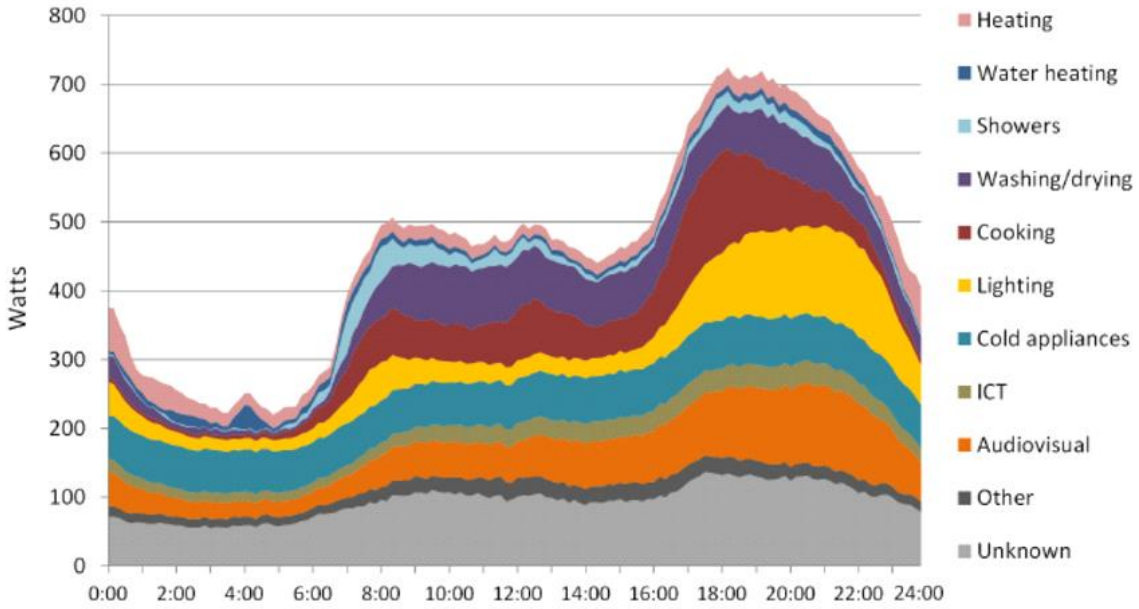
Hourly Energy Consumption



Introduction

Recently, smart building concept has been adapted more frequently as an initiative to create an intelligent space area by taking advantage of the rapid development of computational and communication architecture (Cheng and Kunz, 2009).

This concept is not only limited to Malaysia but other countries as well. General public understanding of smart building concept rotates on the idea of automated process, which is able to automatically control the building's operation through the usage of instrumentation measures and microcontrollers in two-way communication (Qolomany et al., 2019).

Other than automated control, a smart building also consists of an intelligent system which provides energy consumption forecasts as an energy efficiency initiative. This is due to its advantage of yielding economical savings and as a sustainable approach for energy

management to minimize energy wastage (Xu et al., 2018).

A smart energy consumption forecasting is important, especially for buildings as buildings' energy usage is increasing and almost reaches 40% of primary energy use in developed countries (Berardi, 2015).

In Malaysia alone, energy consumption has been increase gradually due to the growth of population. The growth of population lead to the increasing of energy demand in this country and have been estimated to reach 116 million tons of oil equivalents (mtoe) by this year.

Energy provided in Malaysia is influenced by the main fossil fuel sources which included coal, natural gas and fuel oil. Buildings which including commercial, residential and industrial in our country utilises a total of 48% of the electricity that have been created (Hassan et al., 2014).

The increasing of energy consumptions towards buildings from day to day create enforcement to this country in managing and reducing the energy consumption as much as possible in order to improve energy efficiency.

This study is a continuing research from our previous work where previously statistical analysis and k-nearest neighbour (k-NN) method were proposed as the methodologies and SPSS was used as the platform (Mazlan et al., 2020).

In our previous study, only k-NN was proposed as the method to predict energy consumption. It is difficult to know whether the method proposed was the best since there is no comparison had been made.

Hence, another two methods from machine learning are added in

this study.

This research has utilised Microsoft Azure Machine Learning Studio, which is a web service solution for the development of prediction model.

Starting from data analysis until performance evaluation, AzureML has been successfully employed for the implementation of energy demand forecasting.

A major advantage of using Microsoft Azure over SPSS is it is user friendly and easy to use even the user only has basic knowledge in cloud computing and machine learning.

One of the distinguishing features of AzureML was its ability to manoeuvre through a visualization workflow. The workflow that was conducted inside the environment was manipulated through a graphical drag and drop procedure. Other than that, parsing data for experiment was simply done by joining of modules.

Additionally, the platform also supports script packages and algorithms written in external programming language, particularly R programming.

The development of energy consumption predictive models that use statistical analysis and learning methodology possesses several significant challenges.

Attewell and Monaghan (Attewel and Monaghan, 2015) described that statistical prediction is restricted especially in the case of a large dataset with several features, as it requires a higher computational power for modeling.

Other than that, the statistical prediction method itself is comparably weak as it performs better only in the case of stationary time series and high similarity of consumption level (Abdul Karim and Alwi, 2013).

Newsham and Birt (2010) also deduced that time series analysis for electricity consumption forecast performance was unsatisfactory due to variables in the chosen attributes. Moreover, the traditional development of a predictive model is usually based on the trend of maximum demand (kW) consumption only, which is known as a time series method (Xiangyu et al., 2019).

The model development would neglect other electrical parameters such as reactive power changes, which causes the model to be trained only with historical data of maximum demand value. On the contrary, the inclusion of other features of electrical power data would improve the energy consumption prediction (Wei et al., 2019). Therefore, machine learning methodology is preferable when developing a predictive model of electrical consumption.

Changing from the statistical method to machine learning method itself does not solve all the problems with energy consumption prediction.

Missing data that was present on a set of data was well known to cause the performance of the predictive model to deteriorate (Ahmad et al., 2016; Nugroho and Surendro, 2019). This missing data exists usually due to the interconnectivity or sensor problem which is the main complication for smart building energy metering (Ahmad et al., 2016).

Additionally, the development of the machine learning model should utilise a cloud-based service to reduce the dependency of

prediction on the hardware specifications (Mateev, 2019).

Comprehensively, the three critical areas of energy consumption forecast discussed are machine learning prediction methodology, handling of missing data and employment of cloud-based prediction modeling platform which will be the basis of this research.

As the main objective of this research is to develop an energy consumption predictive model for smart commercial building by using several machine learning methods in a cloud-based machine learning platform, this research focuses more on the accuracy of the methodology applied in predicting energy consumption.

Advances in machine learning studies have a tremendous impact on the field of smart building energy management as it is crucial to reduce energy consumption of various types of building from residential buildings to industrial buildings.

Therefore, this study is essential to the following parties such as Ministry of Green Energy and Water (KETTHA) and Malaysia Green Technology Corporations in their determination to analyse energy consumption level of the existing building.

It is also helpful to industrial manufacturing company in predicting electrical loading on their system for long range projection, mapping of capacity versus demand and for factory growth projection.

Last but not least, academician majoring in engineering field to understand the integration of data science and analytics in engineering projects and higher education students to explore the services and possibility of utilising Microsoft Azure Machine Learning Studio for various projects.

Management of missing data

Techniques in handling missing data have been vastly studied before and methodologies have been deduced.

There are two types of methodology that are removing the portion of the data which has missing value and imputation method which is based on close estimation (Hegde et al., 2019). The first method which omit the missing part of data is not feasible as this causes valuable information to be removed (Manly and Wells, 2015).

Without the data, a biased estimation would be made. Therefore, the imputation method is a preferable technique.

Newgard and Lewis (2015) presented several imputation techniques such as Mean Value Imputation, Last Observation Carried Forward, Maximum Likelihood Estimate (MLE) and Multiple Imputation (MI).

The mean value imputation basically substitutes the missing data with the mean value of the dataset. However, this method is not suitable for data which is not strictly random as it will introduce inequality in the data (Kang, 2013).

Another method presented was Last Observation Carried Forward, in which imputation is made for historical data that was collected through (Newgard and Lewis, 2015).

The more advanced methods presented were Multiple

Imputation (MI) and Maximum Likelihood Estimate (MLE) (Newgard and Lewis, 2015).

The Multiple Imputation method substitutes the missing data by gradually supplanting the missing data for every iteration made.

This method utilises statistical analysis based on observed data to handle the uncertainty that is introduced by the missing portion.

An example of a popular MI method is Multiple Imputation Using Chained Equations (MICE) (Azur et al., 2011). Maximum Likelihood Estimate conducts substitution through assumption made by initially identifying the parameters and boundaries based on the distribution of the data.

The imputation would then be made based on the assumed parameters.

This method of imputation was employed by Probabilistic Principal Component Analysis (PPCA). Both advanced imputation methods have been compared by Hegde et al.

(2019) in which imputation method was made on sampled dataset consisting of 87 numeric-converted categorical variables and 29 continuous variables. The study used RMSE metrics to evaluate imputation technique performance.

From the research, the PPCA method showed a much promising result compared to MICE, in which 65% of data

variables were successfully imputed by PPCA and only 38% correct imputation by MICE. This was further supported by Schmitt et al.

wherein the research compared the performance of six imputation methods including PPCA and MICE on a real dataset of various sizes. The result showed that MICE managed to perform well in a small dataset, but in a large dataset case, the MICE method performed poorly.

Support Vector Machine (SVM)

In this research, the Support Vector Machine (SVM) was used with Radial Basis Function (RBF) as its kernel function.

This methodology is usually known as a maximum margin classifier and is utilised to tackle problems regarding classification and regression for a large dataset (Ben-Hur et al., 2008). There are several kernel selections available for SVM method.

In this study, Radial Basis Function (RBF) as shown in equation (5), was chosen due to the broad and non-linear characteristics of the dataset.

is a gamma parameter to determine the spread distribution of the kernel and

is the Euclidean distance between the set of points.

There are 2 tuning parameters for SVM-RBF which are kernel parameter sigma .

and cost parameter (C), that were adjusted for repeated training.

The sigma value plays an important role in getting a good fit model to the data.

Cost parameter is the penalty limit if the data point is misclassified or oversteps maximum margin.

Artificial Nneural Network (ANN

The third methodology in this research for energy consumption prediction was Artificial Neural Network (ANN).

The advantages of using ANN such as its capability to learn complex behaviour, makes it widely used for predictions and pattern recognition (Karunathilake and Nagahamulla, 2017).

ANN model structure consists of a formation of interconnected neurons that have three main layers; input layer, hidden layer, and output layer. By comparing the initial output with the desired output, adjustment of the synaptic weight of each link that connects between the neurons was made until the difference is minimal (minimising Sum Squared Error (SSE)); this would provide regularization for the model (Liu et al., 2019).

The weight is the representation of the priority or importance of the neuron input. For this research, a Multilayer Perceptron Model (MLP) type of ANN structure with error back propagation learning algorithm was used for its network solution structure. In the hidden layer, a suitable non-linear transfer function was used to compute the information accepted by the input layer.

where m is the number of input nodes, n is the number of hidden nodes, f is the Sigmoid Transfer function, is the vector of weights from the hidden layer to the output layer and is the weight from the input to the hidden nodes.

For this research, the hyperparameter tuned was the number of neurons per layer.

This number of neurons denotes the width of the network and its latent space (Weissbart et al., 2019). Another penalizing parameter that was tuned and applied was weight decay.

This parameter is a penalizing method to constrain the complexity of the model and to limit the growth of the model's weight parameter (Gnecco and Sanguineti, 2009).

Step 4: model evaluation (test)

Before inputting the data to the machine learning algorithm, the data was partitioned into two groups whereby 70% of the dataset was used for training and the other 30% was partitioned as testing data groups.

The training groups of data were used to train each machine learning algorithm and generate a predictive model that could output value that matches with the recorded maximum demand data while the rest of the data was held back to be used to test the trained predictive model.

The process is as illustrated in With AzureML, data partitioning for training and testing would not be a hassle and biased as it has built-in support for data division. The partitioning process was straightforward in which selection was made randomly.

This process prevented overfitting, which could cause either underestimation or overestimation of the maximum demand value.

During model training, several models were created with different tuning parameters for each method, in which k-value was adjusted for k-NN tuning; sigma and C parameter were modified for SVM-RBF tuning; and weight decay and hidden unit size were adjusted for ANN-MLP tuning.

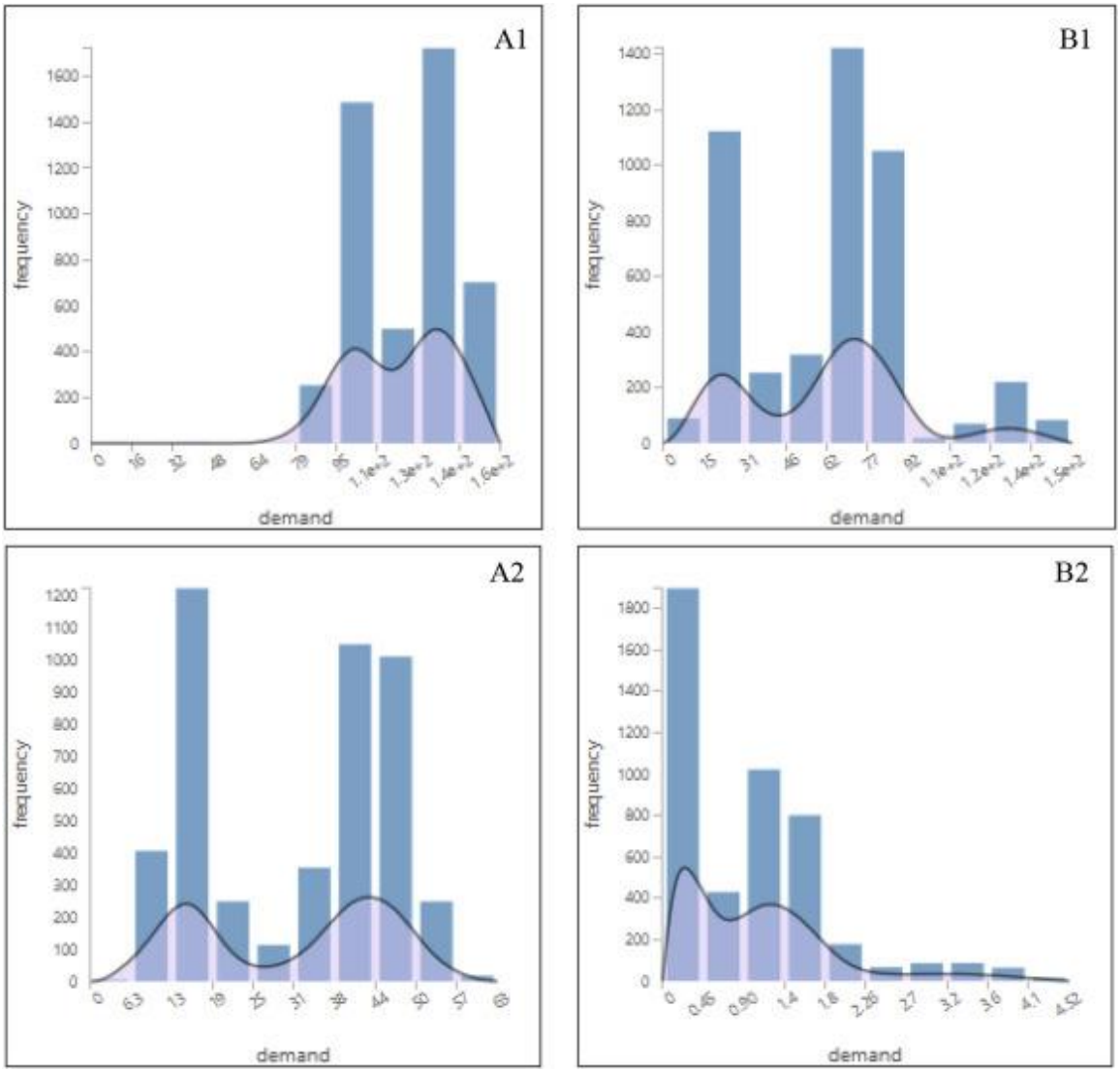
After the repeated tuning finished up to its respective maximum parameters, each model was evaluated based on Root Mean Square Error (RMSE), R-Squared (R2) and Mean Average Error (MAE). The formula is as shown in equations (8), (9), (10), respectively, given that At is the actual recorded values of maximum demand data, and Ft is the predicted values.

Although 3 evaluations were made, only RMSE result was acknowledged as the best model for each method.

After the predictive model using each machine learning algorithm was developed and prediction demand data was generated, they were then evaluated to determine their performance and accuracy.

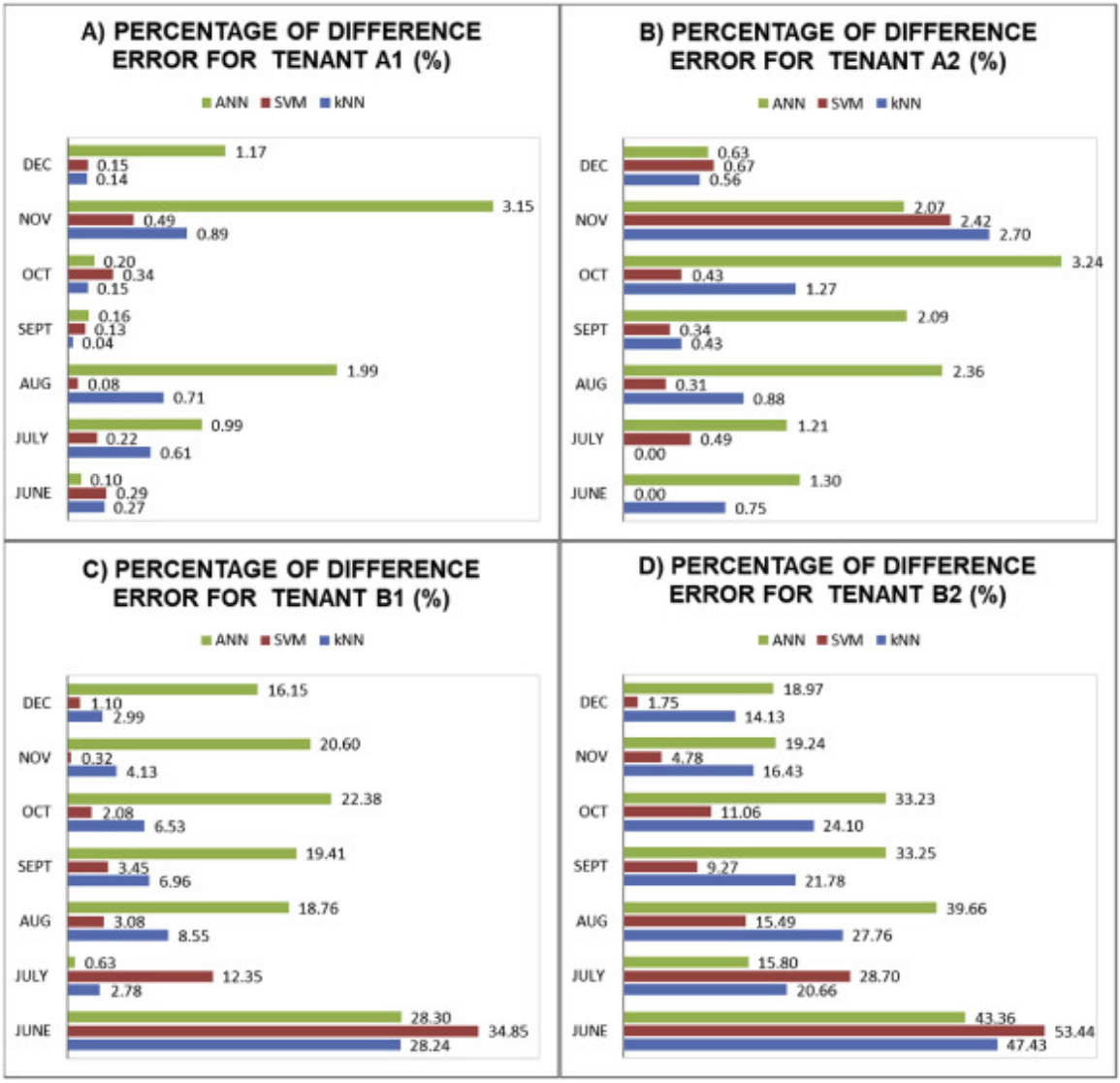
Three methods of evaluation were used which were Root Mean Square Error (RMSE), Normalised RMSE and Mean Absolute Percentage Error (MAPE). The formula for RMSE is as shown in equation (8) and MAPE in equation (11).

Comparison of performance of the methods to different tenants was made by using a normalised RMSE or also known as Coefficient of Variation RMSE (CV RMSE). This metric removes the scale dependent of RMSE (Botchkarev, 2019).



Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.



Program

EXAMPLE:1

```
# We can add multiple fourer series with different k terms - (2*k*pi)
such as k=1,2,3...etc. To generalize the problem,

                                                    # we could have chosen an
                                                    optimal k value for each
                                                    season by trying out some k
                                                    values and choosing the
                                                    values giving
                                                    # the lowest AIC score.
```

```
def add_fourier_terms(df, year_k, week_k, day_k):
    """
    df: dataframe to add the fourier terms to
    year_k: the number of Fourier terms the year period should have. Thus the model will be fit on 2*year_k terms (1 term for
            sine and 1 for cosine)
    week_k: same as year_k but for weekly periods
    day_k:  same as year_k but for daily periods
    """

    for k in range(1, year_k+1):
        # year has a period of 365.25 including the leap year

df['year_sin'+str(k)] = np.sin(2 *k* np.pi * df.index.dayofyear/365.25)

df['year_cos'+str(k)] = np.cos(2 *k* np.pi * df.index.dayofyear/365.25)

    for k in range(1, week_k+1):

        # week has a period of 7

df['week_sin'+str(k)] = np.sin(2 *k* np.pi * df.index.dayofweek/7)
```

```
df['week_cos'+str(k)] =
np.cos(2 *k* np.pi *
df.index.dayofweek/7)


    for k in range(1,
day_k+1):

        # day has period of
24

df['hour_sin'+str(k)] =
np.sin(2 *k* np.pi *
df.index.hour/24)

df['hour_cos'+str(k)] =
np.cos(2 *k* np.pi *
df.index.hour/24)
```

EXAMPLE:2

```
# Fitting Elastic Net Regression model using the lag
variables as the additional inputs ...
```

```
#.. with hours reduced to time_of_day
variables, months to sesonal variables
(winter and summer) and weekdays and
holidays to
#...non_working or workign days.

# Not tuning the elastic net this time
because we will see that there hardly
any scope for improvement in the
excellent results
#..we get

#these columns will be scaled using
StandardScaler
cols_to_transform =
['HourlyDryBulbTemperature',
```

```
'cum_AC_kW', 'year']
# Adding the energy consumption lags to
the columns to transform
list_lags = ['lag'+str(i+1) for i in
range(24)]
cols_to_transform.extend(list_lags)

X_train_lag, X_test_lag, y_train_lag,
y_test_lag = train_test(lag_sdge, \

test_size = 0.15, scale = True, \

cols_to_transform=cols_to_transform)

elastic_net_lag = ElasticNet(l1_ratio
= 1, alpha=0.2)
elastic_net_lag.fit(X_train_lag,
y_train_lag)

#Output model
>> ElasticNet(alpha=0.2, copy_X=True,
fit_intercept=True, l1_ratio=1,
..           max_iter=1000,
normalize=False, positive=False,
precompute=False,
..          random_state=None,
selection='cyclic', tol=0.0001,
warm_star
```

Conclusion

The aim of this project was to identify the variables that influence the generation, the consumption and the price of the electricity in United States.

We have seen that the generation of electricity in American states is driven by the number of commercial and industrial customers. Concerning the electricity consumption, it is influenced by the energy production itself and the amount of commercial customers. Our prediction models are quite accurate and confirmed the results of our exploratory data analysis. About our models, we should not forget that lots of variables can explain the electricity consumption and production as we have seen during the eploratory data analysis, but we only used the most significant ones.

For the structure of the electricity production, we have seen that the energy mix varies tremendously from one region to another and from one state to another. We cannot determine whether a mix defines the price per KWh or not. However, power generation using coal and hydropower is correlated with low energy costs. In addition, KWh prices will be higher for states belonging to the following regions:

- Middle Atlantic
- New England
- West Pacific

Even if we considered to study the fluctuation of the electricity price for the next years, it turned out impossible to achieve a result. Therefore, it would be interesting to analyse accurate external data such as the weather, the cost price per KWh per energy, the political decisions, etc. Those aspects have a direct influence on the price of energy.

Finally, regarding the Californian state, we were able to model the average power per hour over a year. We see that renewable energies are subject to seasonality. The power of renewable energies is highly volatile, which makes them difficult to predict. For this reason, it is more difficult to predict these data in a very short term.