

**DEPARTMENT OF INFORMATION & DECISION SCIENCES**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Course: IDS 476** - **Business Forecasting (Fall 2019)**

**Project Report**

**Time Series Analysis and Forecasting of Household Energy Consumption**

|  |  |
| --- | --- |
| **Project Team** | |
| Sraavya Pathsamatla (662075657) | Ziding Zhao (662024870) |
| Aruna Venkatasubramaniam (678860489) | Amrita Parab (675926967) |
| Subhasree Vaidyanathan (668163416) | Meenaakshi Janardanan (671917004) |
| Rahul Shah (669792302) | Apeksha Sharma (653386893) |

# 

|  |
| --- |
| **Under the Guidance of:** |
| 1. Professor Stanley Sclove |
| 1. Haesung Kim |

# **Abstract**

Data in the form of a time series is emerging as one of the fastest growing categories in databases. Examples such as market fluctuations, smart home monitoring (energy usage pattern), Remaining Useful Life (RUL) estimations from IOT sensor data etc. rely on a form of data that measures how things change over time. Analyzing and monitoring how such a time series data is changing in the past and in the present can allow us to predict how it can change in the future.

Time series data can be analyzed using models ranging from traditional methods such as Autoregressive Integrated Moving Average (ARIMA) to more modern learning techniques such as Long-Short Term Memory Networks (LSTM). Whereas many traditional methods assume the existence of an underlying stochastic model, algorithmic approaches such as LSTM make no claims about the generation process.

# **1.** **Introduction**

The prediction of energy consumption is a key component in the power flow management of the electrical grid. As electric power consumption can be measured over time intervals, for example on a minute basis, an approach to predict future energy consumption is to look at past consumption and their patterns over time to then re-use these patterns for prediction.

* In this project, our goal is to predict values for a time series given the history of 2 million minutes of a household’s power consumption.
* We will use a multi-layered LSTM Recurrent Neural Network to predict the last value of a sequence of values. This will help in forecasting the results based on the historical data.
* We will implement a comparative study between LSTM and RNN to analyze the pattern in the household energy consumption. We want to compare the prediction accuracy of the two models and analyze how well each model forecasts out-of-sample data.
* Finally, we will generate an optimal forecast for the results based on the historical data.

## **1.1. Dataset description**

The data that we are using consists of measurements of electric power consumption in one household with a one-minute sampling rate over a period of almost 4 years.

### **1.1.1. Source of data**

Ref.: UCI Machine Learning Repository

http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption

<https://otexts.com/fpp2/nonlinear-regression.html>

### **1.1.2. Codebook**

### We have a total of 66497 rows and 2 columns. The variables in our dataset are:

1. Datetime
2. COMED\_MW (household global hour-averaged reactive power (in kilowatt))

## **1.2. Purpose of the analysis of these data**

The consumption in the residential sector represents a significant percentage in the total electricity demand all over the world and it is expected to grow. The biggest concern of power system operators is to maintain the balance between generation and load. So, the prediction of energy consumption becomes a key component in the management (e.g. power flow) of the electrical grid.

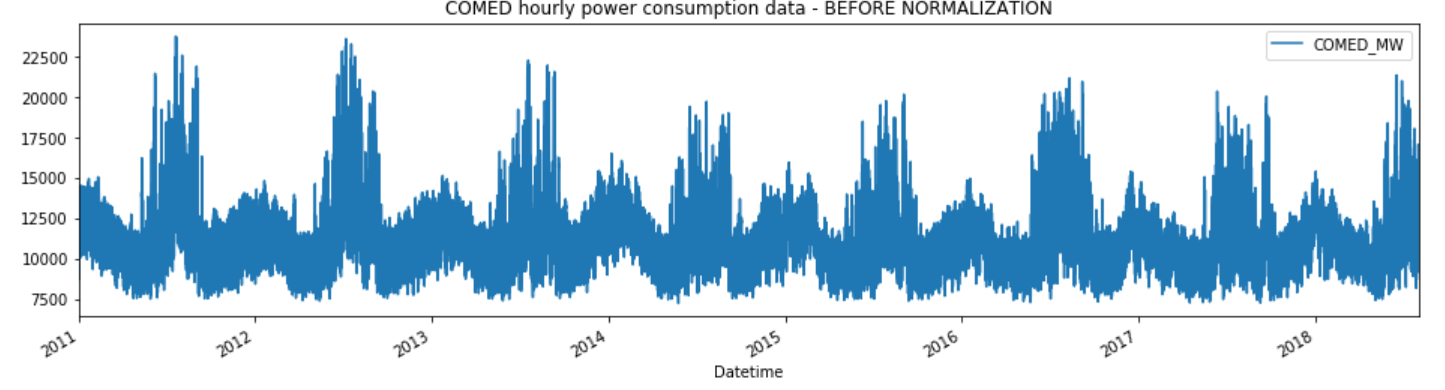
# **2. Statistical Analysis**

## **2.1. Pre-processing of the data**

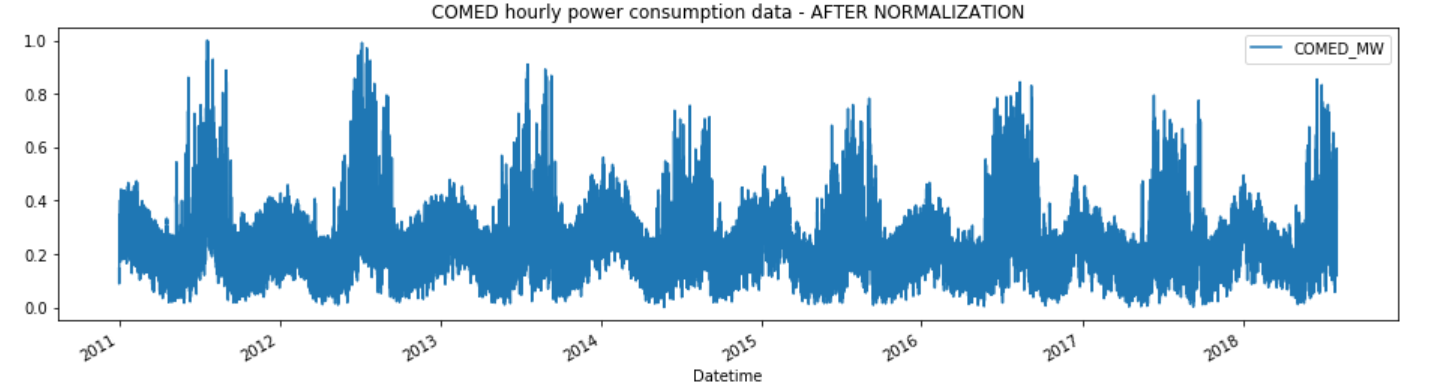
Firstly, the relevant libraries are imported, and data processing is carried out. Data is further normalized using MinMax Scaler and reshaped for analysis with LSTM model. The data for time series sequence prediction problem needs to be scaled when training a neural network, such as a Long Short-Term Memory (LSTM) recurrent neural network.

When a LSTM network is fit on unscaled data that has a range of values (e.g. quantities in the 10s to 100s) it is possible for large inputs to slow down the learning and convergence of your network and in some cases prevent the network from effectively learning your problem. Hence the data has been normalized for the effective optimization of the outputs from LSTM.

Before Normalization:



After Normalization:



## **2.2. Software used**

1. Python 3.7.5
2. Jupyter Notebook

## **2.3. Procedures used**

### **2.3.1 Why LSTM?**

Classical machine learning models based on time series forecasting are much difficult to implement compared to the supervised and unsupervised learning models because of the temporal difference in the data: we work on the data plotted against the same data at a different time step. This makes the process of model-fitting and model evaluation relatively difficult.

ARIMA is a very popular tool in the market, because of the reason that it can be used for almost any kind of time series data and is quite easy to understand and is effective in its implementation. There are 3 main limitations to classical models that can be overcome by deep learning methods, as discussed below.

Limitations of Classical models: (like Holt-Winters, ARIMA - based, other Exponential models)

1. Missing values are not supported.
2. Assuming that the data has linear relationships i.e., showing the trend component has overall downward or upward trend. We can see that the linear trend line shows a generalized downward trend in the data. Whereas the “cubic spline”, which is a non-linear trend line, captures a more intuitive pattern (non-linear pattern) present in the data.
3. These models work on uni-variate data. Most of the models in time series forecasting don’t support multiple variables to be taken as inputs. We focus on only one variable: the outcome of interest is nothing but the future version of the input variable (at a future time step). Exceptions to this case are models like seasonal ARIMA or SARIMA that accept exogenous variables to model the data.

One of the fundamental problems which plagued traditional neural network architectures for a long time was the ability to interpret sequences of inputs which relied on each other for information and context. This information could be temporal information of a sequence which would allow for context on the time based elements of that sequence.

Simply put, traditional neural networks take in a stand-alone data vector each time and have no concept of memory to help them on tasks that need memory.

An early attempt to tackle this was to use a simple feedback type approach for neurons in the network where the output was fed-back into the input to provide context on the last seen inputs. These were called Recurrent Neural Networks (RNNs). Whilst these RNNs worked to an extent, they had a rather large downfall that any significant uses of them lead to a problem called the Vanishing Gradient Problem.This is where the Long Short Term Memory (LSTM) neural network could be used for the following reasons:

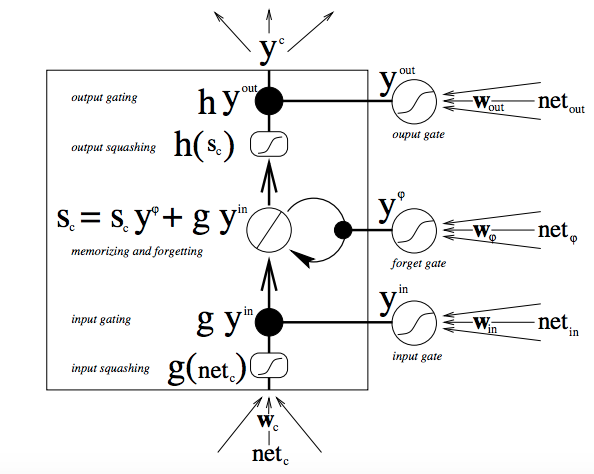
* Multivariate input
* Robustness to noise
* Multivariate output
* Automatic feature extraction, modeling the more complex relationships in the data

### **2.3.2 Long-short-term memory (LSTM) networks:**

Long-short-term memory (LSTM) networks are a special type of recurrent neural networks capable of learning long-term dependencies. LSTMs help preserve the error that can be backpropagation through time and layers. By maintaining a more constant error, they allow recurrent nets to continue to learn over many time steps (over 1000), thereby opening a channel to link causes and effects remotely. This is one of the central challenges to machine learning and AI, since algorithms are frequently confronted by environments where reward signals are sparse and delayed, such as life itself. LSTMs contain information outside the normal flow of the recurrent network in a gated cell. Information can be stored in, written to, or read from a cell, much like data in a computer’s memory. The cell makes decisions about what to store, and when to allow reads, writes and erasures, via gates that open and close. Unlike the digital storage on computers, however, these gates are analog, implemented with element-wise multiplication by sigmoids, which are all in the range of 0-1. Analog has the advantage over digital of being differentiable, and therefore suitable for backpropagation.

Those gates act on the signals they receive, and similar to the neural network’s nodes, they block or pass on information based on its strength and import, which they filter with their own sets of weights. Those weights, like the weights that modulate input and hidden states, are adjusted via the recurrent networks learning process. That is, the cells learn when to allow data to enter, leave or be deleted through an iterative process of making guesses, backpropagating error, and adjusting weights via gradient descent.

The diagram below illustrates how data flows through a memory cell and is controlled by its gates.



# **3. Results and Conclusions**

## **3.1. Model Results and Plot Predictions**

|  |  |
| --- | --- |
| **Models Implemented** | **R-Square Score** |
| **Simple RNN Model** | 84.79% |
| **LSTM Model** | 69.78% |

|  |
| --- |
| **Plot Predictions for Models Implemented** |
| 1. **Predictions for Simple RNN Model** |
|  |
| 1. **Predictions for LSTM Model** |
|  |

From the above results we see that Simple RNN model with a R-Squared Score of about 85% is able to forecast much better on the out-of-sample data compared to the LSTM model. Thus, for this example, Simple RNN model proved to be quite accurate at predicting fluctuations in electricity consumption.

Predicting the energy consumption in dwellings is an essential part in the power management of the grid, as the consumption in the residential sector represents a significant percentage in the total electricity demand.