# TIME SERIES ANALYSIS

TCS HUMAIN ROUND 2

CT/DT Number: CT20182376310

Contestant Name: JAYESH PRASAD

College Name: R.M.D ENGINEERING COLLEGE

# Background

Energy consumption readings for a sample of 5,567 London Households that took part in the UK Power Networks led Low Carbon London project between November 2011 and February 2014.

Readings were taken at half hourly intervals. Households have been allocated to a CACI Acorn group (2010). The customers in the trial were recruited as a balanced sample representative of the Greater London population.

The dataset contains energy consumption, in kWh (per half hour), unique household identifier, date and time, and CACI Acorn group

# Understanding

**Problem**: To Forecast the Energy usage of top three households having maximum number of samples on an hourly basis based on the data given.

It will be of vital importance to forecast the Energy usage or Generation in an area. It serves as a precautionary model to evaluate the future consumption of households, in order to meet the demand of the consumers by generating the power required to satisfy the needs.

# Scope

Given the dataset with columns

'LCLid' – ID of the house

'StdorToU' – Type of Plan

'DateTime' – Half hour period of Date and time

'KWh' – Energy usage of the house in Half hour basis

'Acorn' and 'Acorn-grouped' -- Acorn group

Since we need to forecast the Energy usage, it will be sufficient to utilize the columns 'LCLid', 'DateTime', 'KWh'.

## Out of Scope

We skip the columns ‘STdorToU’, ‘Acorn’, ‘Acorn-grouped’, even though these columns may be used to calculate the power tariff for the household energy consumption it is beyond the scope of the problem and also the rates for the power tariffs are not provided.

## Assumptions

Using only the columns ‘LCLid’, ‘DateTime’, ‘KWh’ for our analysis and forecasting and assuming that the other columns provided are not related to or affecting the forecast.

## Solution Approach

Step 1: Going through the Dataset provided, it has 999970 rows with 30 unique House IDs. It needs a higher RAM and processor to execute the analysis and forecast effectively and easily. Hence, using **Google Colaboratory and Kaggle Kernel** for the problem**.**

Step 2: Uploading the given dataset to Google Colaboratory using in_built function for execution

```
from google.colab import files
files.upload()
```

and can be read using pandas

```
data=pandas.read_csv(‘filename.csv’)
```

whereas in Kaggle Kernel the input file has to be uploaded to input folder and can be read by accessing the path

```
‘../input/Power.csv’
```

And can be read using pandas

```
data = pandas.read_csv('../input/Power.csv')
```

## Step 3: Performing Analysis on the Given Dataset

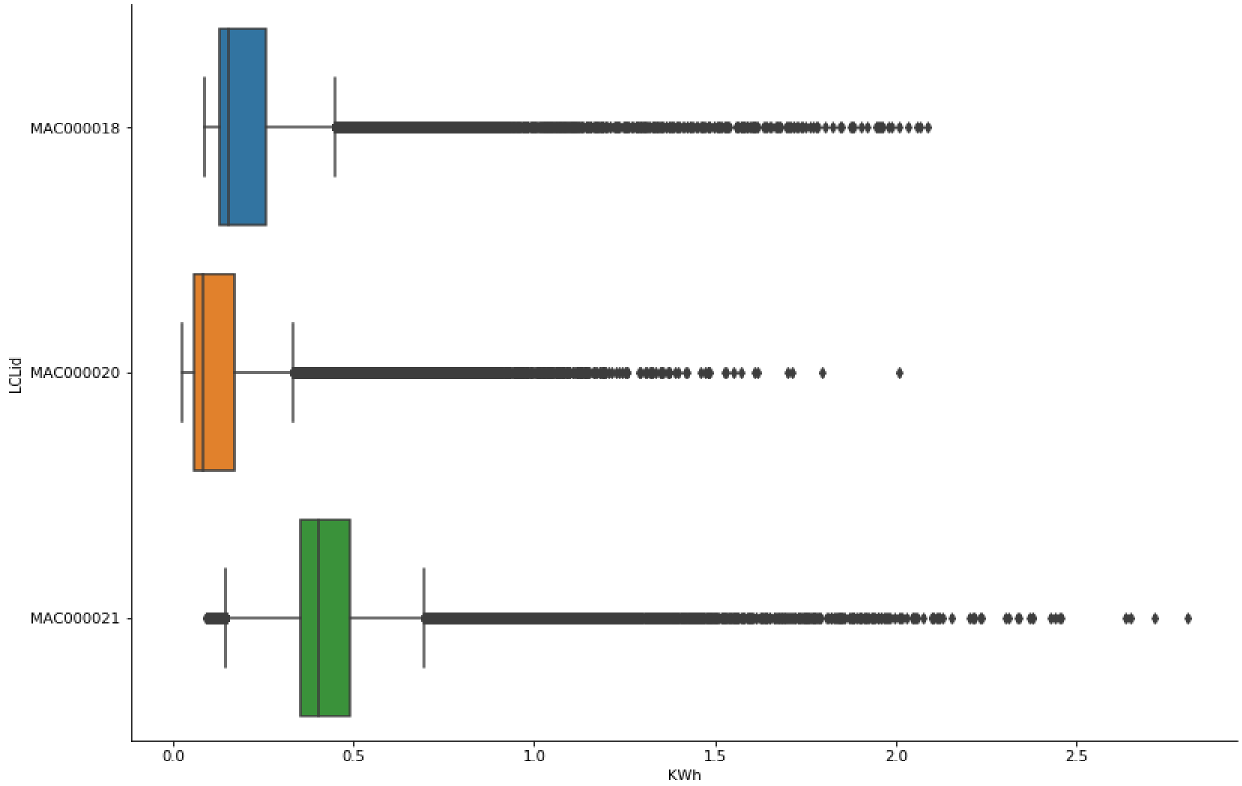
```
DESCRIPTION
                           DateTime      LCLid            KWh
count                        999971     999971  999971.000000
unique                        39095         30            NaN
top     2012-10-20 00:00:00.0000000  MAC000018            NaN
freq                             58      39081            NaN
mean                            NaN        NaN       0.239580
std                             NaN        NaN       0.387533
min                             NaN        NaN       0.000000
25%                             NaN        NaN       0.060000
50%                             NaN        NaN       0.129000
75%                             NaN        NaN       0.255000
max                             NaN        NaN       6.528000
```

## Step 4: Finding the Top Three Maximum Sample HouseIDs

```
THE TOP THREE HOUSEIDs HAVING MAXIMUM SAMPLES ARE

['MAC000018', 'MAC000020', 'MAC000021']

The top three number of samples present are
[39081, 39078, 39078]
```

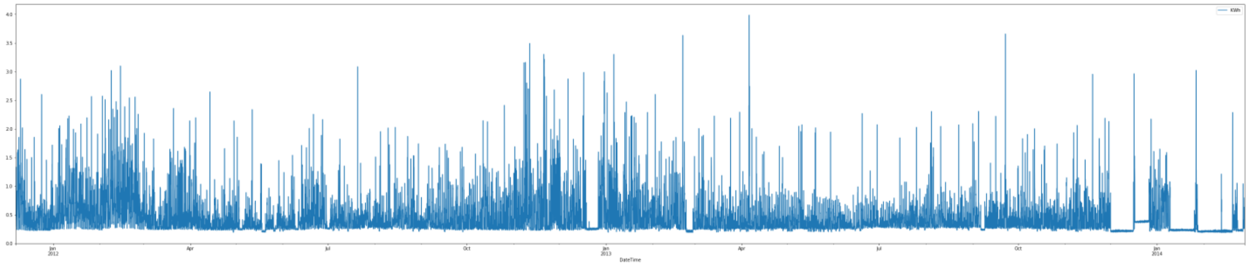


Step 5: Seperating the Dataset into the Three HouseID datasets as “first, second and third”

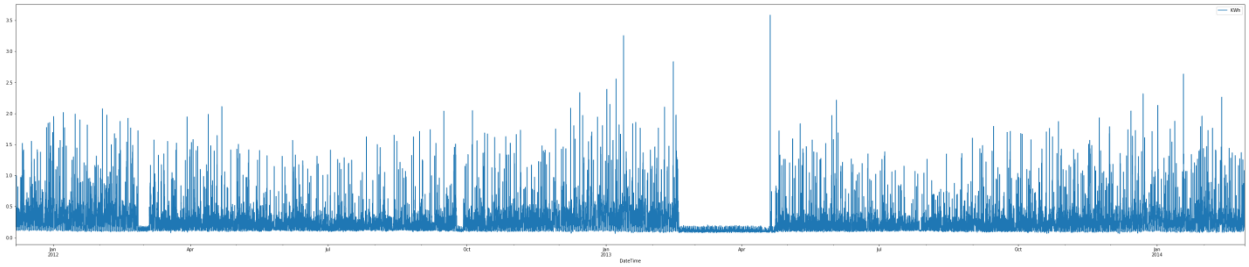
Step 6: Grouping the Data on an hourly basis for hourly forecast

Step 7: Plotting the Datasets for Visualising the Data

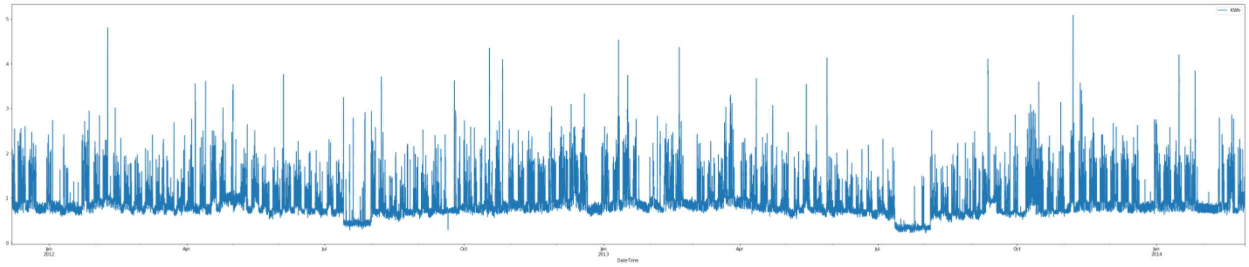
MAC000018



MAC000020

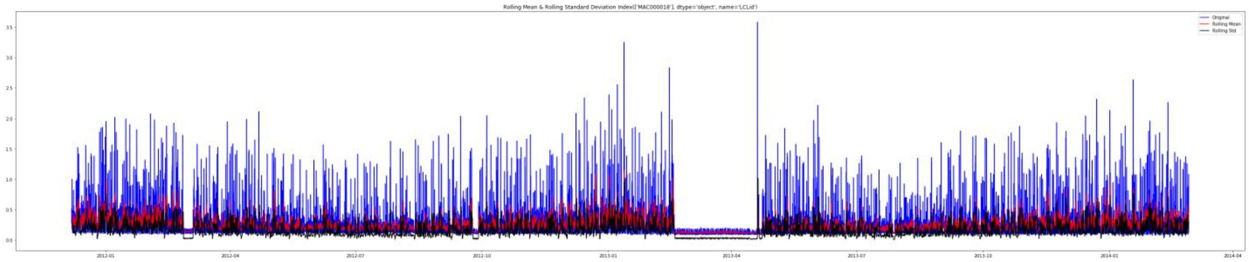


MAC000021

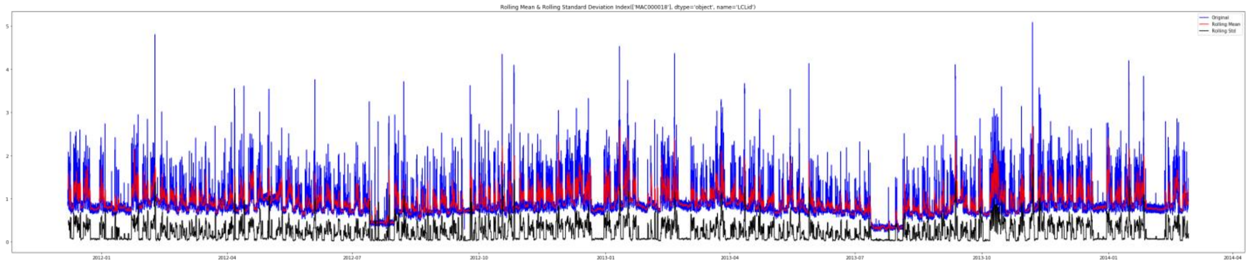


Step 8: Plotting Rolling Mean and Rolling Standard Deviation for the Dataset

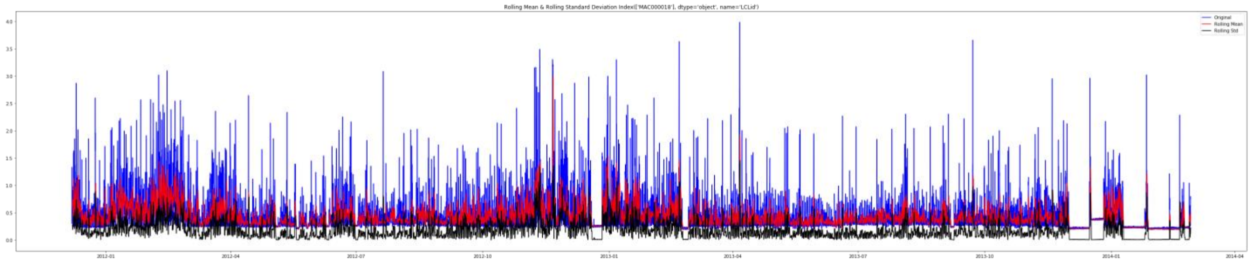
MAC000018



MAC000020



MAC000021

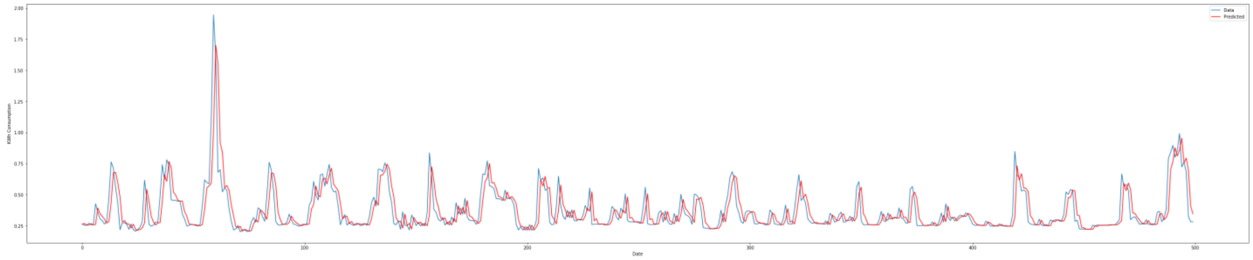


Step 9: Using the TIME-SERIES forecasting technique ARIMA from statistical models

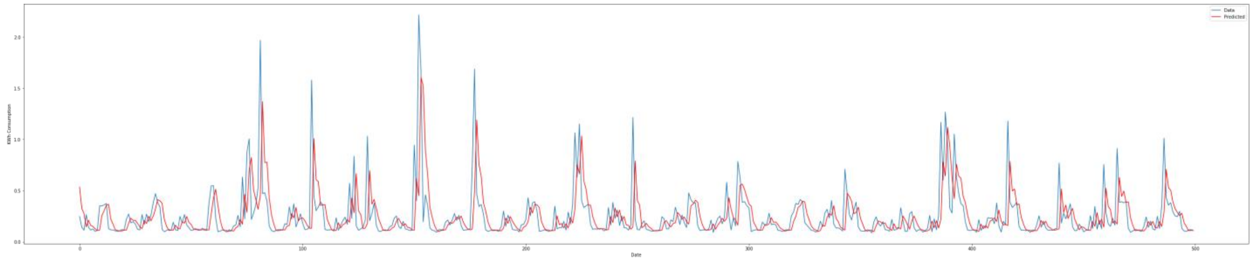
https://www.statsmodels.org/stable/generated/statsmodels.tsa.arima_model.ARIMA.html

## Step 10: Showing the Description, Prediction and Error Graph for each HouseID

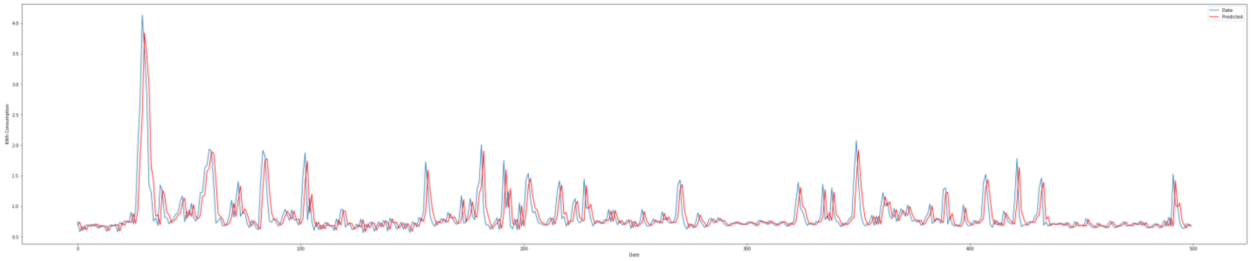
### MAC000018



### MAC000020



### MAC000021

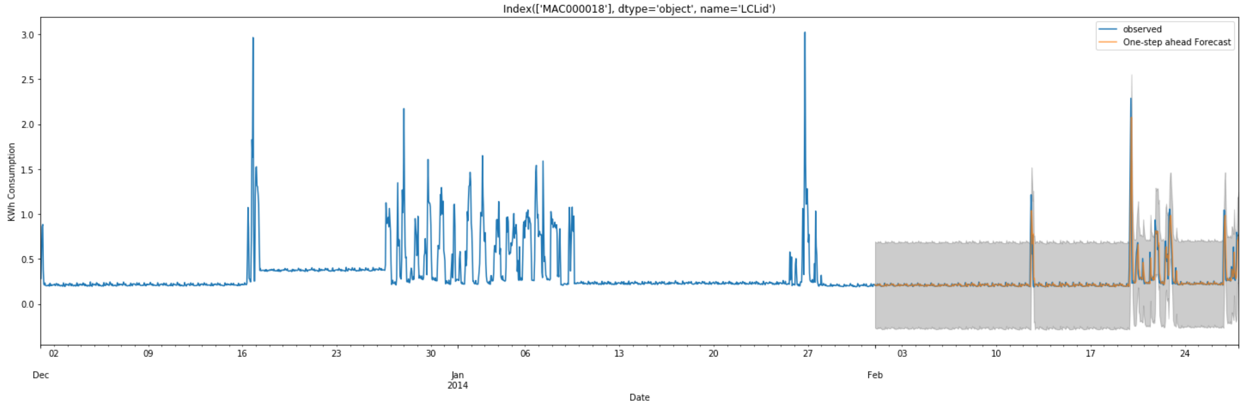


## Step 11: Since ARIMA uses a lot of time for prediction went for SARIMAX

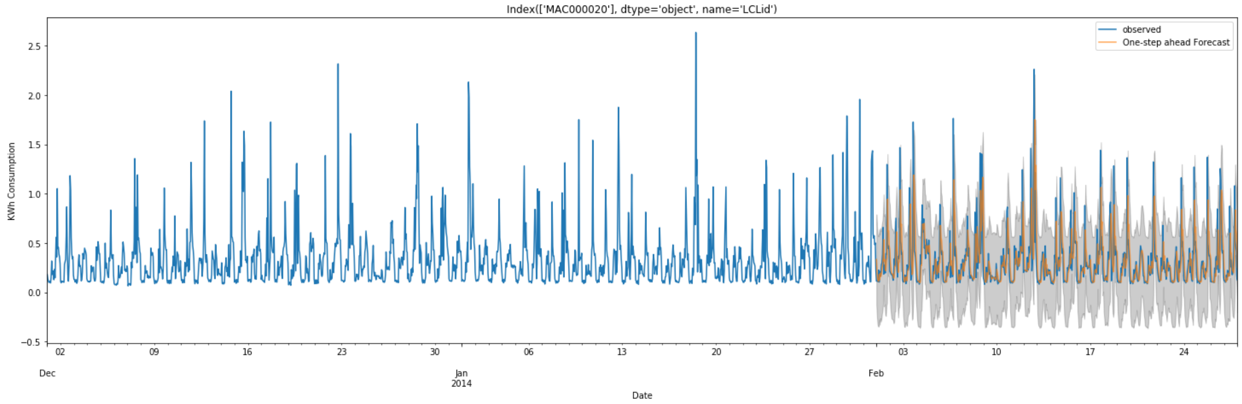
https://www.statsmodels.org/stable/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html

## Step 12: Using Seasonal ARIMAX Predicting and Forecasting values with Mean Squared Error and Root Mean Squared Error for each House ID

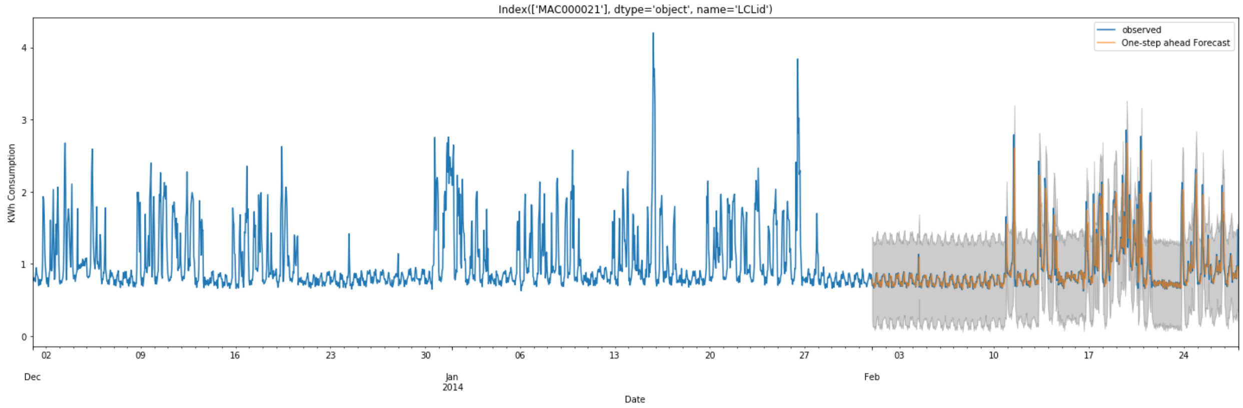
### MAC000018



### MAC000020



### MAC000021



# Implementation Framework

The Analysis part of the Solution was done in Google Colaboratory with Hardware Specifications:

Python3 Google Compute Engine

RAM –14 GB

ROM – 50 GB

Implementation of the Solution was done in Kaggle Kernel with Hardware Specifications:

RAM – 16 GB

ROM – 1 GB

# Solution Submission

GITHUB LINK TO SOLUTION

**https://github.com/jayashprasad8/TCSHumain**

# References

Dataset

https://data.london.gov.uk/dataset/smartmeter-energy-use-data-in-london-households

Visualisation

https://towardsdatascience.com/an-end-to-end-project-on-time-series-analysis-and-forecasting-with-python-4835e6bf050b

SARIMAX statistical model
https://www.statsmodels.org/stable/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html

ARIMA statistical model
https://www.statsmodels.org/stable/generated/statsmodels.tsa.arima_model.ARIMA.html