Parametric and Non-Parametric Time Series Analysis for Power Usage Data

Sajjad Azami and Mahdi Taherahmadi

Cognitive Robotics Lab, Amirkabir University of Technology (Tehran Polytechnic),
No. 424, Hafez Ave., Tehran, Iran. P. O. Box 15875-4413.
{sajjadaazami,14taher}@gmail.com

1 Introduction

The analysis of experimental data that have been observed at different points in time leads to new and unique problems in statistical modeling and inference. In this project, our goal is to predict the power usage of Ontario state, Canada, using recorded usage data from January 2002 to December 2016. We will implement parametric and non-parametric models and compare them with respect to various evaluation metrics. For this purpose, offered models are described and implemented using Python and results are shown in the next sections. Finally, we will provide conclusions about time series analysis based on the current dataset.

2 Related Works

As noted in the previous section, the dataset we are dealing with is a time series record of power usage, so the problem ahead is classified as Time Series Analysis.

A time series is defined as a series of data points indexed (or listed or graphed) in time order. Most commonly, a time series is a sequence taken at successive equally spaced points in time. Time series analysis may be divided into two classes: frequency domain methods and time domain methods. As mentioned before, our goal is to estimate power usage of future periods, so we need to use time domain methods which include auto-correlation and cross-correlation.

Time series are mostly analyzed using a parametric model, in this project we will also implement a non-parametric model. Parametric models assume that the underlying data generating stationary stochastic process has a certain structure. This structure is described by parameters from parameter space . In contrast, a non-parametric model estimates target feature of the process without assuming any particular structure.

Curve Fitting is one of the popular methods in this context. It is the process of constructing a curve, or mathematical function, that has the best fit to a series of data points. Depending on our goal, various settings can be considered for the curve, for example, interpolation when exact fit of data is required or smoothing when we need an approximate fit. Classification is another method for time series analysis. In this approach, we assign each pattern of the signal

to a specific class. This method is useful in applications like hand movement recognition or speech recognition.

Another method is Segmentation. In this approach, the underlying time series is split into a sequence of segments. For example, the audio signal from a conference call can be partitioned into pieces corresponding to the times during which each person was speaking. In time-series segmentation, the goal is to identify the segment boundary points in the time-series, and to characterize the dynamical properties associated with each segment.

3 Implemented Methods and Evaluation

In this section, we are going to implement parametric and non-parametric models and evaluate them. Note that for all the models, we will use about 6% of the data as test set which relates to data points of the year 2016. The rest is used as training set.

3.1 Data Set Characteristics

Before explaining model implementations, we will study some features of the given data set. The plot for 336 data points is shown in fig 1. This period represents two weeks of power usage. By looking at this plot, first, we can understand that this series is stationary. It means the mean is not a function of time. Second, it is homoscedastic, meaning the variance also is not a function of time.

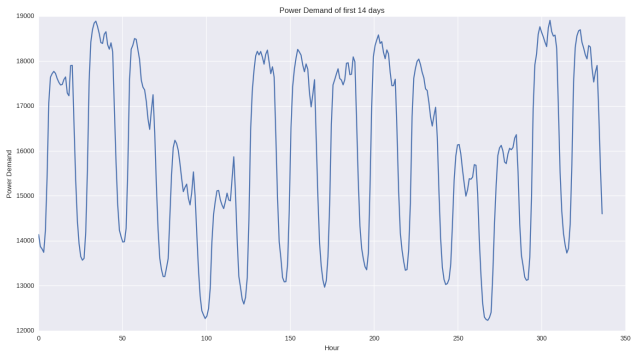

Fig. 1. Power demand of first 14 days.

Next, we need to check correlation. A lag plot checks whether a data set or time series is random or not. Random data should not exhibit any identifiable structure in the lag plot. Non-random structure in the lag plot indicates that the underlying data are not random. In Fig 2, Lag Plot is represented.

It is obvious that there is a strong correlation between time series values from the lag plot.

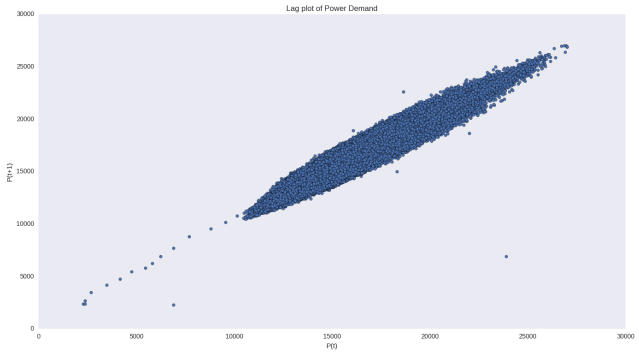

Fig. 2. Lag plot of train data.

The pandas library provides an autocorrelation function for plotting Autocorrelation of data versus lagged values. This is shown in Fig 3.

This can very quickly give an idea of which lag variables may be good candidates for use in a predictive model and how the relationship between the observation and its historic values change over time.

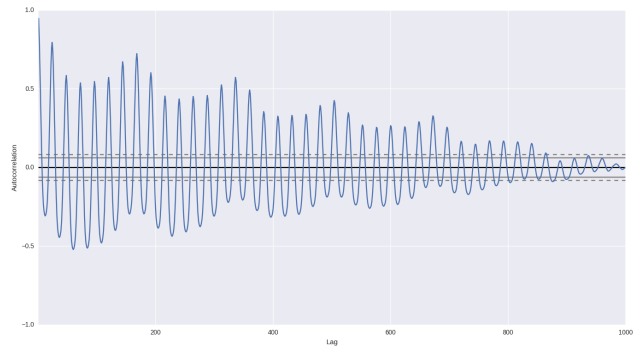

Fig. 3. Autocorrelation of data versus lagged values.

3.2 Persistence Model

We start by implementing a simple non-parametric model, named Persistence Model. The simplest model that we could use to make predictions would be to persist the last observation. We can call this a persistence model and it provides a baseline of performance for the problem that we can use for comparison with an autoregression model.

In this method, as we decrease the period between train and test data the probability of getting better results increases. It means, for example, if we wanted to estimate time point ts power usage, as our train data time period gets closer to t, we will probably get better results. In an extreme case, if we wanted to predict next hours usage, we better have access to last hours usage. But in real life cases, this is not usually feasible, so for this model, we change this period to 48 hours.

After creating the model and fitting it to test set, Mean Absolute Error is calculated by $\frac{1}{n}\sum_{t=1}^{n}|e_t|$. MAE error for this model is 2417.34. Fig 4 shows part of fitted line on test data.

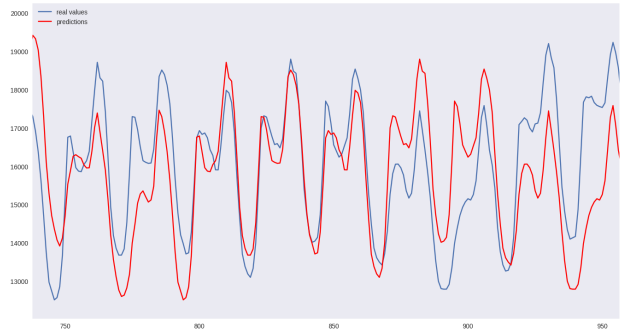

Fig. 4. Part of fitted line versus real values of data.

One of the biggest weaknesses of this model is that it is too sensitive to shock in data. Meaning if we have a sudden change in series, the model will react slow to it and it will increase the error. For example, if time t and t+1 differ largely, in this case in range of more than 4000, the model will fail to predict well. But in this case, as we know we are analyzing power usage, we are aware that there is no sudden change unless a problem occurs like blackouts, which are considered as outliers. Though this can cause error in other contexts like financial time series analysis.

3.3 Auto Regression Model

For the next model, we have implemented an Auto Regression model Linear-Regession class in scikit-learn and manually specifying the lag input variables to use. An autoregression model is a linear regression model that uses lagged variables as input variables. We use 48-hour lag for this model too.

As the name suggests, this is a regression family model. We know that regression models are parametric, because we try to estimate coefficients of the fitted curve, which are the parameters of interest.

Like previous section, we fitted the curve on 94% of data and tested it on 6% which belongs to 2016 data points. Mean absolute error is calculated by formula 1 and the result is 2098.5. A small part of fitted curve is shown in fig 5.

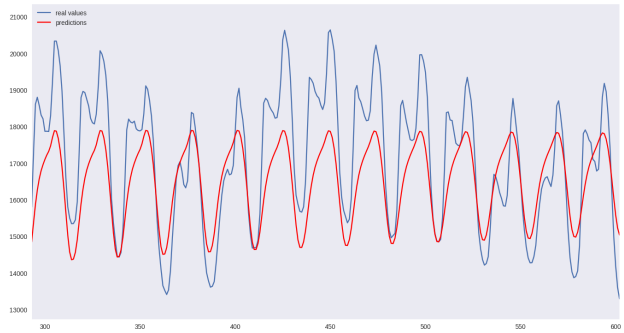

Fig. 5. Part of fitted line versus real values of data.

Coefficients for fitted curve are stored in an array like below:

$$\begin{bmatrix} 1.05e+02 & 1.53e+00 & -7.06e-01 & 1.61e-01 \\ -2.05e-02 & 2.46e-03 & 2.34e-02 & -6.23e-02 \\ & \dots & \dots & \\ 3.53e-02 & -1.47e-02 & -6.66e-03 & 2.16e-02 \\ -7.96e-03 & -1.53e-02 & 4.23e-02 & -1.64e-02 \end{bmatrix} \tag{1}$$

Fig 6 shows the whole fitted curve on 2016 data. This shows us how autoregression models performance weakens over time.

4 Model Comparison and Conclusion

As seen in previous section and with prior knowledge, in time series analysis, parametric models usually perform better than non-parametric ones. But obviously, parametric models are slower to learn, test and implement, especially when data series becomes complex.

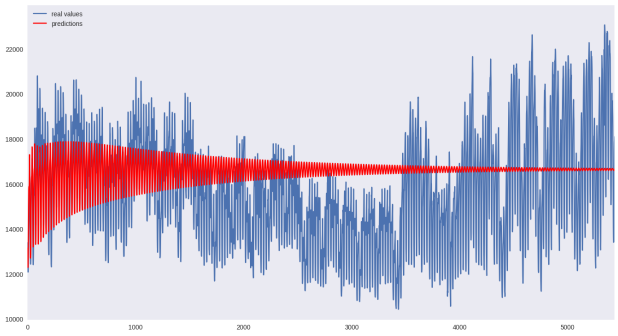

Fig. 6. Whole fitted line using autoregression model.

Finally, we have predicted power usage from 1st January to 10th February 2017. Results are attached to document in csv format.

5 References

1. Arlinghaus, S., 1994. Practical handbook of curve fitting. CRC press.
2. Shumway, R.H. and Stoffer, D.S., 2010. Time series analysis and its applications: with R examples. Springer Science Business Media.
3. "Time Series". En.wikipedia.org. N.p., 2017. Web. 10 Feb. 2017.
4. Qiang, X., Rui-Chun, H. and Hui, L., 2014. Data Mining Research on Time Series of E-commerce Transaction. International Journal of u-and e-Service, Science and Technology, 7(1), pp.9-18.
5. Hrdle, W., Ltkepohl, H. and Chen, R., 1997. A review of nonparametric time series analysis. International Statistical Review, 65(1), pp.49-72.