**DeepAR Electric Vehicle Demand Prediction**

Jose Peeterson

Research Engineer

NTU-Continental Corporate Lab

Table of Contents

[1. Introduction 4](#_Toc124944210)

[1.1 Problem statement 4](#_Toc124944211)

[1.2 Proposed solution 4](#_Toc124944212)

[1.2.1 Vehicle Rebalancing 4](#_Toc124944213)

[1.2.2 Summary of feasible Demand Prediction solutions 5](#_Toc124944214)

[2. Data Wrangling 6](#_Toc124944215)

[2.1 Spatial and Temporal Aggregation 6](#_Toc124944216)

[3. Exploratory Data Visualization and Analysis 8](#_Toc124944217)

[3.1 Input data 8](#_Toc124944218)

[3.2 Data exploration of Target time series 8](#_Toc124944219)

[3.3 Data exploration of Covariate time series 10](#_Toc124944220)

[3.1.2 Time series plot 10](#_Toc124944221)

[3.1.3 Scatter plot 10](#_Toc124944222)

[3.1.2 Seasonality 10](#_Toc124944223)

[3.2.1 Data distribution 12](#_Toc124944224)

[3.2.2 FFT 12](#_Toc124944225)

[3.2.3 Stationarity 13](#_Toc124944226)

[3.2.4 ACF 13](#_Toc124944227)

[3.2.5 PACF 13](#_Toc124944228)

[3.2.6 Relationships between target and covariates 13](#_Toc124944229)

[4. Baselines 14](#_Toc124944230)

[4.1 Persistence Model 15](#_Toc124944231)

[4.2 Statistical Models 15](#_Toc124944232)

[4.2.1 Negative-binomial Autoregression 15](#_Toc124944233)

[5. DeepAR 15](#_Toc124944234)

[5.1 Sequence to Sequence Models 15](#_Toc124944235)

[5.2 Working Principle 15](#_Toc124944236)

[5.3 Data Pre-processing 15](#_Toc124944237)

[5.3.1 Feature Engineering 15](#_Toc124944238)

[5.4 Dataset Creation 15](#_Toc124944239)

[5.5 Architecture 15](#_Toc124944240)

[5.5.1 Block diagram 15](#_Toc124944241)

[5.5.2 Mindmap 16](#_Toc124944242)

[5.5.2 Encoder length considerations 16](#_Toc124944243)

[6. Training 16](#_Toc124944244)

[7. Validation 16](#_Toc124944245)

[7.1 Hyperparameter Tuning 16](#_Toc124944246)

[7.1.2 Optuna 16](#_Toc124944247)

[7.1.3 Observation: s (Check further tuning page) 17](#_Toc124944248)

[7.2 Model Troubleshooting 17](#_Toc124944249)

[7.2.1 Error Analysis 17](#_Toc124944250)

[7.3 Feature Importance 17](#_Toc124944251)

[7.4 Model Explainability 17](#_Toc124944252)

[8. Testing 17](#_Toc124944253)

[8.1 Cluster-wise Results 17](#_Toc124944254)

[8.1.1 Tampines 17](#_Toc124944255)

[8.1.2 Central 18](#_Toc124944256)

[8.1.3 Woodlands 18](#_Toc124944257)

[Region-wise Results 18](#_Toc124944258)

[Tampines 18](#_Toc124944259)

[9. Code Troubleshooting and debugging 18](#_Toc124944260)

[9.1 Code repository 18](#_Toc124944261)

[10. Dashboard Visualization 18](#_Toc124944262)

[11. Gaps and Improvement suggestions 18](#_Toc124944263)

[12. Appendix 19](#_Toc124944264)

[11.1 FAQ 19](#_Toc124944265)

[11.2 Detailed Block Diagram 21](#_Toc124944266)

# Introduction

## 1.1 Problem statement

Station-based EV (Electric Vehicle) systems (human drivers or autonomous) suffer from **spatial-temporal supply-demand imbalance** problem due to the disparity in the popularity of the stations, i.e., there are no EVs at stations where the customers need them, while at the same time there are idle standing EVs at other stations. This leads to sub-optimal fleet utilization, high waiting times for the customers and unmet demands.

## 1.2 Proposed solution

### 1.2.1 Vehicle Rebalancing

To mitigate this problem, there is a requirement for a system that can control the EV distribution among the stations.

More specifically, a system that can calculate which idle standing EV should be removed from the station where it is currently parked (where it has a low chance of getting rented) and moved to which destination station - vehicle rebalancing decisions.

Vehicle rebalancing decision is a complex decision that requires information about the idle standing vehicles, en-route vehicles, upcoming demands, and available parking station capacity to be considered.

Diagram

Description automatically generated

Figure 1: Vehicle Rebalancing Solution block diagram with demand prediction blocks

### 1.2.2 Summary of feasible Demand Prediction solutions

Electric Vehicle (EV) Demand Prediction is necessary to perform effective EV rebalancing and meet future demand to maximise the expected profits at the end of a period. Historical EV demand data is considered as a count time series made up of small positive integer values. This data can be supplemented by information such as present and future weather, vehicle request queues, subscriber driving patterns, locality patterns and date information.

Classical time series Prediction techniques such as ARIMA are mainly tailored for real values that change continuously and exhibit visible trends and seasonality. However, count time series are made-up of sporadic spikes that show no visible patterns to the naked eye. Deeper analysis of the count time series using time series decomposition, Autocorrelation and Partial autocorrelation plots are required to reveal trends, seasonality and autoregressive terms that will help to model the count time series using probabilistic methods.

Count time series prediction models can be broadly classified into Statistical and Machine Learning techniques.

Statistical count time series modelling approaches include historic averaging, exponential smoothing, Poisson and Negative Binomial autoregression and Zero-inflated models. These models have the advantage of being interpretable, but they are not complex enough to capture all the patterns and correlations in the data. These models are also labour intensive as they require manual analysis for model parameter selection. They also require some contextual information provided manually to group the right time series when performing cross correlations between time series from the right clusters that may be geographically distant from each other. Statistical models are generally used as baselines to compare with machine learning techniques.

Machine learning techniques for Demand Prediction include Autoregressive Recurrent Neural Networks that can predict the probability distributions of data at different time intervals. Albeit less interpretable, they can handle cross-correlations between different clusters at different times for all the time series automatically. They can learn the complex patterns in data and improve prediction accuracy.

# Data Wrangling

## 2.1 Spatial and Temporal Aggregation

The instantaneous historic EV inflows and outflows at all charging stations are aggregated every hour. A cluster is then formed by grouping a few neighbouring charging stations. A cluster is chosen such that all stations within the cluster are no farther than 0.5KM from each other. In this way a person making a request from within the cluster can easily walk to the EVs.

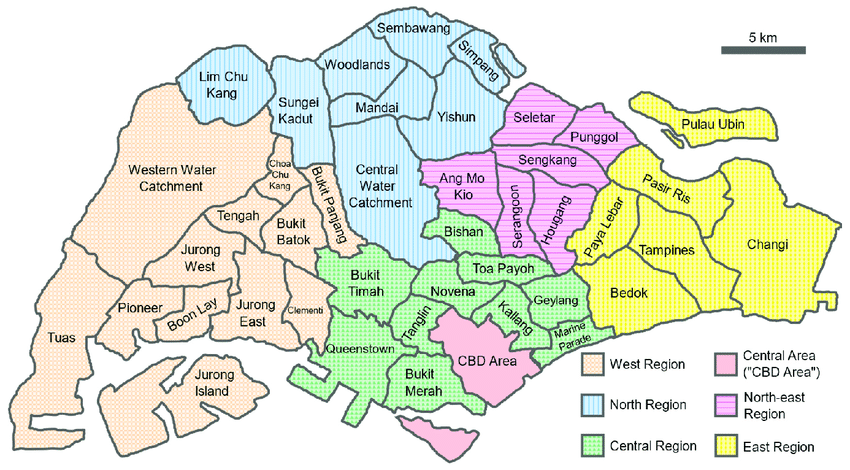


Figure 2: All regions in Singapore

Map

Description automatically generated

Figure 3: Clusters (distinct colour) marked in the Tampines region

Map

Description automatically generated

Figure 4: charging stations (same colour) in each cluster of the Tampines region

EV Demand Prediction is typically performed at a cluster level every 1-hour time steps for a few time steps into the future. Cluster-wise hourly predictions ensure a good trade-off between Demand Prediction accuracy and station-level predictions. The demand data at station-level is very sparse and therefore requires some form of aggregation temporally or multiple station-wise to make any patterns discernible. It is imperative that the models learnt are based on data that show some underlying persistent and learnable patterns so that the predictions will be accurate and consistent.

# Exploratory Data Visualization and Analysis

## 3.1 Input data

In cluster-wise prediction, the target time series is the EV demand time series of a **single** cluster within a region. The covariate time series are weather data, EV supply and date information at the cluster. The covariates also include known lagged versions of the demand time series based on seasonality. The weather data is composed of a categorical time series called weather description and two numerical time series called humidity and temperature. Date information includes 7 categorical time series including hour-of-the-day, day-of-the-week, day-of-the-month, day-of-the-year, week-of-the-year, month-of-the-year, and year.

In region-wise prediction, the target time series is the demand time series of **all** the clusters within a region. The covariates from all the clusters in the region are included.

## 3.2 Exploration of Target time series

Covariate information may not be necessary every time as most of the information needed to predict the target time series can be found within the target time series itself. Therefore, it is necessary to explore the persistent underlying patterns that is propagated in the target time series.

***Identifying these patterns in advance through exploratory studies enables us to cross-check the existence of these patterns in the predictions of the learned models. Thus, ensuring that the models also learn these patterns to make accurate predictions.***

***These exploratory studies aid in troubleshooting wrong predictions from models. It identifies which patterns are not learnt by the model.***

Here onwards, 9 weeks of hourly demand time series of Cluster 175 of the Tampines region is chosen as the target time series for illustration.

Chart, bar chart

Description automatically generated

Figure 5: 9 weeks of Hourly demand time series of cluster 175 in Tampines region

Visually, the target time series does not have a trend, and this is true for most of the clusters in Singapore. The mean is relatively constant around 2 cars for cluster 175 as shown above. Therefore, exploring seasonality in detail is more meaningful for the demand time series.

The following seasonal patterns are explored

1. Hourly seasonality

* FFT and IFFT of demand time series
* Total distribution at each hour of the day. (Boxplot)

1. Daily seasonality

* Hours of same day across weeks
* Total distribution at each day of the week. (Boxplot)
* Trend cycle within a day

1. Weekly seasonality

* Different week across weeks
* Summed hours of days of week across weeks
* Total distribution at each week of the data. (Boxplot)
* Trend cycle within a week
* Trend cycle within a month ~~(~~sub-seasonal charts~~)~~

1. Indirect and direct Correlation due to lags

* ACF
* PACF

### 3.2.1 Hourly seasonality

#### 3.2.1.1 FFT and IFFT of demand time series

**Purpose**: Fast Fourier Transform (FFT) of the hourly demand time series can identify cyclic behaviour like business/economic cycles that may be unique to the cluster and do not follow the regular seasonal patterns. It is difficult to make educated guesses of these patterns. E.g., for residential clusters, there is a regular activity every 5-6 hours indicating that people go to office in the morning, go for lunch in the afternoon, go to shopping in the evening, go for dinner at night and go home late at night and repeat the cycle.

Through FFT, the significant frequencies and their phase can be identified. These frequencies account for the largest peaks of the demand time series.

Reconstructing the demand time series using Inverse Fast Fourier Transform (IFFT) by only including the significant frequencies allows for locating the peak occurrences at specific times of the day and week. This helps to identify the busy times at the cluster. It will be advantageous to ensure sufficient supply at this cluster during these times. The prediction model is expected to not make very low or other erratic predictions at these times.

Graphical user interface, application

Description automatically generated

Figure 6: FFT of demand time series for 9 weeks

The FFT of weekly time series for 9 weeks is given in appendix.

**Observation**:

* Full 9 weeks data
  + Persistent periods: 12, 24,177.33, 4.8, 8.02 hrs
* Weekly data for 9 weeks (in appendix)
  + Persistent periods across weeks: 12, 24, 4.8, 8 hrs

**Inference**:

## 3.3 Exploration of Covariate time series

2) Visualize Correlation between covariate and target time series

Plot scatter plots between covariates

a. Co-integration between lagged supply and demand

b. Do same Visualization as target time series for all the covariate time series as well. Try to associate non-seasonal high demand with covariate.

c. Add bubbles to scatterplot especially for target vs categorical to show how many . This will show which category has high correlation with which demand value.

d. Lagged covariate series may have better correlation.

LOOKUP(SUM([Clstr 171] ),-1)

### 3.1.2 Time series plot

### 3.1.3 Scatter plot

Purpose:

Observation:

Inference:

### 3.1.2 Seasonality

#### 3.1.2.1 Seasonal heatmaps

Purpose:

Observation:

Inference:

#### 3.1.2.2 Week-on-Week seasonality

Purpose:

Observation:

Inference:

#### 3.1.2.3 Same-day seasonality

##### Tree-maps

Purpose:

Observation:

Inference:

#### 3.1.2.4 Trend Cycle

##### Trends

Purpose:

Observation:

Inference:

#### 3.1.2.5 Spatial and Temporal Segregation of cluster demand

The cluster-level demand prediction every hour is segregated/down-sampled into 5-minute intervals at every stations.

##### 3.1.2.5.1 Persistence model

Due to strong weekly seasonality, the demand after down-sampling is in same proportion to the demand in the previous week.

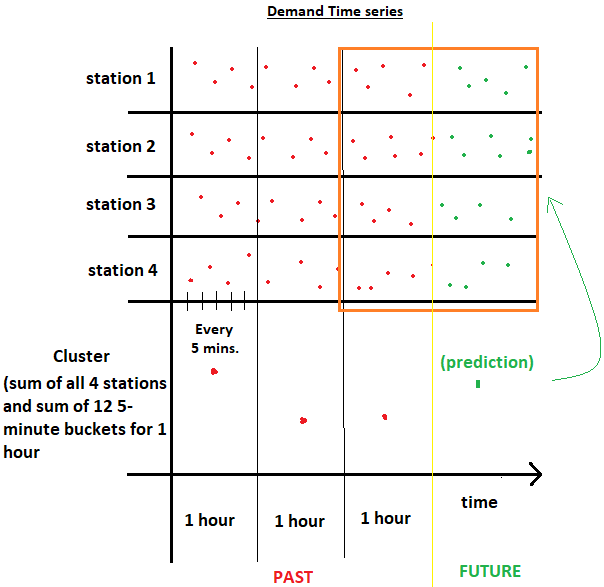


Figure 7: Illustration of segregation of hourly cluster-level demand prediction

##### 3.1.2.5.2 XGBOOST model

### 3.2.1 Data distribution

Purpose:

Observation:

Inference:

### 3.2.2 FFT

#### 3.2.2.1 Cyclical behviour

Purpose:

Observation:

Inference:

#### 3.2.2.2 IFFT

Identify peak times

Purpose:

Observation:

Inference:

### 3.2.3 Stationarity

Purpose:

Observation:

Inference:

### 3.2.4 ACF

Purpose:

Observation:

Inference:

### 3.2.5 PACF

Purpose:

Observation:

Inference:

### 3.2.6 Relationships between target and covariates

#### Co-integration between lagged supply and demand

#### 1 hour lag, 2 hour lag, 3 hour lag

Purpose:

Observation:

Inference:

#### Cross-correlation

Purpose:

Observation:

Inference:

#### Correlation Analysis

Purpose:

Observation:

Inference:

#### Frequency distribution

Purpose:

Observation:

Inference:

#### Scatter plots

:

Observation:

Inference:

#### Hypothesis Testing

Purpose:

Observation:

Inference:

# Baselines

## 4.1 Persistence Model

## 4.2 Statistical Models

### 4.2.1 Negative-binomial Autoregression

#### Model formulation

#### Test results

# DeepAR

## 5.1 Sequence to Sequence Models

## 5.2 Working Principle

## 5.3 Data Pre-processing

### 5.3.1 Feature Engineering

## 5.4 Dataset Creation

## 5.5 Architecture

### 5.5.1 Block diagram

Draw a better block diagram for the offline model without strea

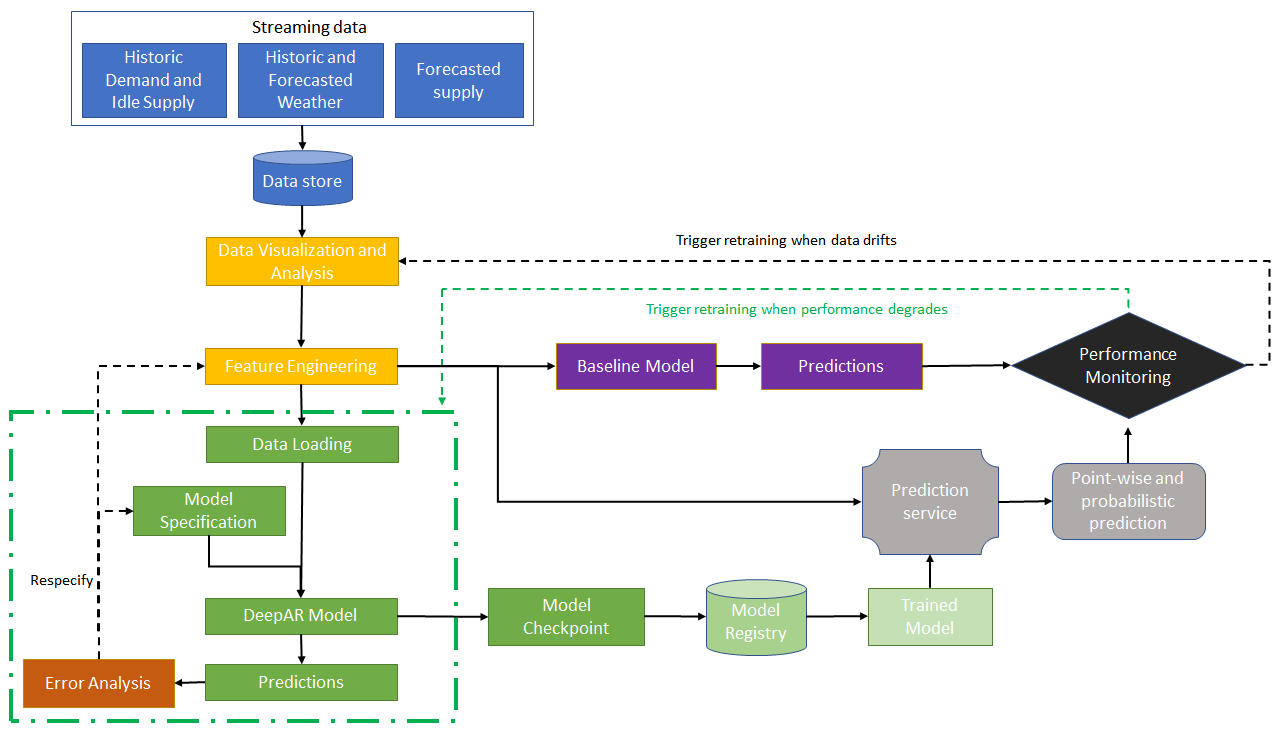


Figure 8:DeepAR block diagram

Refer to Detailed block diagram in appendix.

### 5.5.2 Mindmap

Mind map of the entire process is summarized in the appendix.

### 5.5.2 Encoder length considerations

# Training

# Validation

## 7.1 Hyperparameter Tuning

### 7.1.2 Optuna

### 7.1.3 Observation: s (Check further tuning page)

## 7.2 Model Troubleshooting

### 7.2.1 Error Analysis

#### Missing seasonality

#### Model Selection

AIC/BIC, ensures model simplicity and avoids overfitting.

Does extra parameters justify the improvement?

Reduce embedding dimension. Increase number of neurons.

## 7.3 Feature Importance

## 7.4 Model Explainability

# Testing

## 8.1 Cluster-wise Results

### 8.1.1 Tampines

#### 8.1.1.1 Probabilistic forecast

#### 8.1.1.2Residual Analysis

##### 8.1.1.2.1 DeepAR prediction:

##### 8.1.1.2.2 Historic Average prediction

### 8.1.2 Central

### 8.1.3 Woodlands

## Region-wise Results

### Tampines

# Code Troubleshooting and debugging

## 9.1 Code repository

Code repository is located at : [Github\_DeepAR](https://github.com/JosePeeterson/DeepAR_demand_prediction)

# Dashboard Visualization

# Gaps and Improvement suggestions

1. Include holidays as categorical time series

# Appendix

## 11.1 FAQ

1. What is the difference between Autocorrelation and FFT?
   1. ACF: Any instance Correlated by all lags. FFT: any instance is correlated by the same point in the period.

## Diagram Description automatically generated11.2 Detailed Block Diagram

## FFT of weekly demand time series (9 weeks)

Graphical user interface

Description automatically generated with medium confidence

Histogram

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

A picture containing text

Description automatically generated

Text

Description automatically generated with medium confidence