

## **M1-FA2.1: Chapter Project (Data Exploration and Summary)**

In this project, the researchers were tasked to explore and summarize airline flights using any appropriate algorithm that could predict airline price based on selected variables. Attributes such as *Airline*, *Source*, *Destination*, *Time of Departure and Arrival*, and *Date of Journey* were taken into consideration as to how they might affect the dependent variable *Price*. The results of the relationship between the independent and dependent variables were expressed through visualizations in RStudio.

### **I. Installing Packages and Dependencies**

To prepare the RStudio environment for the codes that will be run, the following packages and their dependencies were initially installed and unpacked.

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("psych")
install.packages("caret")
install.packages("tseries")
install.packages("forecast")

library(ggplot2)
library(dplyr)
library(psych)
library(caret)
library(tseries)
library(forecast)
```

As seen in the figure above, the *ggplot2* and *dplyr* packages from the previous chapter project were installed. These were the same packages used in the [Regression Model chapter project](#) for data visualizations and data manipulation. The new packages (i.e. *caret*, *tseries*, and *forecast*) that will be used in this assignment will later help implement the time series analysis model that will predict the airline price based on a specified variable. After the packages and libraries have finished installing, the console should look like the figure below.

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

  filter, lag

The following objects are masked from 'package:base':

  intersect, setdiff,
  setequal, union

> library(psych)

Attaching package: 'psych'

The following objects are masked from 'package:ggplot2':

  %+%, alpha

> library(caret)
Loading required package: lattice
> library(tseries)
Registered S3 method overwritten by 'quantmod':
  method      from
as.zoo.data.frame zoo

'tseries' version:
0.10-53

'tseries' is a package
for time series analysis
and computational
finance.

See
'library(help="tseries")'
for details.

> library(forecast)
This is forecast 8.21
Need help getting started? Try the online textbook FPP:
http://otexts.com/fpp2/
```

Then, to set a proper working directory for the program and its referenced file, the code snippet below was run.

```
# Set a working directory to store all the related datasets and files
setwd("C:/Users/Anton/Documents/3Q2223/CS174/Module 1/M1-FA2.1 Chapter Project (Data Exploration and Summary)")
```

Within the specified working directory, there exists the dataset that will be used in this project. Using the *read.csv()* function, the dataset was imported to the data frame variable *flights.df*.

```
# Import the Data_Train.csv dataset using the read.csv() function
flights.df <- read.csv("Data_Train.csv", stringsAsFactors = TRUE)
```

## II. Data Preparation

Before the data is explored to identify its correlations, the data was cleaned and prepared through format conversion, irrelevant and redundant data removal, and missing data identification. Since the dataset contained date values, those were converted to the appropriate date format using the **as.Date()** function.

```
> flights.df$Date_of_Journey = as.Date(flights.df$Date_of_Journey, "%d/%m/%y")
```

Then, to remove any redundant data and columns, the **subset()** function was run. The code snippet below shows the line of code that was run to remove the redundant data. One specific attribute was also set to NULL as its column did not contain any information that will help in fulfilling the objective of the assignment.

```
> flights.df = subset(flights.df, Airline != "Trujet" & Airline != "Vistara Premium economy" & Airline != "Jet Airways Business" & Airline != "Multiple carriers Premium economy")
> flights.df$Route <- NULL
```

To confirm that the data has had its specified values set to NULL, the *table()* function was run to reveal the following output.

```
> table(flights.df$Airline)
```

Air Asia	319	Air India	1752	GoAir	194	Indigo	2053	Jet Airways	3849
Jet Airways Business	0	Multiple carriers	1196	Multiple carriers Premium economy	0	SpiceJet	818	Trujet	0
Vistara	479	Vistara Premium economy	0						

Lastly, to identify if there are any missing values, the **colSums()** function was run. The figure below shows that the specified attributes do not have missing values.

```
> # Identify missing values
> colSums(is.na(flights.df))
```

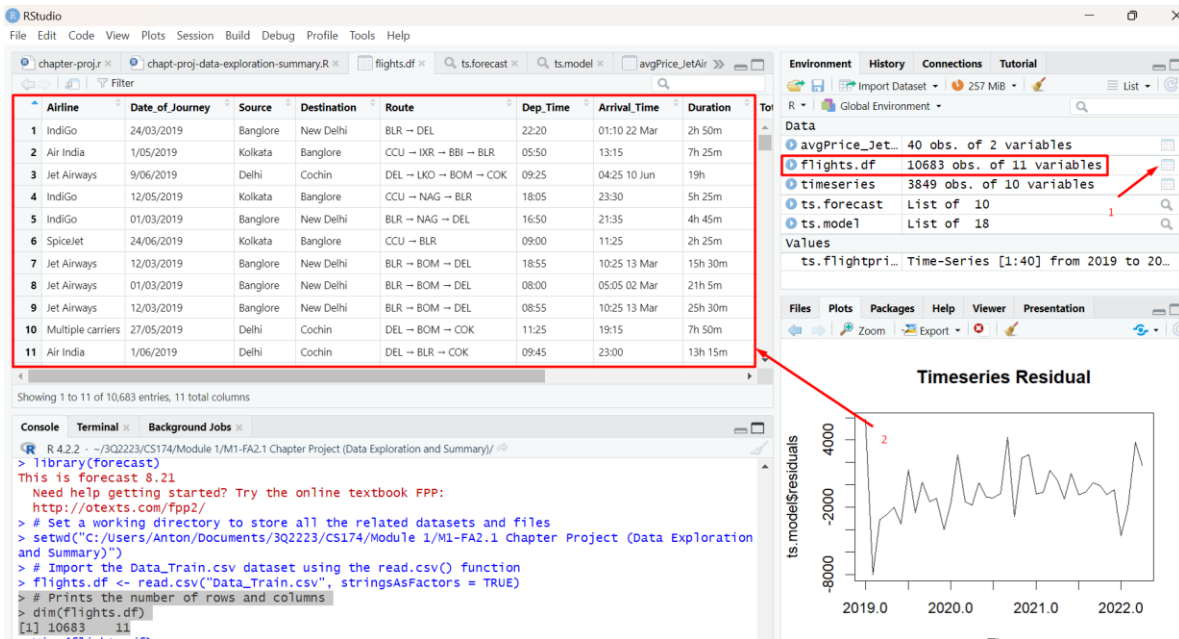
Airline	Date_of_Journey	Source	Destination	Dep_Time	Arrival_Time
0	0	0	0	0	0
Duration	Total_Stops	Additional_Info	Price		
0	0	0	0		

### III. Data Exploration and Summary

After the dataset had been imported and cleaned, the data was explored through manipulation, sorting, and printing specified values of the table onto the console. The first function that was used to explore the data was the **dim()** function, which displays the number of rows and columns of its parameter. Its output should look like the code snippet below.

```
> # Prints the number of rows and columns
> dim(flights.df)
[1] 10683 11
```

The function's output states that the dataset has **10,683 entries with 11 variables**. The *flights.df* data frame can also be viewed in RStudio on the Source column.



The next function used was the **head()** function. This function will print the first few lines of data on the console, allowing the members of the group to see the attribute names that they will use during further manipulating of the data.

```
> view(flights.df)
> # Prints the first few lines of the data on the console
> print(head(flights.df))
  Airline Date_of_Journey Source Destination Route Dep_Time Arrival_Time
1  IndiGo    24/03/2019  Banglore  New Delhi  BLR → DEL    22:20 01:10 22 Mar
2  Air India  1/05/2019  Kolkata  Banglore  CCU → IXR → BBI → BLR    05:50      13:15
3  Jet Airways  9/06/2019   Delhi  Cochin  DEL → LKO → BOM → COK    09:25 04:25 10 Jun
4  IndiGo    12/05/2019  Kolkata  Banglore  CCU → NAG → BLR    18:05      23:30
5  IndiGo    01/03/2019  Banglore  New Delhi  BLR → NAG → DEL    16:50      21:35
6  SpiceJet   24/06/2019  Kolkata  Banglore  CCU → BLR    09:00      11:25
  Duration Total_Stops Additional_Info Price
1  2h 50m    non-stop      No info  3897
2  7h 25m      2 stops      No info  7662
3  19h      2 stops      No info 13882
4  5h 25m      1 stop      No info  6218
5  4h 45m      1 stop      No info 13302
6  2h 25m    non-stop      No info  3873
```

Since the main objective of the assignment is to understand which variables affect the airline price, the data was sorted in ascending order in terms of the attribute *Price* to see what kind of relationship the dependent variable has with the other variables. To sort the flights, the **order()** function was used to sort the data from the flight of the lowest cost to that of the highest cost.

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
4067	SpiceJet	21/03/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:10 22 Mar	1h 25m	non-stop	No info	1759
4275	SpiceJet	27/03/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:10 28 Mar	1h 25m	non-stop	No info	1759
4840	SpiceJet	3/04/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 02 Apr	1h 30m	non-stop	No info	1759
10514	SpiceJet	27/03/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:05	1h 20m	non-stop	No info	1759
1514	Jet Airways	27/03/2019	Mumbai	Hyderabad	BOM → HYD	02:55	04:25	1h 30m	non-stop	In-flight meal not included	1840
229	SpiceJet	21/05/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965
388	SpiceJet	18/06/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
656	SpiceJet	3/05/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
1473	SpiceJet	21/05/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
1581	SpiceJet	24/06/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 25 Jun	1h 30m	non-stop	No check-in baggage included	1965
1653	SpiceJet	9/05/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965
1719	SpiceJet	24/05/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965
1993	SpiceJet	6/06/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 07 Jun	1h 30m	non-stop	No check-in baggage included	1965
2103	SpiceJet	27/06/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
2416	SpiceJet	18/05/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 19 May	1h 30m	non-stop	No check-in baggage included	1965
2427	SpiceJet	18/03/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:10 19 Mar	1h 25m	non-stop	No check-in baggage included	1965
2725	SpiceJet	21/05/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 22 May	1h 30m	non-stop	No check-in baggage included	1965
3506	SpiceJet	15/05/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 16 May	1h 30m	non-stop	No check-in baggage included	1965
3578	SpiceJet	1/04/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
3726	SpiceJet	27/05/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 28 May	1h 30m	non-stop	No check-in baggage included	1965
4535	SpiceJet	3/06/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
4596	SpiceJet	3/05/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965
5328	SpiceJet	3/04/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965
5743	SpiceJet	6/05/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965
5789	SpiceJet	9/05/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 10 May	1h 30m	non-stop	No check-in baggage included	1965
5941	SpiceJet	6/06/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
6143	SpiceJet	15/06/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
6688	SpiceJet	3/04/2019	Mumbai	Hyderabad	BOM → HYD	13:15	14:45	1h 30m	non-stop	No check-in baggage included	1965
7201	SpiceJet	1/05/2019	Mumbai	Hyderabad	BOM → HYD	22:45	00:15 02 May	1h 30m	non-stop	No check-in baggage included	1965
7293	SpiceJet	12/05/2019	Mumbai	Hyderabad	BOM → HYD	05:45	07:15	1h 30m	non-stop	No check-in baggage included	1965

Showing 1 to 31 of 10,683 entries, 11 total columns

As seen in the figure of the data frame above, the dataset has been reorganized according to price. Although the dataset is too saturated to be fully comprehended by the naked eye, one can immediately recognize a correlation between the price, airline, source, and destination. Unfortunately, looking at the dataset alone is not enough nor practical to understand the data. To enable a more comprehensive approach, the succeeding section will portray these relationships using data visualizations.

The figure below shows a summary of the *flights.df* data frame.

```
> summary(flights.df)
```

Airline	Date_of_Journey	Source	Destination	Dep_Time
Jet Airways :3849	Min. :2020-03-01	Bangalore:2191	Bangalore :2871	18:55 : 233
IndiGo :2053	1st Qu.:2020-03-27	Chennai : 380	Cochin :4522	17:00 : 227
Air India :1752	Median :2020-05-15	Delhi :4522	Delhi :1264	07:05 : 204
Multiple carriers:1196	Mean :2020-05-04	Kolkata :2871	Hyderabad: 696	10:00 : 203
SpiceJet : 818	3rd Qu.:2020-06-06	Mumbai : 696	Kolkata : 380	07:10 : 202
Vistara : 479	Max. :2020-06-27		New Delhi: 927	20:00 : 185
(Other) : 513				(Other):9406

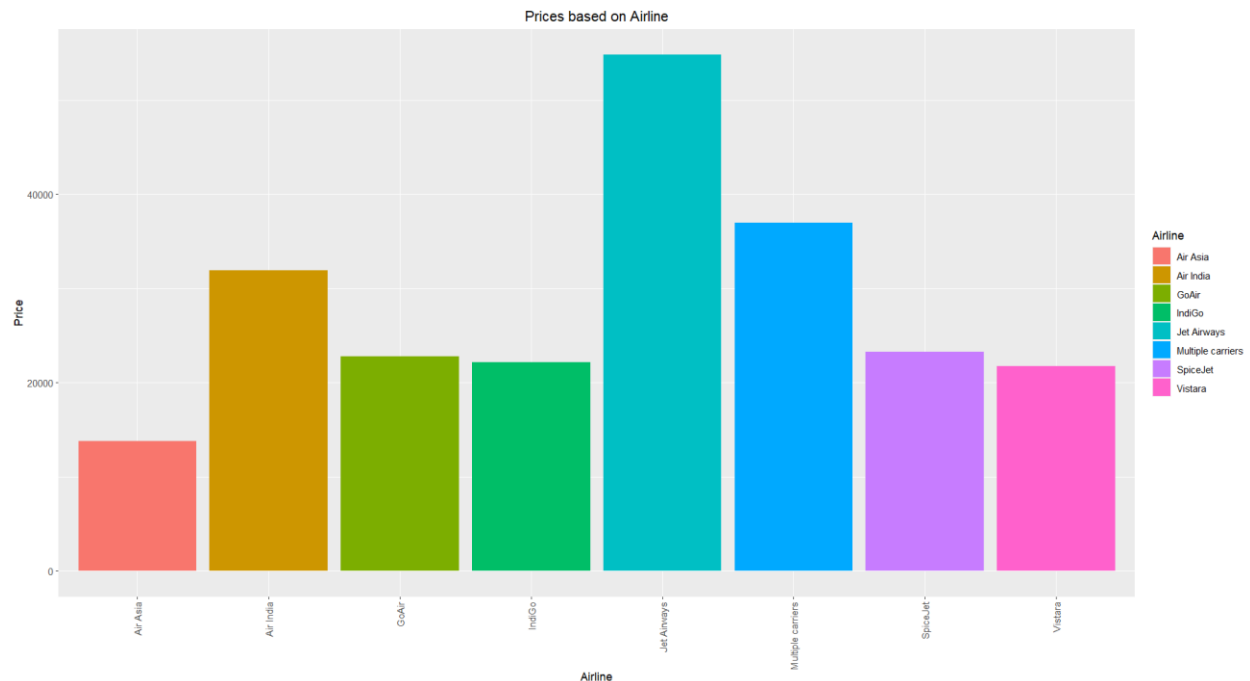
Arrival_Time	Duration	Total_Stops	Additional_Info
19:00 : 423	2h 50m : 549	: 1	No info :8325
21:00 : 357	1h 30m : 386	1 stop :5607	In-flight meal not included :1982
19:15 : 331	2h 45m : 337	2 stops :1518	No check-in baggage included: 320
16:10 : 154	2h 55m : 337	3 stops : 45	1 Long layover : 19
12:35 : 121	2h 35m : 328	4 stops : 1	Change airports : 7
20:45 : 112	3h : 261	non-stop:3488	No Info : 3
(Other):9162	(Other):8462		(Other) : 4

Price
Min. : 1759
1st Qu.: 5267
Median : 8372
Mean : 9057
3rd Qu.:12373
Max. : 54826

#### IV. Data Visualizations and Analysis

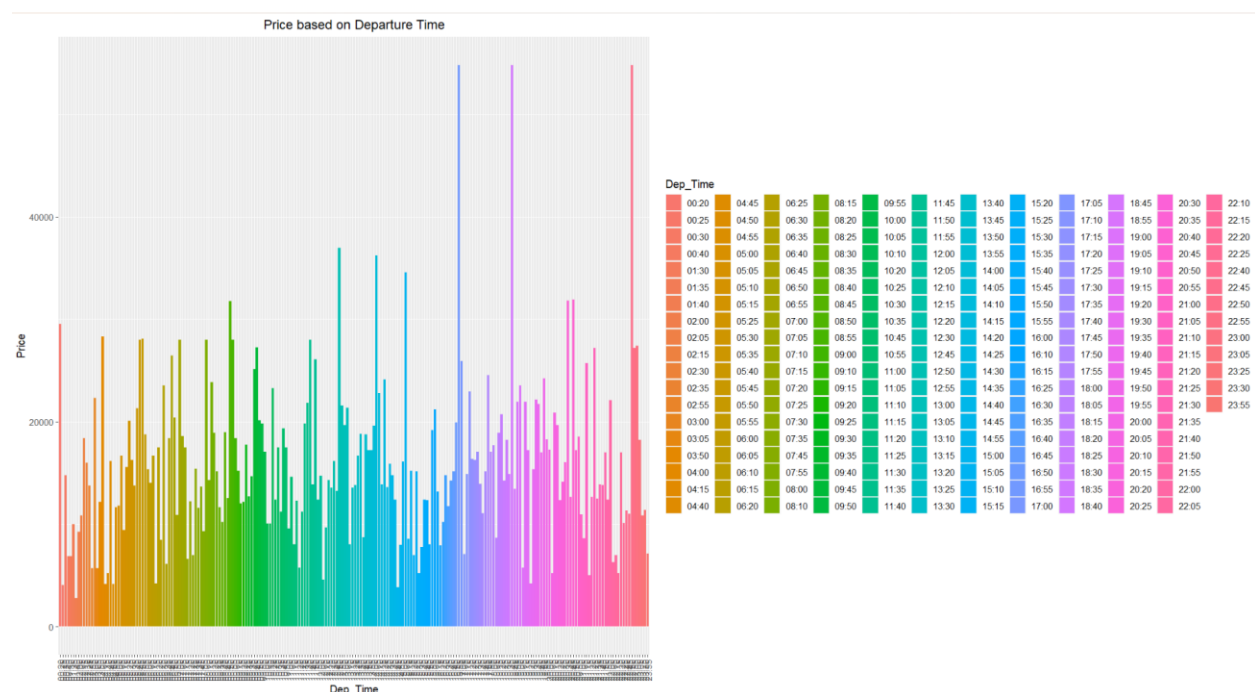
As mentioned in the previous section, correlations between the price and the airline, source, and destination were recognized by simply looking at the sorted dataset. The first correlation that was identified was the one between *Price* and *Airline*, whose relationship can be better understood through the figure below.



**Price based on Airline**

According to the *Prices based on Airline* bar graph, the Jet Airways airline has the highest flight cost with a price of nearly six thousand. On the other hand, the airline with the least flight cost is *Air Asia* with a price of less than two thousand. Since Jet Airways airline has the highest cost in flight tickets, its price data will be extracted to be grouped using the **group\_by()** function with the *Date\_of\_Journey* attribute to derive the average price of Jet Airways. The *Date\_of\_Journey* attribute will be set as the independent variable that will affect the dependent variable *Price* because it is the only other numerical variable available in the dataset that could be compared with the cost of the flight tickets. (The results of the correlation and forecast will be further discussed in the Time Series Analysis section.)

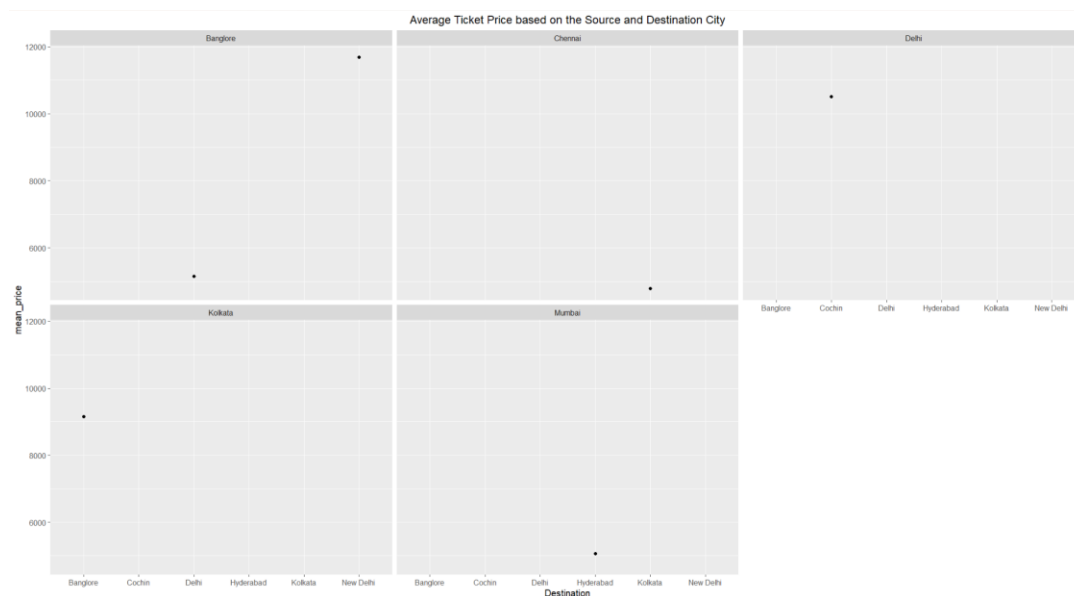
Other correlations were explored, as evidenced by the figures below, but were too saturated and unrelated to *Price*. Furthermore, attributes *Dep\_Time* and *Arrival\_Time* would be unreliable variables to consider given that any delays in flight schedule does not alter nor affect the price of a flight.



Another correlation observed was that between *Price* and *Source* and *Destination*. To understand the relationship between these values, the **mean\_price** in terms of *Source* and *Destination* was calculated. According to the table below, flights from Bangalore to New Delhi usually cost more, followed by flights from Delhi to Cochin.

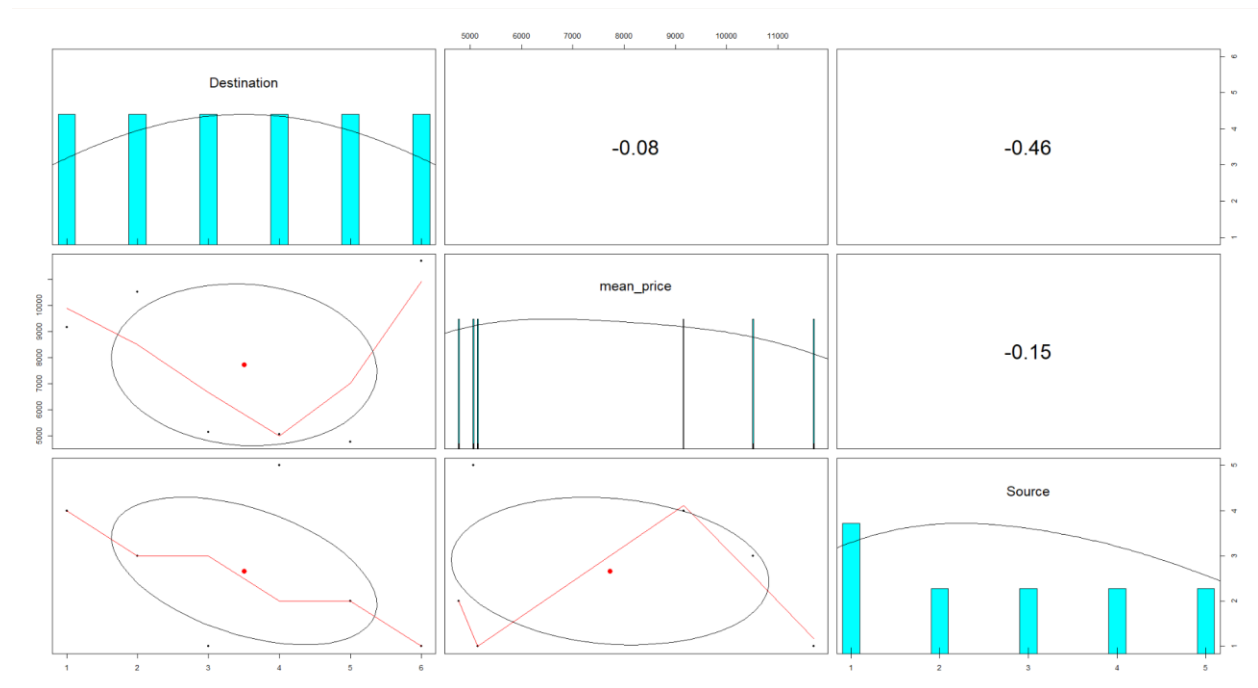
	Source	Destination	mean_price
1	Banglore	Delhi	5143.266
2	Banglore	New Delhi	11698.104
3	Chennai	Kolkata	4778.484
4	Delhi	Cochin	10519.729
5	Kolkata	Banglore	9158.389
6	Mumbai	Hyderabad	5061.030

**price\_comp**



**Average Ticket Price based on the Source and Destination City**

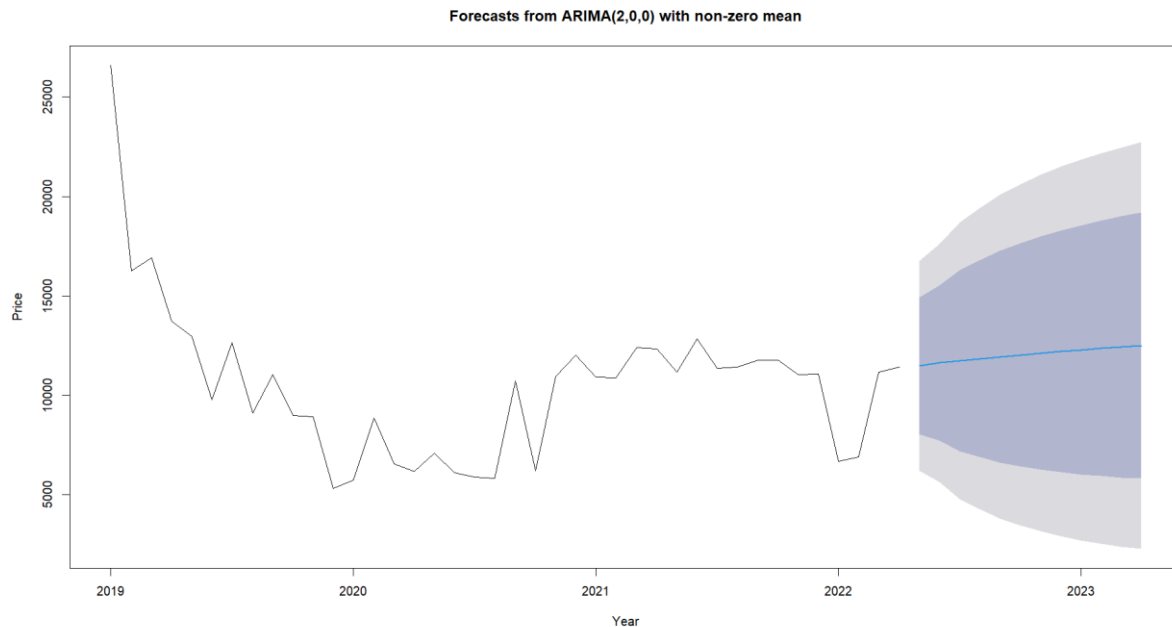
The correlation between the Destination, mean\_price, and Source can also be further understood by referring to the figure below.



**Correlation between Destination, mean\_price, and Source**



## V. Airline Price Prediction using Auto Arima Timeseries



We used time series using AUTO ARIMA to forecast the airline price for Jet Airways in 2023. We chose Jet Airways because it had the highest prices among other airlines. The above graph plot estimated forecasted values of airline price if it continues to be widespread this year. Based on the code snippet below, we predicted that the airline ticket price would be 12,502.64 in April 2023.

Forecasts:

	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
May 2022	11494.35	8042.651	14946.05	6215.433	16773.27
Jun 2022	11633.27	7723.252	15543.28	5653.415	17613.12
Jul 2022	11736.89	7191.049	16282.74	4784.624	18689.16
Aug 2022	11846.02	6900.503	16791.54	4282.504	19409.54
Sep 2022	11944.36	6632.871	17255.85	3821.140	20067.58
Oct 2022	12039.09	6434.783	17643.40	3468.041	20610.15
Nov 2022	12127.72	6270.007	17985.43	3169.122	21086.32
Dec 2022	12211.69	6138.963	18284.42	2924.254	21499.13
Jan 2023	12290.82	6032.070	18549.56	2718.890	21862.74
Feb 2023	12365.55	5945.893	18785.22	2547.529	22183.58
Mar 2023	12436.07	5876.284	18995.86	2403.743	22468.40
Apr 2023	12502.64	5820.533	19184.74	2283.240	22722.04

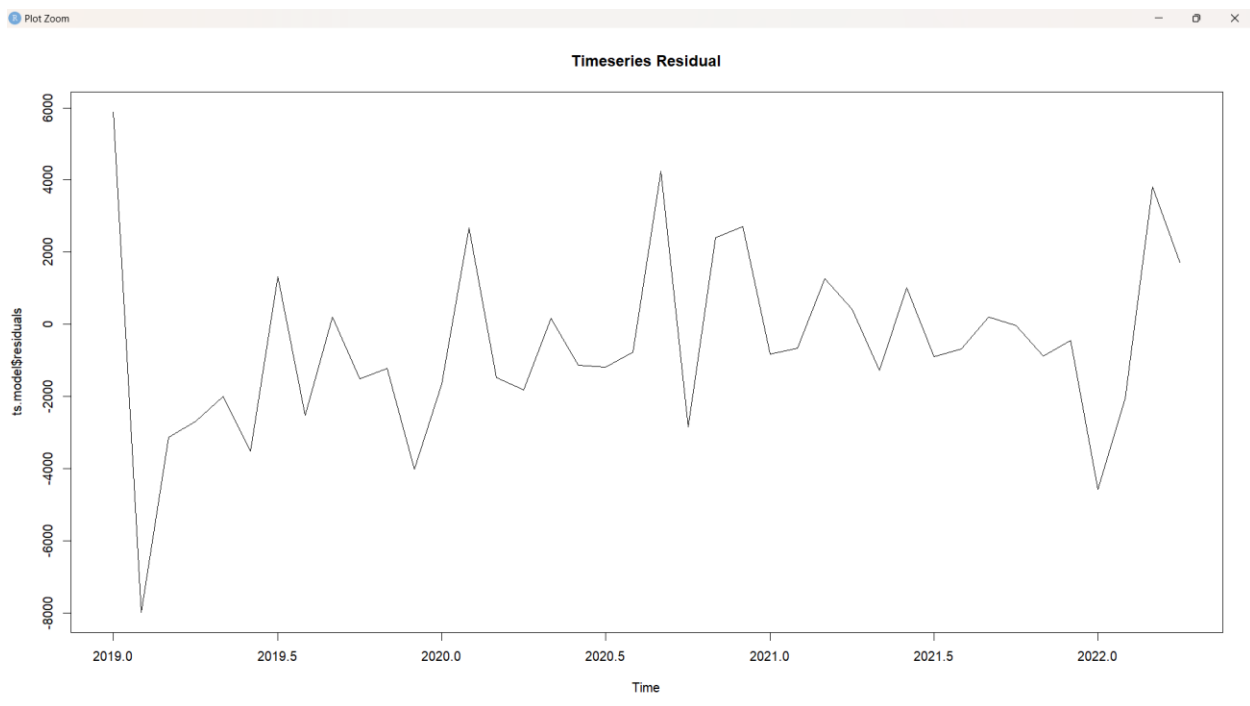
Next, we needed to ensure that our model is stationary, meaning it does not depend on time or trends. A stationary time series is required for better prediction of the airline price. To achieve this goal, we used Augmented Dickey-Fuller (ADF) to identify whether the time series is stationary. Based on the results below, our time series is stationary and expected since we have used AUTO ARIMA for our time series.

```
> #Hypothesis Testing using ADF  
> adf.test(ts.forecast$fitted, alternative = "stationary")
```

### Augmented Dickey-Fuller Test

```
data: ts.forecast$fitted  
Dickey-Fuller = -2.9616, Lag order = 3, p-value = 0.1961  
alternative hypothesis: stationary
```

Finally, we checked to see if the model accurately represented the information in the data. The model's residuals help determine whether our model possesses the fundamental characteristics. As shown in the graph below, the mean of the residuals is almost zero, and the residuals series exhibits no discernible correlation. Except for the one outlier shown in 2019, we can view the residual variance as constant because the residuals' time plot demonstrates that the residuals' variation is mainly constant across the historical data.



## References

- 3.3 *Residual Diagnostics / Forecasting: Principles and Practice* (2nd Ed).  
[otexts.com/fpp2/residuals.html](https://otexts.com/fpp2/residuals.html).
- N. (n.d.). *GitHub - notrichbish/airline-ticket-price-prediction: This project covers both Simple Linear Regression and Multiple Linear Regression which are used in prediction airline flight ticket. Moreover, Correlation analysis and Timeseries analysis are performed as well. This project is built as a fulfillment for my masters degree.* GitHub.  
<https://github.com/notrichbish/airline-ticket-price-prediction>
- Robert Kabacoff - [robk@statmethods.net](mailto:robk@statmethods.net). (n.d.). *Quick-R: Date Values*.  
<https://www.statmethods.net/input/dates.html>
- Sharda, R., Delen, D., & Turban, E. (2020). *Analytics, Data Science, and Artificial Intelligence: Systems for Decision Support, Global Edition*.
- “Why Does a Time Series Have to Be Stationary?” *Cross Validated*,  
[stats.stackexchange.com/questions/19715/why-does-a-time-series-have-to-be-stationary](https://stats.stackexchange.com/questions/19715/why-does-a-time-series-have-to-be-stationary).