NUS Business School Honors Dissertation

Peng Seng Ang

AY 2019/2020 Semester 2

Abstract

Many real industry problems involves spatio-temporal demand prediction (e.g predicting demand at a certain time across different locations). Most of the demand data are not continuous variables but instead count variables, like number of orders and number of transactions. This paper studies how we can apply time series model to count variables, using VAR models to exploit spatial correlations as well as introduce a 3-step approach to improve forecast accuracy.

1 Introduction

The motivation for this paper comes from Sheng Liu (2018), where their focus is to optimise assignments of orders to drivers in order to minimize the total delay of all drivers. The dataset used in both Sheng Liu (2018) and this paper are from a food service provider in China that allows customer to place orders before a cutoff time in the day (e.g 10.00am) and the customer can expect to recieve their orders by a deadline (e.g anytime from 10.30am to 11.45am).

The problem is not only restricted to the abovementioned food provider. With the rise of e-commerce and the food ordering and delivery services, like GrabFood and FoodPanda,

demand prediction and driver assignment problem would be an everyday concern for them too.

In reality, demand is never deterministic and hence, having an accurate forecast of demand for the food service providers would help them more effectively and efficiently assign orders to drivers to improve the overall delivery time. As such, the focus of this paper would be to accurately model and forecast the demand at the different locations and time. Currently, most Autoregressive (AR) or Autoregressive Integrated Moving Average (ARIMA) models only consider temporal features when predicting demand. However, we believe including spatial features between the data points might improve forecast accuracy. This paper would focus on and explore models that include both spatial and temporal features to improve forecast accuracy.

2 Literature Review

Marina Knight and Nason (2016) describes and shows how they implemented a network autoregressive moving average model to model the number of cases of Mumps in UK counties. In their example, they also showed that they might achieve a better result by modelling the series separately as univariate time series, also suggested in Matthew A. Nunes (2015) since the neighbouring counties does not provide a substantial amount of explanatory power. Other related literature includes de Luna and Genton (2005) where they propose a model building strategy for spatially sparse but temporally rich data.

BigVAR and tscount are some libraries that would be used in this paper. Tobias Liboschik (2017) provides the mathematical background and implementation of Generalised Linear Models (GLM) for count time series as a library (tscount) in R while William Nicholson (2017) extensively descibes the background and implementation of the VAR models and BigVAR library for multi-variate time series.

There are also various research on different methods used to predict spatio-temporal demand.

One example would be Abolfazl Safikhani (2017), where they proposed using generalized spatio-temporal autoregressive model for predicting taxi demands across locations in New York City.

3 Data

The data source used was an operational dataset from a food delivery service provider from Shanghai that includes delivery information for a 2-month period from 10 August 2015 to 30 September 2015 (excluding Saturdays) in 2015. The provider only provides delivery service for 90 minutes during lunchtime and the dataset has split the data into 15-minute time periods, and as such, each day would only consists of demand data for 6 time periods. Hence, our dataset has 839 locations with demand count data, in integer, for 204 time periods in total.

To include other exogenous variables, data from https://www.worldweatheronline.com/shanghai-weather-history/shanghai/cn.aspx was used to include weather and rainfall data as well as encoding of the weekadys for all the respective days.

3.1 Exploratory Analysis

We would first do some exploratory analysis and check if there are any interesting relationships between the variables.

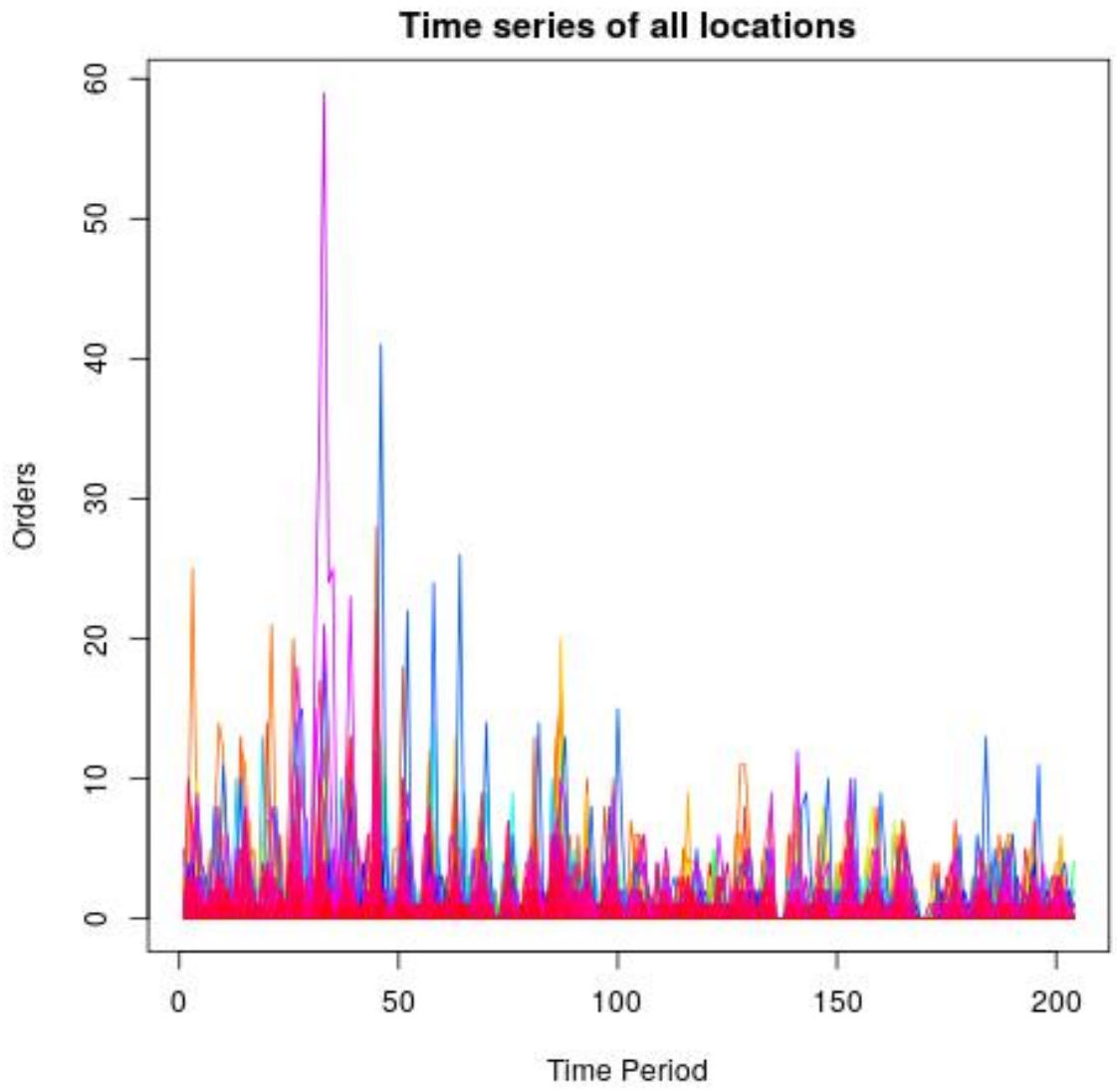

Figure 1: Time series of all locations in the dataset. It can be observed that most locations have very low number of orders across the time period.

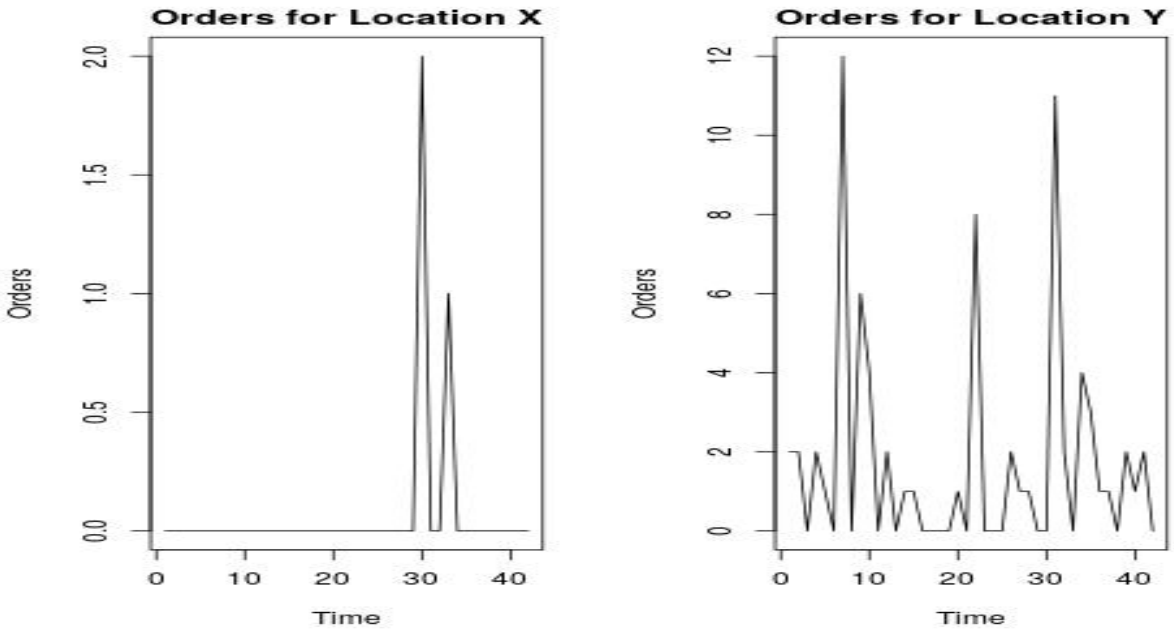

Figure 2: Most locations have very sparse time series (left) while some have relatively more dense time series (right)

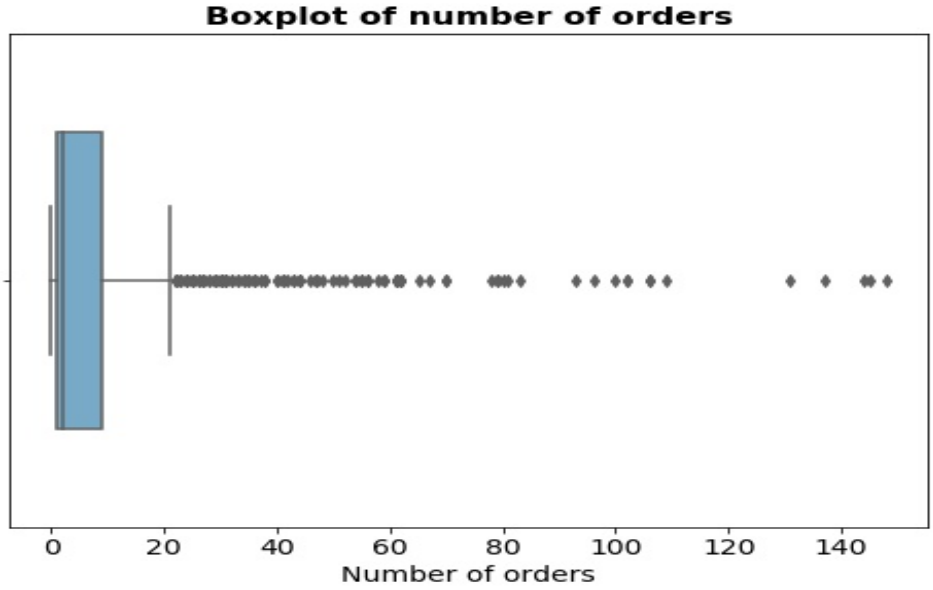

Figure 3: Boxplot of counts

We can see from the boxplot in Figure 3 that most of the locations have extremely low number of non-zero orders and further analysis showed that about 335 locations have just a maximum of one non-zero order throughout the 204 time periods.

Analysis on some exogenous factors were performed too.

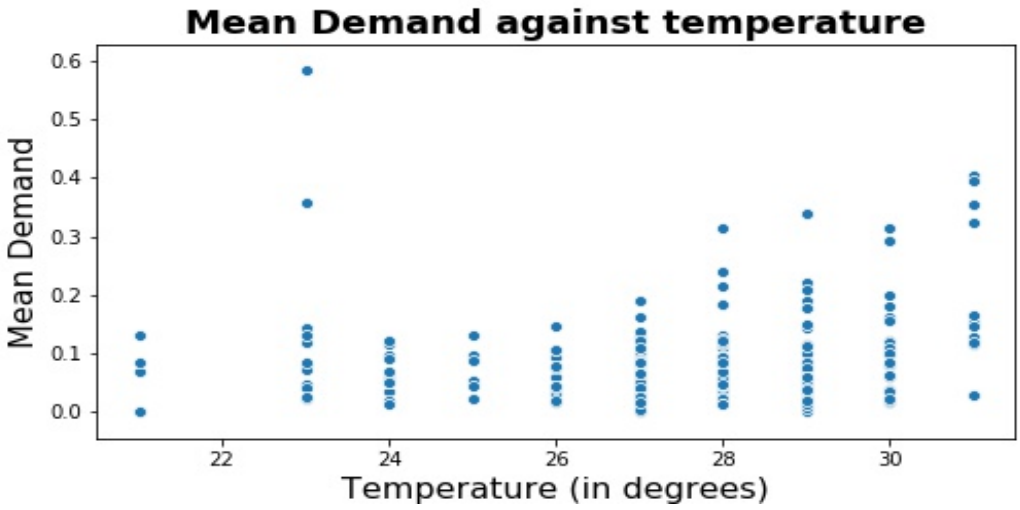

Figure 4: Scatter plot of mean counts against temperature

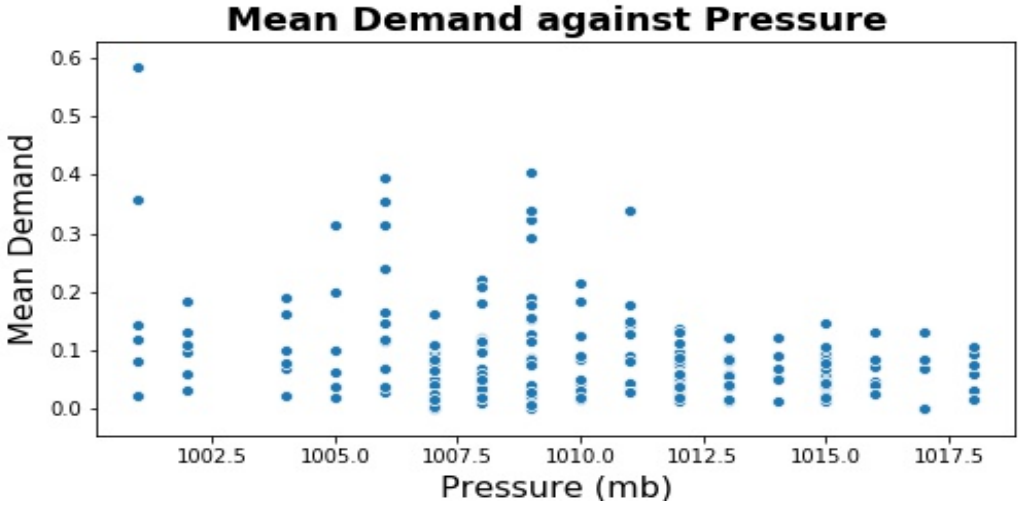

Figure 5: Scatter plot of mean counts against pressure

The scatter plot in Figure 4 visually display a slight positive relationship between temperature and mean demand across all locations whereas Figure 5 visually display a slight negative relationship between pressure and mean demand across all locations.

3.2 Train-Test Split

From Figure 1 in Section 3.1, the data is very sparse as there are many locations that have no demand counts for the majority of the time period. Hence, to get a better idea of how our models would work, only locations with at least 50 non-zero counts across the time period would be used initially, leaving us with 42 locations that meet this criteria. After the model is tested on the smaller dataset, we would then use the full dataset for location with at least one non-zero count to test and fit our model.

The dataset was then split into training and test set by considering the first 33 days as the training set and the next 1 day as the test set. Our training set would then have 198 demand data for each location and test set would have 6 demand data for each location.

4 Baseline Model

In this section, we would build a simple baseline model. Following which, we would try other different spatial temporal time series models and compare the results to the baseline model.

4.1 Metric Used

The main metric that would be used for comparison would be Mean Squared Forecast Error (MSFE), which is calculated by:

$$MSFE = \frac{1}{n}\sum_{t=1}^{n} \|\hat{y}_t - y_t\|_2^2$$

where n is the number of data points, $\hat{y_t}$ is the predicted demand at time t and y_t is the actual demand at time t.

4.2 ARIMA models

Autoregressive Integrated Moving Average (ARIMA) models are one of the most commonly used models for time series (Z. Asha Farhath (2016)). ARIMA models are made up of 3 processes, mainly the Autoregressive (AR) process, the Integrated (I) process and the Moving Average (MA) process (Jamal Fattah (2018)). The AR process assumes that each observation can be expressed as a linear combination of its past values. An AR(x) process would mean using x lagged values. The MA process assumes that each observation can be expressed as a linear combination of its current error term as well as its past error terms. The Integrated Process states that the time series can undergo differencing to ensure that the series is stationary. A MA(x) process would mean using x number of past observations. Hence, an ARIMA model is usually represented by ARIMA(p,d,q), where p represents the number of autoregressive terms, d represents the number of differences needed for stationarity, and q represents the number of lagged forecast errors.

4.3 Baseline ARIMA Result

As a baseline model, each of the locations was assessed individually and a suitable ARIMA model was built for each location. Auto-arima function from Python was used to implement this. The out-of-sample MSFE for this baseline model on the 42 locations is **73.29**. A sample forecast plot is shown below:

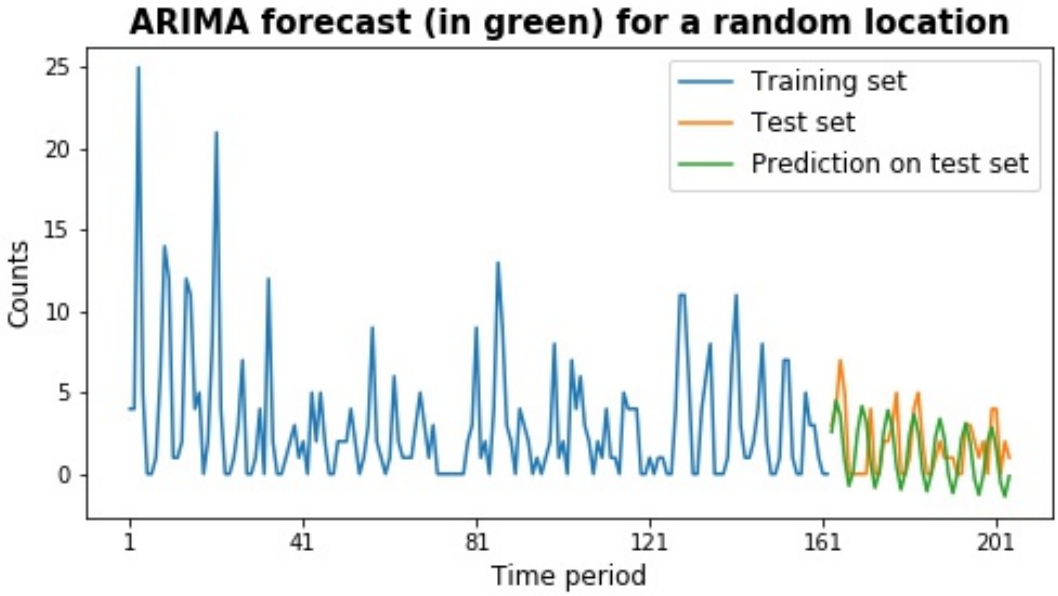

Figure 6: ARIMA forecast on a random location

5 GLM Model

The dataset that we are using follows a count time series, which means the observations are non-negative integers. A flexible and commonly used model for count time series is the Generalized Linear Model (GLM) Nelder JA (1972). GLM normally take the form of:

$$g(\lambda_t) = \eta^T X_t$$

Using the R package from Tobias Liboschik (2017), the GLM used would be an extension of the above equation and can be expressed in the form of:

$$g(\lambda_t) = \beta_0 + \sum_{k=1}^{p} \beta_k \tilde{g}(Y_{t-i_k}) + \sum_{l=1}^{q} \alpha_l g(\lambda_{t-j_l}) + \eta^T X_t$$

where g represents a link function and $\tilde{g}$ represents a transformation function. η represents a parameter vector that corresponds to the covariates.

5.1 Model Implementation

Since there are many locations which have values that are all 0 throughout all the time period, the GLM model would run into an error if applied on those. Hence, only locations with at least 1 non-zero value would be considered. Similar to before, each of the locations was assessed individually and a suitable GLM model was fitted for each location. The out-of-sample MSFE for this baseline model is **82.42**.

5.2 Model Diagnostics

To validate and verify if our fitted model is adequate, model checking would be performed by performing the following residual analysis.

5.2.1 Residuals plots

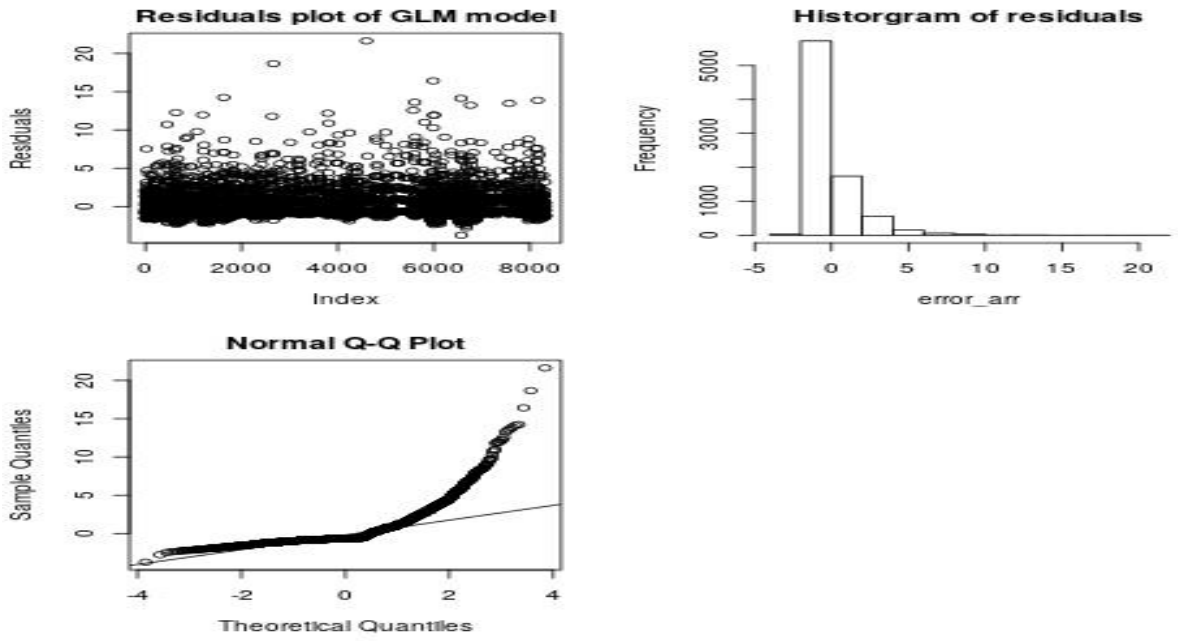

Figure 7: Residuals for GLM Model

Figure 7 diagnostic plots shows that while the residuals roughly randomly scattered, the GLM model produces residuals that does not follow the normal distribution well. This is expected as we assume that the distribution is poisson and not normal.

5.2.2 Residuals against Predicted values

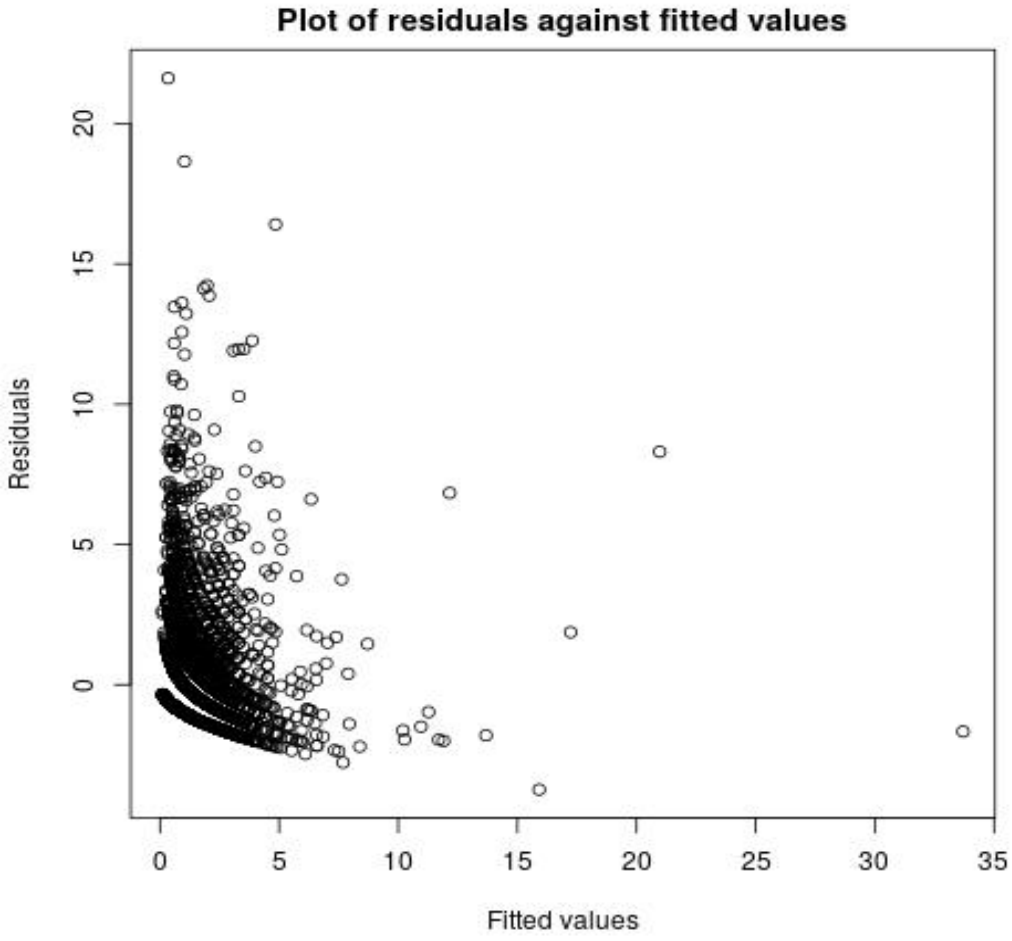

Figure 8: Residuals against Predicted values for GLM Model

The above plot of residuals against the predicted values suggests heteroscedasticity between residuals, or non-constant variance between the residuals as the predicted value increases, which is expected in a Poisson GLM, as mentioned in Molenaar and Bolsinova (2017). For

a normal regression model, this is a bad sign but since we are assuming that our count data follows a Poisson distribution, the residuals are bound to display heteroscadasticity.

5.3 Model Limitation

The MSFE of the GLM model is better than that of the baseline model. However, it would be relatively expensive and time-consuming as every location has to be individually fitted to a GLM model. Also, it only uses its past values and does not take into account data from the other locations. The next section explores another type of model which would use data from other locations.

6 VAR Model

6.1 Motivation

As with many spatio-temporal demand prediction, spatial features could be an important feature if there exist spatial correlation. To check if there exist spatial correlation between the locations, a correlation heatmap would be plotted. For clear illustration purposes, instead of showing the matrix of correlation for all 839 locations, only the 42 locations with at least 50 non-zero counts would be shown.

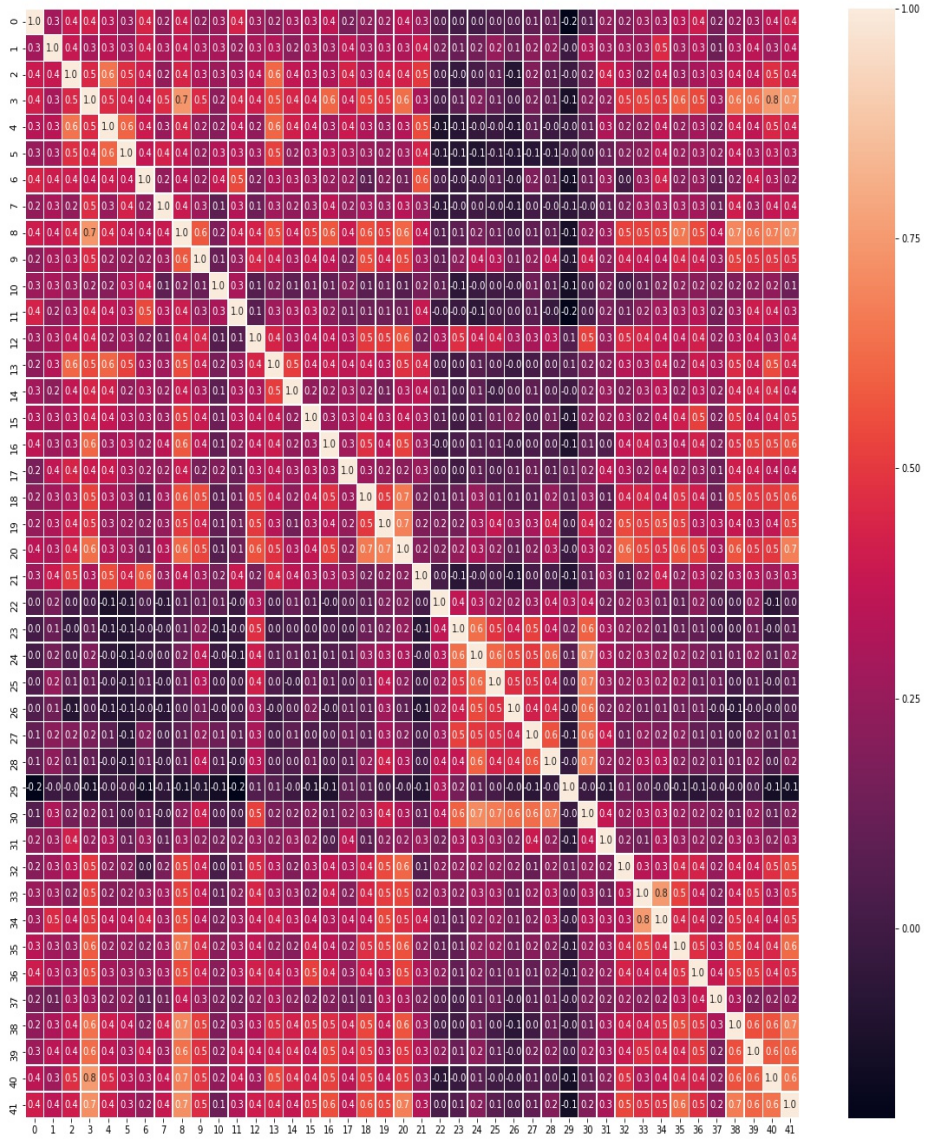

Figure 9: Spatial Correlation Heatmap on the locations with at least 50 non-zero counts. Axes represent the 42 locations

This provides the motivation to use the existence of spatial correlation and explore the usage of VAR model.

6.2 Details of VAR Model

Vector Autoregressive (VAR) models are the most commonly used model for multivariate time series, particularly in economics and financial time series as shown in Bjrnland (2000). VAR models are very similar to multivariate linear regression models and methods used to perform inferencing on linear regression models can also be applied to VAR models. VAR(p) represents a VAR model of order p if the time series can be written as:

$$y_t = v + \sum_{i=1}^{p} \phi_i y_{t-i} + \alpha_t$$

where p is the number of lagged endogenous variables used, y_t is the value at time t, v is a constant vector, ϕ_i are coefficient matrices for $i > 0$ and α_t are independent and identically distributed random vectors.

6.3 Stationarity Condition

For a univariate time series, it is important for the time series to be transformed into a stationary series and Augmented Dickey-Fuller (ADF) test can be used to perform unit root test for stationarity, as shown in Zhijie X. (1998) and Mushtaq (2011). For a multi-variate time series, if the series are unit-root non-stationary, applying the VAR model could lead to spurious regression, as shown in Baumhl (2009).

6.3.1 Cointegration

Box (1977) shows that it is possible to linearly combine various unit-root nonstationary time series to form a stationary series. The term Cointegration, first mentioned in Granger (1983),

states that although some or all the time series might be unit-root nonstationary individually, these time series can be said to be cointegrated if there exists a possible linear combination of them that would form a stationary series. Intuitively, 2 series are cointegrated if they move together and the distance between them remain stable over time.

6.3.2 Johansen Test for Cointegration

While Cointegrated Augmented Dickey Fuller Test, commonly used for Pairs Trading, can be used, it is only able to be applied on 2 separate series. In our dataset, we have 839 locations at least, hence we would apply the popular approach to cointegrating tests for VAR model, called the Johansen's Cointegration Test. However, one limitation is that it can only be used to check for cointegration between a maximum of 12 variables. For further elaboration on the Johansen's Cointegration Test, please refer to Johansen (1991). Since our dataset has 839 variables (locations), we are unable to accurately calculate the significant values of more than 12 variables and hence unable to determine correctly the number of cointegration vectors needed.

6.3.3 Differencing

Instead, differencing would be performed on every series to make the series levels stationary. After differencing, ADF test was performed on each series and the following result shows that all the series are stationary.

The time series of each location have been checked and are stationary. While it is possible to perform differencing on every series, it might cause over-differencing, as mentioned in Tsay (2014).

6.4 VARX Model

VAR models can also be extended to include exogenous variables. A VARX(p,s) (with exogenous variables) model can be expressed as:

$$y_t = v + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=1}^{s} \beta_j x_{t-j} + \alpha_t$$

where p is the number of lagged endogenous variables used, s is the number of lagged exogenous variables used, y_t is the value at time t, v is a constant vector, ϕ_i are coefficient matrices for endogenous coefficient matrix for $i > 0$, β_i are coefficient matrices for exogenous coefficient matrix for $i > 0$ and α_t are independent and identically distributed random vectors.

Our dataset uses additional exogenous variables like temperature, wind, gust, cloud, humidity, precipitation, pressure as well as one-hot encoding of the day of the week. Our dataset now would have 839 endogenous variables/locations and 13 exogenous variables.

6.5 Model Checking

To validate and verify if our fitted model is adequate, model checking would be performed by performing the following residual analysis:

6.5.1 Whiteness of Residuals

To ensure our fitted model is adequate, the resiudals should behave like a white noise series. The plots below shows the distribution of our residuals for the VAR model and the VARX model.

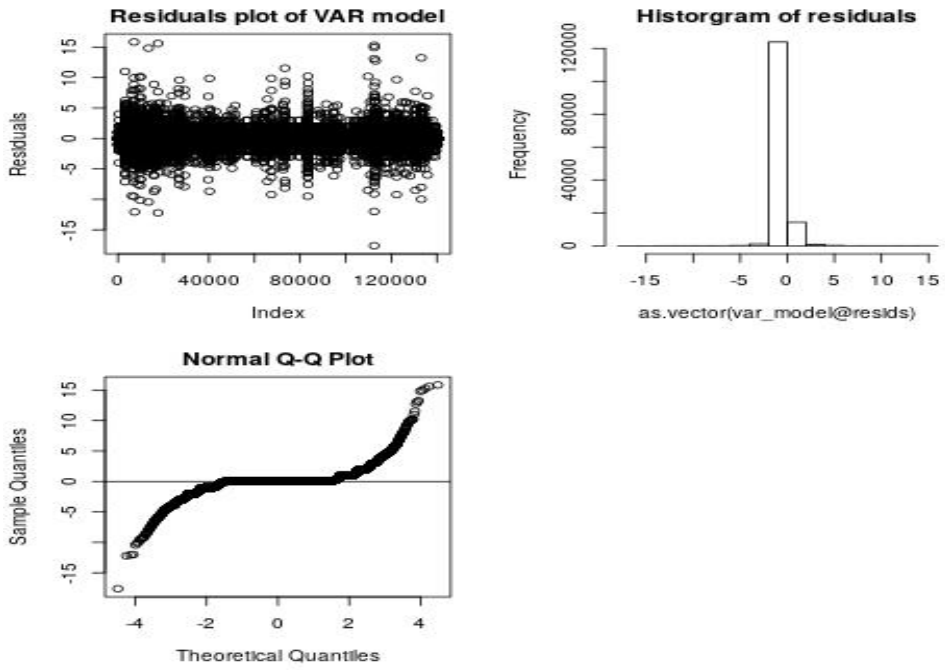

Figure 10: Residuals for VAR Model

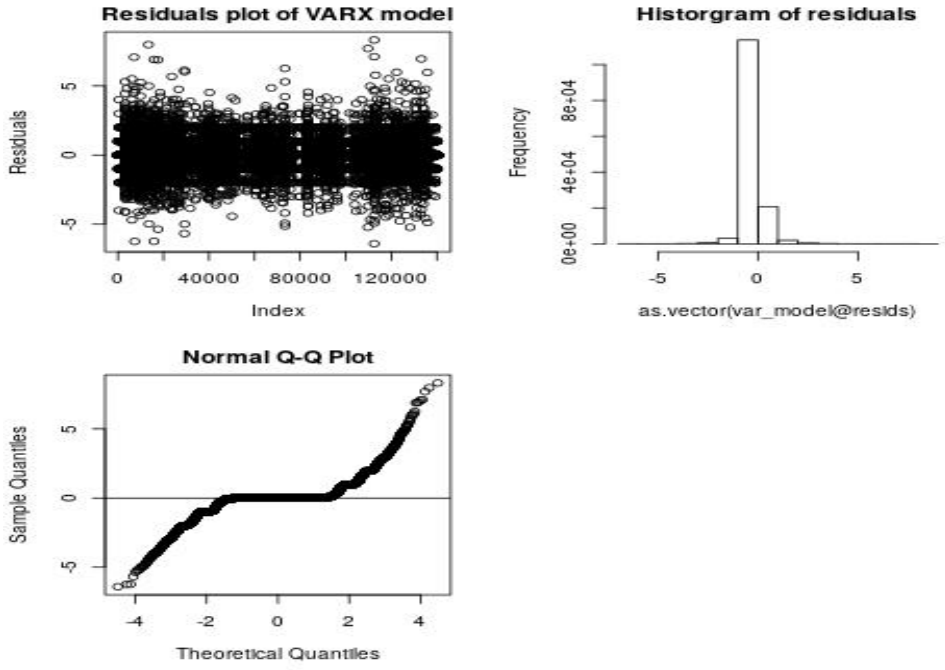

Figure 11: Residuals for VARX Model

From Figure 11 and 12, we can see that the residuals are mostly randomly scattered and they roughly follow a normal distribution, although it performs rather badly on the lower end and higher end of the outliers.

6.5.2 Residuals against Fitted Values

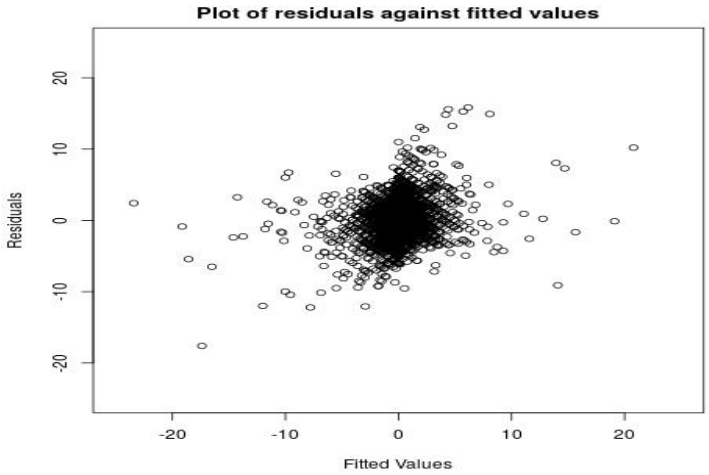

Figure 12: Residuals against Fitted Values for VAR Model

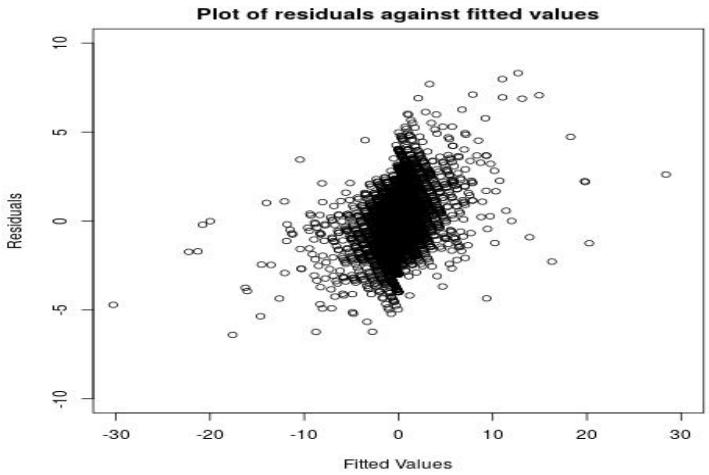

Figure 13: Residuals against Fitted Values for VARX Model

Figure 13 and 14 shows that heteroscadasticity occurs at the lower and higher ends of the predicted values, whereas in the middle range, the residuals tend to be roughly randomly scattered.

6.6 Results

BigVAR library in R was used to implement the VAR models. The MSFE for the VAR model is 76.72 and MSFE for VARX model is 75.63. This would imply that the exogenous variables used do have a moderate amount of explanatory power.

7 3-step Approach

We could then explore an alternative 3-step approach. We would first cluster the locations, then use VAR/VARX to predict the total demand of each cluster, and then assign the total demand to each individual locations. Similar methods can be found in Paul W.Murraya (2015) and Chi-Jie Lu (2014)

7.1 Clustering of locations

A simple K-means clustering was first done on the locations using their time series data in the training set.

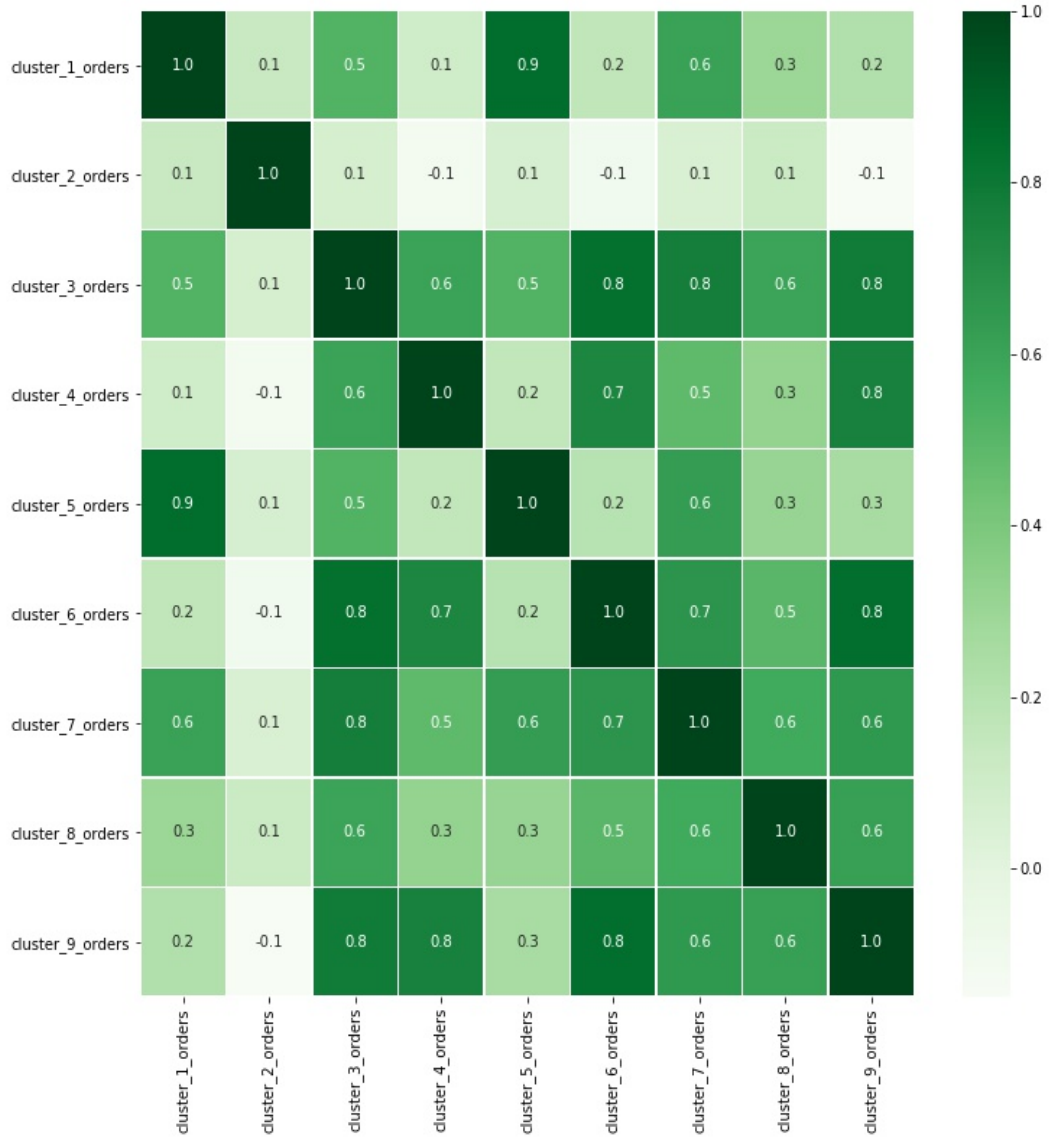

Figure 14: Correlation heatmap for 9 clusters

We can see that there exists storng correlation between certain cluster (like cluster 1,5 and cluster 6,9).

3 different kind of distance metrics are explored to perform K-means clustering.

7.1.1 Using longitude/latitude as the distance metric

Firstly, we would cluster the locations using their geographical longitude and latitude. K-means clustering is performed using the euclidean distance of the locations as the distance metric.

7.1.2 Using correlation as the distance metric

We then also explored using correlation between the time series as the distance metric for K-means clustering.

7.1.3 Using Dynamic Time Warping as the distance metric

Finally, we used Dynamic Time Warping (DTW) Distance as the distance metric for K-means clustering.

DTW is one of the commonly-used algorithm for measuring similarity between time series. More details on DTW can be found in Senin (2017)

7.2 Predicting total demand for each cluster

For each type of clustering, a VAR model was then trained on the clusters and the model was used to predict the total demand for each of the clusters.

7.3 Assigning total cluster demand to individual locations

The predicted demand for each cluster would then be reallocated to each individual location in the cluster by:

$$y_i = D_{C(i)} * \frac{(\sum_{j=1}^{n} x_{ij})}{\sum_{l \epsilon C(i)} \sum_{j=1}^{n} x_{lj}}$$

where y_i represents the predicted demand for location i, C(i) represents the cluster which location i belongs to, $D_C(i)$ represents the total demand (from training set) of the cluster C(i), $x_i j$ represents the training set demand at location i at time j

7.4 Results

We also test the effect of different number of cluster groups on the MSFE on our full dataset.

	3 Clusters	**6 Clusters**	**9 Clusters**	**12 Clusters**
Clustering using euclidean distance of latitude/longitude	61.18	**60.67**	61.03	62.24
Clustering using correlation between time series as distance metric	62.22	**59.41**	61.49	61.35
Clustering using DTW as distance metric	64.00	64.37	62.68	**62.20**

8 Results and Conclusion

The table below summarises the results of our models.

	MSFE (Locations with at least 50 non-zero counts)	**MSFE (All locations)**
ARIMA	47.60	72.48
GLM	38.16	82.42
VAR	42.76	76.72
VARX	41.73	75.63
3-step approach (Using K-means on geographical location)	40.43	60.67
3-step approach (Using K-means on correlation of time series)	39.49	**59.41**
3-step approach (Using K-means on DTW of time series)	40.10	62.20

We can see that the VAR and VARX model performs better than the GLM models on the full dataset, suggesting that the spatial relationship between locations are useful in forecasting. Also, VARX performs better than VAR model in both cases, suggesting the exogenous variables have some explanatory power and do improve the forecast accuracy.

Applying the 3-step forecast also gives a much improved result. This might be due to using a more reliable and reasonable model to predict total demands for just 9 clusters and then re-assigning it to individual locations, rather than having a model to predict all 839 locations, which would intuitively have higher error rate. The method of using VAR on all locations produced a relatively worse result, which is similar to the findings from Abolfazl Safikhani (2017), which states that a simple VAR model would not perform as well for high-dimensional data,

9 Limitations and Future Work

Our 3-step approach model display decent MSFE result. Future work could include using neural network, such as Long Short-Term Memory (LSTM) models or Convolutional LSTM models to use spatial-temporal features to forecast the demand across locations.

10 Appendix

10.1 Distribution Plots of Exogenous Variables

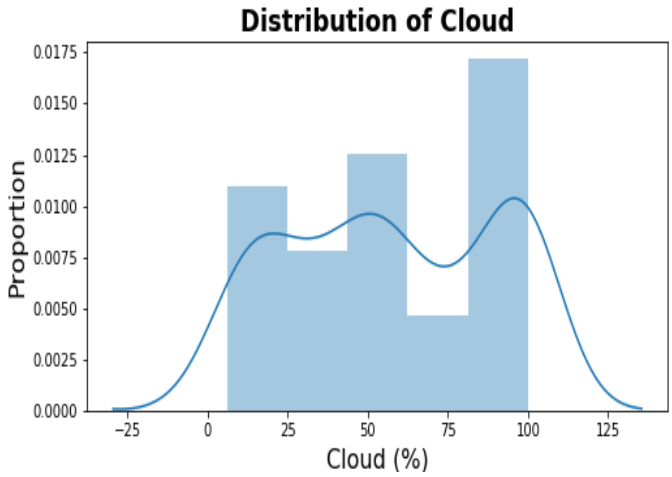

Figure 15: Distribution of Cloud

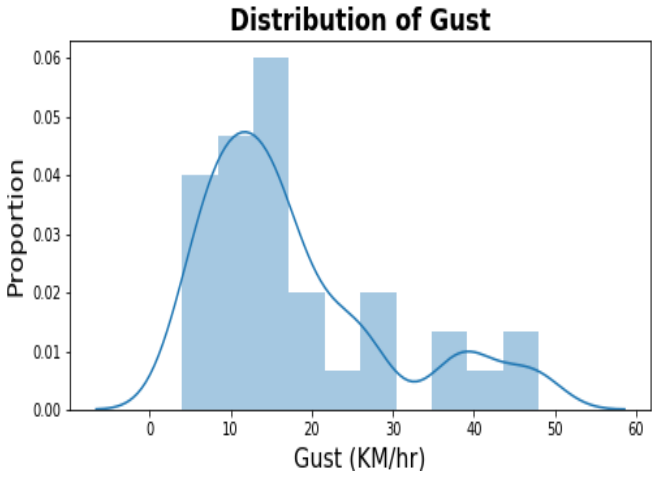

Figure 16: Distribution of Gust

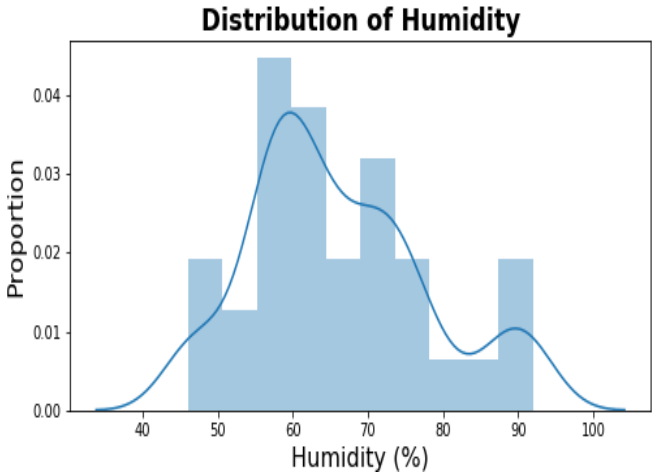

Figure 17: Distribution of Humidity

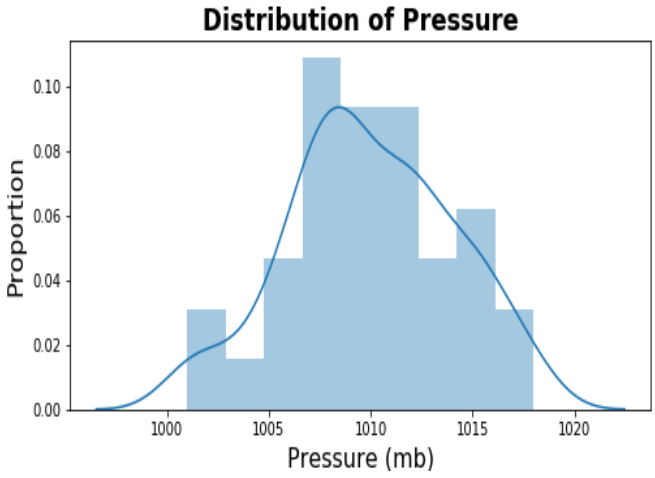

Figure 18: Distribution of Pressure

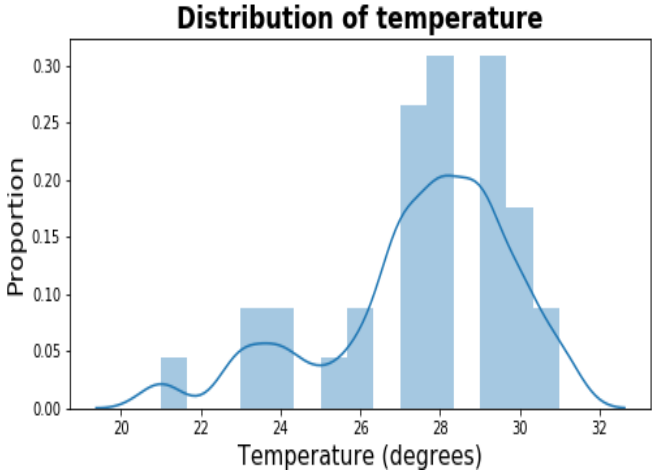

Figure 19: Distribution of Temperature

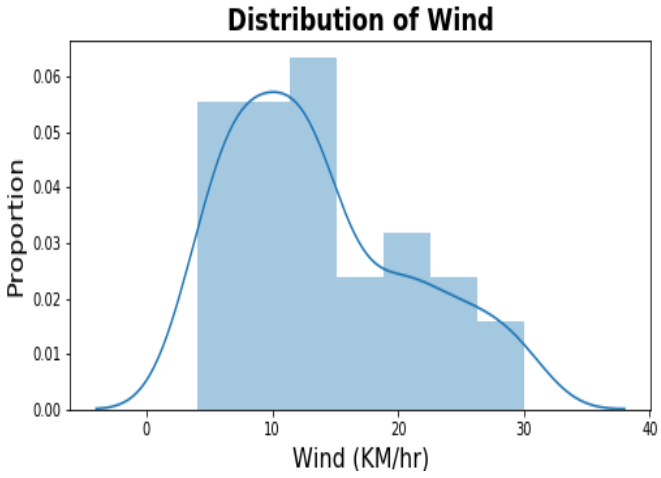

Figure 20: Distribution of Wind

10.2 Mean Demand Against Exogenous Variables

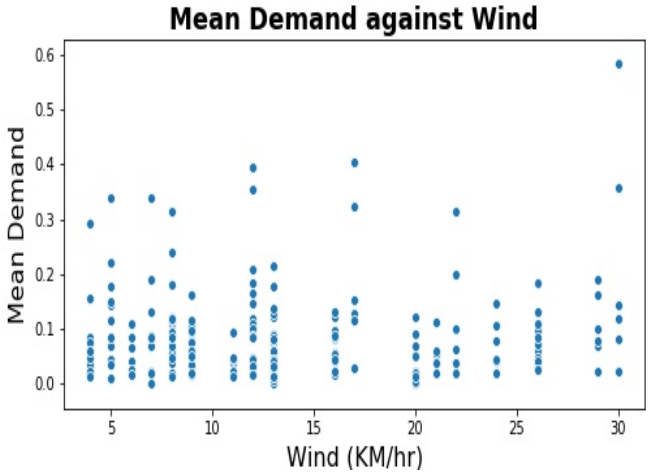

Figure 21: Mean Demand against Wind

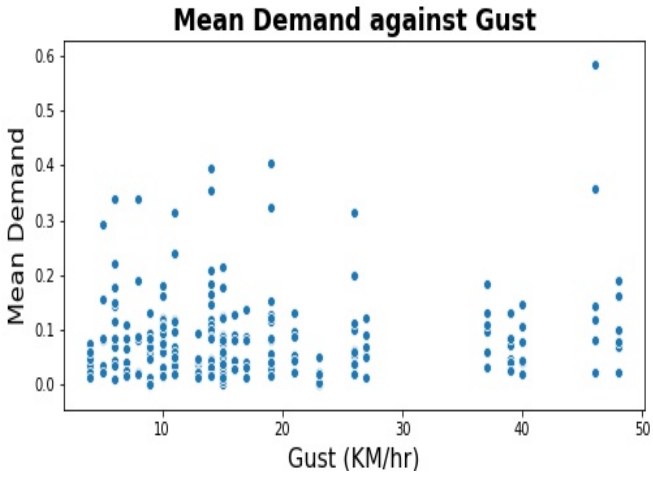

Figure 22: Mean Demand against Gust

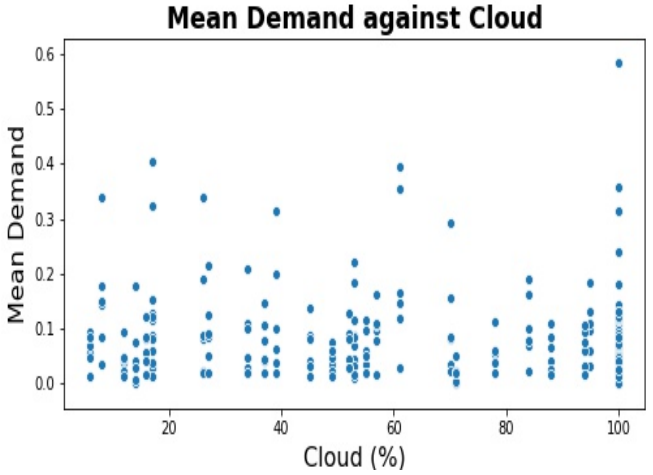

Figure 23: Mean Demand against Cloud

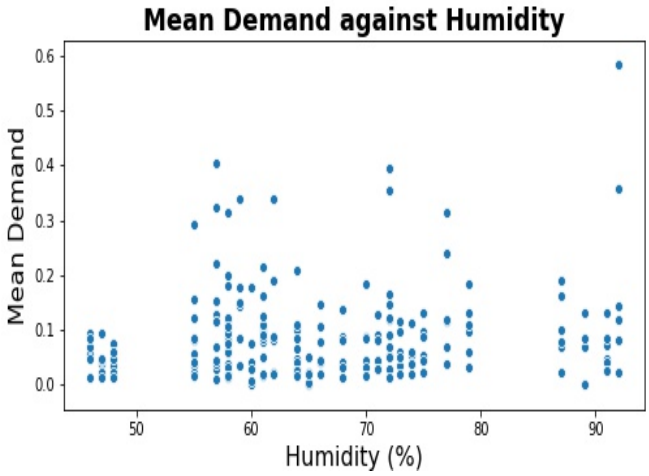

Figure 24: Mean Demand against Humidity

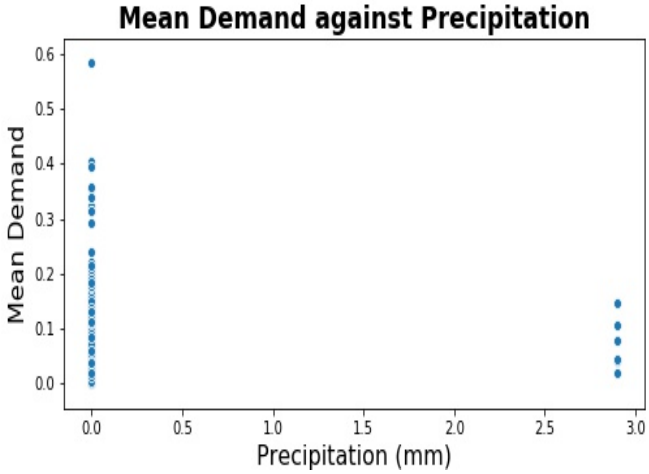

Figure 25: Mean Demand against Precipitation

References

Abolfazl Safikhani, Camille Kamga, S. M. S. S. F. B. M. (2017). Spatio-temporal modeling of yellow taxi demands in new york city using generalized star models.

Baumhl, Eduard, L. . (2009). Stationarity of time series and the problem of spurious regression.

Bjrnland, H. C. (2000). Var models in macroeconomic research.

Box, G. E. P., T. G. C. (1977). A canonical analysis of multiple time series. biometrika64(2):355365.

Chi-Jie Lu, C.-C. C. (2014). A hybrid sales forecasting scheme by combining independent component analysis with k-means clustering and support vector regression.

de Luna, X. and Genton, M. G. (2005). Predictive spatio-temporal models for spatially sparse enviromental data.

Granger, C. (1983). Co-integrated variables and error-correcting models, ucsd discussion paper 83-13.

Jamal Fattah, Latifa Ezzine, Z. A. H. E. M. A. L. (2018). Forecasting of demand using arima model.

Johansen, S. (1991). Estimation and hypothesis testing of cointegration vectors in gaussian vector autoregressive models.

Marina Knight, M. N. and Nason, G. (2016). Modelling, detrending and decorrelation of network time series.

Matthew A. Nunes, Marina I. Knight, G. P. N. (2015). Modelling and prediction of time series arising on a graph.

Molenaar, D. and Bolsinova, M. (2017). A heteroscedastic generalized linear model with a nonnormal speed factor for responses and response times.

Mushtaq, R. (2011). Augmented dickey fuller test.

Nelder JA, W. R. (1972). generalized linear models. journal of the royal statistical society a, 135(3), 370384.

Paul W.Murraya, Bruno Agardb, M. A. (2015). Forecasting supply chain demand by clustering customers.

Senin, P. (2017). Dynamic time warping algorithm review.

Sheng Liu, Long He, Z.-J. M. S. (2018). On-time last mile delivery: Order assignment with travel time predictors.

Tobias Liboschik, Konstantinos Fokianos, R. F. (2017). tscount: An r package for analysis of count time series following generalized linear models.

Tsay, R. S. (2014). Multivariate time series analysis with r and financial applications.

William Nicholson, David Matteson, J. B. (2017). Bigvar: Tools for modeling sparse high-dimensional multivariate time series.

Z. Asha Farhath, B. Arputhamary, L. A. (2016). A survey on arima forecasting using time series model.

Zhijie X., P. C. P. (1998). An adf coefficient test for a unit root in arma models of unknown order with empirical applications to the us economy.