Data Preprocessing Code Screenshots</b></td></tr>
<tr><td>Loading Data</td><td>



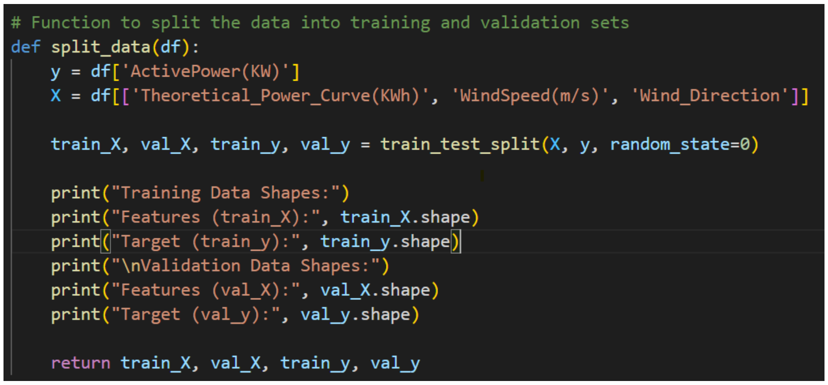
```
# Function to load and preprocess the data
def load_and_preprocess_data(path):
    df = pd.read_csv(path)
```

</td></tr>
<tr><td>Handling Missing Data</td><td>nill</td></tr>
<tr><td>Data Transformation</td><td>



```
df.rename(columns={
    'Date/Time': 'Time',
    'LV ActivePower (kW)': 'ActivePower(KW)',
    'Theoretical_Power_Curve (KWh)': 'Theoretical_Power_Curve(KWh)',
    'Wind Speed (m/s)': 'WindSpeed(m/s)',
    'Wind Direction (°)': 'Wind_Direction'
}, inplace=True)
```

</td></tr>
<tr><td>Feature Engineering</td><td>



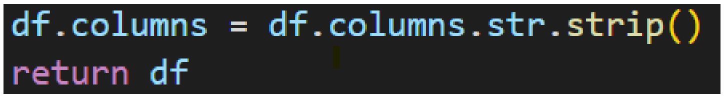
```
# Function to split the data into training and validation sets
def split_data(df):
    y = df['ActivePower(KW)']
    X = df[['Theoretical_Power_Curve(KWh)', 'WindSpeed(m/s)', 'Wind_Direction']]

    train_X, val_X, train_y, val_y = train_test_split(X, y, random_state=0)

    print("Training Data Shapes:")
    print("Features (train_X):", train_X.shape)
    print("Target (train_y):", train_y.shape)
    print("\nValidation Data Shapes:")
    print("Features (val_X):", val_X.shape)
    print("Target (val_y):", val_y.shape)

    return train_X, val_X, train_y, val_y
```

</td></tr>
<tr><td>Save Processed Data</td><td>



```
df.columns = df.columns.str.strip()
return df
```

</td></tr>
</table>