**DeepAR Electric Vehicle Demand Prediction**

Jose Peeterson

Research Engineer

NTU-Continental Corporate Lab

Table of Contents

[1. Introduction 4](#_Toc124423080)

[1.1 Problem statement 4](#_Toc124423081)

[1.2 Proposed solution 4](#_Toc124423082)

[1.2.1 Vehicle Rebalancing 4](#_Toc124423083)

[1.2.2 Summary of feasible Demand Prediction solutions 5](#_Toc124423084)

[2. Data Wrangling 6](#_Toc124423085)

[2.1 Spatial and Temporal Aggregation 6](#_Toc124423086)

[3. Exploratory Data Analysis 6](#_Toc124423087)

[3.1 Data Visualization 6](#_Toc124423088)

[3.1.1 Scatter plot 6](#_Toc124423089)

[3.1.2 Seasonal heatmaps 6](#_Toc124423090)

[3.1.3 Week-on-Week seasonality 6](#_Toc124423091)

[3.1.4 Same-day seasonality 6](#_Toc124423092)

[3.1.5 Trend Cycle 6](#_Toc124423093)

[3.2 Data Analysis 6](#_Toc124423094)

[3.2.1 Data distribution 6](#_Toc124423095)

[3.2.2 FFT 6](#_Toc124423096)

[3.2.3 Stationarity 7](#_Toc124423097)

[3.2.4 ACF 7](#_Toc124423098)

[3.2.5 PACF 7](#_Toc124423099)

[3.2.6 Relationships between target and covariates 7](#_Toc124423100)

[4. Baselines 7](#_Toc124423101)

[4.1 Persistence Model 7](#_Toc124423102)

[4.2 Statistical Models 7](#_Toc124423103)

[4.2.1 Negative-binomial Autoregression 7](#_Toc124423104)

[5. DeepAR 8](#_Toc124423105)

[5.1 Sequence to Sequence Models 8](#_Toc124423106)

[5.2 Working Principle 8](#_Toc124423107)

[5.3 Data Pre-processing 8](#_Toc124423108)

[5.3.1 Feature Engineering 8](#_Toc124423109)

[5.4 Dataset Creation 8](#_Toc124423110)

[5.5 Architecture 8](#_Toc124423111)

[5.5.1 Encoder length considerations 8](#_Toc124423112)

[6. Training 8](#_Toc124423113)

[7. Validation 8](#_Toc124423114)

[7.1 Hyperparameter Tuning 8](#_Toc124423115)

[7.1.2 Optuna 8](#_Toc124423116)

[7.1.3 Observations (Check further tuning page) 8](#_Toc124423117)

[7.2 Model Troubleshooting 8](#_Toc124423118)

[7.2.1 Error Analysis 8](#_Toc124423119)

[7.3 Feature Importance 8](#_Toc124423120)

[7.4 Model Explainability 8](#_Toc124423121)

[8. Testing 9](#_Toc124423122)

[8.1 Cluster-wise Results 9](#_Toc124423123)

[8.1.1 Tampines 9](#_Toc124423124)

[8.1.2 Central 9](#_Toc124423125)

[8.1.3 Woodlands 9](#_Toc124423126)

[Region-wise Results 9](#_Toc124423127)

[Tampines 9](#_Toc124423128)

[9. Dashboard Visualization 9](#_Toc124423129)

# 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.

Map

Description automatically generated

Map

Description automatically generated

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 Analysis

## 3.1 Data Visualization

### 3.1.1 Scatter plot

### 3.1.2 Seasonality

#### 3.1.2.1 Seasonal heatmaps

#### 3.1.2.2 Week-on-Week seasonality

#### 3.1.2.3 Same-day seasonality

##### Tree-maps

#### 3.1.2.4 Trend Cycle

##### Trends

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

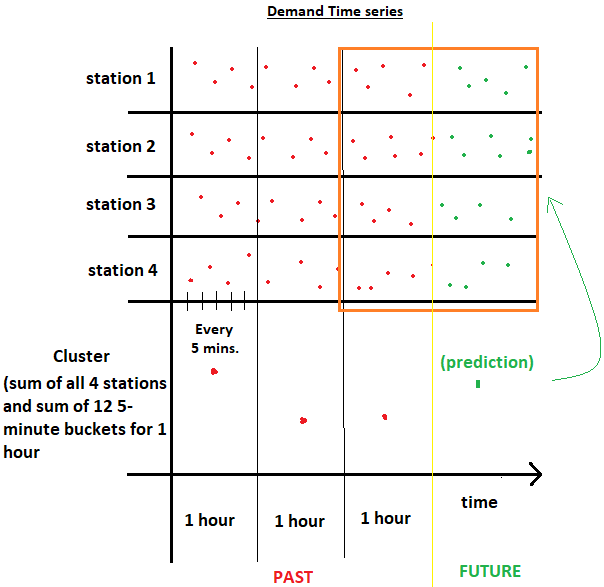


Figure 2: Segregation of hourly cluster-level demand prediction

## 3.2 Data Analysis

### 3.2.1 Data distribution

### 3.2.2 FFT

#### Cyclical behviour

#### IFFT

Identify peak times

### 3.2.3 Stationarity

### 3.2.4 ACF

### 3.2.5 PACF

### 3.2.6 Relationships between target and covariates

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

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

#### Cross-correlation

#### Correlation Analysis

#### Frequency distribution

#### Scatter plots

#### Hypothesis Testing

# 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 Encoder length considerations

# Training

# Validation

## 7.1 Hyperparameter Tuning

### 7.1.2 Optuna

### 7.1.3 Observations (Check further tuning page)

## 7.2 Model Troubleshooting

### 7.2.1 Error Analysis

## 7.3 Feature Importance

## 7.4 Model Explainability

# Testing

## 8.1 Cluster-wise Results

### 8.1.1 Tampines

#### Probabilistic forecast

#### Residual Analysis

### 8.1.2 Central

### 8.1.3 Woodlands

## Region-wise Results

### Tampines

# Code Troubleshooting and debugging

# Dashboard Visualization

# 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.