Source: https://www.tiagoholidays.com/assets/images/service-banner.jpg

airbnb - Bookings & Cancellations Data analysis

June 2019

Francihelena Uzcategui

Table of Contents

- Introduction
- Data Wrangling
- Data Cleaning and Exploratory Data Analysis
- Limitations
- References

Introduction

The AIRBNB datasets have univariate and bivariate data to analyze — relations between entities and cancellations, entities and profits, and root cancellation and country.

Data wrangling

The data used for the analysis is from AIRBNB. There are four tables, each of them with information about bookings, cancellations, listing, and entities (100MB).

This report has modifications from the original queries to avoid plagiarism.

Data cleaning and Exploratory Data Analysis

There are no critical amount of null values. There are several columns with ID information, and it works as primary keys; to avoid duplicates, we used the statement DISTINCT(), HAVING COUNT(), and JOINS.

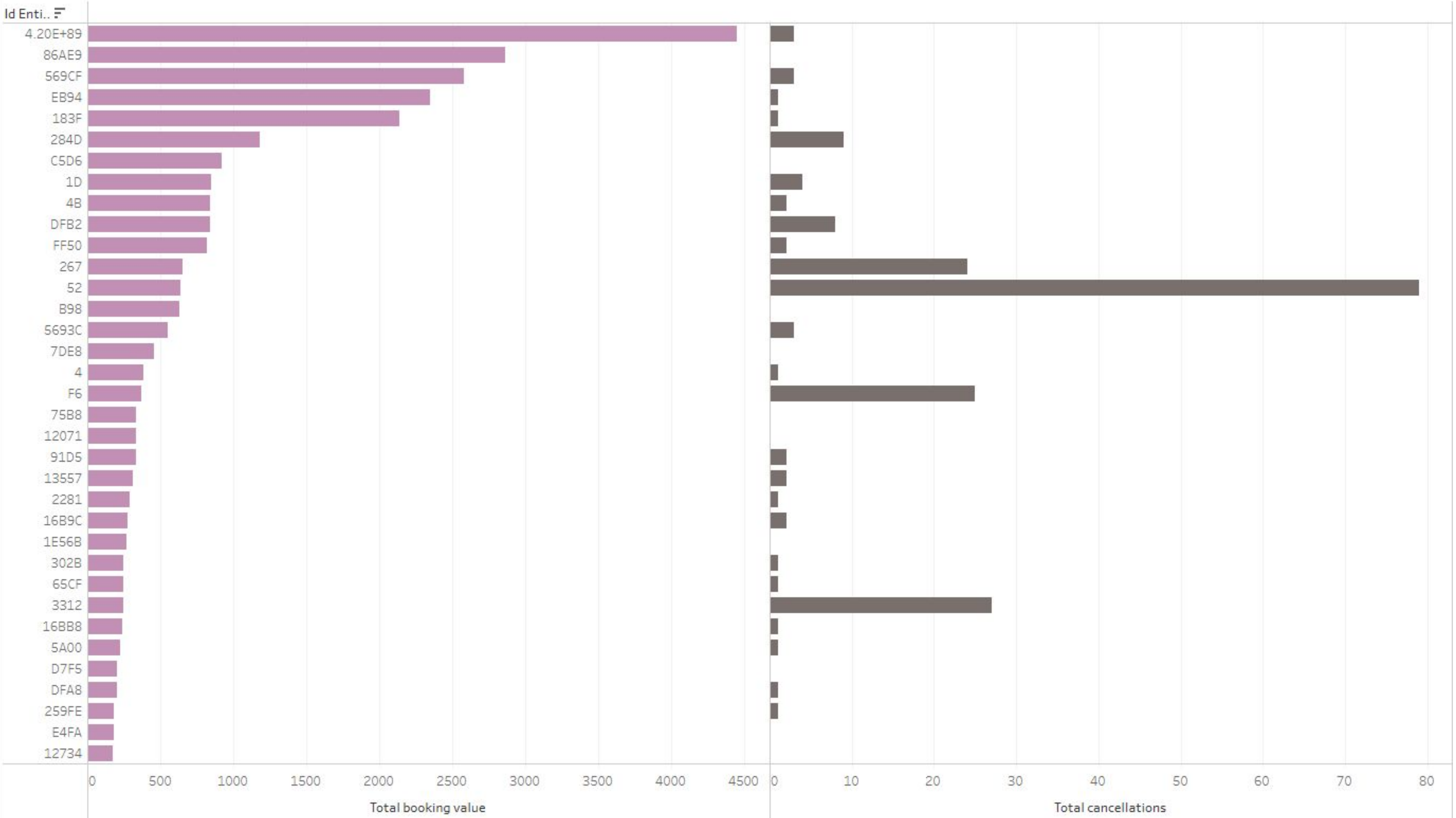
We use SQLite to develop the queries, access to the tables, and compiled the target information.

Identify the entities with at least 50 bookings in 2018 based on total bookings and total cancellations.

The pair of graphs show all the entities with at least 50 bookings. The bar graph on the left depicts all these entities by the total booking values; it means the profits. Contrary, the chart on the right side shows the same properties by the full cancellations.

The properties with at least 50 bookings have high total booking values and highest-demand, and it is inversely proportional to cancellations.

Entities with at least 50 bookings: comparison between Total booking value & Total cancellations



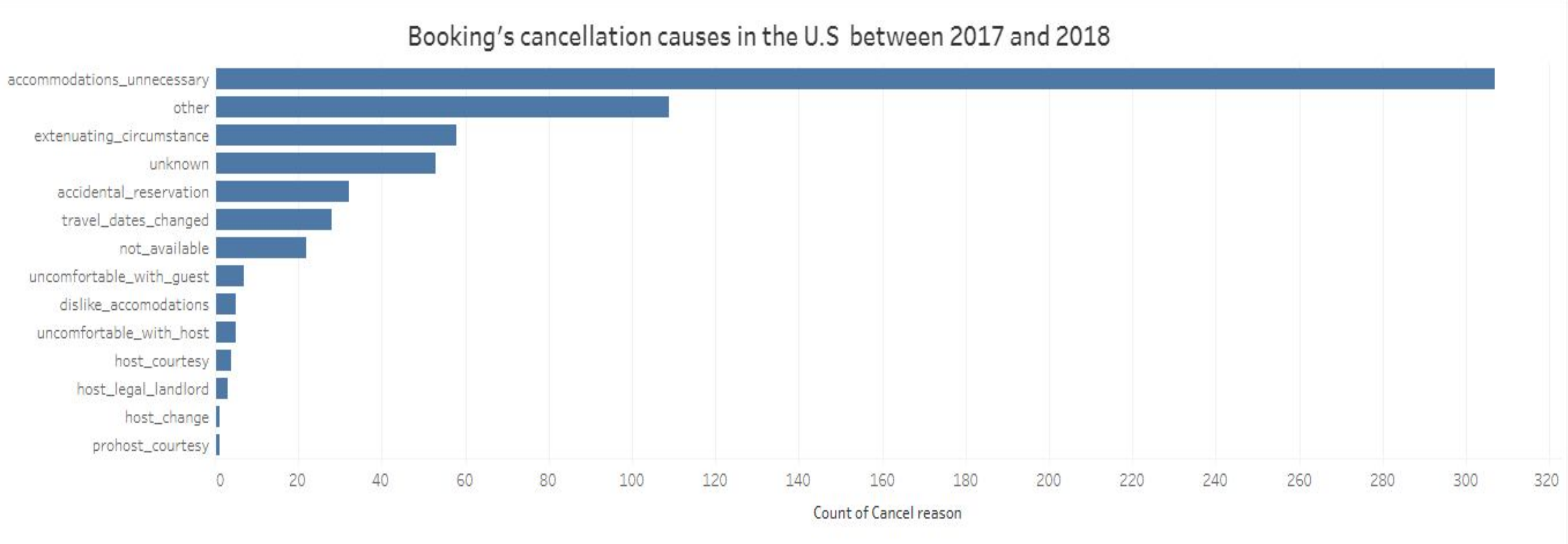
1.- SQL query

```
SELECT
      fct_bookings.field1 AS id_entity,
      fct_bookings.field15 AS Year2018,
      --fct_bookings.field2 AS id_reservation,
      CAST(COUNT(fct_bookings.field2) AS INT) AS Total_bookings,
      --SUM(fct_bookings.field7) AS m_nights_booked, -- it works for each entity
      fct_bookings.field9 AS Total_booking_value,
      --fct_cancellations.field1 AS id_reservation,
      COUNT(fct_cancellations.field1) AS Total_cancellations,
      ROUND(COUNT(fct_cancellations.field1),2)/ ROUND(COUNT(fct_bookings.field2),2) AS Cancellation_rates,
      SUM(fct_bookings.field9) / SUM(fct_bookings.field7) AS Price_per_night,
      --(COUNT(fct_bookings.field2)*100)/(SUM(fct_bookings.field2)) AS Percentage_entitys_booking
      --(SELECT total_bookings*100/(CAST(COUNT(fct_bookings.field1) AS INT)) ashhhhhhh),
      ROUND(COUNT(fct_bookings.field2),2)*100/ROUND(44271) AS Percentage_entitys_booking
FROM fct_bookings
LEFT JOIN fct_cancellations
ON fct_bookings.field2=fct_cancellations.field1
WHERE (fct_bookings.field15 BETWEEN "2018-01-01" AND "2018-12-31")
GROUP BY 1
HAVING COUNT(fct_bookings.field2) >= 50
Order BY 3 DESC;
```

What are the top three cancellation reasons in the U.S between 2017 and 2018?

As shown in the below bar graph, the leading root causes of cancellation were accommodations unnecessary, other, and extenuating circumstance.

For us, the word accommodations unnecessary is that the guest house doesn’t have proper amenities for the clients.



2.- SQL query- Part A

```sql
SELECT
    DISTINCT fct_bookings.field1 AS ID_listing,
      SUM(fct_bookings.field9) AS Total_booking_value,
      fct_bookings.field10 AS Checking_date,
    strftime('%Y', fct_bookings.field15) AS Year,
    strftime('%m', fct_bookings.field15) AS month,
    strftime('%d', fct_bookings.field15) AS Day,
    datetime(fct_bookings.field14, 'unixepoch') AS Timestamp_when_booking_was_made,
    dim_listings.field4 AS Country,
    fct_cancellations.field6 AS Cancel_reason
FROM fct_bookings
INNER JOIN dim_listings -- para encontrar el pais
      ON fct_bookings.field6=dim_listings.field1
INNER JOIN fct_cancellations -- para encontrar el Id_listing
      ON dim_listings.field1=fct_cancellations.field2
WHERE (Checking_date BETWEEN "2017-01-01" AND "2018-12-31") AND (dim_listings.field4= "US")
GROUP BY 1,5,3
ORDER BY 4,5,6 DESC;
```

2.- SQL query- Part B

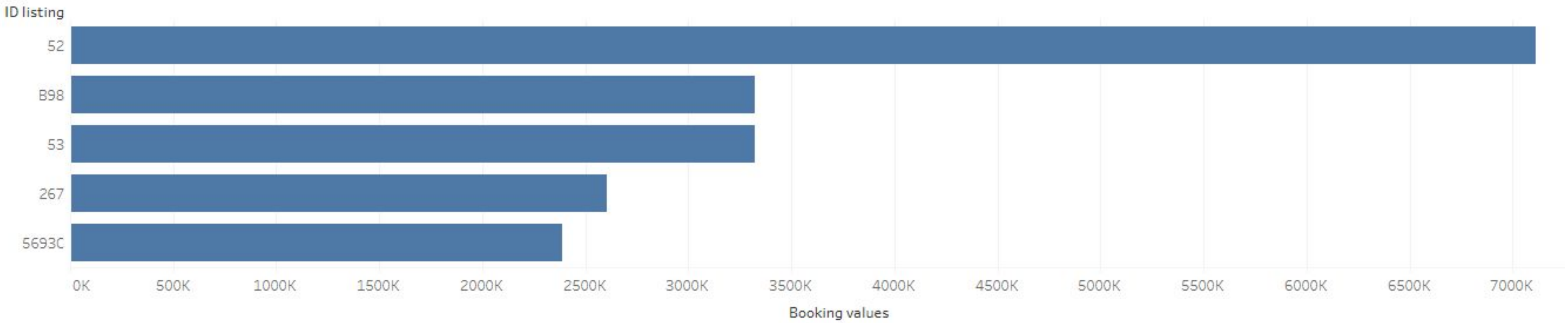
```
SELECT
      prompt3_2.ID_listing,
      prompt3_2.Total_booking_value,
      prompt3_2.Checking_date,
      prompt3_2.Timestamp_when_booking_was_made,
      prompt3_2.ds,
      prompt3_2.Year_booking_date,
      prompt3_2.month_booking_date,
      prompt3_2.Day_booking_date,
      strftime('%H:%M:%S', prompt3_2.Timestamp_when_booking_was_made) AS time_booking_date,
      prompt3_2.Country,
      prompt3_2.Cancel_reason
FROM prompt3_2
WHERE (Checking_date BETWEEN "2017-01-01" AND "2018-12-31")
GROUP BY 3,9
ORDER BY 6,7,8,9 ASC;
```

What are the top five entities with best economic performance in 2018 YTD?

The bar graph depicts that the entity - 52 had the best performance of profits with 7000,000 booking value, representing twice more than the closer two entities to it.

Opposite, the entity - 5693C had the lowest profit, approximately 2,400,000 booking value.

Top five entities with the best economic performance in 2018 YTD



3.- SQL query

```
SELECT
      DISTINCT fct_bookings.field1 AS ID_listing,
      SUM(fct_bookings.field9) AS Booking_values,
      fct_bookings.field15 AS "2018_YTD"
FROM fct_bookings
WHERE ("2018_YTD" BETWEEN "2018-01-01" AND "2018-12-31")
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5;
```

Limitations

The SQL queries required information from the four tables provided; thus, we must use Joins. We verified several times the type of Join to use, after several tries we accessed and obtained to the target columns.

References

- AIRBNB tables -bookings, cancellations, listing, and entities (100MB)
- SQLite strftime Function. Retrieved from http://www.sqlitetutorial.net/sqlite-date-functions/sqlite-strftime-function/
- Date And Time Functions. Retrieved from https://www.sqlite.org/lang_datefunc.html
- Year to Date (YTD) Retrieved from https://www.investopedia.com/terms/y/ytd.asp