Anomaly Detection and Outlier Analysis in the RITA Dataset:

A Case Study on 2019 Flight Performance

Introduction

In the realm of aviation, ensuring the punctuality and reliability of flight operations is crucial for both airlines and passengers. With millions of flights operating annually, even minor delays can cascade into significant disruptions, affecting everything from customer satisfaction to operational efficiency. Anomaly detection and outlier analysis play a vital role in identifying unusual patterns within flight data, allowing stakeholders to address inefficiencies and improve performance. This project aims to analyze the **RITA dataset (Reporting Carrier On-Time Performance)**, which provides comprehensive flight performance data spanning from 1987 to 2019. We will focus specifically on the 2019 dataset, represented by the file Flights1_2019_1.csv, to detect anomalies in flight operations. Anomalies in this context may include unexpected delays, irregularities in operational processes, or inconsistencies in recorded data. Identifying these anomalies is essential for enhancing decision-making, optimizing resource allocation, and gaining a deeper understanding of the factors influencing on-time performance. By applying exploratory data analysis alongside advanced anomaly detection algorithms, our goal is to uncover insights that can inform strategies for improving efficiency within the aviation industry. Through a systematic approach, this project will not only identify outliers but also validate the results to ensure their reliability. The methodologies used, the findings derived, and their implications for future flight operations and management will be discussed in detail throughout this report.

I. *Understanding and Analyzing the Dataset: Dictionary, Exploration, and Visualization*

1. *Data dictionary*

VARIABLES	TYPE	DESCRIPTION
YEAR	Integer	Represents the year of the flight. Since the data is only for 2019, it will always have the value of 2019.
DAY OF WEEK	Integer	Represents the day of the week (1 = Monday, ..., 7 = Sunday).
FL DATE	Date	Represents the flight date in the format *yyyymmdd*
ORIGIN_AIRPORT_ID	Integer	Unique identification number for the origin airport. (13487,13485....)
ORIGIN_AIRPORT_SEQ_ ID	Integer	An identification number assigned to identify a unique airport at the given point of time (1348702,1349505...)
ORIGIN_CITY_MARKET_ ID	Integer	An identification number for the city of departure (31650, 33495...)
ORIGIN_CITY_NAME	String	The name of the departure city (New Orléans, Minneapolis, Portland...)
DEST_AIRPORT_ID	Integer	The unique identification number for each destination's airport (12953, 12478...)
DEST_AIRPORT_SEQ_ID	Integer	An identification number assigned to identify a unique destination's airport at the given point of time (1295304, 1247805....)
DEST_CITY_MARKET_ID	Integer	An identification number for the city of destination (31703, 30198....)
DEST_CITY_NAME	String	The name of the destination's city (New York, Cincinnati...)
DEST_STATE_ABR	String	The abbreviations of destination states (KY, NY...)
DEP_DELAY	Integer	Difference in minutes between scheduled and actual departure time. Negative values indicate early departures. Example: -5 (departed 5 minutes early), 10 (departed 10 minutes late).

ARR_TIME	Integer	Actual arrival time in local time (Format: *hhmm*)
ARR_DELAY	Integer	Difference in minutes between scheduled and actual arrival time. Negative values indicate early arrivals. Example: 0 (on time), 25 (25 minutes late).
ARR_DELAY_NEW	Integer	Non-negative version of "ARR_DELAY". Early arrivals are set at 0. Example: 0 (on time or early), 30 (30 minutes late).
ARR_DEL15	Integer	Indicates if the arrival delay is 15 minutes or more. (Indicator: 1 = Yes, 0 = No)

2. Data exploration

Our dataset, *Flights1_2019_1.csv*, consists of 583,985 rows and 18 columns. It contains information about flights between various cities in the United States in 2019.

• *Head of data*

	YEAR	DAY_OF_WEEK	FL_DATE	ORIGIN_AIRPORT_ID	ORIGIN_AIRPORT_SEQ_ID	ORIGIN_CITY_MARKET_ID	ORIGIN_CITY_NAME	DEST_AIRPORT_ID	DEST_AIRPORT_SEQ_ID
0	2019	6	2019-01-19	13487	1348702	31650	Minneapolis, MN	11193	1119302
1	2019	7	2019-01-20	13487	1348702	31650	Minneapolis, MN	11193	1119302
2	2019	1	2019-01-21	13487	1348702	31650	Minneapolis, MN	11193	1119302
3	2019	2	2019-01-22	13487	1348702	31650	Minneapolis, MN	11193	1119302
4	2019	3	2019-01-23	13487	1348702	31650	Minneapolis, MN	11193	1119302

DEST_CITY_MARKET_ID	DEST_CITY_NAME	DEST_STATE_ABR	DEP_DELAY	ARR_TIME	ARR_DELAY	ARR_DELAY_NEW	ARR_DEL15
33105	Cincinnati, OH	KY	-10.0	18 :32	-25.0	0.0	0.0
33105	Cincinnati, OH	KY	-4.0	18 :25	-37.0	0.0	0.0
33105	Cincinnati, OH	KY	-9.0	18 :45	-17.0	0.0	0.0
33105	Cincinnati, OH	KY	-4.0	18 :39	-23.0	0.0	0.0
33105	Cincinnati, OH	KY	-6.0	18 :50	-12.0	0.0	0.0

When we observe this table, we can note that all variables are simple except two of them. These are **ARR_DELAY_NEW** and **ARR_DEL15. ARR_DEL15** has two categories: **1** if arrival delay is 15 minutes or more, **0** otherwise. Concerning **ARR_DELAY_NEW,** the values are 0 if arrival delay is negative, and arrival delay itself otherwise. Since we have a large database, we will try to calculate the null values, reduce data and choose the most important variables for our analysis.

• *Missing values*

Our dataset contains:

No missing data in the following columns: YEAR, DAY_OF_WEEK, FL_DATE, ORIGIN_AIRPORT_ID, ORIGIN_AIRPORT_SEQ_ID, ORIGIN_CITY_MARKET_ID, ORIGIN_CITY_NAME, DEST_AIRPORT_ID, DEST_AIRPORT_SEQ_ID, DEST_CITY_MARKET_ID, DEST_CITY_NAME, DEST_STATE_ABR.

Missing data in columns named **DEP_DELAY** with 16,355 missing values, **ARR_TIME** with 17,061 missing values and **ARR_DELAY ARR_DELAY_NEW**, **ARR_DEL15** with 18,022 missing values each.

The missing data primarily affects the columns related to flight delays and arrival times. For each of these variables, the null values represent only **3%** of data. This proportion is very small, not to say negligible. So, we decided to remove all null values to carry out our analysis. Initially at 583985, the number of observations is returned to **565963. We judged that the variables ARR_DELAY, DEP_DELAY, ORIGIN_AIRPORT_ID, DEST_AIRPORT_ID and ARR_TIME** are the most relevant therefore we will use them for the rest of our study.

• *Summary of the Data*

	YEAR	DAY_OF_WEEK	FL_DATE	DEP_DELAY	ARR_DELAY	ARR_DELAY_NEW	ARR_DEL15
count	583985.0	583985.000000	583985	567630.000000	565963.000000	565963.000000	565963.000000
mean	2019.0	3.835626	2019-01-15 23:02:31.604578816	9.766091	4.257506	13.654539	0.185917
min	2019.0	1.000000	2019-01-01 00:00:00	-47.000000	-85.000000	0.000000	0.000000
25%	2019.0	2.000000	2019-01-08 00:00:00	-6.000000	-16.000000	0.000000	0.000000
50%	2019.0	4.000000	2019-01-16 00:00:00	-3.000000	-7.000000	0.000000	0.000000
75%	2019.0	5.000000	2019-01-24 00:00:00	5.000000	7.000000	7.000000	0.000000
max	2019.0	7.000000	2019-01-31 00:00:00	1651.000000	1638.000000	1638.000000	1.000000
std	0.0	1.921899	NaN	48.626941	51.159511	47.488893	0.389040

When we analyze this summary, we can notice that flights are much delayed at departure than at arrival because the average of delays at departure (9.77 minutes) is higher than the average of delays on arrival (4.25 minutes)**.** We also noticed that approximately 18.59% of flights have an arrival delay of more than 15 minutes and 81.4% of flights have an arrival delay of less than 15 minutes. The highest value for DEP_DELAY and ARR_DELAY is very far from the average, suggesting the existence of one or more high aberrant values (outliers). Departure delays are often caused by factors such as runway congestion, logistical issues (boarding, refueling) or safety procedures. At busy airports, departures are more affected by slot management and flight sequences.

3. *Data visualization*

After this exploration, we had to naturally produce several visualizations with these variables to deepen our understanding of this database. On one hand, **Figure 1.1** suggests that when we analyze **the graph of arrival delays**, we note that the distribution is positively skewed. Most of the data is clustered around 0, suggesting that flights are likely arriving on time or with a slight delay. The long queue on the right indicates that some flights are experiencing significant delays (200 minutes). Moderate delays are common (0-50 minutes). On the other hand, the histogram of **distribution of departure delays** shows the majority of flights depart on time or with slight delays, as indicated by the peak around zero. There are also some flights departing early (negative values). The distribution is skewed, with a tail extending toward larger delays, suggesting that while rare, some flights experience significant delays (up to 150-200 minutes).

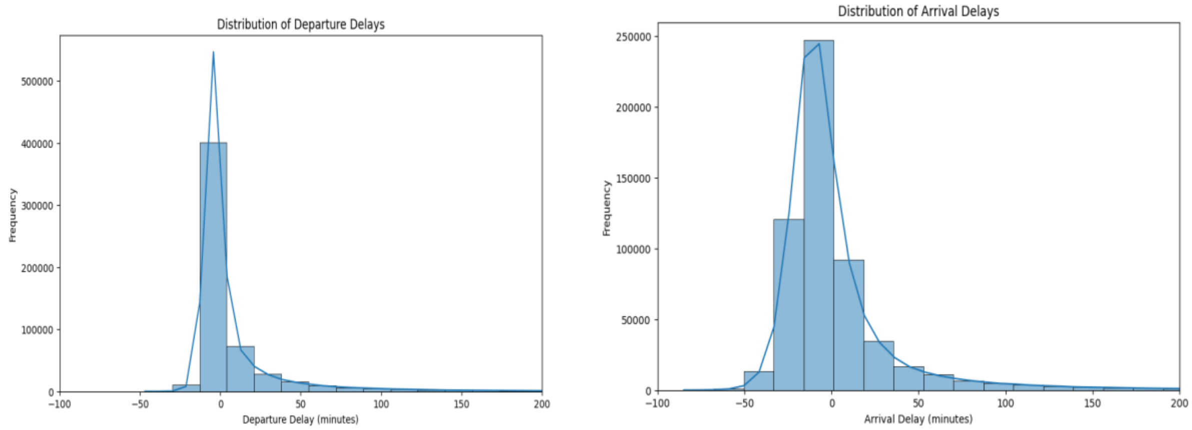


***Figure 1.1:* Distribution of Departure and Arrival Delays**

The graph in **Figure 1.2** below shows the evolution of the average departure delays throughout the month of January. Two notable peaks are observed. The first occurred on January 21, while the second took place on January 24. We will later analyze the causes of these outliers in the following sections of the report.

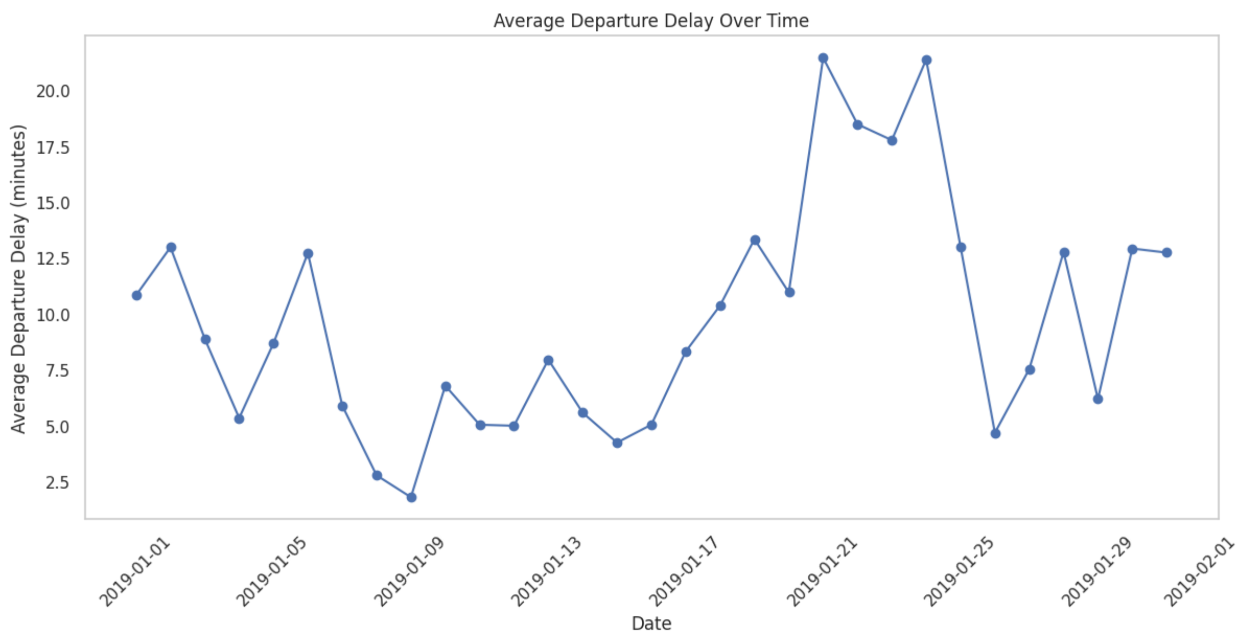
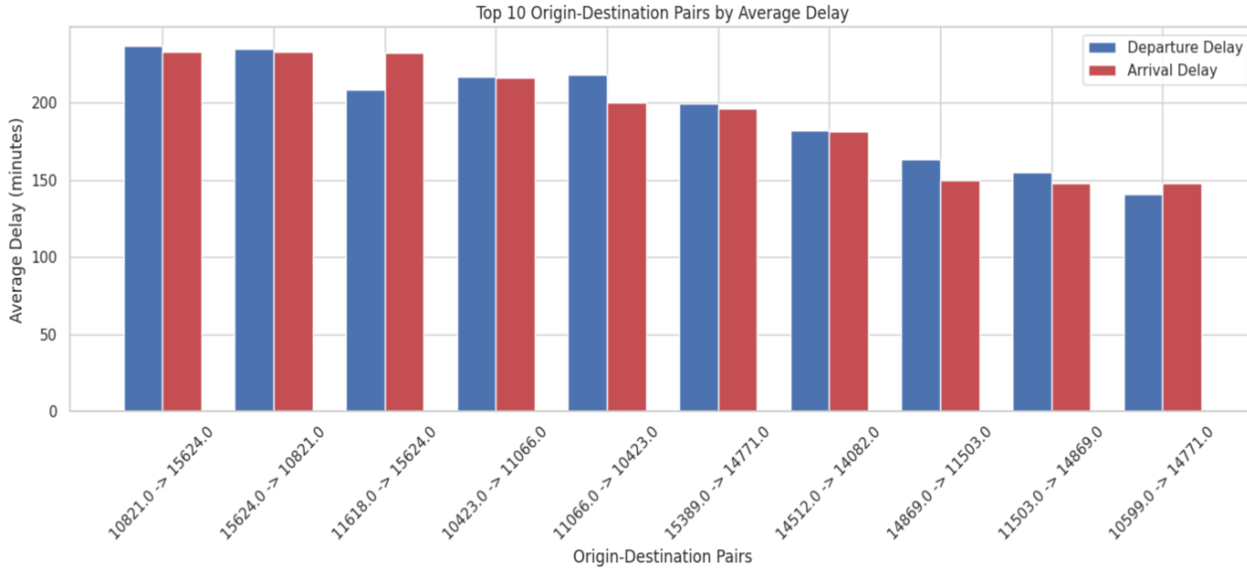


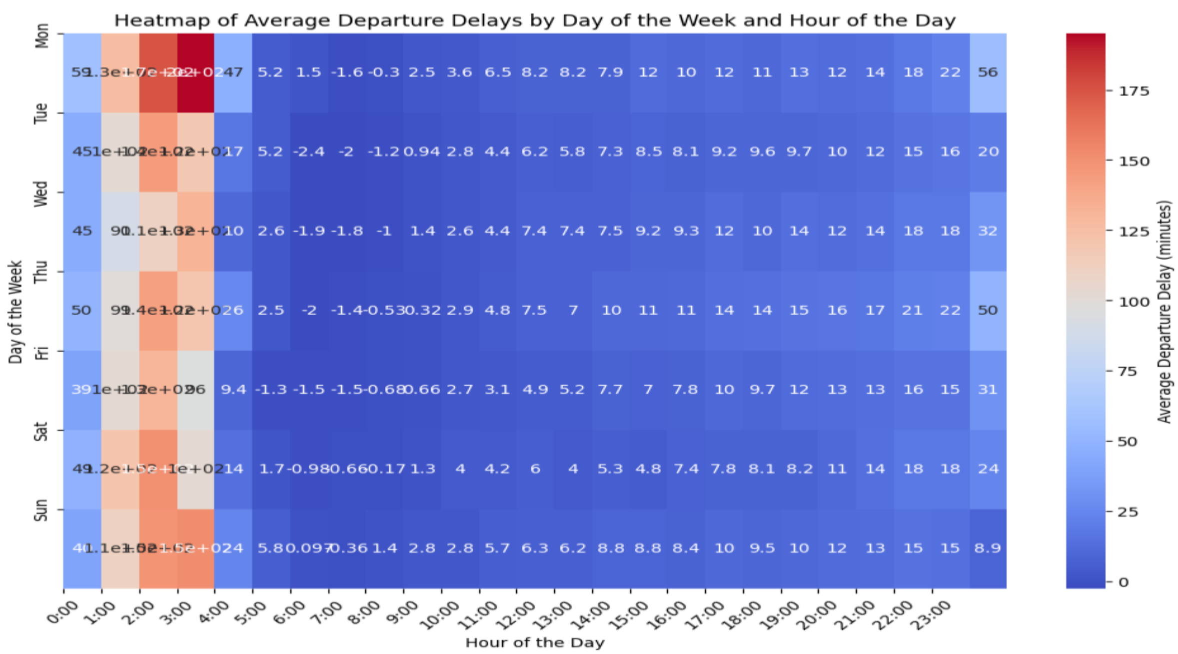
***Figure 1.2:* Trends in Average Departure Delays**

The following **Figure 1.3** represents the 10 flights with the highest delays (both departure and arrival) in the dataset. The data is grouped by Origin-Destination pairs and then aggregated by the average. It shows that the airport pair ID: 10821 (Origin) to ID: 15624 (Destination) recorded the highest average delays (Departure and Arrival).



<u>Figure 1.3:</u> Top 10 Origin-Destination Pairs by Average Delay

The next heatmap **Figure 1.4** shows that average departure delays are notably higher between 1 AM and 3 AM, particularly on Mondays. This indicates that flights departing during this time window, especially at the start of the week, experience more delays. This pattern may be attributed to factors such as the resumption of operations after the weekend, logistical constraints, or specific weather conditions.

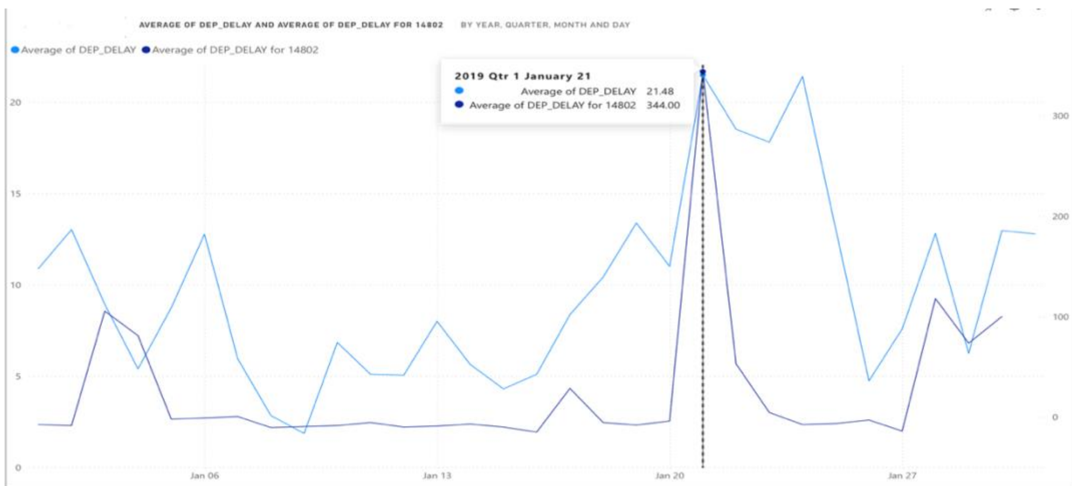


<u>Figure 1.4:</u> Heatmap of Average Departure delays by days of the week and Hour of the Day

II. <u>Anomalous Observations: Identification and Justification</u>

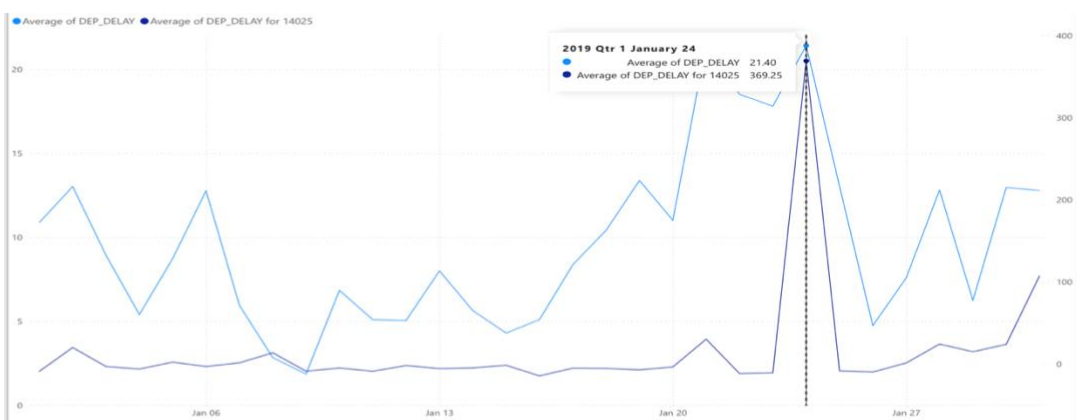
In this section, we begin by analyzing the causes of the peaks observed in the time series shown in Figure1.2.

- For the first peak on January 21, we plotted a time series comparing the evolution of average departure delays for airport **14802** with the overall average departure delays across all airports (Fig 2.1). On this date, we observed the following: Average DEP_DELAY: **21.48**, Average DEP_DELAY for **14802**: **344.00**. The results reveal that the average departure delays at airport **14802** were exceptionally high at certain times, which likely contributed to the spike in the overall average departure delays for all airports. We obtained similar results for the following airports: **12003**, **11898**, **15295**, and **14543**. The average departure delays for these airports were significantly higher compared to the overall average, which likely contributed to the increase in the overall departure delay average across all airports on this day.



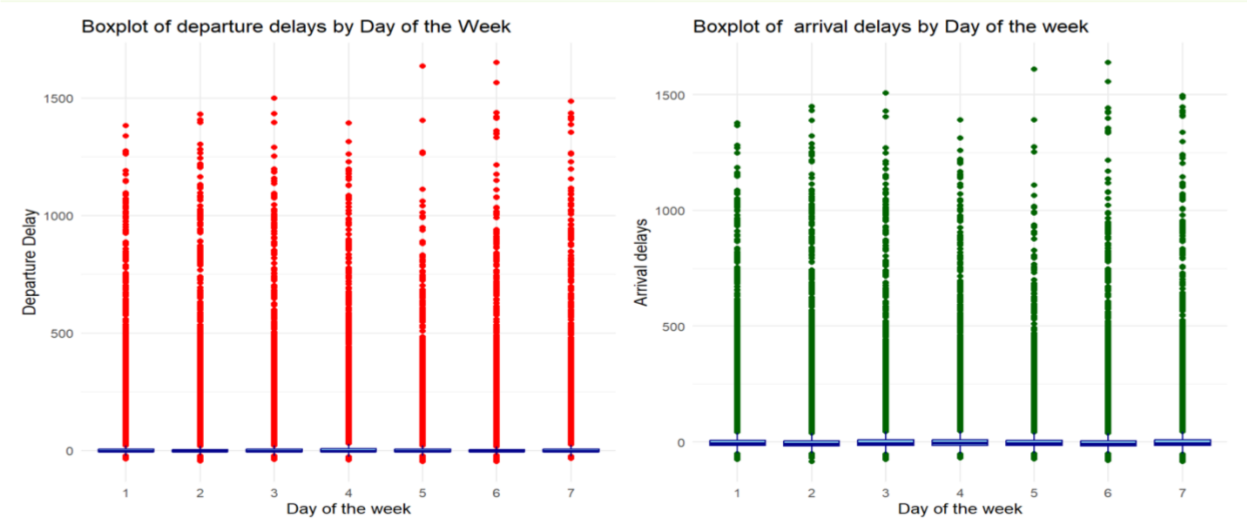
<u>Fig 2.1</u>: Time Series of Average Departure Delays for Airport 14802 and All Airports

• For the second peak on January 24, we performed the same analysis for airport **14025 (**Fig 2.2). The results were as follows: Average DEP_DELAY: **21.40**, Average DEP_DELAY for **14025**: **369.25.** Similarly to the first peak, the results show that the average departure delays at airport 14025 were exceptionally high at certain times, which likely contributed to the increase in the overall average departure delays for all airports.



<u>Fig 2.2</u>: Time Series of Average Departure Delays for Airport 14025 and All Airports

When analyzing the graphs in Figure 2.3, we observe that the distribution of average departure and arrival delays is not very normal. Both boxplots show many extreme values, particularly on Friday and Saturday. These are likely outliers, but we need to further analyze them using more precise tools to confirm this hypothesis. For now, we can consider them as anomalies.



<u>Fig2.3:</u> Boxplot of departure and arrival delays depending on the Day of Week.

III. Dimensionality Reduction

For the size reduction, we have grouped the average departure times and arrival times by origin airport and destination airport. We got 5532 points that correspond to pairs (airport of origin, airport of destination). We will use this new database **Origin_Destination_delay.csv** to detect anomalies.

• *Head of the new dataset:*

	ORIGIN_AIRPORT_ID	DEST_AIRPORT_ID	AVG_DEP_DELAY	AVG_ARR_DELAY	TOTAL_FLIGHTS
0	10135	10397	7.059701	3.970149	67
1	10135	11057	3.227848	1.721519	79
2	10135	11433	20.611765	14.400000	85
3	10135	11697	44.000000	35.750000	4
4	10135	13930	36.228571	28.857143	35

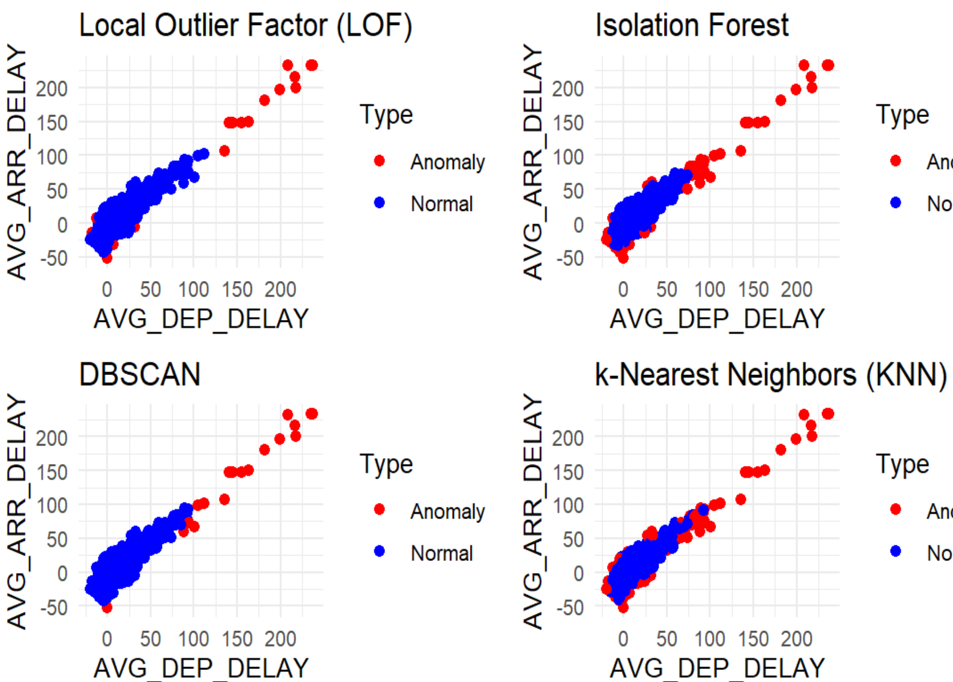
• *Summary of the new dataset:*

	ORIGIN_AIRPORT_ID	DEST_AIRPORT_ID	AVG_DEP_DELAY	AVG_ARR_DELAY	TOTAL_FLIGHTS
count	5532.000000	5532.000000	5532.000000	5532.000000	5532.000000
mean	12759.687816	12757.095083	10.268851	4.896444	102.307122
std	1542.515754	1541.933636	15.252626	16.881113	117.644996
min	10135.000000	10135.000000	-20.000000	-51.888889	1.000000
25%	11292.000000	11292.000000	2.279231	-4.031634	28.000000
50%	12889.000000	12889.000000	6.750000	1.453730	62.000000
75%	14107.000000	14107.000000	14.608696	10.219192	135.000000
max	16218.000000	16218.000000	237.000000	233.000000	1209.000000

The route with the highest number of flights in the summary of the dataset corresponds to a flight from **Los Angeles, CA (ID: 12892)** to **San Francisco, CA (ID: 14771)**, with a total of **1,209 flights**. The average departure delay for this route is approximately **21.93 minutes**, while the average arrival delay is about **17.89 minutes**.

IV. ***Anomaly Detection Techniques***

We will focus now on detecting anomalies. For this task, we are implementing four methods which are **DBSCAN**, **Isolation Forest**, **Local Outlier Factor**, and **KNN (k-Nearest Neighbors). These** methods, each providing a unique perspective, will help us to identify airports that have atypical behaviors. To obtain robust results, we set a strict criterion: "A trajectory will be considered as an anomaly only if it is identified as such by at least three of our four methods". This approach could make it possible to avoid false detections.



The tables below list the anomalies detected according to our criterion.

A tibble: 22 × 5

ORIGIN_AIRPORT_ID <dbl>	DEST_AIRPORT_ID <dbl>	AVG_DEP_DELAY <dbl>	AVG_ARR_DELAY <dbl>	Final_Anomaly <lgl>
14869	11503	163.00000	150.000000	TRUE
15041	13930	101.00000	67.000000	TRUE
15389	14771	199.14286	196.285714	TRUE
15624	10821	235.00000	233.000000	TRUE
15624	11618	144.50000	147.500000	TRUE

A tibble: 22 × 5

ORIGIN_AIRPORT_ID <dbl>	DEST_AIRPORT_ID <dbl>	AVG_DEP_DELAY <dbl>	AVG_ARR_DELAY <dbl>	Final_Anomaly <lgl>
10423	11066	217.00000	216.000000	TRUE
10599	14771	140.50000	148.000000	TRUE
10721	10800	31.44828	-5.444444	TRUE
10821	15624	237.00000	233.000000	TRUE
11042	14112	-18.25000	-13.750000	TRUE
11066	10423	218.00000	200.000000	TRUE
11447	12519	87.67857	59.000000	TRUE
11503	14869	155.00000	148.000000	TRUE
11618	15624	208.50000	232.500000	TRUE
11884	12892	135.75000	107.000000	TRUE
12264	14321	105.06897	98.586207	TRUE
12889	10408	0.00000	-51.888889	TRUE
13241	13930	111.67857	101.464286	TRUE
13830	14869	93.94737	72.684211	TRUE
14122	14679	7.00000	-31.000000	TRUE
14122	14747	-12.38710	6.967742	TRUE
14512	14082	181.80000	181.000000	TRUE

1-17 of 22 rows Previous 1 2 Next

V. ***Validation of the Results***

Each algorithm independently classified each data point as either normal or anomalous. Since the results of anomaly detection can vary between algorithms due to differences in their underlying methodologies, we adopted a majority voting approach to achieve a more robust and reliable classification. The majority voting approach combines the predictions of the four algorithms. For each data point, we counted the number of algorithms that classified it as an anomaly. If at least 3 out of 4 algorithms identified a point as an anomaly, it was classified as an anomaly under the majority vote. Otherwise, the point was classified as normal.

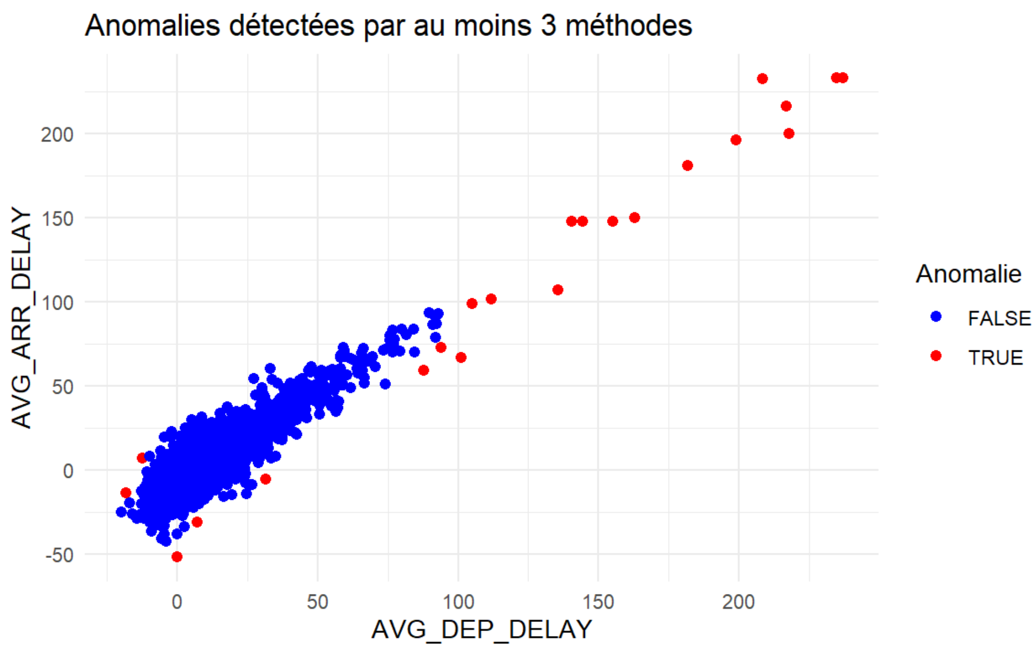


***Fig5.1:* Anomalies detected according to our criterion.**

To summarize, we had 22 trajectories which are considered anomalies**.** This can be explained by delays in departures or on arrival very often due to several factors. We can validate the results because an airport will be considered an anomaly if it's identified as such by three detection methods. However, we want to point out that the reduction of data could influence the results.

Conclusion

Anomaly detection is a complex study because of all the parameters it uses. In this analysis, we applied several anomaly detection algorithms to identify unusual observations in our dataset, which contains information about flight delays (**AVG_DEP_DELAY and AVG_ARR_DELAY**). The algorithms used are: **DBSCAN**, **ISOLATION FOREST, LOCAL OUTLIER FACTOR AND K-NEAREST NEIGHBOR.** Each algorithm used has its advantages and disadvantages, but the criterion used makes it possible to identify anomalies in a robust and reliable way. Data visualization and exploration allowed us to emphasize that delays in flight departures are recurrent, certainly because of the weather, the number of flights and several factors.

Appendix

- ## *Key code:*

- **Dimension Reduction by grouping by (Origin – Destination Airport)**

```r
library(dplyr)

# Regrouper les délais par aéroport d'origine et de destination
resultat <- flights_2019_clean %>%
  group_by(ORIGIN_AIRPORT_ID, DEST_AIRPORT_ID) %>%  # Remplace par les noms appropriés
  summarise(
    avg_departure_delay = mean(DEP_DELAY, na.rm = TRUE),  # Moyenne des délais de départ
    avg_arrival_delay = mean(ARR_DELAY, na.rm = TRUE),      # Moyenne des délais d'arrivée
    total_flights = n()                                   # Nombre total de vols
```

- **Anomaly detection Algorithm (Example of LOF)**

```r
# LOF (Local Outlier Factor)
lof_scores <- lofactor(cleaned_data[, c("AVG_DEP_DELAY", "AVG_ARR_DELAY")], k = 20)
cleaned_data$LOF_ANOMALY <- lof_scores > quantile(lof_scores, 0.997)  # TRUE pour les anomalies

# Isolation Forest
iso_forest <- isolation.forest(cleaned_data[, c("AVG_DEP_DELAY", "AVG_ARR_DELAY")], ntrees = 100)
iso_scores <- predict(iso_forest, cleaned_data[, c("AVG_DEP_DELAY", "AVG_ARR_DELAY")], type = "score")
cleaned_data$ISO_ANOMALY <- iso_scores > quantile(iso_scores, 0.991)  # TRUE pour les anomalies

```

- **Boxplot**

```r
# Diagramme à moustaches du retard au depart
plot1<-ggplot(flights_2019_clean, aes(x = as.factor(DAY_OF_WEEK), y = DEP_DELAY)) +
  geom_boxplot(fill = "skyblue", color = "darkblue", outlier.color = "red", outlier.shape = 16) +
  labs(title = "Boxplot of departure delays by Day of the Week",
       x = "Day of the week",
       y = "Departure Delay") +
  theme_minimal()
# Diagramms à moustaches pour le retard d'arrivée
plot2<-ggplot(flights_2019_clean, aes(x = as.factor(DAY_OF_WEEK), y = ARR_DELAY)) +
  geom_boxplot(fill = "skyblue", color = "darkblue", outlier.color = "darkgreen", outlier.shape = 16) +
  labs(title = "Boxplot of  arrival delays by Day of the week ",
433:1   Chunk 10 ÷                                                R Markdown ÷
```

- ## *Contributions:*

  - **Mor Fall SYLLA**: Introduction, Data exploration, Data visualization, Anomalous Observations: Identification and Justification, Dimensionality Reduction, Anomaly Detection Techniques (DBSCAN), Appendix.
  - **Arnauld Yannick BOYARM:** Conclusion, Data Dictionary, Data visualization, Isolation Forest algorithm, Appendix
  - **Faizatou DEME**: Layout, Data visualization, KNN algorithm and LOF algorithm