NAAN MUDHALVAN

MEASURE ENERGY CONSUMPTION

Development Part 2:

This research was done as part of a research seminar course at LAB University of Applied Science (fall 2022). The report was written using the thesis template of LAB University.

Abstract

The purpose of this research was to predict energy consumption using the data of Finland's transmission system operator. The objective of this project was to test if a machine learning model can yield good enough results in a complex forecasting problem, exploring machine learning techniques and developing a data-driven model for forecasting energy. The data contained 6-year hourly electrical consumption in Finland, and it is a univariate time series, as it is seasonal. We used a long-short term memory (LSTM) model to train the data. The model was evaluated using root mean squared error (RMSE) to be directly comparable to energy readings in the data. The result shows that electricity consumption can be predicted using machine learning algorithms so we can use the results to deploy renewable energy, plan for high/low load days, and reduce wastage from polluting on reserve standby generation.

Model Implementation

The data was imported from Finland's transmission system operator as a CSV file and then exported to a GitHub repository. There was a total number of 52965 observations and 5 variables in this dataset and no missing values were found. The minimum load volume is 5341 MWh, and the maximum load volume is 15105 MWh along with an average volume of 9488.750519 MWh. The data is univariate time series, where there is a need for one column to present time and another one to present energy consumption. For predicting day consumption, data were down-sampled using resample function. This function changed the data from hourly frequency to daily frequency. Training the LSTM model was done using the training set and the validation dataset for testing the results through the training process. The learning algorithm worked through the entire training dataset 60 times (Epoch), and the model weights were updated after each batch where the batch size is 20.

HISTORICAL DATA: When historical data are given by steps of 15 minutes, forecasts are required by steps of 15 minutes. When historical data are given by steps of 1 hour, forecasts are required by steps of 1 hour. When historical data are given by steps of 1 day, forecasts are required by steps of 1 day. • Timestamp - The time of the measurement • Value - A measure of consumption for that building
WEATHER FORECAST: It can be obtained for the OpenWeatherStation Website for the weather forecast of the place where the building is located. • Timestamp - The time of the measurement • Temperature - The temperature as measured at the weather station
BUILDING DATA: • DAY_OF_WEEK]IsDayOff - True if DAY_OF_WEEK is not a working day • Number of employees everyday
FEATURE EXTRACTION: Feature Extraction: In this phase we need to validate the predictive power of new features as well as existing features. There are many techniques applied to validate the feature importance such as correlation analysis, ensemble and tree based model based feature importance. Feature Transformation/Derivation: during the validation with a baseline model some of the feature may require transformation. These transformations include log transformation, Standard Scaling (SS) and Min Max Scaling (MMS). After literature survey and consultation with subject matter expert, a set of most desirable features for electricity load/consumption forecasting were listed. : a) Past consumption pattern: electrical consumption pattern cannot change abruptly until unless some major changes happen at the place. So past consumption pattern carries information for future consumption pattern.
b) Calendar: month, day of week, c) Demography: The population of building can affect the consumption pattern d) Geography:, temperature, etc. If temperature is high, people will use more electrical appliance and similarly when temperature is low.

ALGORITHMS AND MODELS TO BE USED FOR THE PREDICTION ANALYSIS:

LINEAR REGRESSION We'll train a Linear Regression model for predicting building energy consumption based on historical enregy data, several weather variables, hour of the day, day of the week, weekends and holidays. The accuracy obtained is 63.74% USING NEURAL NETWORKS: LSTMs (or long-short term memory networks) allow for analysis of sequential or ordered data with long-term dependencies present. Traditional neural networks fall short when it comes to this task, and in this regard an LSTM will be used to predict electricity consumption patterns in this instance. The accuracy obtained for this model is 97.12%

So we can conclude that deep neural networks such as LSTMs are more useful for prediction of energy consumption.

missing data was restructured based on the compressed details as predicted data points.

The imputation method was also evaluated to determine its performance. The resultant cleaned data was then further pre-processed using standardisation. Standardisation or also known as Z-score normalisation is a transformation to change the observed data to have characteristics of standard normal distribution in which the mean is 0, and the standard deviation is 1. This transformed the data to be equally distributed above and below the mean value by using the formula in equation

where � is mean and � is standard deviation.

The standardisation process under Caret Package consists of 2 steps which are centring and scaling. The centring transformation computes the mean for a feature and subtracts it from each data point of the feature. On the other hand, the scale transformation computes the standard deviation for a feature and divides the output from centring transformation with the standard deviation.

3.3. Step 3: model development (training)

This research used a supervised machine learning methodology to predict energy consumption. After data was prepared, it was then inputted into the learning algorithm. Different feature combinations were fed into the algorithm to generate a candidate for the predictive model. Before using the data to create and train the model, data partitioning was done to separate the data into two groups – a training group and a testing group.

The predictive modeling for this research used a classification method to predict discrete variables instead of regressive prediction. As Azure ML does not have k-Nearest Neighbour and Artificial Neural Network for classification, the modeling function in Caret R package was utilised for all prediction to ensure uniform execution. Three types of machine learning algorithm were used for this research which were Artificial Neural Network (ANN-MLP), k-Nearest Neighbour (k-NN), and Support Vector Machine (SVM-RBF). Fig. 1 below shows the process after the data preparation until the generation of the predictive model.

```
import numpy as np # linear algebra
```

```
import pandas as pd # data processing
```

```
df =
pd.read_csv('household_power_consumption
.txt', sep=';',
                 parse_dates={'dt' :
['Date', 'Time']},
infer_datetime_format=True,
                 low_memory=False,
na_values=['nan','?'], index_col='dt')


df.shape
```

The dataset contains 2,075,259 rows and 7 columns, let’s take a look at the number of null values:

```
df.isnull().sum()
```

Global_active_power   25979
Global_reactive_power   25979
Voltage   25979
Global_intensity   25979
Sub_metering_1   25979
Sub_metering_2   25979
Sub_metering_3   25979
dtype: int64

We have so many null values in the dataset, I will fill these null values with the mean values:

```
df = df.fillna(df.mean())
```

Data Visualization

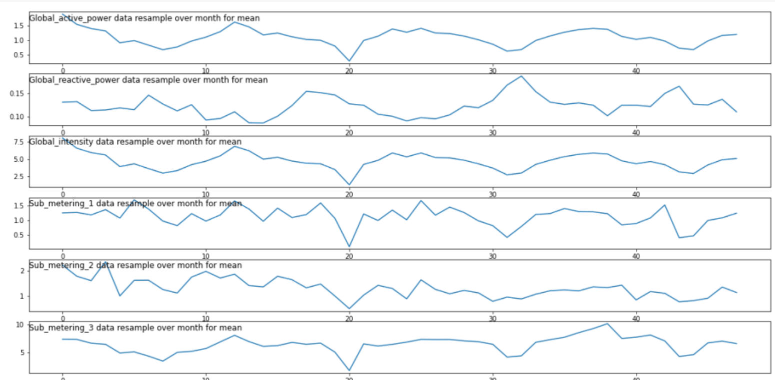
Let’s have a look at the data more closely by visualizing it:

```
import matplotlib.pyplot as plt
```

```
i = 1

cols=[0, 1, 3, 4, 5, 6]

plt.figure(figsize=(20, 10))

for col in cols:

   plt.subplot(len(cols), 1, i)


plt.plot(df.resample('M').mean().values[
:, col])

   plt.title(df.columns[col] + ' data
resample over month for mean', y=0.75,
loc='left')

   i += 1

plt.show()
```

view raw

energy consumption.py hosted with ❤ by GitHub

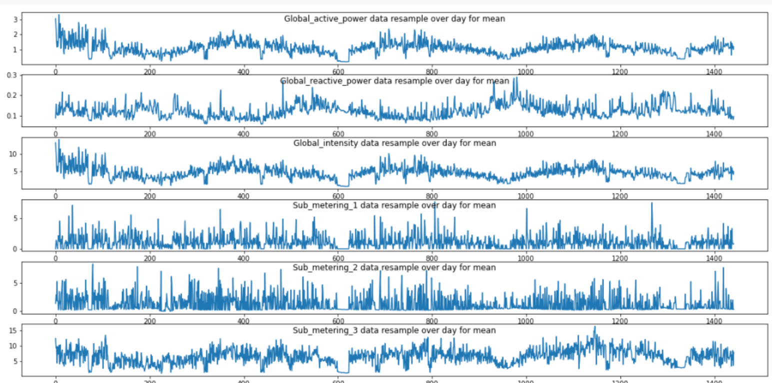



```
i = 1
```

```
cols=[0, 1, 3, 4, 5, 6]

plt.figure(figsize=(20, 10))

for col in cols:

   plt.subplot(len(cols), 1, i)


plt.plot(df.resample('D').mean().values[
:, col])

   plt.title(df.columns[col] + ' data
resample over day for mean', y=0.75,
loc='center')

   i += 1

plt.show()
```

view raw

energy consumption.py hosted with ❤ by GitHub




```
i = 1


cols=[0, 1, 3, 4, 5, 6]

plt.figure(figsize=(20, 10))

for col in cols:
```

```
    plt.subplot(len(cols), 1, i)

    plt.plot(df.resample('H').mean().values[:, col])
    plt.title(df.columns[col] + ' data resample over hour for mean', y=0.75, loc='left')
    i += 1
plt.show()
```