# Forecasting Total Passengers for Flights to and from Charlotte and Raleigh/Durham, NC

Maria Duran Rondon, Neal Lockhart, Ixchel Peralta-Martinez

M.S Data Science

University of North Carolina Wilmington

Course Number: DSC 551

May 1, 2024

**Table of Contents**


**Abstract** ..... 3

**1. Introduction** ..... 4

**2. Methodology** ..... 4

2.1 Data Manipulation ..... 4

2.2 Model Creation ..... 5

**3. Analysis** ..... 11

3.1 Exploratory Data Analysis ..... 11

3.2 Spatial Analysis: Geocoding and Mapping Flight Data ..... 14

**4. Forecasting Results** ..... 16

**5. Discussion** ..... 17

**6. Conclusion and Future Work** ..... 18

## Abstract

This report presents a forecast of total passenger for flights coming from or going to Charlotte and Raleigh/Durham North Carolina for Jan. 2020 – Jun. 2020 using data from January 2000 to December 2019. Focusing on the primary air traffic hubs in North Carolina, the research involves extensive data manipulation and the application of various forecasting models including Exponential Smoothing State Space Model (ETS) and Seasonal ARIMA. The findings indicate that specific ARIMA configurations provide the most reliable forecasts. The research was executed in the R programming language, employing a range of techniques from data cleaning to exploratory data analysis and model selection. This report details the outcomes of the comprehensive analysis and explores the strategic implications of these results.

## 1. Introduction

This project aims to employ predictive forecasting techniques to accurately project airline passenger traffic for Charlotte and Raleigh/Durham in the United States from Jan. 2020 - Jun. 2020. The analysis utilizes data spanning from January 2000 to December 2019, focusing on variables such as location and time to build the models. Charlotte and Raleigh-Durham were selected as the primary study locations because they are central to North Carolina's air travel, together handling 91.69% of the state's total air traffic. Exploratory data analysis was conducted to delve into the details of the flight data. Additionally, a geographic map was created to visualize the spatial distribution of the data.

## 2. Methodology

### 2.1 Data Manipulation

The U.S. Monthly Air Passenger dataset is a comprehensive collection of data that includes detailed information about the number of airline passengers per month, flight origins and destinations, and the specific months and years of those flights, spanning from January 2000 to December 2019. The dataset provides extensive details, such as the names of the origin and destination cities, state abbreviations and names, country codes and names, destination airport codes, and the timing of flights. Due to the vast scope of the dataset, which includes numerous origin and destination countries, our analysis was refined to focus solely on flights within the United States, specifically targeting the state of North Carolina. We concentrated on Raleigh and Charlotte, as these cities account for 91.69% of North Carolina's total air traffic during the study period. We used R filtering methods using the dplyr package to filter our data down to just North Carolina flights coming to/from Charlotte or RDU. After narrowing our scope, the dataset included 217,683 flights with no significant missing values. There was a minor gap between five missing airline IDs, but these were not substantial enough to compromise the data's overall integrity.

We aggregated passenger data by month and year, creating a time series object from this consolidated information. To assess our model's accuracy, we split the data into training and testing sets, allocating 90% of the data for training and 10% for testing. For the spatial component, we utilized the Google Maps API using the ggmap package in R to obtain precise latitude and longitude coordinates for the unique city names. These coordinates were then integrated back into the dataset.

## 2.2 Model Creation

The data was aggregated by calculating the total number of passengers monthly and annual. A seasonal plot was then used to examine the presence of seasonal trends across different years, identifying noticeable increases in passenger numbers during March and May and a decline in September. Subsequently, the Seasonal and Trend decomposition using Loess (STL) was applied to the training dataset. As seen in *Figure 1.1* decomposition reveals a clear fluctuation within the original time series data, along with a discernible long-term upward trend. This trend component smoothed out the seasonal variations, displaying a gradual increase over time. The seasonal plot highlighted a consistent and repeating seasonal pattern, with peaks occurring annually. After extracting the trend and seasonal components, the remainder component appeared random, capturing the irregularities and non-systematic fluctuations in the data.

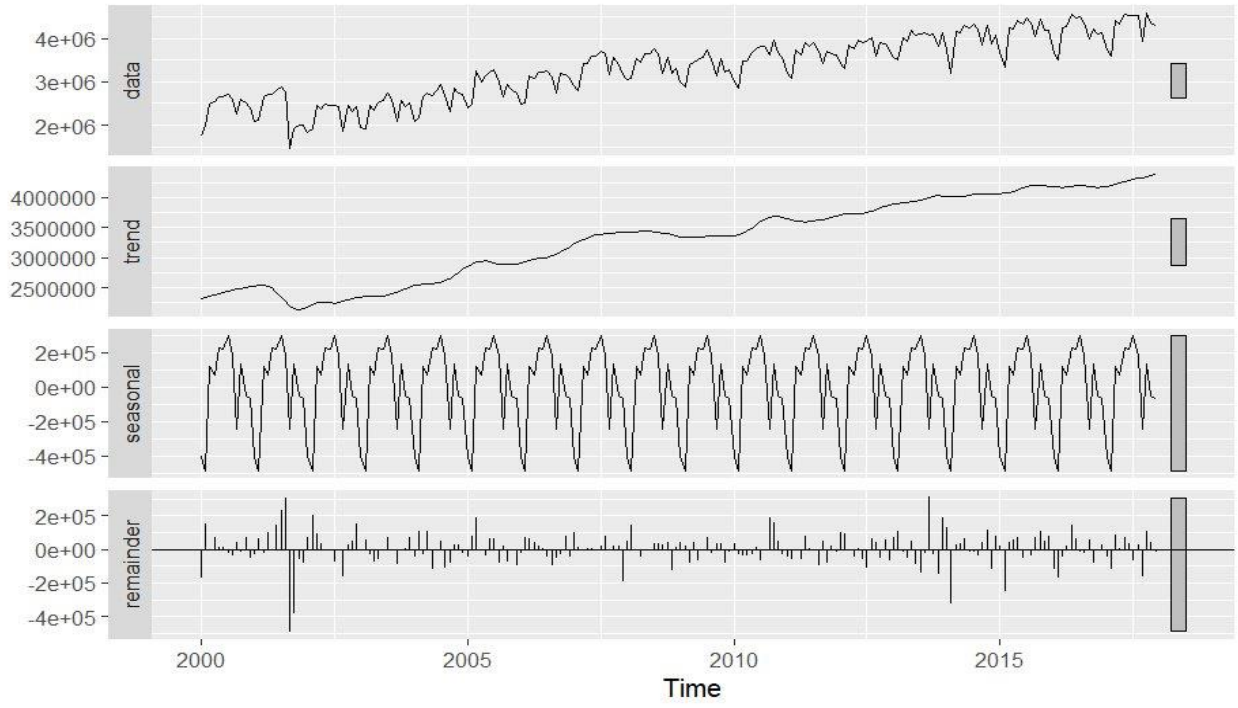



*Figure 2.2.1* Decomposition on the Training Data

To ensure consistent variance across the data, a variance stabilization transformation was conducted. The Box-Cox transformation was selected for its effectiveness in normalizing data distributions and stabilizing variance. An optimal lambda ($\lambda$) value of 1.127672 was chosen to achieve the best variance stabilization, which was then integrated into our forecasting functions. Four distinct forecasting techniques were implemented to establish a benchmark and evaluate various simplistic approaches for time series prediction. The Naive Forecast serves as a baseline by assuming future values will replicate the last observed value. The Seasonal Naive method adjusted for recurring seasonal patterns, while the Drift Forecast incorporated a linear trend component. The Average Forecast predicted future values by computing the historical mean. Each model applied the Box-Cox transformation with the determined optimal lambda to the training data with aggregated monthly and yearly passenger data. The Root Mean Square Error (RMSE) was calculated for each model and stored for comparison on training data.

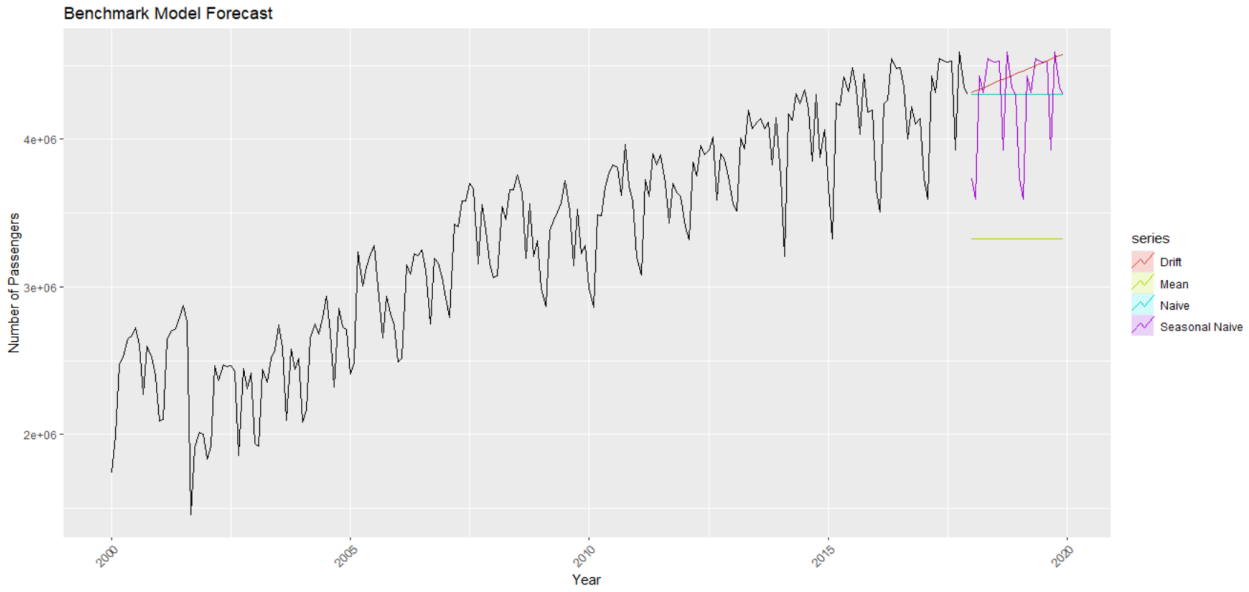



*Figure 2.2.2* Forecast of the four benchmark models and training data

For the time series forecasting of passenger data, an Exponential Smoothing State Space Model (ETS) was also implemented to capture the underlying patterns of trend, seasonality, and error. The ETS model, specifically the ETS(A, A, A) version, was chosen based on its ability to model data with additive error, trend, and seasonal components, for the given characteristics of the dataset. The model was identified using the ets() function in R, configured to automatically select the best fitting model based on information criteria (BIC). The chosen model was an ETS(A, A, A), both the trend and seasonal components were best modeled as additive effects, which aligns with the observed seasonal peaks and overall upward trend in passenger numbers. The model also implemented the Box-Cox transformation, with a lambda value of 1.127672. The Root Mean Square Error (RMSE) was calculated for each model and stored for comparison on training data.

To effectively implement ARIMA models for forecasting, it was essential first to establish the stationarity of the time series data. Initial assessments using the KPSS Unit Root Test indicated non-stationarity; the test statistic of 4.1306 significantly exceeded the critical value of 0.463 at the 5% significance level. This result confirmed that the time series was not stationary, needs a transformation to achieve stationarity. This involved applying seasonal differencing. After applying one seasonal differencing, a subsequent KPSS test showed a test statistic of 0.1466, which is below the 5% critical value, confirming that the series had become stationary with this adjustment.

To examine the stationary data, the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) on *Figure 2.2.3* were visualized using the ggtsdisplay function in R. The plots revealed significant autocorrelations at initial lags (1-7) and a notable yearly pattern at lag 12, suggesting an AutoRegressive (AR) process with a strong seasonal influence. The PACF displayed a prominent spike at lag 1 and additional seasonal spikes at lags 12 and 24, indicating the need for an AR component and seasonal adjustments.

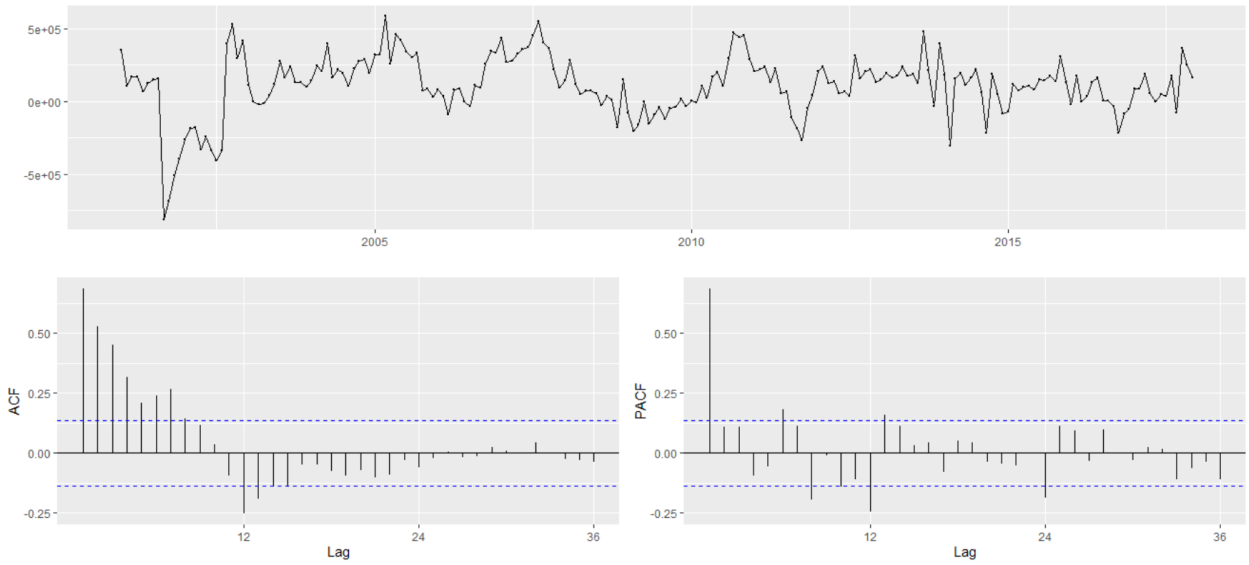


*Figure 2.2.3* Autocorrelation Function and Partial Autocorrelation Function

Based on these insights, a Seasonal ARIMA model should be used to accommodate both non-seasonal and seasonal dependencies. The auto.arima() function was utilized to automate the selection of the best fitting model, using the Akaike Information Criterion (AIC) for optimization due to its efficacy in predictive modeling contexts. The function was configured with stepwise set to false, D set to 1 to account for one seasonal difference, and d set to 0 indicating no non-seasonal differences. This function selected an AR model that includes a seasonal component ARIMA(1,0,1)(2,1,1)[12] with drift Box Cox transformation: lambda= 1.127672.

To explore further, another ARIMA model was configured, this time employing the Bayesian Information Criterion (BIC) for a more regularized model selection approach. This function selected an AR model that includes a seasonal component ARIMA(1,0,1)(0,1,1)[12] with drift with drift Box Cox transformation: lambda= 1.127672. Each model's performance was quantitatively evaluated, with the Root Mean Square Error (RMSE) calculated and stored for comparison across all models on the training data.

## 2.3 Model Selection

Comparing the RMSE values obtained from the training data to determine the top-performing models, it was found that the primary candidates were two ARIMA models and one ETS model show in the *Table 2.1.3*.

| | ARIMA(1,0,1)(2,1,1)[12] | ARIMA(1,0,1)(0,1,1)[12] | ETS(A,A,A) |
|---|---|---|---|
| *RMSE* | *291369.3* | *288072.5* | *631790.2* |

*Table 2.3.1* RSME of ARIMA Models and ETS Model

A check on the residuals for each model was done using the checkresiduals function in R with a lag of 24. The Ljung-Box test results and visual inspections indicated that the ETS(A,A,A) model displayed autocorrelation in its residuals, suggesting a potential need for further model refinement or specification adjustments. The ARIMA(1,0,1)(2,1,1)[12] with drift  Was above the .05 significance level indicating the residuals were white noise. The ARIMA(1,0,1)(0,1,1)[12] with drift exhibited the least autocorrelation among the models, indicating a robust fit to the time series data. Residuals from the ARIMA(1,0,1)(0,1,1)[12] with drift model were well-centered around zero, suggesting an effective fit. Autocorrelation values primarily stayed within the confidence intervals, reflecting minimal autocorrelation. Ljung-Box Test: A p-value of 0.07997 confirmed the lack of significant autocorrelation, supporting the model's capability in capturing the dynamics of the time series without leaving residual patterns.

A cross-validation process was implemented to further test the models over different data segments and forecast horizons. This method allowed for an extensive evaluation of each model's predictive accuracy. The ARIMA(1,0,1)(0,1,1)[12] with drift consistently showed the lowest RMSE, proving its efficiency in accurately forecasting seasonal patterns. Cross-validation was also performed on ETS model ETS(A,A,A).Despite its suitability for the data characterized by additive trends and seasonality, the ETS model did not perform as well, suggesting that its residuals might still contain autocorrelation or the need for further tuning.

The final model selected was the ARIMA(1,0,1)(2,1,1)[12] with drift, for, despite a weaker cross validation score than the lower order ARIMA model, the score was close enough to indicate there was no significant overfitting and therefore would provide greater predictive power.  In

addition to this, the model's residuals were indeed white noise. This model employs settings such as approximation=TRUE, seasonal=TRUE, ic="aic", stepwise=FALSE, d=0, and D=1. It demonstrates a superior in-sample RMSE, indicating a better fit to the training data and suggesting that it effectively captures the dataset's underlying patterns, which are crucial for accurate forecasting. Although its cross-validation RMSE is slightly higher than that of the ARIMA(1,0,1)(0,1,1)[12], the difference is minimal, indicating that the ARIMA(1,0,1)(2,1,1)[12] generalizes well to unseen data. The use of approximation=TRUE speeds up the model fitting process, a significant benefit for large datasets, while also reducing computational demands without substantially compromising accuracy.

Selected Model Equation:

$$y_t' = \epsilon_t + 0.8711y_{t-1}' - 0.3289\epsilon_{t-1} + 0.0403y_{t-12}' - 0.0508y_{t-24}' - 0.7172\epsilon_{t-12} + 69592.891 \cdot t$$

# 3. Analysis

## 3.1 Exploratory Data Analysis

In this section of the exploratory data analysis, we delve into various aspects of flight and passenger data to uncover patterns and insights that can inform future decisions and strategies. We begin by visualizing the top 15 airlines based on the number of flights they operate, followed by a look at the top 15 destination cities. Additionally, we explore specific data from Charlotte and Raleigh/Durham airports in North Carolina and analyze passenger numbers over different months and years.

Figure 3.1.1shows a visualization of the top 15 airlines sorted by the total number of flights. This is achieved by grouping the data by carrier name and counting the flights. The bar chart is

enhanced with a gradient color scale from light blue to blue, emphasizing airlines with a higher number of flights. US Airways Inc., PSA Airlines Inc., and Southwest Airlines Co. are the leading carriers with significantly higher flight operations compared to others.

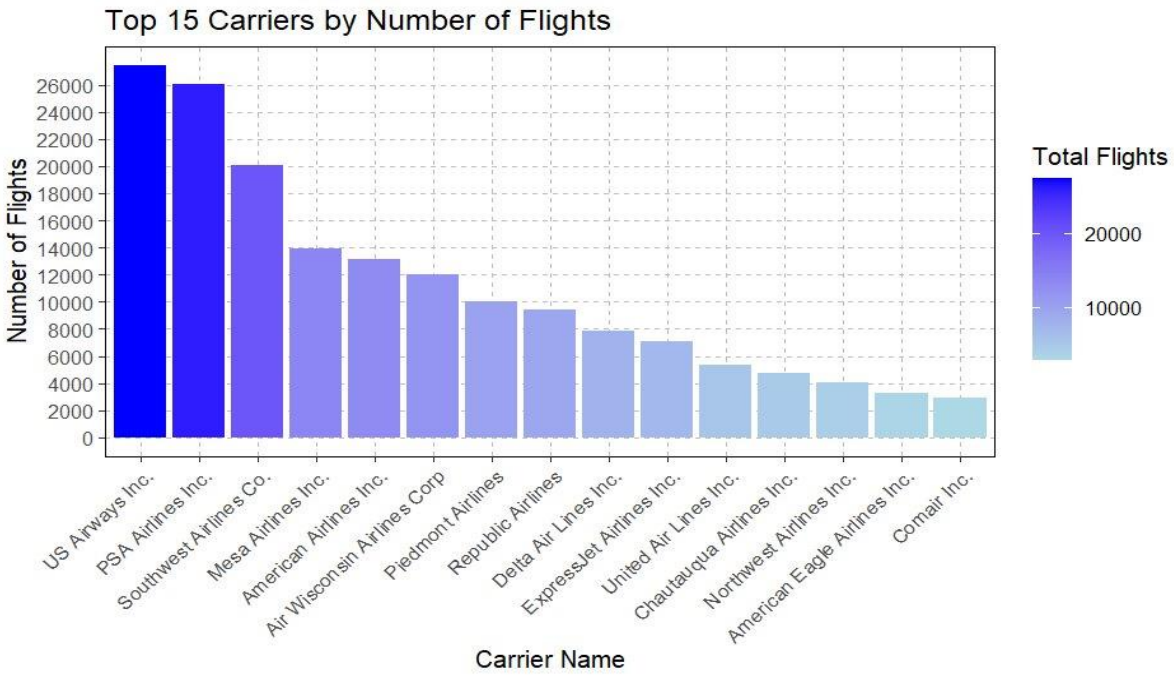



*Figure 3.1.1* Top 15 Airlines Sorted by the Total Number of Flights

Like the carriers, we analyze the top destination cities in *Figure 3.1.2*. The data is grouped by destination city name, and the number of flights is counted to identify the top 15 cities. This bar chart uses a color fill based on the destination city, providing a visual differentiation between cities. Common destinations between the two airports are highlighted in different colors (blue for Charlotte, red for Raleigh/Durham) against a gray background for other destinations. This dual-bar chart layout allows for an immediate visual comparison between the two airports, emphasizing shared and unique routes.

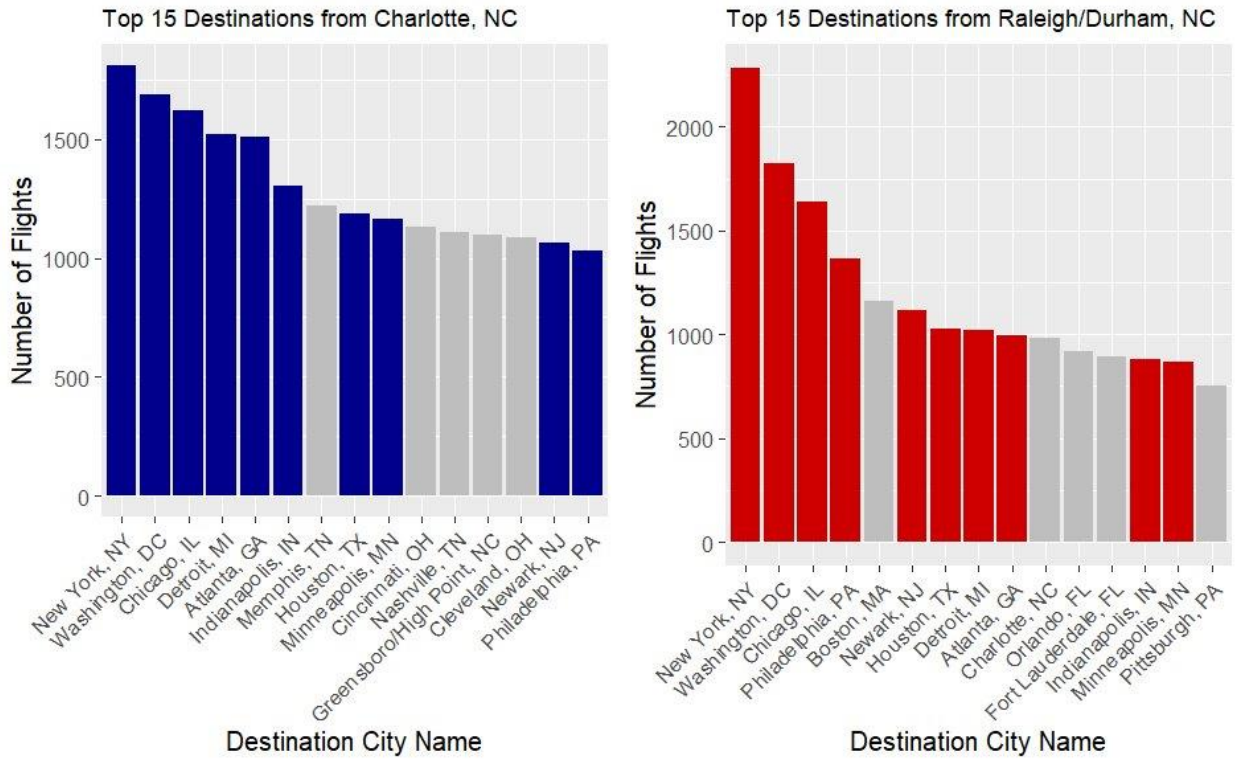



*Figure 3.1.2* Top 15 Destinations From Charlotte and Raleigh, North Carolina

Finally, a heatmap to easily track passenger numbers over various months and years. In this visualization, data is organized by year and month, and the total number of passengers for each period is calculated. The heatmap displays this information using a color gradient ranging from yellow to purple. Light yellow represents months with fewer passengers, and dark purple indicates months with more passengers. This color coding clearly shows trends in passenger traffic, with dark purple spots highlighting peak travel times, usually later in the year. The gradient shift from yellow to purple across the heatmap helps to quickly spot years with increasing travel activity In the heatmap, we observe that darker purple shades, indicating higher passenger volumes, begin to appear more frequently after the year 2013.

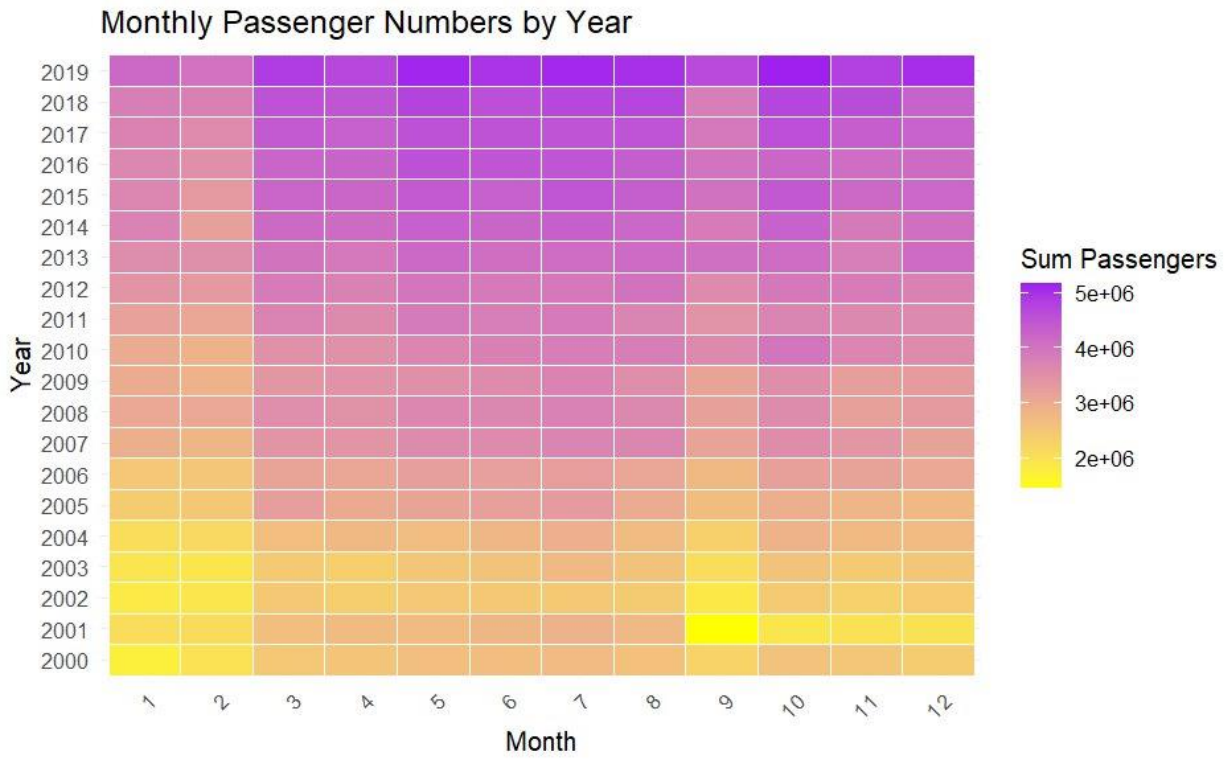



*Figure 3.1.3 Heatmap to Showing Number of Passengers for Month and Year*

## 3.2 Spatial Analysis: Geocoding and Mapping Flight Data

In this section of the analysis, we focus on enhancing our dataset with spatial components to better understand geographic trends in flight data. The process begins by isolating city names from our primary dataset and creating a new data frame to handle these unique geographical identifiers. We then use the Google Maps API to geocode each city, transforming city names into precise latitude and longitude coordinates.

Once we have the geographic coordinates, we incorporate them back into our main dataset, enabling us to conduct spatial analyses and visualizations. The dataset was broken into two-parts flights originating from or destined to Charlotte and Raleigh/Durham, North Carolina. This approach helps in visualizing not only the volume of flights but also their geographical distribution

across the U.S. We also generate maps using the sf (simple features) package in R, which facilitates the integration of complex geographic data with traditional data frames. By plotting these coordinates, we create visual representations that highlight flight traffic patterns and densities. In *Figure 3.2.1*, there is a noticeable concentration of heavy traffic along the East Coast, with many passengers. To better present the flight data between Charlotte and Raleigh and other destinations, we are implementing a more interactive and intuitive mapping solution using the Shiny and Leaflet libraries in R shown in *Figure 3.2.3*. This approach allows users not only to see static data points but also to interact dynamically with the information. The user interface of our Shiny app includes options to select views based on direction (to or from Charlotte/Raleigh), as well as filters for specific years, months, or airlines.

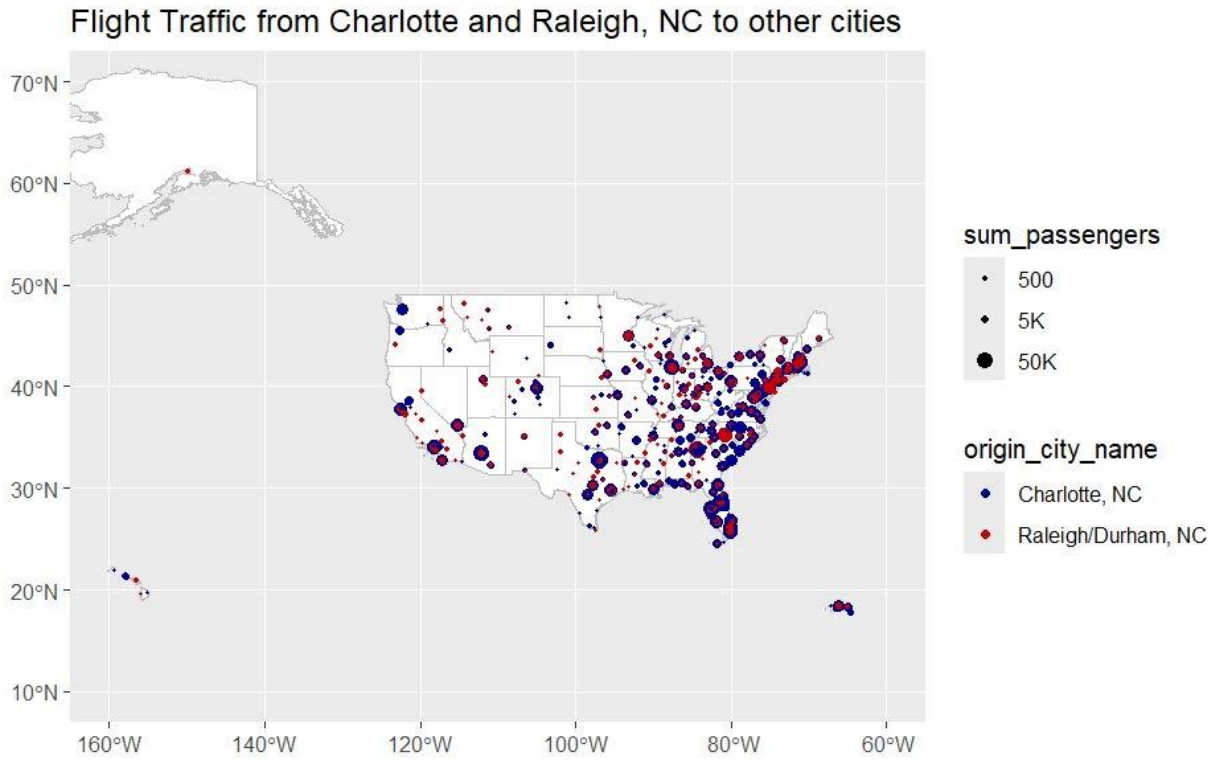



*Figure 3.2.1 Flight Traffic from charlotte and Raleigh/Durham, NC*

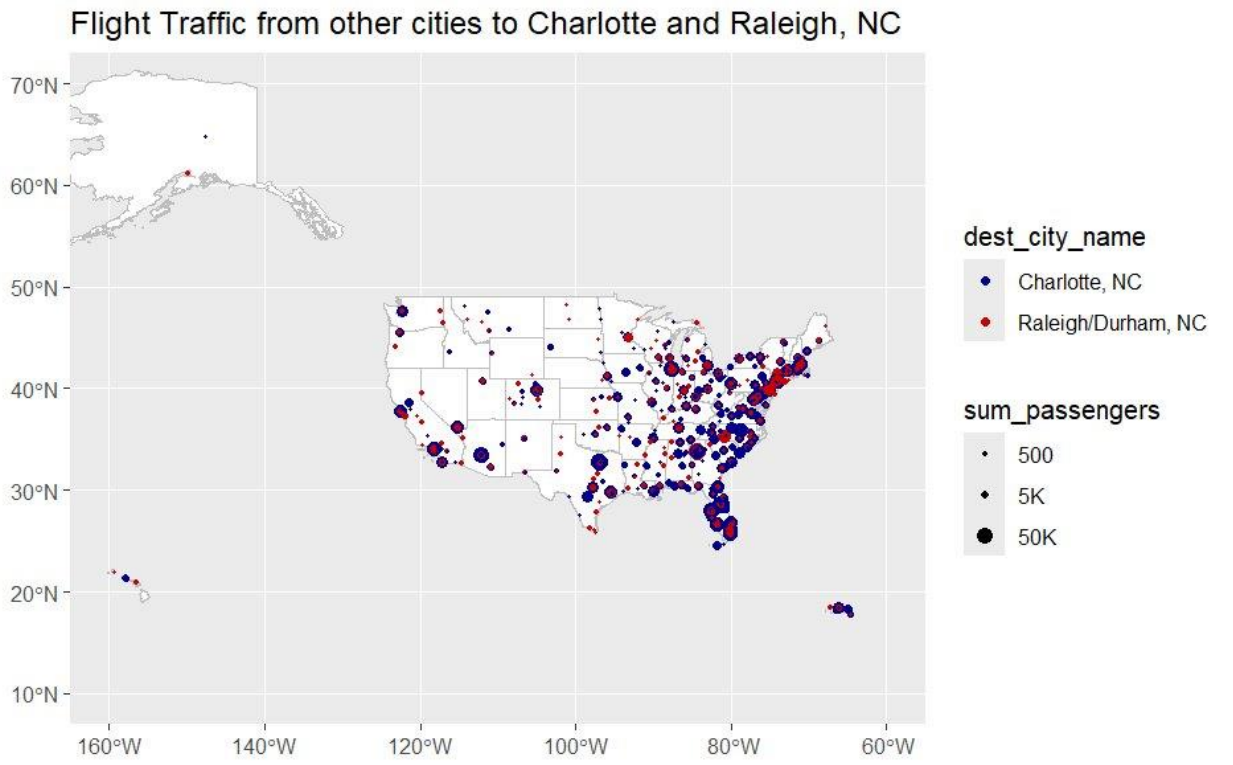



*Figure 3.2.2 Flight Traffic to charlotte and Raleigh/Durham, NC*

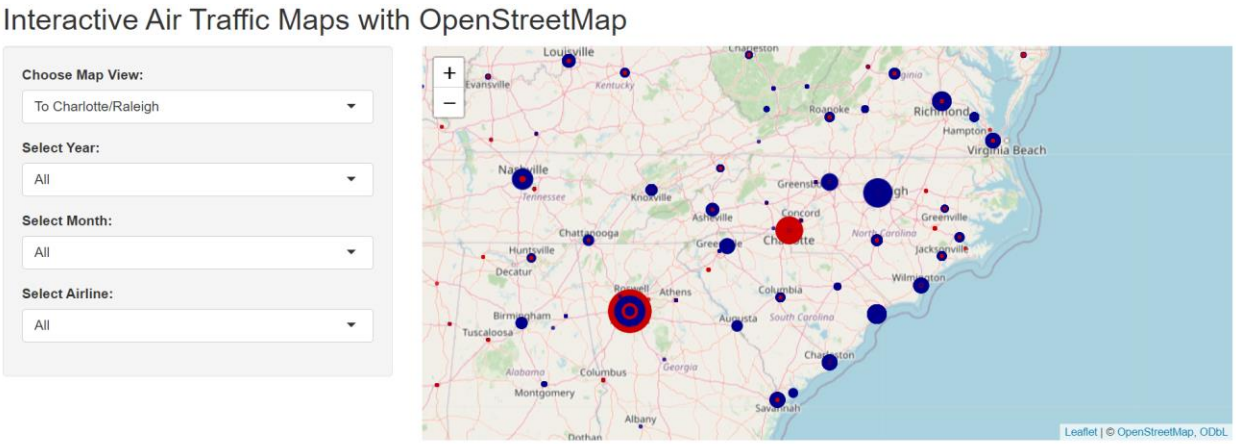
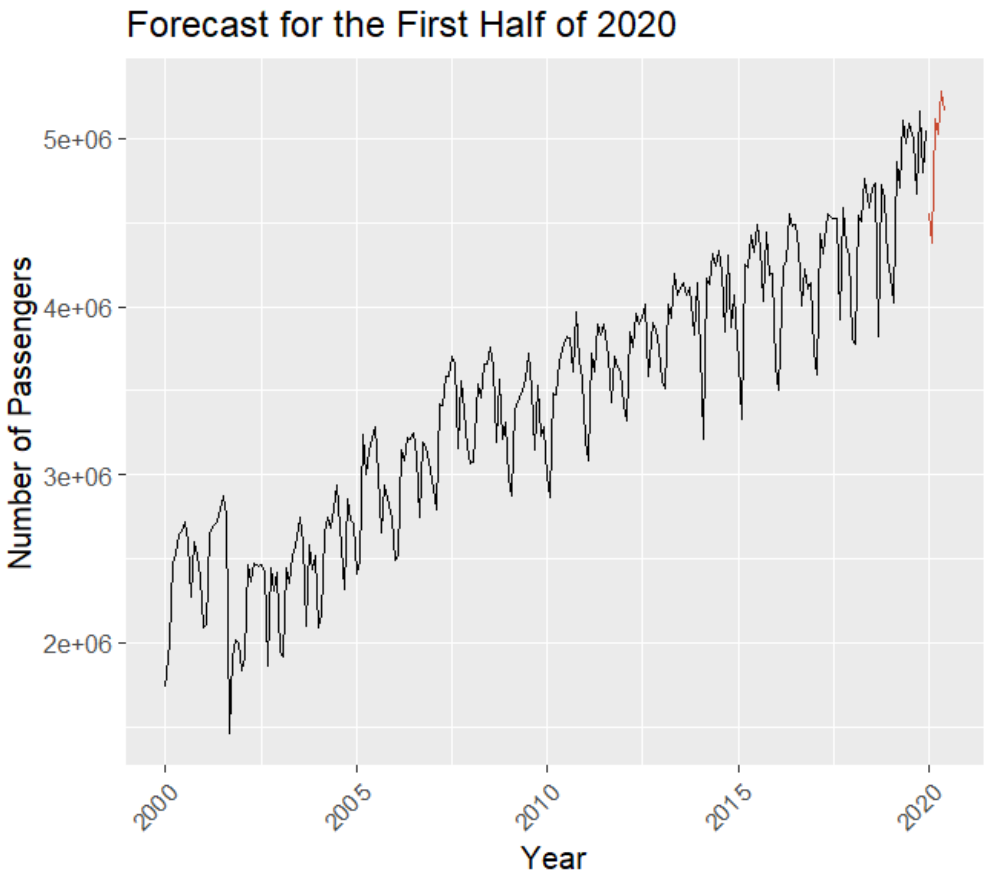



*Figure 3.2.3 R Shiny Interactive Air Traffic Map*

4. Forecasting Results

| Month, Year | Passenger Forecast |
|---|---|
| January 2020 | 4567622 |

| | |
|---|---|
| February 2020 | 4389766 |
| March 2020 | 5126686 |
| April 2020 | 5025187 |
| May 2020 | 5282171 |
| June 2020 | 5169308 |




## 5. Discussion

The actual passenger numbers during 2020 are likely to deviate significantly from this forecast due to the unprecedented impact of the COVID-19 pandemic. This impact includes widespread travel restrictions, lockdowns, and a significant reduction in demand for travel services.

## 6. Conclusion and Future Work

The project successfully developed robust predictive models to forecast airline passenger traffic, particularly focusing on the interplay between Charlotte and Raleigh-Durham. The comparison of various models, such as ETS and multiple configurations of ARIMA, highlighted the nuances in their predictive abilities. The Seasonal ARIMA model, particularly ARIMA(1,0,1)(2,1,1)[12] with drift, emerged as the most effective, demonstrating high accuracy in both training and cross-validation phases. This model's capability to handle seasonal variations and trends ensures reliable forecasts, making it an invaluable tool for stakeholders in the aviation sector in North Carolina. The integration of spatial analysis using the Google Maps API and R's Shiny app further enriched the visualization and interpretability of the data, enhancing the practical value of the research. Future research should explore the integration of additional variables that may impact air traffic, such as economic indicators, weather conditions, and socio-political events, to enhance the models' comprehensiveness and accuracy. Lastly, the development of a real-time data updating system in the forecasting model could adapt to rapidly changing air travel patterns, particularly relevant in post-pandemic recovery scenarios.