Smart Home Dataset With Weather Information

Group 2 - Storm Troopers

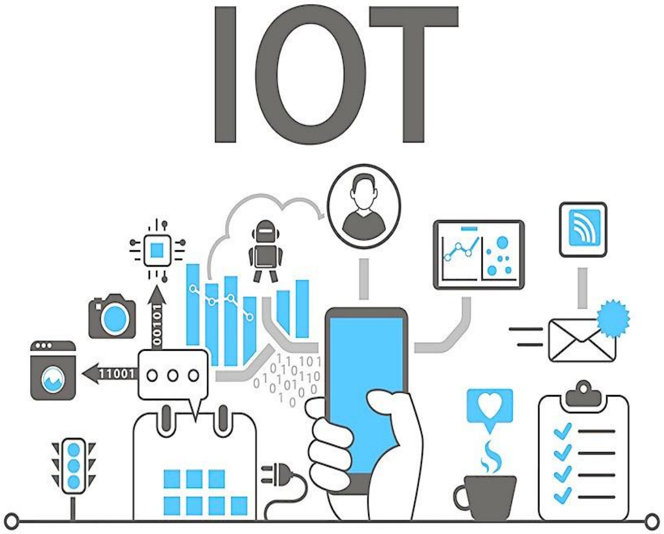
1.	Vandana Chandola	:	014748604
2.	Nikhila Churia	:	014546116
3.	Haasitha Pidaparthi	:	012669254

Overview

1. Introduction
2. Data Pre-processing
3. Data Transformation
4. Data Visualization
5. Correlation between features
6. Time-Series Analysis
7. Factors in Time-Series Analysis
8. Anomaly Detection
9. Time-Series Forecasting

Introduction

- IoT brings together everything at home under one umbrella which has the potential to monitor and remote control such as air conditioning, alarm system, lighting, heating, ventilation, telephone system, tv, etc. to enhance our comfort and security with low energy consumption.
- The home is a specific environment, and energy management is one of the IoT use cases with which energy being sent out or consumed can be monitored.



Dataset

- The dataset we have used for this project is titled “Smart Home Dataset with Weather Information”, which has been downloaded from Kaggle.
- The dataset has 32 columns and more than 500,000 readings with the time-span of 1 minute of the energy used (in kW) by the appliances of a smart home, and the weather conditions of that area at that time.

```
]: home_df.shape[0]
]: 503911
```

```
: home_df.shape[1]
: 32
```

Data Pre-processing

- Removing invalid rows
- Converting attribute values into proper format for analysis.(eg.UNIX timestamp to date format)
- Data imputation such as replacing missing weather information with the next valid observation.
- Identifying and removing unwanted and duplicate columns.
- Resampling the dataset into daily and monthly subsets based on the analysis requirements.

Data Transformation

- Split the dataset into ‘enery_data’ and ‘weather_data’
  - ‘energy_data’ - gen, use, Dishwasher, Furnace, Home office, Fridge, Wine cellar, Garage door, Kitchen, Barn, Well, Microwave, Living room
  - ‘weather_data’ - Temperature, humidity, visibility, apparentTemperature, pressure, windSpeed, windBearing, dewPoint

	temperature	humidity	visibility	apparentTemperature	pressure	windSpeed	windBearing	dewPoint
2016-01-01 05:00:00	36.14	0.62	10.0	29.26	1016.91	9.18	282.0	24.4
2016-01-01 05:01:00	36.14	0.62	10.0	29.26	1016.91	9.18	282.0	24.4
2016-01-01 05:02:00	36.14	0.62	10.0	29.26	1016.91	9.18	282.0	24.4
2016-01-01 05:03:00	36.14	0.62	10.0	29.26	1016.91	9.18	282.0	24.4
2016-01-01 05:04:00	36.14	0.62	10.0	29.26	1016.91	9.18	282.0	24.4

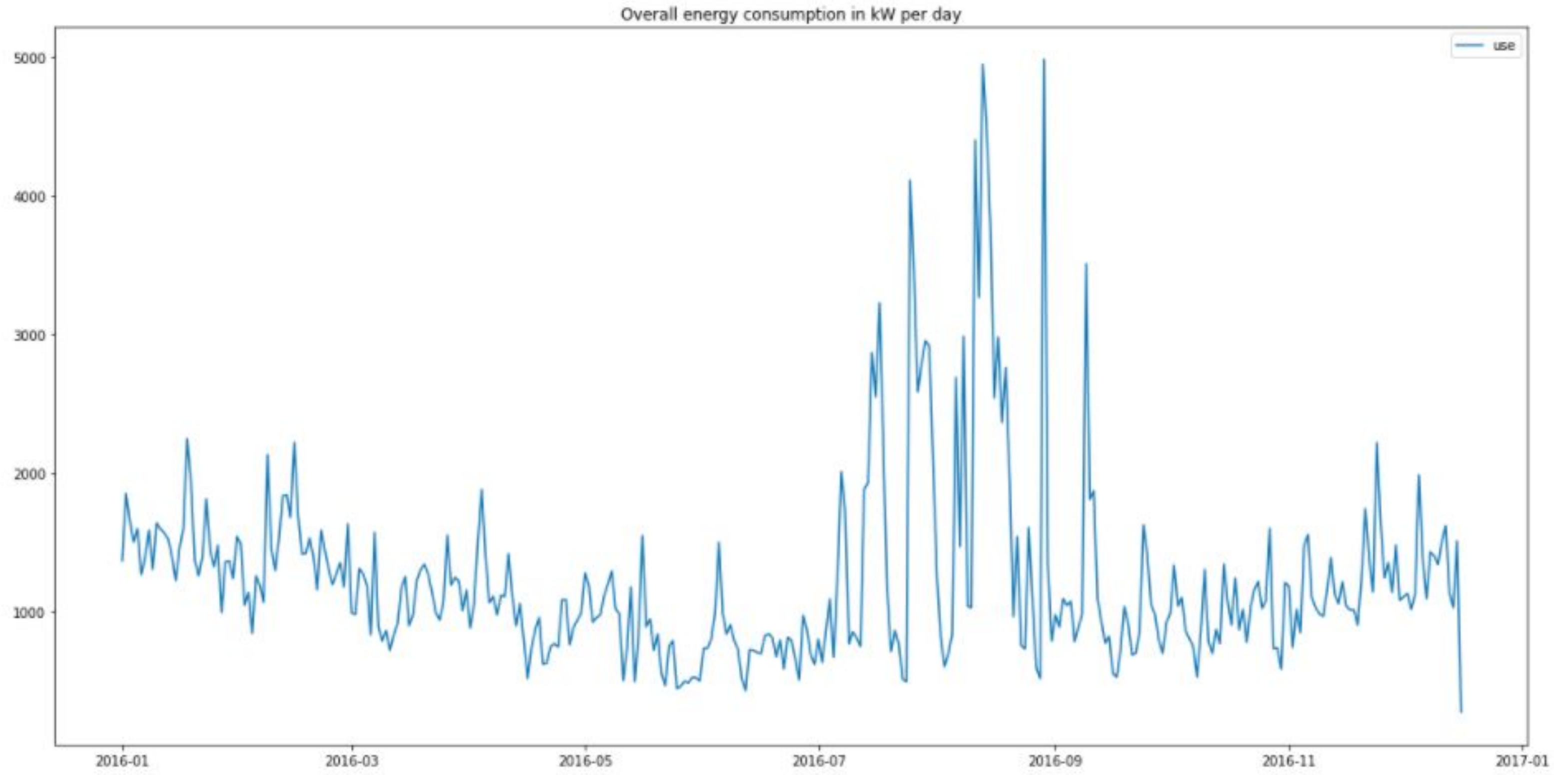
Data Transformation

- Split ‘energy_data’ into 3 subsets
  - energy_per_day
  - energy_per_week
  - energy_per_month

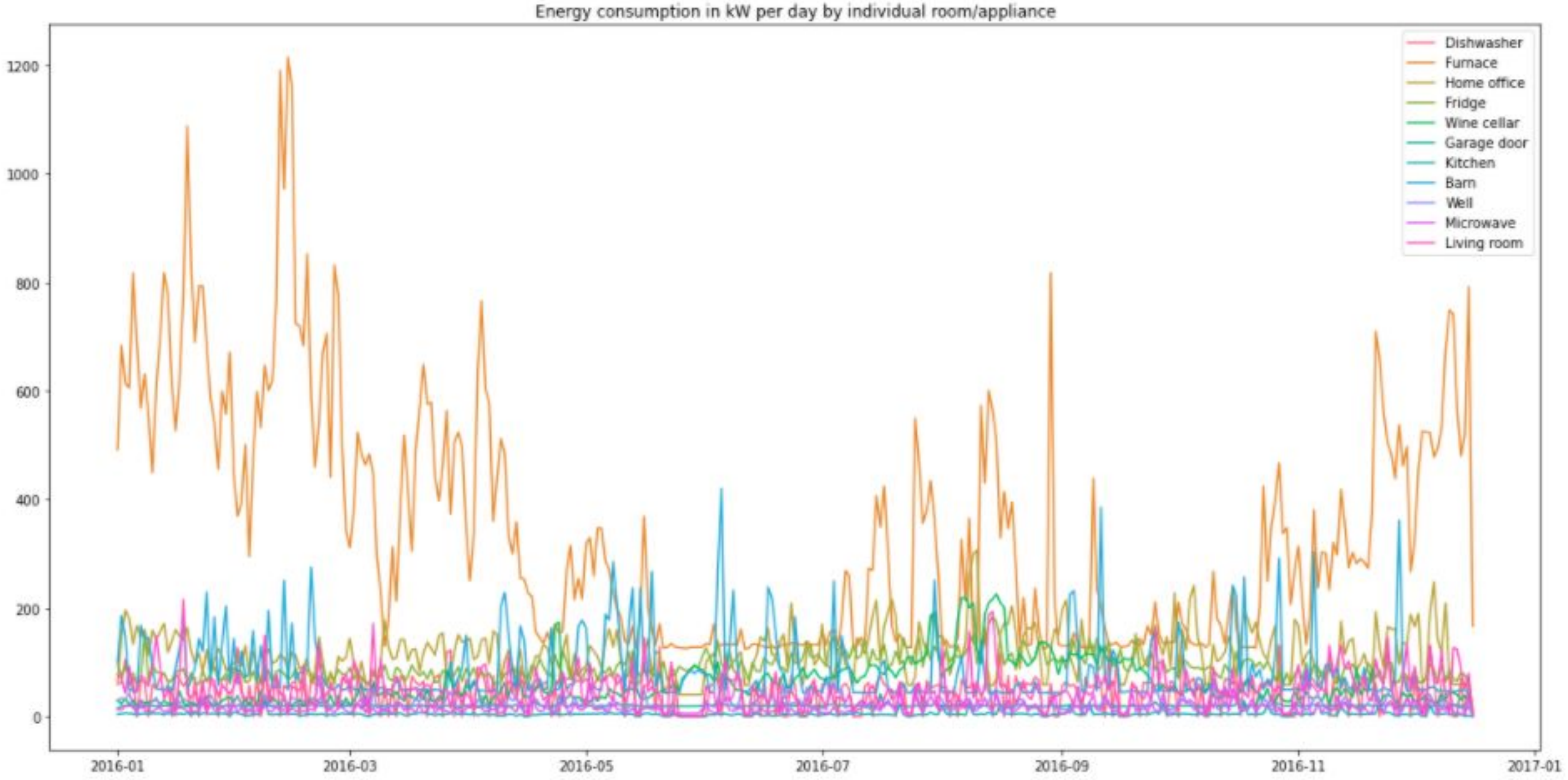
	gen	use	Dishwasher	Furnace	Home office	Fridge	Wine cellar	Garage door	Kitchen	Barn	Well	Microwave	Living room
2016-01-31	2596.565467	46256.153317	1613.572299	20272.447383	4043.940117	2436.271883	840.762533	605.041667	154.011083	3314.010383	766.110601	429.532350	1921.156633
2016-02-29	2704.221700	41558.035267	1399.090831	19171.333067	2850.642583	2225.080050	832.488483	572.159733	137.273728	2817.221550	741.079083	388.025434	1408.870900
2016-03-31	3795.807367	34026.880883	1506.501997	13046.526433	3511.736400	2393.101050	1268.479517	614.368167	134.469697	1791.915983	731.827333	426.910884	1585.980033
2016-04-30	3893.534950	29662.845900	1443.403725	9393.876000	3173.250717	2454.983017	1548.467600	627.425083	103.812260	2626.763767	658.356017	488.510350	1571.712033
2016-05-31	3670.712050	25550.843150	1180.812253	5957.877471	2768.990462	2648.659933	1561.469854	617.532683	99.109124	3321.740146	450.940233	406.917284	1179.055583

Data Visualization

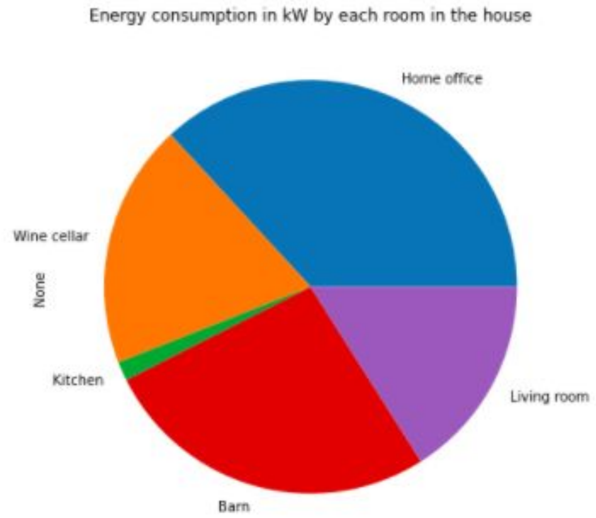
- Line Plot - Overall energy consumption per day (in kW)
  - Per week
  - Per month
- Line Plot - Energy consumption per day for each room/appliance (in kW)
  - Per week
  - Per month
- Pie Chart - Energy Consumption by each room
- Pie Chart - Energy Consumption by each appliance

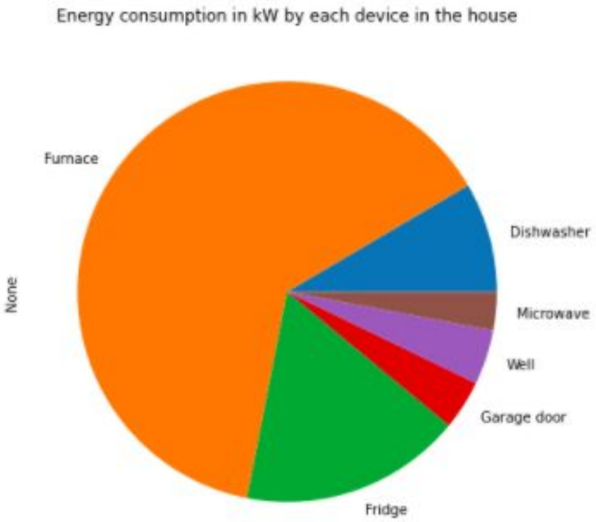


According to this plot, the months of August and September seem to have the highest energy consumption



This line plot indicates that Furnace has the highest energy consumption compared to other rooms/appliances



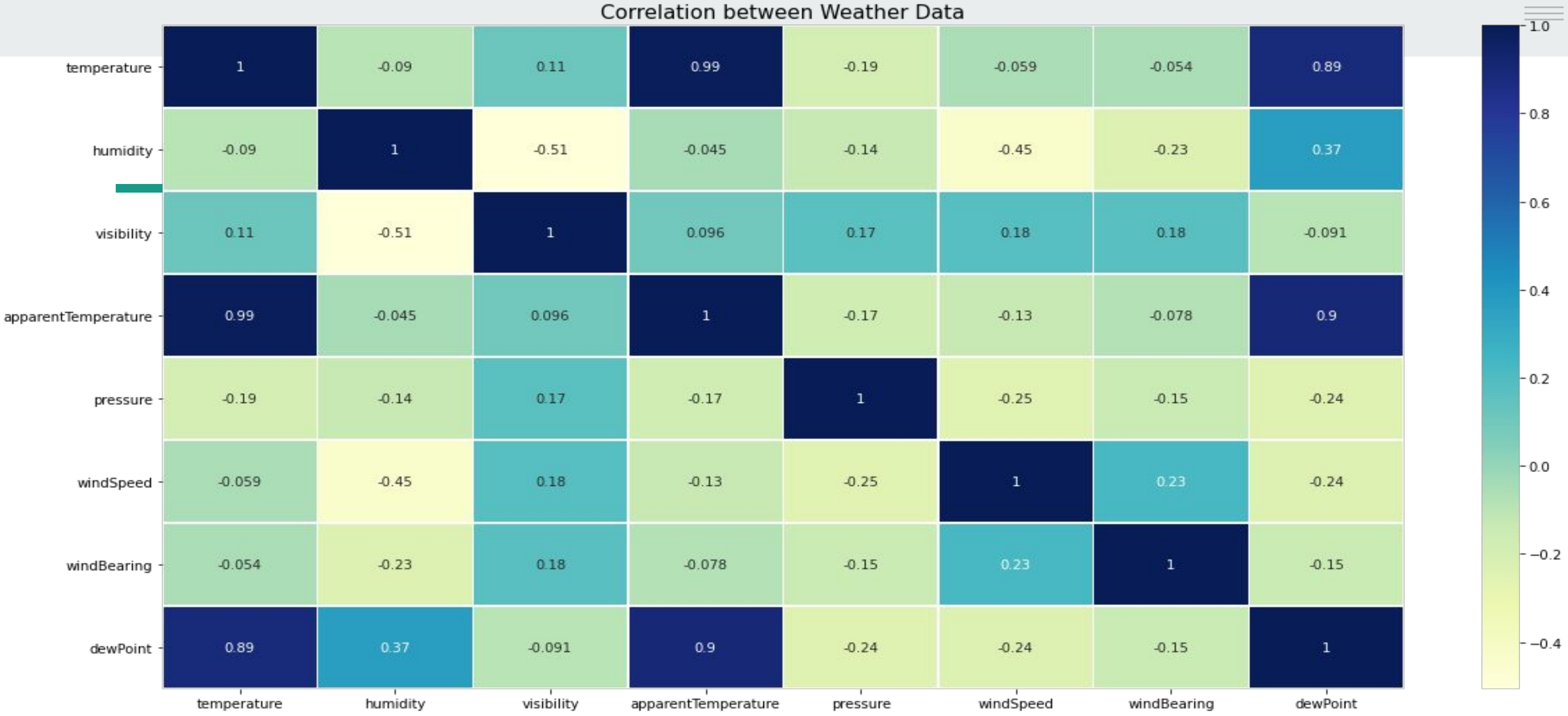



```
Energy consumed in kW per month by each room in the house
Home office    40961.277962
Wine cellar    21233.079320
Kitchen         1644.062194
Barn           29493.633696
Living room    17794.476716
dtype: float64
Energy consumed in kW per month by each device in the house
Dishwasher      15806.409510
Furnace        118917.457954
Fridge          32026.710600
Garage door      7124.847316
Well             7882.163852
Microwave        5534.440218
dtype: float64
```

Pie chart that shows the energy consumption by room and appliances

Correlation

- Indicates the relationship between features, values ranging from -1 to 1.
- There are two key components of a correlation value:
    - **magnitude** : The larger the magnitude (closer to 1 or -1), the stronger the correlation
    - **sign** : Negative indicates inverse correlation. Positive indicates regular correlation.
- Here we use correlation to identify whether there is any kind of significant relationship between appliances, between weather data, and overall (i.e. interdependence between weather data and appliances).



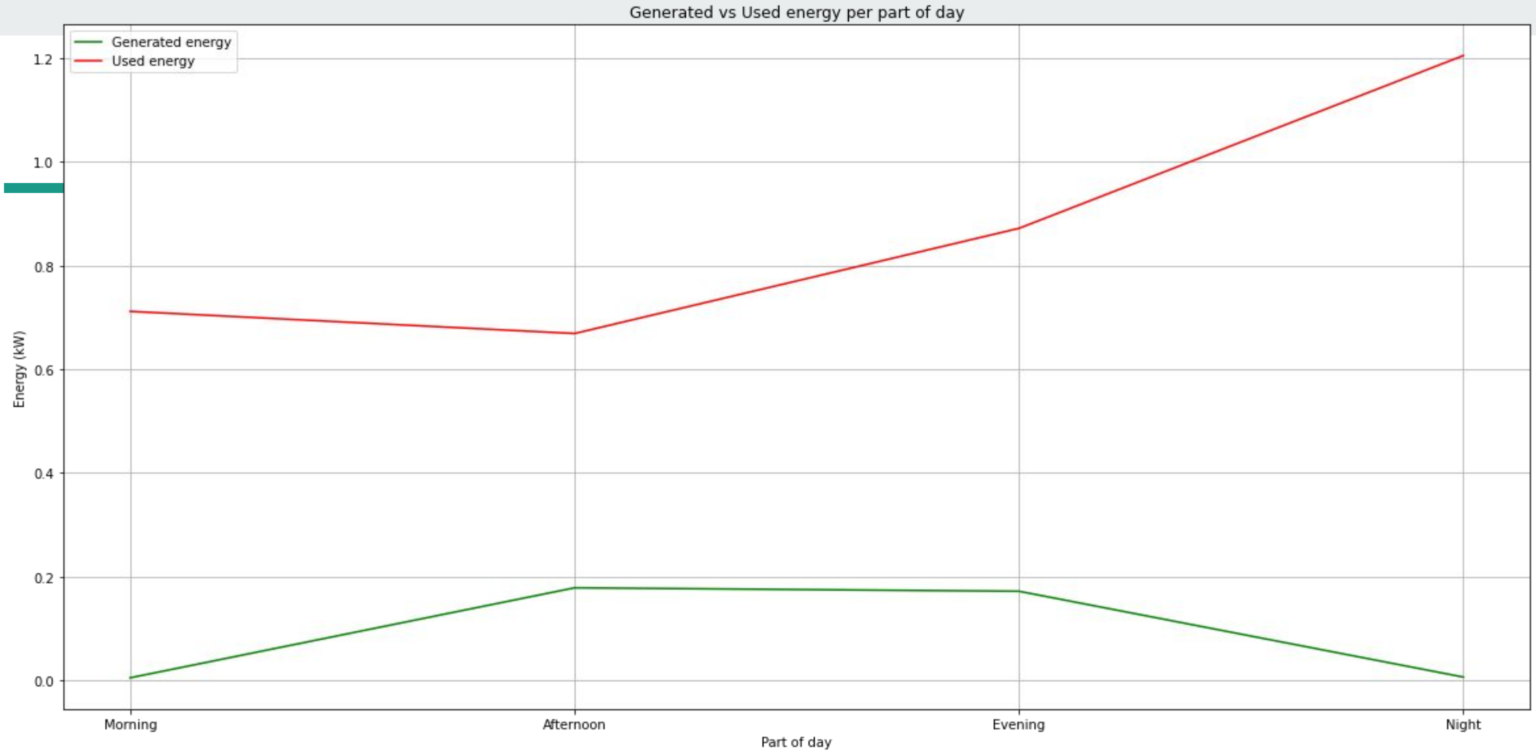
- Strong positive correlation observed between **temperature, apparentTemperature** and **dewPoint**.
- Relationships observed between other features as well, but not as significant.

- **Correlation of energy usage by appliances**
  - No significant relationship, positive or negative, was observed between energy usage by appliances.
  - Safe to presume that energy usage of one appliance doesn’t affect another.

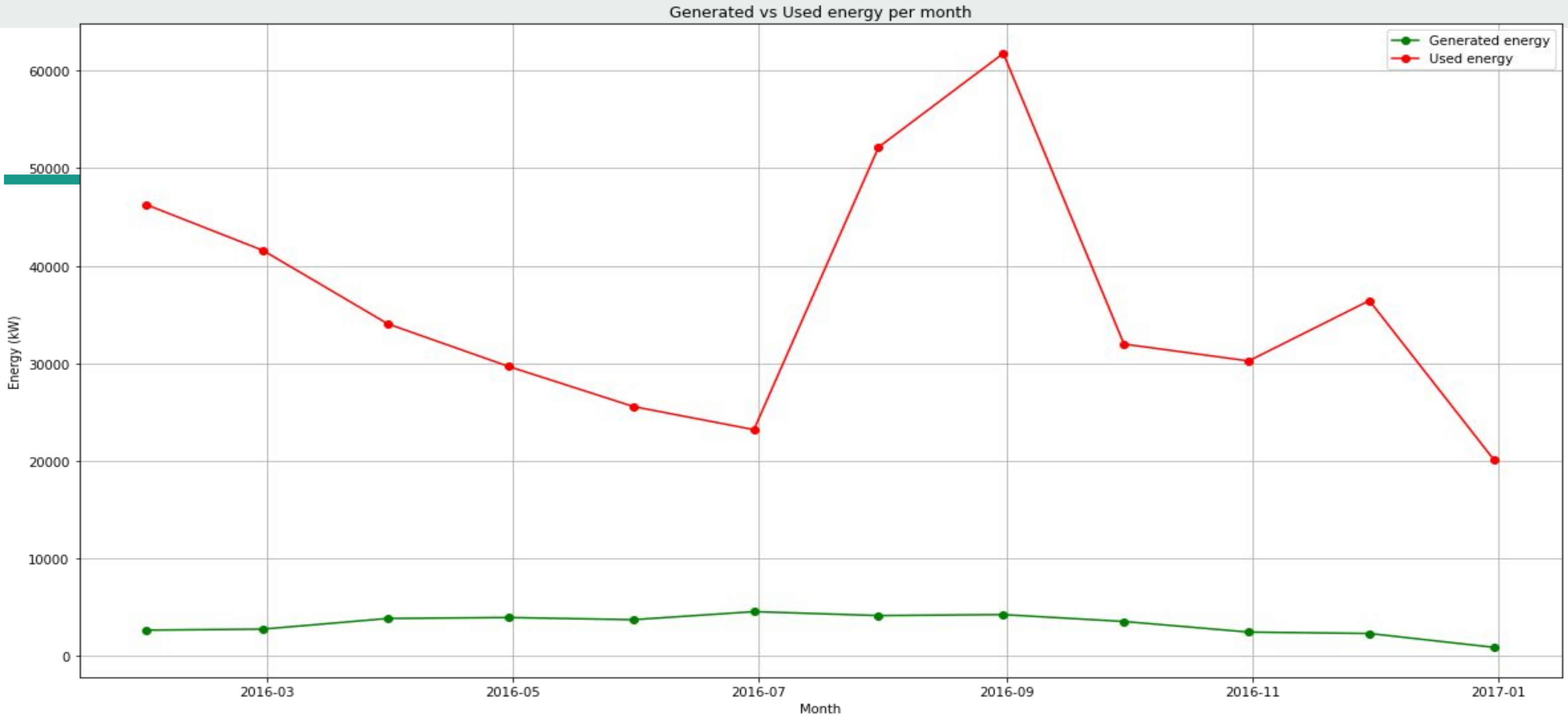
- **Correlation between all data**
  - Weak correlation between **wine cellar** and weather features like **dewPoint(0.3), apparentTemperature(0.29)** and **temperature(0.29).**
  - Relationships observed between other features as well, but not as significant.

Time Series Analysis

- Our dataset contains minute-by-minute observation of generated and used energy.
- We try to identify different time basis to observe the generated vs used energy relation.
- Observations done per time of day (morning, afternoon, evening, night), per day, per week and per month.
- This is done to identify patterns and/or inconsistencies in overall energy generation and consumption.



- **Per time of day:**
  - Energy generated is high during afternoon and evening, and energy used is high during evening and night.



- **Per day/ week/month:**
  - Energy generated is highest during the time period between August and September.
  - Significantly high in months of February, March, April and December.

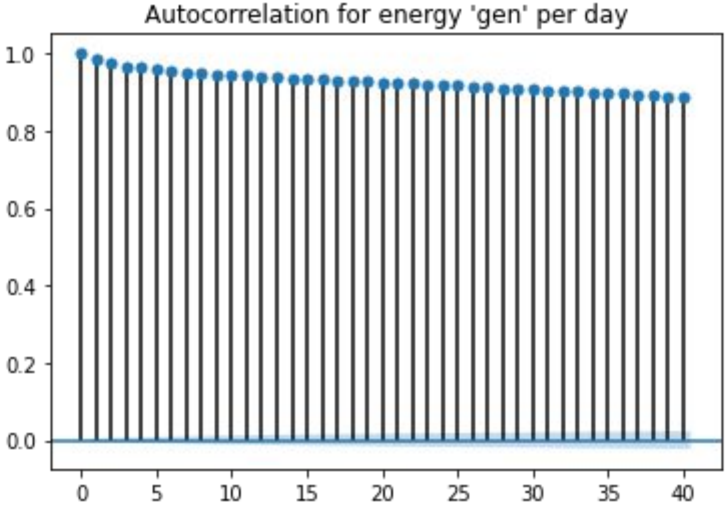
Factors in Time-Series Analysis

3 main factors in Time-Series Analysis:

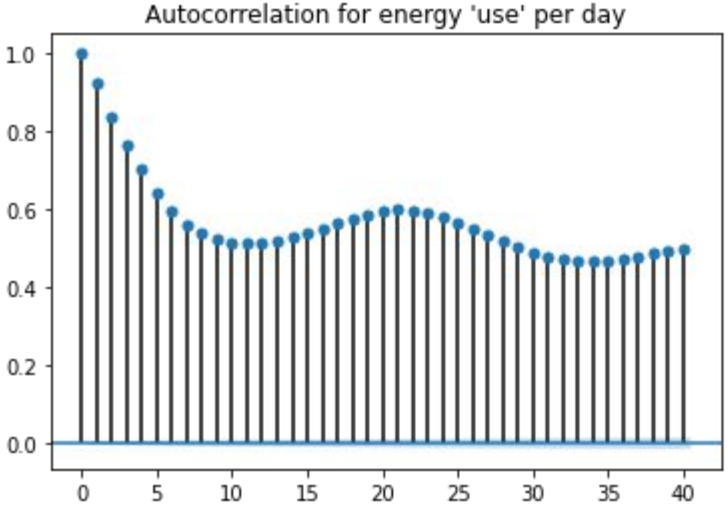
- **Autocorrelation** - is there a tendency of observations and patterns to repeat?
- **Seasonality** - do observations and patterns repeat at regular intervals?
- **Stationarity** - how little the mean and variance of a series change over time?

Autocorrelation

- Autocorrelation measures a set of current values against a set of past values to see if they correlate.
- We calculate the correlation for current time-series observations with observations of previous time steps called **lags**.
- For example, one might expect the energy usage at the 1st minute of the day to be more similar to the usage at 2nd minute rather than at a minute during mid-day.
- Data that has strong autocorrelation is not random, and provides high predictability.
- In this experiment, we use autocorrelation to determine the randomness and predictability of energy generation and usage.



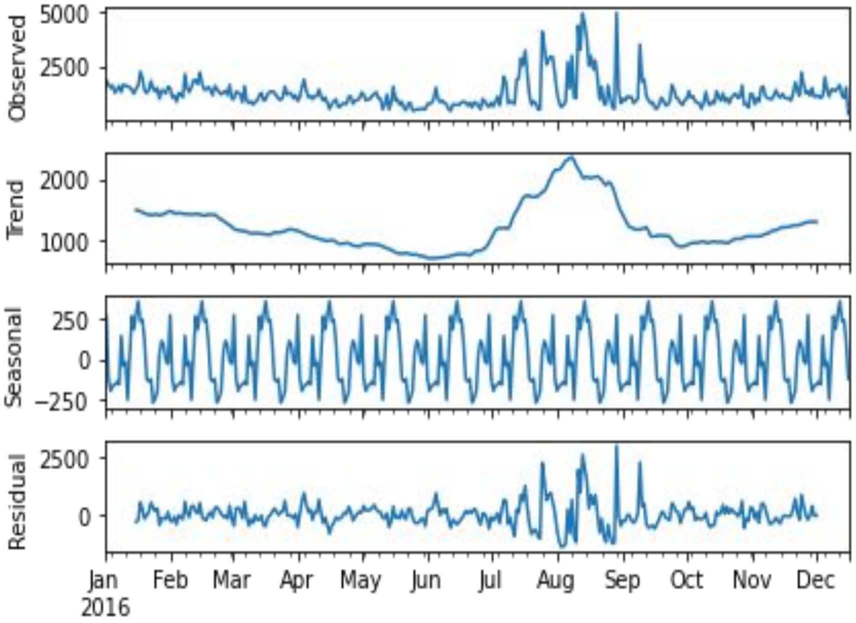
- Autocorrelation = 0.41
- Lags = 40



- Autocorrelation = 0.61
- Lags = 40

Provides good predictability if modeled properly.

Seasonality



- The repeating short-term cycle in the series.
- The dataset used for this is the energy usage on a daily basis, and it showed somewhat of a cyclic behavior on a monthly basis.
- Therefore, used seasonal decomposition with parameter freq=30.
- Seasonal behavior displayed.

Stationarity

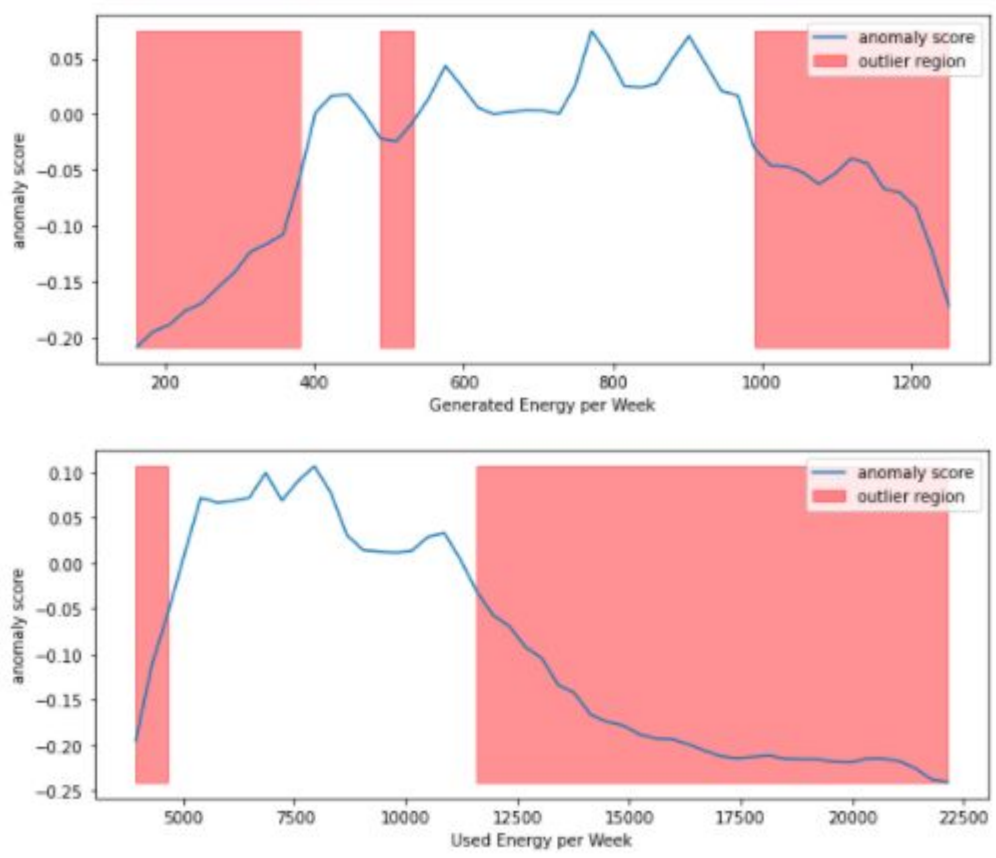
- Summary statistics calculated on the time series are consistent over time i.e. the mean or the variance.
- To observe stationarity, split the dataset into two exact halves.
- For each subset, calculate mean and variance.
- Compare the values.
- If they differ, and the difference is statistically significant, the time-series is non-stationary and needs to be made stationary.
- It was observed that the values differ but are essentially in the same ballpark.

```
mean1=0.113294, mean2=0.121690
```

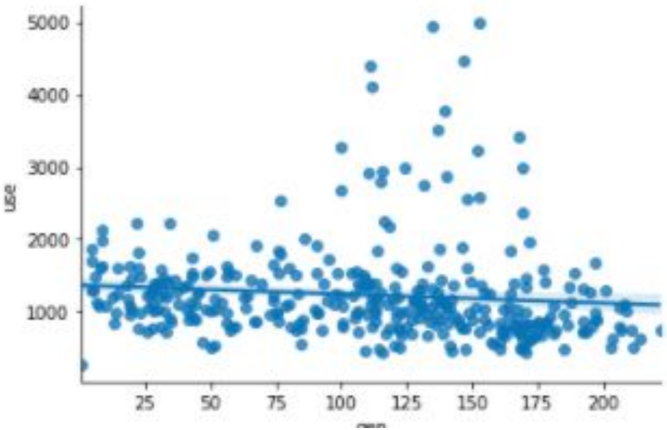
```
variance1=0.092859, variance2=0.212700
```

Anomaly Detection

- Isolation Forest
  - unsupervised learning algorithm that identifies anomaly by isolating outliers in the data
  - isolates the outliers by randomly selecting a feature from the given set of features and then randomly selecting a split value between the max and min values of that feature
  - detect anomalies faster and require less memory compared to other algorithms

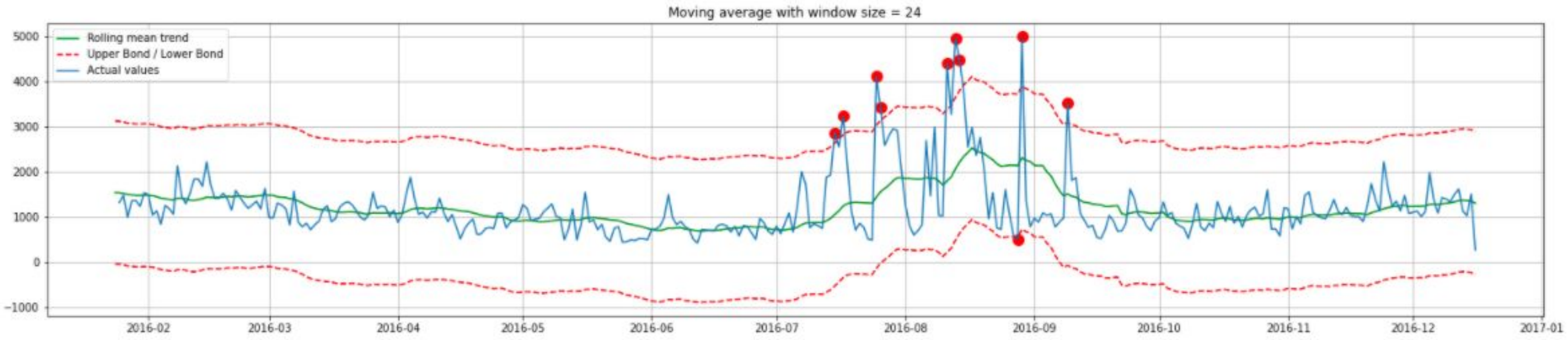


Anomaly Detection using Isolation Forest



Multivariate Anomaly Detection

- **Moving Average technique**
  - a calculation to analyze data points by creating series of averages of different subsets of the full data set
  - commonly used with time series data to smooth out short-term fluctuations and highlight longer-term trends or cycles
  - it is often used in technical analysis of financial data, like stock prices and in economics to examine gross domestic product, employment or other macroeconomic time series
  - Higher the window, smoother the curve



Time series forecasting using ARIMA model

- We have used Autoregressive Integrated Moving Average or ARIMA model for predicting the future values of used energy (‘use’).
- The model has been trained on the resampled data of 20000 records using 70% as the training data and 30% as the test data.

```
predicted=0.791036, expected=0.792383
predicted=0.790207, expected=0.768883
predicted=0.765278, expected=0.771917
predicted=0.775596, expected=0.769117
predicted=0.770476, expected=0.764117
predicted=0.764030, expected=0.763067
Time taken to train the model in seconds=  3571.433491230011
Test MSE: 0.082
```

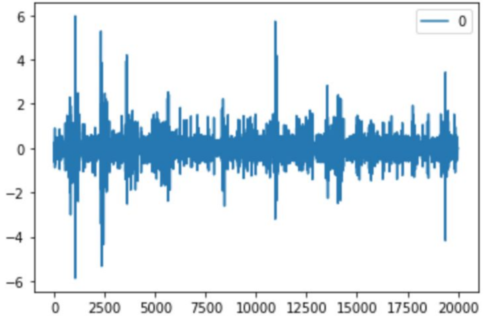
Residual analysis of ARIMA model

Residuals are useful in checking whether a model has adequately captured the information in the data. A good forecasting method will yield residuals with the following properties:

1. The residuals are uncorrelated. If there are correlations between residuals, then there is information left in the residuals which should be used in computing forecasts.

```
from pandas import DataFrame
#We get the information if the model is accurate fr
residuals= DataFrame(model_fit.resid)
residuals.plot()
```

<matplotlib.axes._subplots.AxesSubplot at 0x7f51f8100b38>



2. The residuals have zero mean. If the residuals have a mean other than zero, then the forecasts are biased.

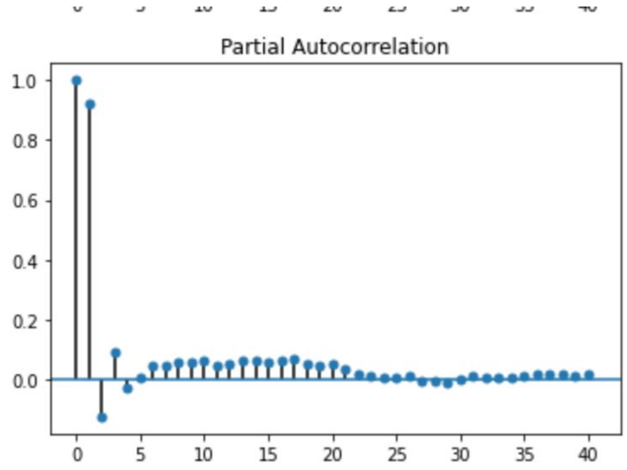
```
residuals.describe()
```

Mean= 0.0000003816986

	0
count	1.999800e+04
mean	3.816986e-07
std	3.205345e-01
min	-5.876675e+00
25%	-6.830416e-02
50%	-2.082544e-03
75%	6.833898e-02
max	5.969465e+00

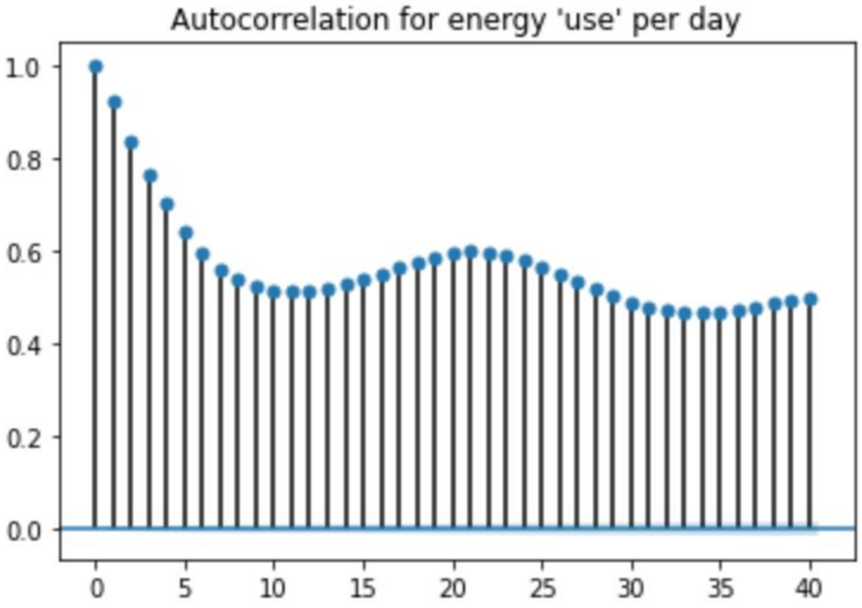
ARIMA model parameters:

The p and q parameters have been decided upon by observing the PACF and ACF plots respectively. Also we have used first order differencing in the data and hence d=1.



```
print("Autocorrelation for 'use' = ", energy_per_day['use'].autocorr())
fig = plot_acf(energy_data['use'], lags=40, title="Autocorrelation for ene
plt.show()
```

```
Autocorrelation for 'use' =  0.6107009825029095
```



Conclusion

- Time-Series Analysis is used to understand trends and inconsistencies in time-related data.
- We have used it to detect abnormalities and forecast data.