IÉSEG SCHOOL OF MANAGEMENT

New York City Flights Delay Analysis (2013)

Business Reporting Tools Group Assignment

Members

Arias Toribio Brigth
Diaz Gonzalez Manjarez Fernando
Karthikeyan Arrunkkumar

Lille, France, 2022

TECHNICAL REPORT

This technical report aims to outline the factors and reasons for NYC Flight delays in 2013 using the dataset obtained from R 'nycflights13' package, which contains 5 separate datasets (flights, airlines, airports, planes and weather). There are different reasons for flight delays - controllable (e.g. the airline used) and uncontrollable (e.g. weather), etc.. We have analysed the different reasons of flight delays using the existing data's / by joining different tables using SQL code in SQLite Studio and export the table results as an .CSV file / adding additional data's from external sources. Finally, we have used all these data's in Tableau for creating the story and visualization.

To organize the data from the different sources and to use it to graph it in tableau we have created the following additional tables:

- `Airplane_Route_types`
- `Arrival_Delays_all_airports`
- `Delays_airline_airport`
- `Delays_all_airports`
- `Flight_delays_avg_per_airline_size`
- `Flight_delays_per_airline`
- `Flight_delays_per_airline_size`
- `Flight_delays_per_airport`
- `Flight_delays_per_month`
- `Max_per_Month`
- `No_of_engines`
- `No_of_flights_origin`
- `No_of_flights_per_airline`
- `No_of_seats`
- `Summary_delays`
- `Table Dealys`
- `DelayClean`
- `Flights`

We have created these tables using the code **INNER JOIN…ON** and selecting the variables that we needed to explore. We have also assigned a name to each table and saved them under the library "flights" using this piece of code:

```
create table NameOfTable AS
```

To create some of these joins we had to create temporal tables that helped us making the query fast. Below are the SQL codes we have used to create the above mentioned tables.

```
/* Airplane_Route_types:*/

CREATE TABLE Airplane_Route_types AS
SELECT
CASE WHEN f.distance <= 1500 THEN 'Short-Haul Flights'
     WHEN f.distance > 1500 AND f.distance <= 2200 THEN
'Medium-Haul Flights'
     ELSE 'Long-Haul Flights' END AS Plane_types,
COUNT(f.flight) AS Total_flights,
SUM(f.dep_delay >0 OR f.arr_delay >0) AS Delayed_flights,
ROUND(SUM(f.dep_delay>0 OR f.arr_delay>0)*1.0 /COUNT(f.flight),2) AS
Perc_delayed_flights
FROM flights AS f
GROUP BY 1
ORDER BY 4 DESC;


/* Arrival_Delays_all_airports:*/

CREATE TABLE Arrival_Delays_all_airports AS
SELECT a.name AS airport_name, a.alt, a.lon, SUM(f.arr_delay) AS Delay
FROM flights AS f, Airports AS a
GROUP BY 1,2;

/* Delays_airline_airport:*/

CREATE TABLE delays_airline_airport AS
SELECT a.origin, a.dest, b.name AS Name, b.carrier,
SUM(a.dep_delay>0) AS No_of_Delayed_Flights,
ROUND(SUM(a.dep_delay>0)*1.0/COUNT(*),2)AS Percent_of_dep_delayed_flights,
COUNT(*) AS total_no_of_flights
FROM flights AS a, Airlines AS b
WHERE a.carrier = b.carrier
GROUP BY 1,2,3
ORDER BY 1,3 DESC;

/* Flight_delays_avg_per_airline_size:*/

CREATE TABLE flight_delays_avg_per_airline_size AS
SELECT
CASE WHEN p.engine in ('Turbo-fan','Turbo-jet') THEN 'Big Flights'
     ELSE 'Small Flights' END AS Engine, a.name AS Airline, a.carrier,
p.manufacturer, ROUND(AVG(f.dep_delay) , 0.01) AS Avg_dep_delayed_flights
FROM Airlines AS a, flights AS f, planes AS p
WHERE a.carrier = f.carrier AND
```

```
      p.tailnum = f.tailnum
GROUP BY 1,2,3
ORDER BY 1,4 DESC;

/* Flight_delays_per_airline:*/

CREATE TABLE flight_delays_per_airline AS
SELECT
CASE WHEN p.engine in ('Turbo-fan','Turbo-jet') THEN 'Big Flights'
     ELSE 'Small Flights' END AS Engine, a.name AS Airline,
f.origin AS Airports, a.carrier, p.manufacturer,
ROUND(SUM(f.dep_delay>0)*1.0/COUNT(*),2)AS Percent_of_dep_delayed_flights,
ROUND(AVG(f.dep_delay),0.01) AS Avg_dep_delays
FROM Airlines AS a, flights AS f, planes AS p
WHERE a.carrier = f.carrier AND
      f.tailnum = p.tailnum
GROUP BY 1,2,3,4
ORDER BY 1,2;

/* Flight_delays_per_airline_size:*/

CREATE TABLE flight_delays_per_airline_size AS
SELECT
CASE WHEN p.engine in ('Turbo-fan','Turbo-jet') THEN 'Big Flights'
     ELSE 'Small Flights' END AS Engine, a.name AS Airline, a.carrier,
p.manufacturer,
ROUND(SUM(f.dep_delay>0)*1.0/COUNT(*),2)AS Percent_of_dep_delayed_flights,
COUNT(*) AS total_no_of_flights
FROM Airlines AS a, flights AS f, planes AS p
WHERE a.carrier = f.carrier AND
      p.tailnum = f.tailnum
GROUP BY 1,2,3
ORDER BY 1,3 DESC;

/* Flight_delays_per_airport:*/

CREATE TABLE flight_delays_per_airport AS
SELECT
CASE WHEN p.engine in ('Turbo-fan','Turbo-jet') THEN 'Big Flights'
     ELSE 'Small Flights' END AS Engine, a.name AS Airports_Name, f.origin
AS Airports, p.manufacturer, ROUND(SUM(f.dep_delay>0)*1.0 / COUNT(*),2) AS
Percent_of_dep_delayed_flights,
ROUND(AVG(f.dep_delay),0.01) AS Avg_dep_delays
FROM Airports AS a, flights AS f, planes AS p
WHERE a.faa = f.origin AND
      f.tailnum = p.tailnum
GROUP BY 1,2,3
ORDER BY 1,2;
```

```
/* Flight_delays_per_month:*/


CREATE TABLE flight_delays_per_month AS
SELECT month AS Month, origin AS Origin, COUNT(flight) AS No_of_flights,
ROUND(AVG(dep_delay),2) AS Average_Delays
FROM flights
GROUP BY 1;

/* Max_per_Month:*/

CREATE TABLE Max_per_month AS
SELECT a.month, a.day, a.Avg_dep_delay
FROM (  SELECT Month, Day, ROUND(AVG(dep_delay),2) AS Avg_dep_delay
        FROM flights
        GROUP BY 1,2) AS a
GROUP BY month
HAVING a.Avg_dep_delay = MAX(a.Avg_dep_delay);

/* No_of_engines:*/

CREATE TABLE No_of_engines AS
SELECT
CASE WHEN (p.engines=1) THEN 'One Engine'
     WHEN (p.engines=2) THEN 'Two Engines'
     WHEN (p.engines=3) THEN 'Three Engines'
     ELSE   'Four   Engines'   END)   AS   No_of_engines,   p.manufacturer,
COUNT(f.flight) AS Total_flights,
SUM(f.dep_delay>0      OR      f.arr_delay>0)      AS      Delayed_flights,
ROUND(SUM(f.dep_delay>0   OR   f.arr_delay>0)*1.0   /COUNT(f.flight),2)   AS
Perc_delayed_flights
FROM flights AS f, planes AS p
WHERE f.tailnum = p.tailnum
GROUP BY 1
ORDER BY 5 DESC;

/* No_of_flights_origin:*/

CREATE TABLE no_of_flights_origin AS
SELECT origin, SUM(dep_delay>0) AS no_of_delay_dep_flights, COUNT(origin)
AS total_flights,
ROUND(SUM(dep_delay>0)*1.0/COUNT(origin),2) AS Perc_of_delays_origin
FROM flights
GROUP BY 1
```

```
ORDER BY 3 DESC;


/* No_of_flights_per_airline:*/

CREATE TABLE no_of_flights_per_airline AS
SELECT a.name AS Airline, a.carrier, COUNT(b.flight) AS Total_no_of_flights,
SUM(b.dep_delay>0 OR b.arr_delay>0) AS Delayed_flights,
SUM(b.dep_delay<=0 AND b.arr_delay<=0) AS Ontime_flights
FROM Airlines AS a, flights AS b
WHERE a.carrier=b.carrier
GROUP BY 1,2
ORDER BY 1 DESC, 2 DESC, 3 DESC;


/* No_of_seats:*/

CREATE TABLE No_of_seats AS
SELECT
CASE WHEN (p.seats<=50) THEN '2-50 Seats'
     WHEN (p.seats<=150) THEN '51-150 Seats'
     WHEN (p.seats<=250) THEN '151-250 Seats'
     WHEN (p.seats<=350) THEN '251-350 Seats'
     ELSE '351-450 Seats' END) AS no_of_seats,
COUNT(f.flight) AS Total_flights, SUM(f.dep_delay>0 OR f.arr_delay>0) AS
Delayed_flights,
ROUND(SUM(f.dep_delay>0 OR f.arr_delay>0)*1.0 /COUNT(f.flight),2) AS
Perc_delayed_flights
FROM flights AS f, planes AS p
WHERE f.tailnum = p.tailnum
GROUP BY 1
ORDER BY 4 DESC;

/*Summary_delays:*/

CREATE TABLE Summary_delays AS
SELECT a.carrier, a.name, f.origin, f.month, f.day, f.hour, COUNT(f.flight)
AS sum_of_flights, SUM(dep_delay) AS total_dep_delay, AVG(dep_delay) AS
Avg_dep_delay
FROM flights AS f, Airlines AS a
WHERE a.carrier = f.carrier
GROUP BY 1,2,3,4,5,6;

/*Union all the tables of each month of 2013, as well as filtering only the
origin airports "EWR", "JFK" and "LGA":*/
CREATE TABLE DELAYS AS

SELECT *
```

```
FROM January
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM February
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM March
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM April
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM May
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM June
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM July
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM August
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
```

```
FROM September
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM October
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM November
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA"

UNION

SELECT *
FROM December
WHERE Origin="EWR" OR Origin="JFK" OR Origin="LGA";


/*Filter the columns that we will need to join the database to the flights table:*/

CREATE TABLE Delayclean AS
SELECT Month, Dayofmonth, Flight_Number_Reporting_Airline, origin, dest, Tail_Number,cancelled,diverted,carrierdelay,weatherdelay,NasDelay,Security Delay,LateAircraftDelay
FROM DELAYS;

/*Flights:*/

CREATE TABLE Flights AS
SELECT year, month, day, dep_time, sched_dep_time, dep_delay, arr_time, sched_arr_time, arr_delay, carrier, flight, tailnum, origin, dest, air_time, distance, hour, minute, time_hour,(strftime('%d', flights.time_hour) || "/" || strftime('%m', flights.time_hour) || "/" || strftime('%Y', flights.time_hour)) AS DATE
FROM flights;
```

We have created a complete story with all these data's using Tableau visualization and the complete story has been shared in Tableau Public. The link to access the Tableau Public is as below,

Group Assignment - Flight Delay Analysis | Tableau Public