Alice Tang
Dr. Fulton
ADEC 7460
4 December 2022

Midterm Paper: Recruit Restaurant Visitor Forecasting Competition

I. Introduction

We were tasked with participating in Kaggle's Recruit Restaurant Visitor Forecasting competition for the midterm. The context of the competition is set in Japan, where participants were given restaurant and visitor data in order to predict the future number of daily visitors. Knowing this information provides an incredibly valuable tool for restaurant businesses. In particular, accurate visitor predictions could aim to help with one of the most uncertain aspects of the business— how to prepare.

Being ready for the amount of customers that come in each day include purchasing the correct amount of stock, where no more and no less is used, as well as scheduling enough workers for the day. It is difficult to prepare these things without knowing the number of visitors. Better preparation, in turn, increases the overall efficiency for restaurants. Once the proper number of resources are determined, more effort can be focused on brainstorming ways of giving consumers a better experience— one without any sort of delay in service, and one with a better dining aspect. Accurately predicting the number of visitors leads to a win-win situation for both restaurants and consumers, making it an important problem to analyze.

II. Data

Ultimately, being presented wth many datasets at once was overwhelming. One of the biggest challenges for me was choosing the best datasets that would give me useful information in predicting visitors accurately. In the end, I only chose to examine the AirRegi datasets with air_visit_data, air_store_info, and date_info. I did not choose to look at any Hot Pepper Gourmet (HPG) datasets, because I did not think it would offer a substantial amount of improvement to our model. This is because if we only look at HPG reservations, the amount of reservations do not always equate to visitors. Customers can easily cancel reservations or simply not show up. For these reasons, I was more curious in examining datasets with the official number of visitors, the visit date, and the store information. I also decided to use the dataset concerning holidays, because the dates that we have to predict (April 23rd - May 31st), falls within the timeframe of Japan's Golden Week. Because of this, it was crucial to also examine the relationship between holidays and number of visitors.

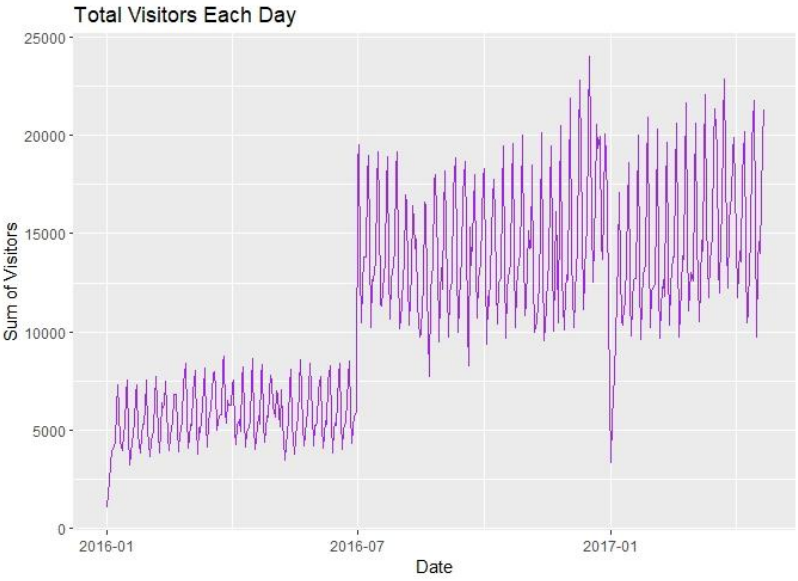
A: air_visits Dataset

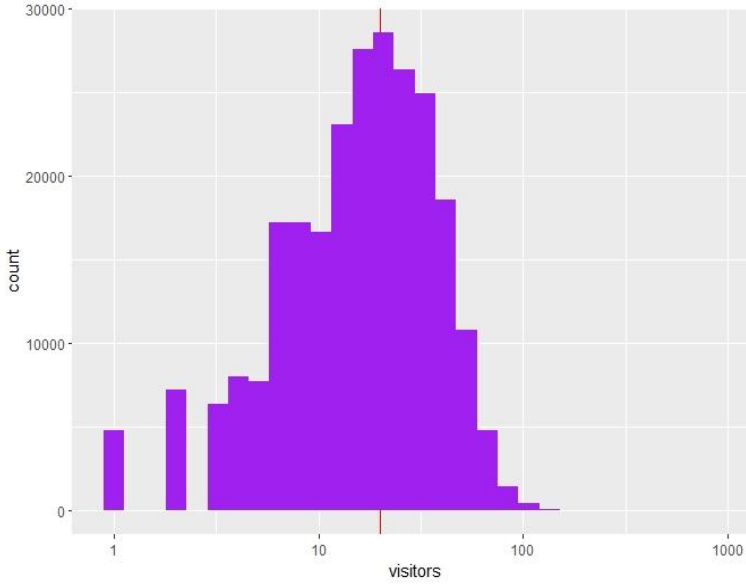
```
> summary(air_visits)
 air_store_id        visit_date          visitors
 Length:252108      Length:252108      Min.    :  1.00
 Class :character   Class :character   1st Qu.:  9.00
 Mode  :character   Mode  :character   Median : 17.00
                                       Mean    : 20.97
                                       3rd Qu.: 29.00
                                       Max.    :877.00
```

```
> glimpse(air_visits)
Rows: 252,108
Columns: 3
$ air_store_id <chr> "air_ba937bf13d40fb24", "air_ba937bf13d40fb24", "air_ba937bf13d40fb24", "air_ba937bf13d40fb24",
$ visit_date   <chr> "2016-01-13", "2016-01-14", "2016-01-15", "2016-01-16", "2016-01-18", "2016-01-19", "2016-01-20"
$ visitors     <int> 25, 32, 29, 22, 6, 9, 31, 21, 18, 26, 21, 11, 24, 21, 26, 6, 18, 12, 45, 15, 19, 15, 32, 3, 26,
```

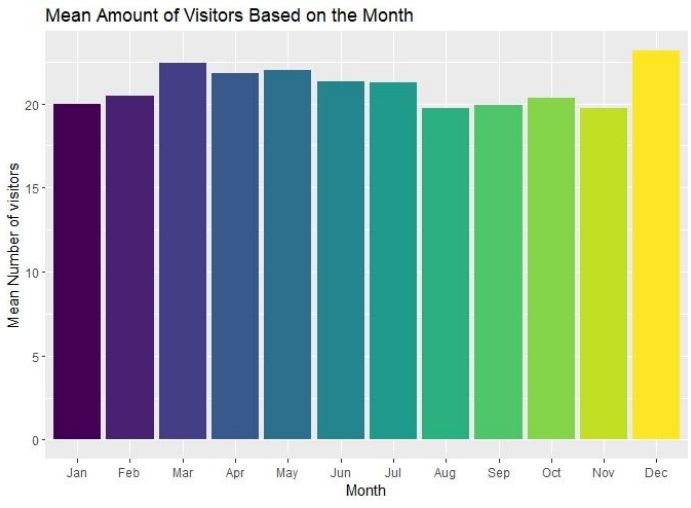
Taking a look at the summary of the air_visits dataset, there are only 829 different restaurants, and the mean visitors is only 20.97— a much smaller number than I had imagined.

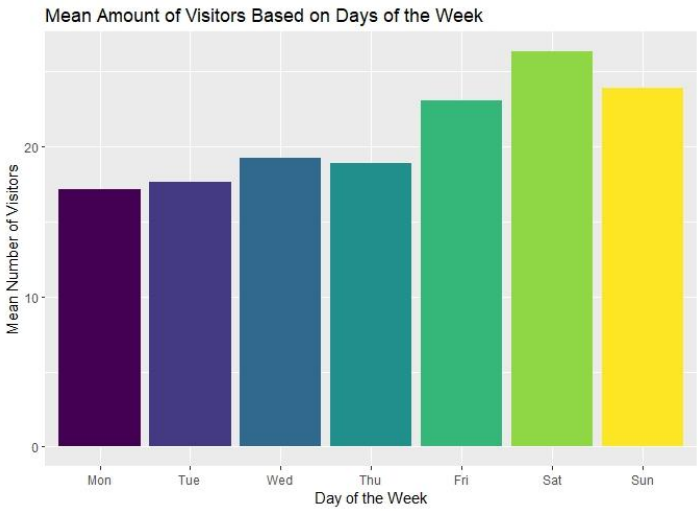
Before graphing anything, I had to check for NA values and convert the visit_date variable from chr to date.





From these graphs, we can make multiple observations. The first is that there appears to be a periodic pattern that could correlate to a weekly cycle. We can also notice that restaurant visitors per day tend to peak around 20 (red line in the graph on the right). Lastly, we see a huge increase after 2016— perhaps due to certain structural changes— and a huge dip in 2017. This will be important to keep in mind when we form our models.

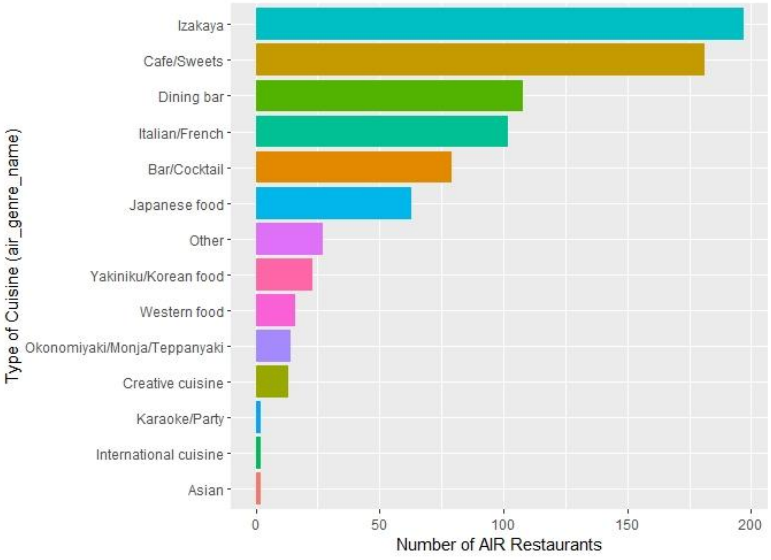


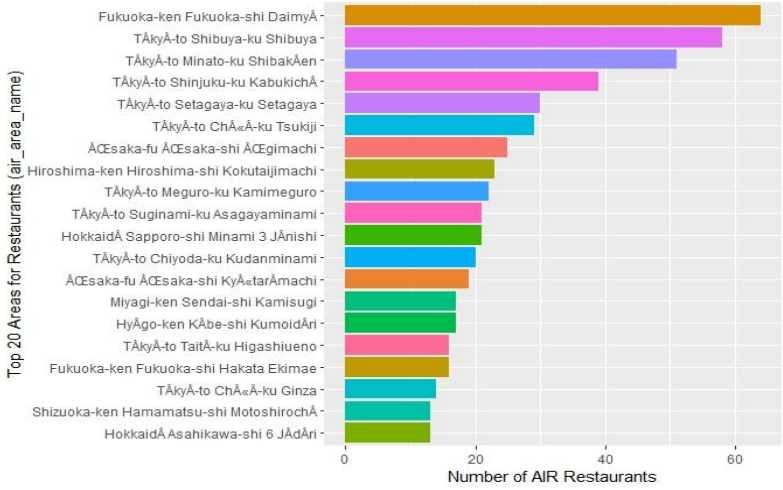


In the graphs above, we can understand that restaurant visitors peaks on the weekends (Fri-Sun), which makes sense due to people having more free time on the weekends. There is also a variation in the number of visitors for each month. December seems to be the most popular month for visits, but March-May are also consistently popular. This could be due to more free time and holidays being celebrated in these months.

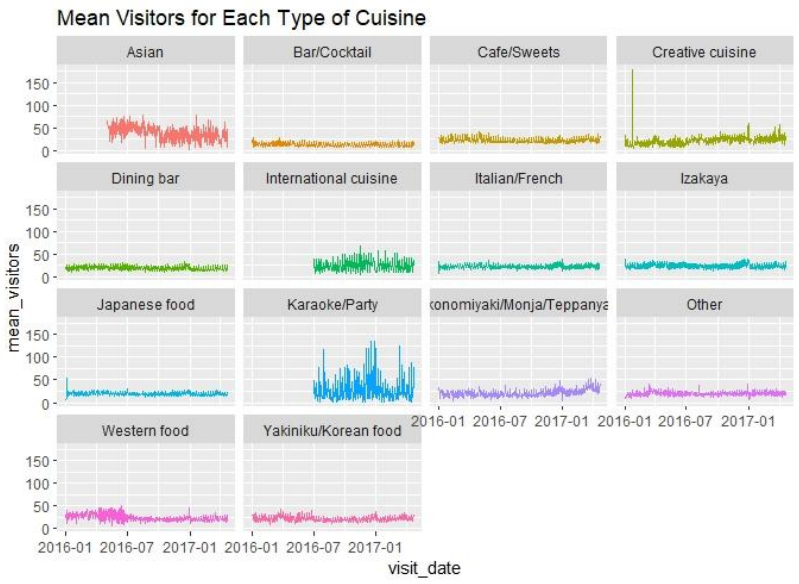
B: air_store Dataset

Analyzing the air_store dataset to understand the different types of cuisine restaurants offer and the location distributions is very important. Before plotting any graphs, I made sure to check for NA values again.





From these graphs, we can clearly conclude that Izakaya cuisine has the most number of restaurants. Fukuoka is also the most populated area with restaurants. The following five most popular areas are all in the Tokyo vicinity as well.



The third graph on the right was created when combining the air_visits dataset together with the air_store. From this graph, we notice that the average values per day range from 10-100 customers for each type of cuisine. There is not a large amount of variation with each type of cuisine. It is diligent to note that we do see larger variations and more "noise" in certain genres such as Karaoke due to lower number of visitors.

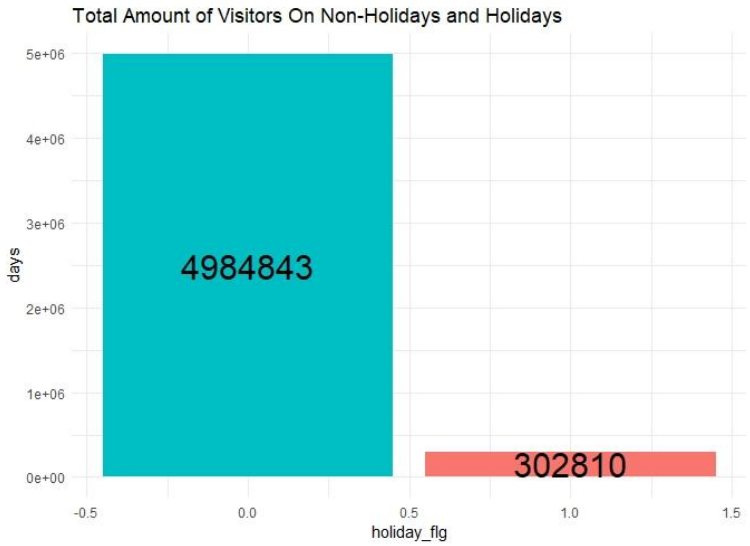
C: date_info Dataset

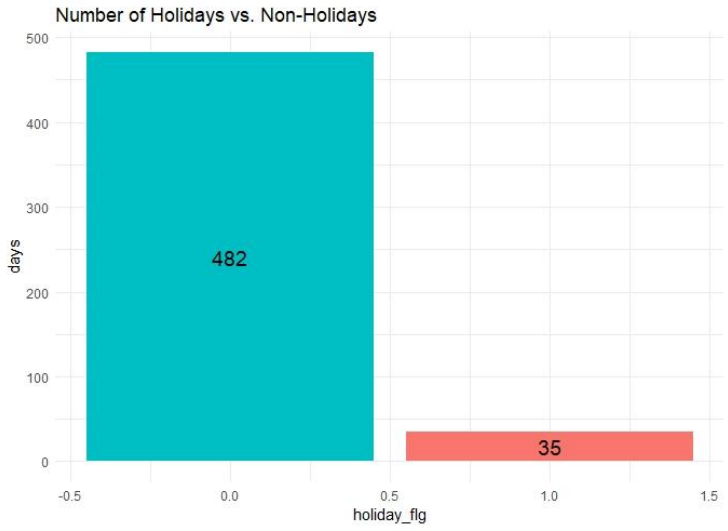
When viewing the summary results, we see that holiday_flg is a dummy variable where "1" symbolizes a holiday and "0" symbolizes a non-holiday.

```
> summary(holidays)
 calendar_date        day_of_week         holiday_flg      
 Min.   :2016-01-01   Length:517         Min.   :0.0000  
 1st Qu.:2016-05-09   Class :character   1st Qu.:0.0000  
 Median :2016-09-15   Mode  :character   Median :0.0000  
 Mean   :2016-09-15                      Mean   :0.0677  
 3rd Qu.:2017-01-22                      3rd Qu.:0.0000  
 Max.   :2017-05-31                      Max.   :1.0000  
```

```
> glimpse(holidays)
Rows: 517
Columns: 3
$ calendar_date <date> 2016-01-01, 2016-01-02, 2016-01-
$ day_of_week   <chr> "Friday", "Saturday", "Sunday", "
$ holiday_flg   <int> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1,
```

Before visualizing the data, I checked for NAs and changed the day_of_week variable from chr to date. I also chose to combine the date_info dataset with the air_visits dataset so I could determine if there was a difference in the average number of visitors when it is a holiday versus a non-holiday.





Holidays account for around 7% in our data only. The number of holidays do not seem to make a significant impact on the average number of visitors.

III. Literature

An Application of ARIMA Model to Forecast the Dynamics of COVID-19 Epidemic in India

In a study of the COVID-19 pandemic in India, ARIMA models were used to track and examine underlying the patterns in the frequency and number of COVID-19 cases. When applying different forecasting models in this situation, researchers affirm that the ARIMA model had the lowest MAPE of all methods when forecasting. The ARIMA model was proved to be useful due to it's simplicity and strong capability (Katoch & Sidhu) to explain the dataset.

Researchers plan to use the ARIMA model in order to make decisions and provide objective forecasts for confirmed cases of COVID-19 for the future.

Forecasting the Number of Incoming Tourists using ARIMA Model: Case Study from Armenia

In this case study, the ARIMA model was used to analyze patterns and to forecast the future number of tourists the demand. The paper mentions the various different situations where the ARIMA model was used to predict similar situations and their demands. The results showed that the ARIMA model outperformed other models and overall had above average performance. Through their forecasting with the ARIMA model, researchers had the ability to provide their own policy suggestions and recommend steps toward improving tourism.

fETSmcs: Feature-based ETS model component selection

This paper analyzes the utility of ETS models when confronted with different types of business decisions and dealing with large amounts of time series data. The authors mention that their ETS model selection approach was able to reduce computational costs by predicting model components from pre-trained classifiers (Qi et al.) which proved to be a very valuable tool.

Building Energy Prediction Using Artificial Neural Networks: A Literature Survey

Researchers used artificial neural network models to create an energy prediction. They found that ANN models had a unique ability to be able to model complex relationships without any sort of expert knowledge. ANN was the method with the most potential in increasing accuracy and performance. The researchers were able to implement ANNs and predict building energy demand or consumption— ultimately increasing efficiency.

Stock Market Forecasting Using Recurrent Neural Network

In a paper regarding stock market forecasting, they affirmed that NN models were extremely helpful in predicting values because of their flexibility in comparison to other traditional statistical methods. The author affirmed that NN models were widely used in the business field due to their ability to handle incomplete or noisy data, as well as not requiring prior assumptions of the distribution of data (Gao).

IV. Types of Models

Since a large majority of our class was spent on discussing ARIMA and ETS models, I instantly knew that I wanted to use these models. It also turns out that these models have also been used in cases of predicting demand and other variables. I was curious to see the results of which performed better.

While I was researching other types of models I stumbled across the Neural Network (NN) model. Essentially, neural networks are a “subset of machine learning” (IBM) and this method is heavily influenced by the human brain. Neural networks behave in a way that are similar to the way biological neurons work together. I became intrigued to how this model would

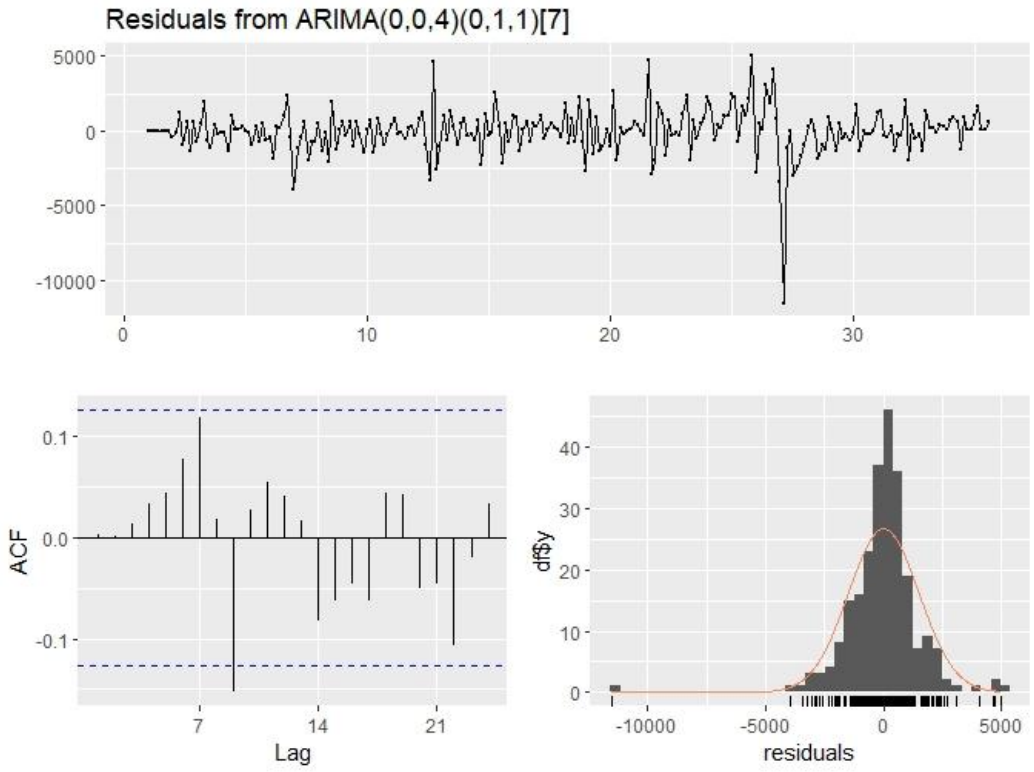
perform on forecasting visitors for our competition and thought it would be a valuable addition. I had faith that it would perform well because of how NN models can proficiently handle missing/noisy data as well as complex functions (Gao).

V. Formulation

Before building the models, I prepared the final dataset to be used by adding a holidays column to the data and set up the aggregate. Before splitting the training and testing dataset, I decided to filter the dates from after July 2016. Earlier when examining the data, I noticed that there was a huge spike in July 2016 most likely due to structural changes, so I was more curious in creating the training and testing dataset after this change. Splitting the datasets in this way is especially important because only using an average of restaurant visitors may not fully capture the essence of this change.

A: ARIMA Model

The first model I examined was the ARIMA model. I formulated this model by using the auto.arima function and set the frequency to "7" due to the weekly periodic cycle. The model defaulted to ARIMA(0,0,4). Below are the results.



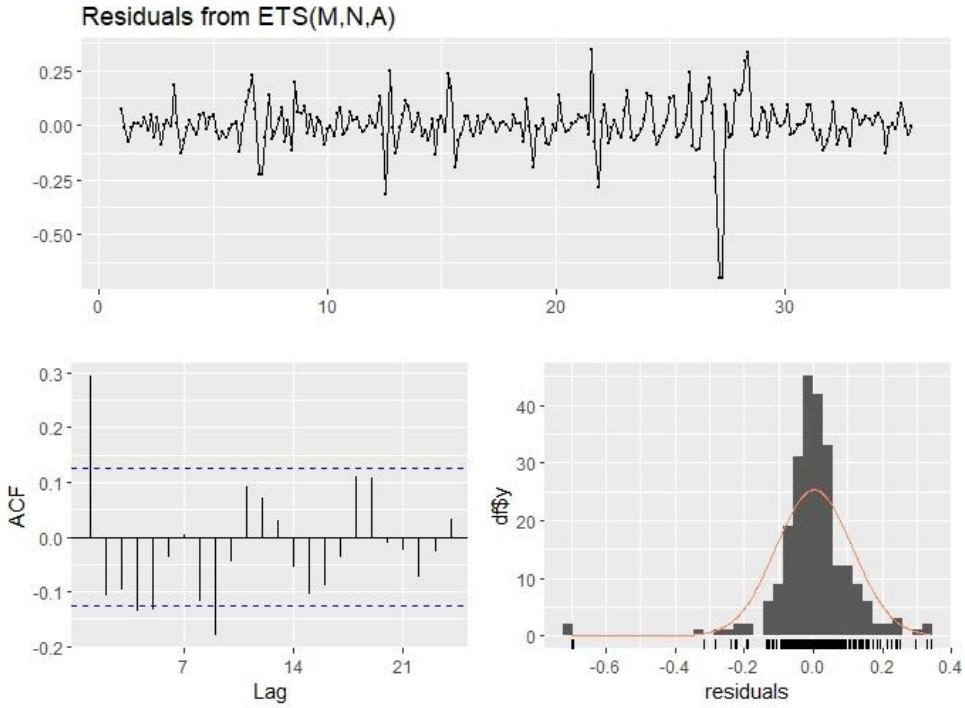
```
Series: ts(cut1.train$all_visitors, frequency = 7)
ARIMA(0,0,4)(0,1,1)[7]

Coefficients:
         ma1     ma2     ma3     ma4     sma1
      0.7704  0.3076  0.1800  0.1024  -0.9301
s.e.  0.0660  0.0830  0.0819  0.0574   0.0420

sigma^2 = 2354973:  log likelihood = -2070.99
AIC=4153.98   AICc=4154.34   BIC=4174.76

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE        ACF1
Training set -9.392344 1496.221 975.8477 -1.508499 7.611166 0.6294639 0.003326546
```

```
        Ljung-Box test

data:  Residuals from ARIMA(0,0,4)(0,1,1)[7]
Q* = 15.199, df = 9, p-value = 0.08562

Model df: 5.   Total lags used: 14
```

We obtained a RMSE of 1496.22, an MAE of 975.85, and an AIC of 4153.98. There are a few outliers in the residuals, but they are still normally distributed. After, I decided to forecast the ARIMA model 53 days into the future til the end of the test data to be able to compare it with other models.

B: ETS Model

The next model I created was the default ETS model, which would allow an adjust for seasonality. I set the frequency to "7" to account for the weekly periodic cycle and set the model to 'ZZZ' — signifying automatic selection. The most optimal ETS model was (M,N,A). The results are as shown below.



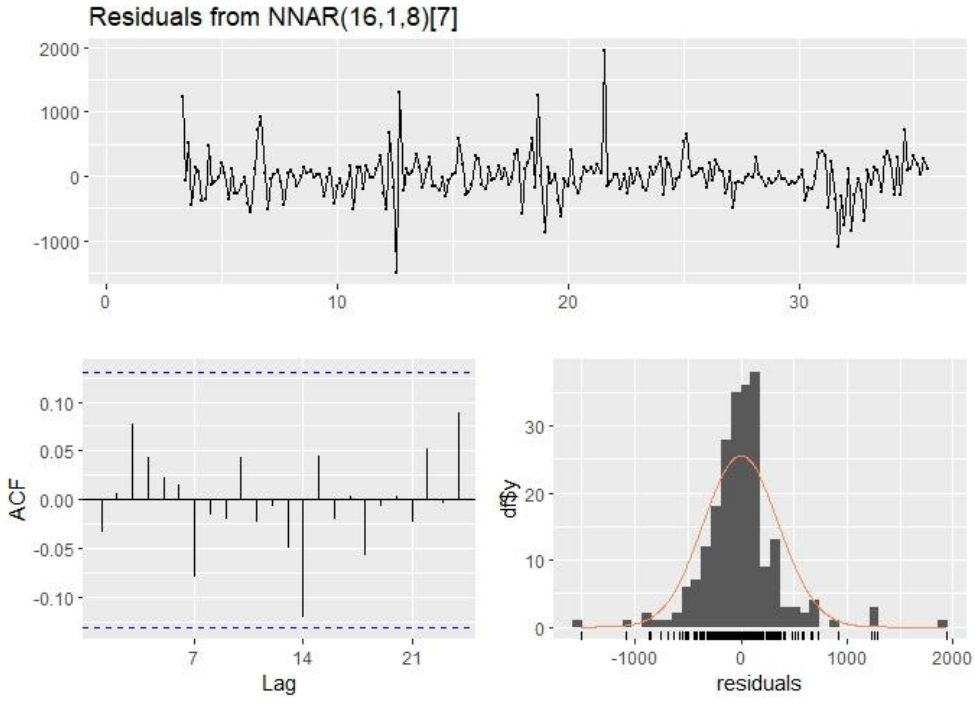
```
ETS(M,N,A)

Call:
 ets(y = ts(cut1.train$all_visitors, frequency = 7), model = "ZZZ")

  Smoothing parameters:
    alpha = 0.3479
    gamma = 1e-04

  Initial states:
    l = 14451.5607
    s = -543.4404 -727.7131 -2170.374 -3835.75 -494.0252 4882.413
           2888.889

  sigma:  0.1124

     AIC     AICc      BIC
4912.763 4913.711 4947.694

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE      ACF1
Training set 9.097243 1701.823 982.1447 -1.869905 8.162042 0.6335257 0.3596927
```

```
        Ljung-Box test

data:  Residuals from ETS(M,N,A)
Q* = 52.576, df = 5, p-value = 4.11e-10

Model df: 9.   Total lags used: 14
```

We obtained a RMSE of 1701.82, an MAE of 982.15, and an AIC of 4913.71 for our default ETS model. Similarly to the ARIMA model, I forecasted 53 days into the future to determine the accuracy.

C: Neural Network Model

Lastly, I used the nnetar() function to create a neural network model. This function works by fitting a neural network model to a time series with lagged values as inputs. The frequency was set to "7" to account for the weekly periodic cycle. The results are as follows.




```
> summary(cut1.nnet)
          Length Class        Mode
x         243    ts           numeric
m           1    -none-       numeric
p           1    -none-       numeric
P           1    -none-       numeric
scalex      2    -none-       list
size        1    -none-       numeric
subset    243    -none-       numeric
model      20    nnetarmodels list
nnetargs    0    -none-       list
fitted    243    ts           numeric
residuals 243    ts           numeric
lags       16    -none-       numeric
series      1    -none-       character
method      1    -none-       character
call        2    -none-       call
```

From these results we can conclude that although the residuals are normally distributed, there are more outliers than both the ARIMA and ETS models. Luckily, the frequency of [7] was still apparent in this model.

D: Comparing the Results

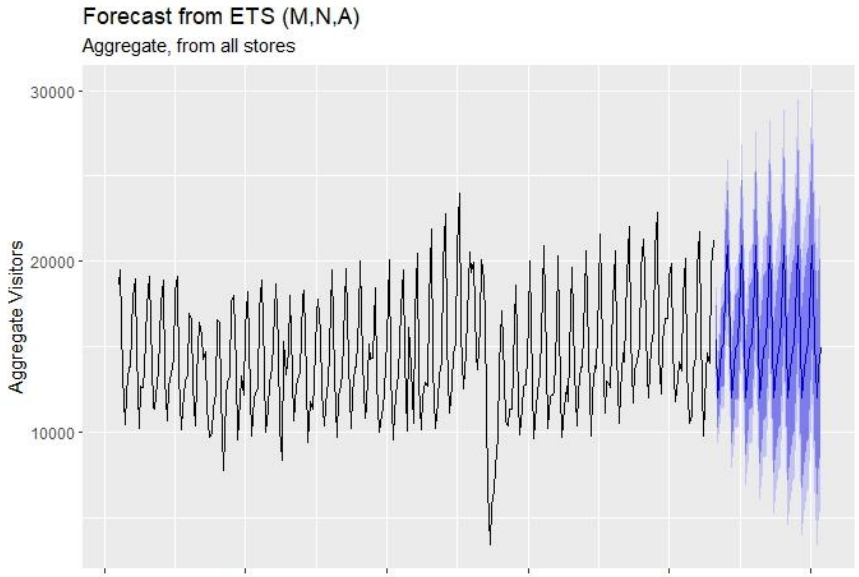
To compare the results, I created a matrix that would contain RMSE and MAE as evaluation metrics to determine the best model.

	RMSE-train	MAE-train	RMSE-test	MAE-test
ARIMA Model	1496.2214	975.8477	1955.9368	1601.5425
ETS Model	1701.8227	982.1447	1356.9200	1052.2063
Neural Networks	350.2691	228.9949	2138.1982	1496.2214

The Neural Networks (NN) model performed the best in all of the train datasets. It had a much lower RMSE of 350.27 and MAE of 229 compared to the ARIMA and ETS models. However, on unseen data NN performed terribly, which was shocking to me. The ETS (M,N,A) model reigned superior due to having the lowest RMSE of 1356.92 and MAE value of 1052.21 in comparison to the others.

The ETS (M,N,A) is our final model of choice. The results and forecast are as follows.

```
ETS(M,N,A)

Call:
 ets(y = ts(cut2$all_visitors, frequency = 7), model = "ZZZ")

  Smoothing parameters:
    alpha = 0.3451
    gamma = 2e-04

  Initial states:
    l = 14498.7956
    s = -674.6785 -915.6947 -2200.504 -3803.51 -479.7646 5053.785
           3020.367

  sigma:  0.1066

     AIC     AICc      BIC
6020.765 6021.537 6057.669

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE      ACF1
Training set 13.01053 1623.717 954.0923 -1.589591 7.667284 0.6302864 0.3486499
```



VI. Performance and Accuracy

My submission had a private score of 0.61463 and a public score of 0.58304. I think that this was a relatively solid attempt, and a decent score considering I did not use all the datasets given. I believe that using the forecast matrix which contained multiple models (ETS, ARIMA, Neural Network) helped with the accuracy of the attempt, versus just using one model.

VII. Limitations

The most apparent limitations are the unaccounted for factors that were not shown in the dataset. I do think that the information in the datasets given were not expansive enough for this task, which may explain some inaccuracy. There are other factors that impact visitors (and are very hard to predict) such as the weather, demand, and current food trends. Especially because the impact of social media is so prevalent and can impact the business of a restaurant (for better or for worse), taking trends into account would be a crucial step in creating a better forecast.

Furthermore, I only focused on creating univariate models for this assignment. However because multivariate models lead to further insight on the relationship between multiple variables, I believe that using multivariate models would have improved the results. In this aspect, my models are insufficient but not wrong.

VIII. Conclusion

Overall, this competition was difficult but fun to explore. I enjoyed applying the skills and methods learned from class to real-world situations— which is hard to conceptualize at times. Finding the best model for forecasting was challenging, but I also learned and was able to see how differently models perform on training and unseen data. Though the ARIMA and Neural Networks models performed substantially better on the training data, the ETS model outshined the others on the testing data. It was compelling to be able to see how this could be possible firsthand.

If I reattempted this competition in the future, I would improve my model by including more relevant datasets and trying other forecasting methods. I focused more on ARIMA, ETS, and NN models for this assignment, but there are many other models I could have tried. I would be interested in using the Holt-Winters or Prophet method next time. Additionally, I would have included the different types of cuisines in my model. To do this, I would create a model for each genre of cuisine and adjust it in accordance. It would have been fascinating to examine the intersection between certain trends at the time and different genres of food and see if this would affect visitor counts for certain restaurants.

Works Cited

Gao, Qiyuan. *Stock Market Forecasting Using Recurrent Neural Network*. University of Missouri-Columbia, May 2016, https://mospace.umsystem.edu/xmlui/bitstream/handle/10355/56058/research.pdf.

IBM Cloud Education. "What Are Neural Networks?" *IBM*, 17 Aug. 2020, https://www.ibm.com/cloud/learn/neural-networks.

Katoch, Rupinder, and Arpit Sidhu. "An Application of ARIMA Model to Forecast the Dynamics of COVID-19 Epidemic in India." *Sage Journals*, International Management Institute, New Delhi, 8 Mar. 2021, https://journals.sagepub.com/doi/full/10.1177/0972150920988653.

Lu, Chujie, et al. "Building Energy Prediction Using Artificial Neural Networks: A Literature Survey." *Research Gate*, 26 Nov. 2021, https://www.researchgate.net/publication/356555138_Building_Energy_Prediction_Using_Artificial_Neural_Networks_A_Literature_Survey.

Qi, Lingzhi, et al. "FETSmcs: Feature-Based ETS Model Component Selection." *International Journal of Forecasting*, Science Direct, 29 July 2022, https://www.sciencedirect.com/science/article/abs/pii/S0169207022000954#!

Tovmasyan, Gayane. "Forecasting the Number of Incoming Tourists Using ARIMA Model: Case Study from Armenia." *Armg Publishing*, Sumy State University, 13 Sept. 2021, https://armgpublishing.com/journals/mmi/volume-12-issue-3/article-12/.