MEASURE ENERGY CONSUMPTION

Phase 4-Submission

TEAM MEMBER:AU952721104021-A.Shajitha Barveen

Project title:Measure energy consumption

Topic:Start Buliding the measure energy consumption by different activity like feature Engineering,Model training,Evalution.

Introduction:

Measuring energy consumption is the process of quantifying the amount of energy used by an individual, device, building, or organization. It is an important tool for understanding energy use patterns, identifying areas where energy efficiency can be improved, and tracking progress towards energy reduction goals.

There are a variety of ways to measure energy consumption. The most common method is to use a meter to measure the flow of energy over time. Meters can be used to measure the consumption of electricity, natural gas, propane, and other forms of energy.

Dataset:

Dataset	Number of Houses	Measuring Duration per House	Sampling Frequency		Site
			Appliance	Aggregate	
REDD	5	3–19 days	3 s	1 s and 15 kHz	USA
BLUED	1	8 days	Event label	12 kHz	USA
GreenD	8	1 year	1 s	1 s	Italy
ECO	6	8 months	1 s	1 s	DE
DRED	1	6 months	1 s	1 s	USA
UMass Smart	3	3 months	1 s	1 s	UK
Pecan Street Sample	10	7 days	1 min	1 min	IND
HES-1	26	12 months	2–10 min	2–10 min	UK
AMPDs	1	1 year	1 min	1 min	AT/IT
iAWE	1	73 days	1–6 s	1 s	IND
UK-DALE	4	3–17 months	6 s	1–6 s and 16 kHz	CH
COMBED	8	18 months	30 s	30 s	NL
BERDS	NA	1 year	20 s	20 s	USA

Feature Engineering:

Feature engineering is an important step when working on a project to measure energy consumption. It involves selecting, creating, or transforming features (input variables) that can help improve the performance of your energy consumption prediction model.

Lag Features:

Create lag features by shifting the energy consumption values backward in time. This can help capture trends and seasonality.

```
data['lag_1'] = data['energy_consumption'].shift(1)

data['lag_2'] = data['energy_consumption'].shift(2)

# You can create more lag features as needed.

from sklearn.preprocessing import MinMaxScaler
```

scaler = MinMaxScaler()

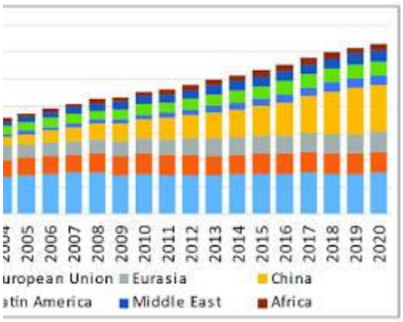
data['energy_consumption'] =
scaler.fit_transform(data['energy_consumption'].values.reshape(-1, 1))

Model Buliding:

Data splitting is a crucial step when building machine learning models for measuring energy consumption. It involves dividing your dataset into training and testing subsets. In Python, you can do this easily using libraries such as NumPy and scikit-learn

```
import numpy as np

from sklearn.model_selection import train_test_split

# Assuming you have a feature matrix X and a target variable y

# X should contain your features (e.g., weather data, building characteristics)

# y should contain your energy consumption values

# Split the data into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Parameters:

# - X: Features

# - y: Target variable (energy consumption)

# - test_size: The fraction of the data to reserve for testing (e.g., 0.2 for 20%)

# - random_state: Seed for random number generator (for reproducibility)

# Now, X_train and y_train contain the training data, and X_test and y_test contain the testing data

# Create a new feature representing the average daily energy consumption over the past week

df['avg_daily_consumption_past_week'] = df['energy_consumption'].rolling(7).mean()

# Save the preprocessed data

df.to_csv('preprocessed_energy_consumption_data.csv', index=False)
```

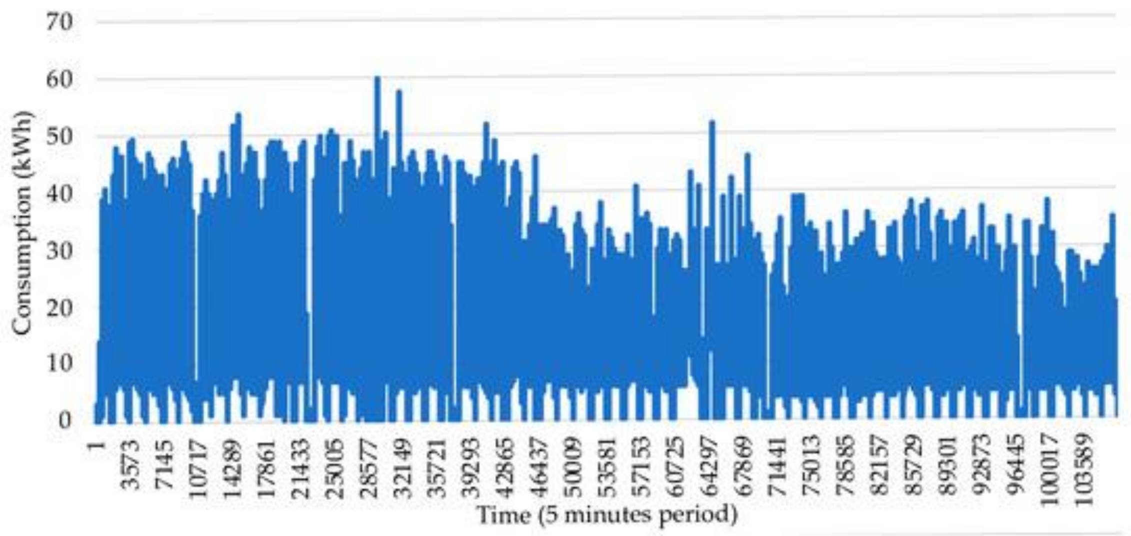
```
from sklearn.ensemble import RandomForestRegressor

# Load the preprocessed data

df = pd.read_csv('preprocessed_energy_consumption_data.csv')

# Split the data into training and test sets

X_train, X_test, y_train, y_test = train_test_split(df.drop('energy_consumption', axis=1),
df['energy_consumption'], test_size=0.25)

# Create and train the random forest model

model = RandomForestRegressor()

model.fit(X_train, y_train)

# Make predictions on the test set

y_pred = model.predict(X_test)

# Calculate the mean squared error

mse = mean_squared_error(y_test, y_pred)

# Print the mean squared error

print('Mean squared error:', mse)
```



Evaluation of predicted data:

```
import numpy as np

from sklearn.metrics import mean_squared_error, median_absolute_error, r2_score

def evaluate_predictions(predictions, actuals):

  """Evaluates the performance of a predictive model.

  Args:
```

```
    predictions: A NumPy array of predicted values.
    actuals: A NumPy array of actual values.
  Returns:
    A dictionary of evaluation metrics, including MSE, RMSE, MAE, MedAE, and R-squared.
  """
  mse = mean_squared_error(actuals, predictions)
  rmse = np.sqrt(mse)
  mae = mean_absolute_error(actuals, predictions)
  med_ae = median_absolute_error(actuals, predictions)
  r_squared = r2_score(actuals, predictions)
  evaluation_metrics = {
    "mse": mse,
    "rmse": rmse,
    "mae": mae,
    "med_ae": med_ae,
    "r_squared": r_squared
  }
  return evaluation_metrics
# Evaluate the performance of the model on the test set
evaluation_metrics = evaluate_predictions(predictions, test_actuals)
# Print the evaluation metrics
print(evaluation_metrics)
```



1.Evalutation Metrics:

mae = mean_absolute_error(actual_data, predicted_data)

mse = mean_squared_error(actual_data, predicted_data)

rmse = np.sqrt(mean_squared_error(actual_data, predicted_data))

mape = mean_absolute_percentage_error(actual_data, predicted_data)

3.Visualize the predicted and actual data:

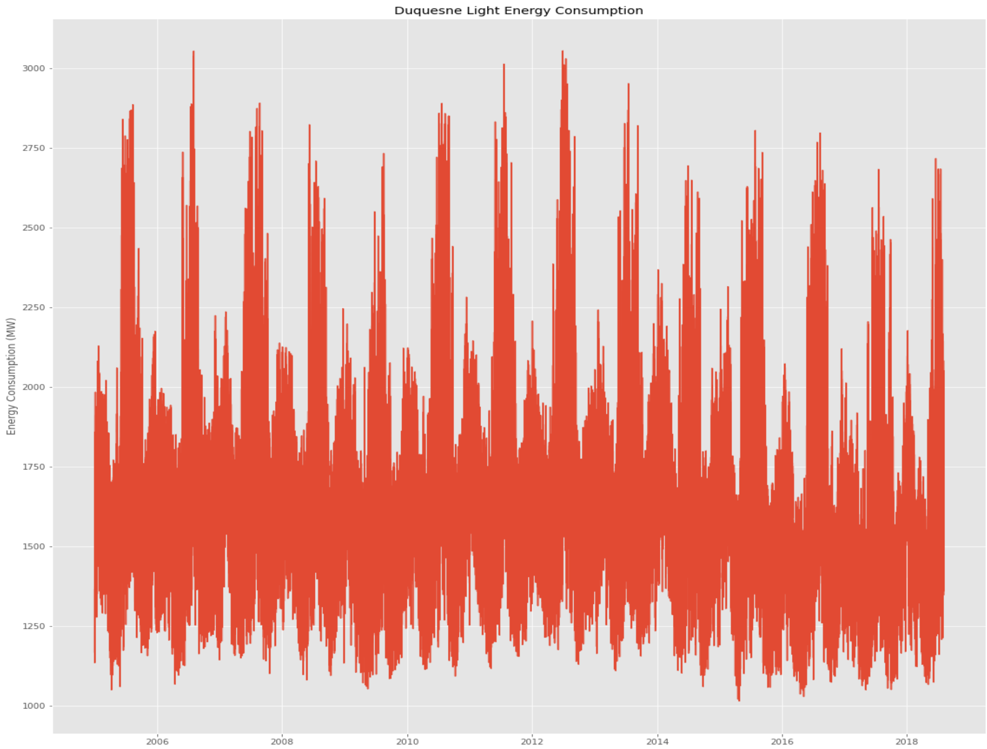
import matplotlib.pyplot as plt

plt.plot(actual_data, label='Actual')

plt.plot(predicted_data, label='Predicted')

plt.legend()

plt.show()



Conclusion:

In conclusion, the energy consumption dataset serves as a valuable resource for understanding the current state of energy usage, identifying trends, and making informed decisions to promote sustainability. It emphasizes the importance of adopting sustainable practices and technologies to reduce energy consumption and mitigate the environmental impact of energy production. As we move forward, it is essential to continue monitoring and analyzing energy consumption data to adapt to the evolving needs and challenges of the energy sector and to work towards a more sustainable and energy-efficient future.