MEASURE ENERGY CONSUMPTION

A group of power lines with lines connecting

Description automatically generated

# import the libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

# customize the style

pd.options.display.float\_format = '{:.5f}'.format

pd.options.display.max\_rows = 12

# load the data

filepath = '../input/hourly-energy-consumption/PJME\_hourly.csv'

df = pd.read\_csv(filepath)

print("Now, you're ready for step two")

Step 2 - Explore the data

To better understand the data, I need to create a graph to see the change in PJM Energy over time.

# turn data to datetime

df = df.set\_index('Datetime')

df.index = pd.to\_datetime(df.index)

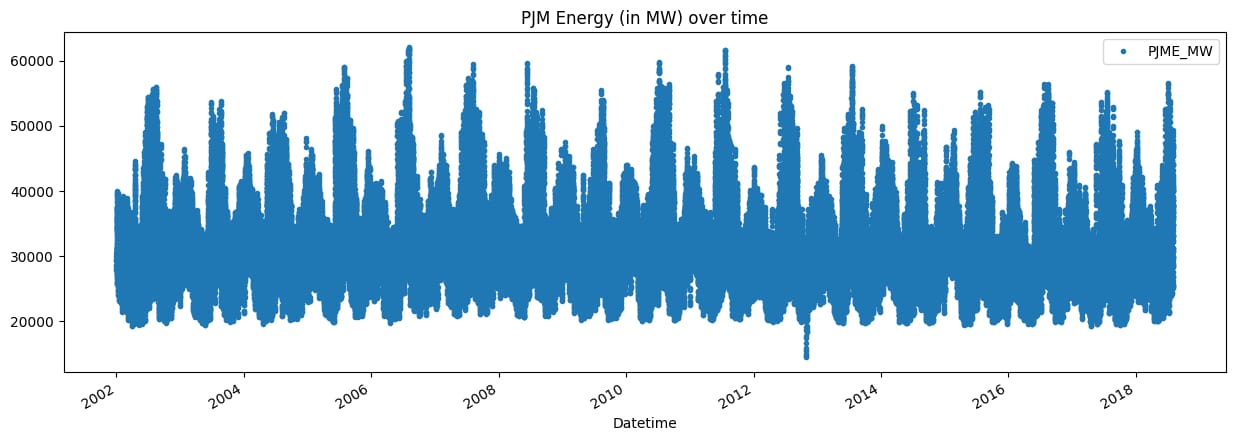
# create the plot

df.plot(style='.',

figsize=(15, 5),

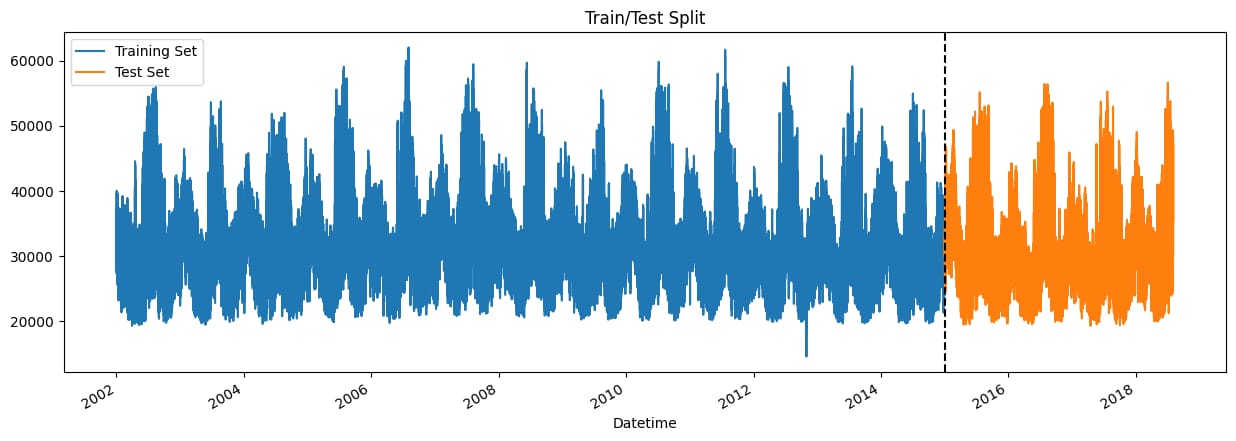
title='PJM Energy (in MW) over time')

plt.show()



Step 3 - split the data

Everything prior to January 2015 will be our training data and keep our test data as the following dates.



# train / test split

train = df.loc[df.index < '01-01-2015']

test = df.loc[df.index >= '01-01-2015']

Step 4 - Feature engineering

We're going to create some time features using the Datetime index. After that, we'll explore the distributions of Hourly and Monthly megawatt usage.

# feature creation

def create\_features(df):

df = df.copy()

df['hour'] = df.index.hour

df['dayofweek'] = df.index.dayofweek

df['quarter'] = df.index.quarter

df['month'] = df.index.month

df['year'] = df.index.year

df['dayofyear'] = df.index.dayofyear

df['dayofmonth'] = df.index.day

df['weekofyear'] = df.index.isocalendar().week

return df

df = create\_features(df)

df = create\_features(df)

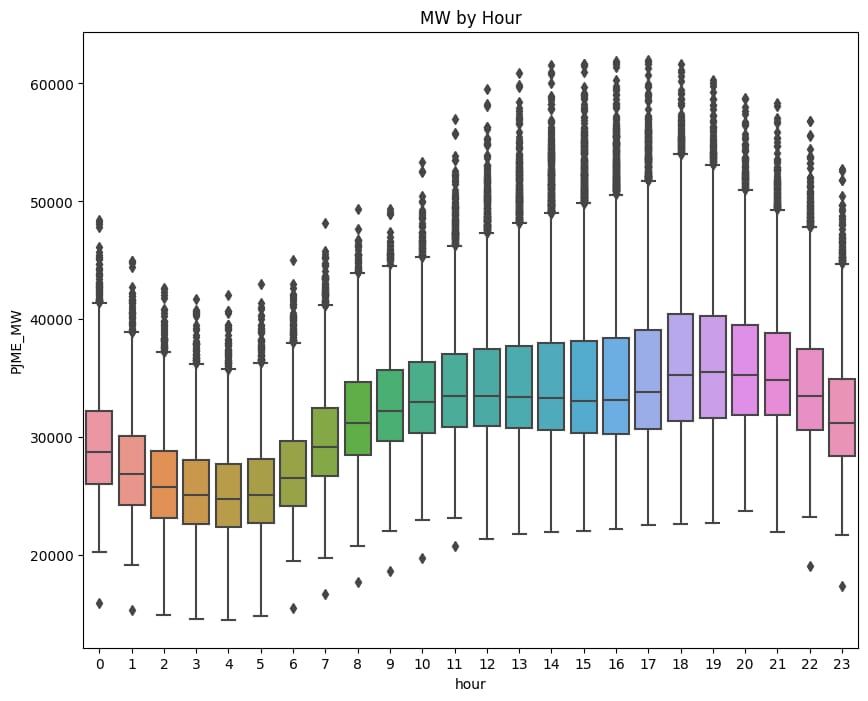
# visualize the hourly Megawatt

fig, ax = plt.subplots(figsize=(10, 8))

sns.boxplot(data=df, x='hour', y='PJME\_MW')

ax.set\_title('MW by Hour')

plt.show()



NOW VISUALISING MEGAWATTS BY MONT

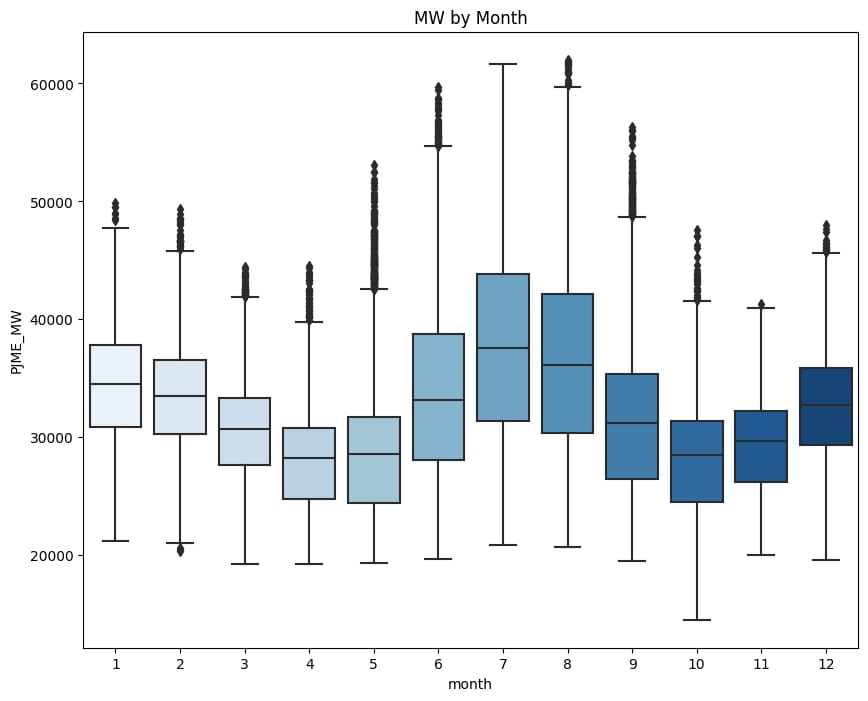
# viaualize the monthly Megawatt

fig, ax = plt.subplots(figsize=(10, 8))

sns.boxplot(data=df, x='month', y='PJME\_MW', palette='Blues')

ax.set\_title('MW by Month')

plt.show()



The monthly usage tends to peak here two times in the winter season, then in the fall and sprint it has lower and another peak in the middle of summer.

# RMSE Score

score = np.sqrt(mean\_squared\_error(test['PJME\_MW'], test['prediction']))

print(f'RMSE Score on Test set: {score:0.2f}')

RMSE Score on Test set: 3721.75

# R2 Score

from sklearn.metrics import r2\_score

r2 = r2\_score(test['PJME\_MW'], test['prediction'])

print("R-squared (R2) Score:", r2)

R-squared (R2) Score: 0.6670230260104328

CONCLUSION

Energy Audit is thus the key to a systematic approach for decision-making in the area of Energy Management.

Energy audit identifies where energy is being consumed and assesses energy saving opportunities so you get to save money where it counts the most.

An energy audit improves and compares the energy efficiency of plant against "Best practices"

