Client Report - Are we missing JSON on our flight?

Course DS 250
Conner Crook

Elevator pitch

Flight delays happen more than we would like and there are many reasons why flights can get delayed. This data shows us different reasons why flights can be delayed and what trends we can see to help us avoid delays.

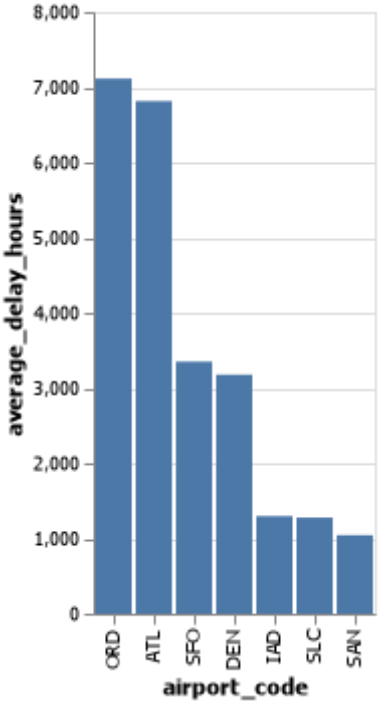
GRAND QUESTION 1

Which airport has the worst delays? How did you choose to define “worst”? As part of your answer include a table that lists the total number of flights, total number of delayed flights, proportion of delayed flights, and average delay time in hours, for each airport.

_To decide which airport had the worst delays, I decided that the airport with the highest average delay hours would be the worst. Chicago O'hare International Airport (ORD) had the worst delays of the listed airports. It had an average of 7115.67 hours of delays. Jackson Atlanta International Airport (ATL) also had a high number of delay hours at 6816.15 hours delayed. _

TECHNICAL DETAILS

```
worst_chart = (alt.Chart(worst_delay)
    .encode(
        alt.X('airport_code', sort=alt.EncodingSortField(field='airport_code', op='count')),
        y = 'average_delay_hours')
    .mark_bar())
```




```
print(worst_delay
    .head(20)
    .to_markdown(index=False))
```

airport_code	total_flights	total_delays	proportion_delay	average_delay_hours
ORD	3597588	830825	23.09	7115.67
ATL	4430047	902443	20.37	6816.15
SFO	1630945	425604	26.1	3352.33
DEN	2513974	468519	18.64	3178.46
IAD	851571	168467	19.78	1298.42
SLC	1403384	205160	14.62	1278.2
SAN	917862	175132	19.08	1044.98

GRAND QUESTION 2

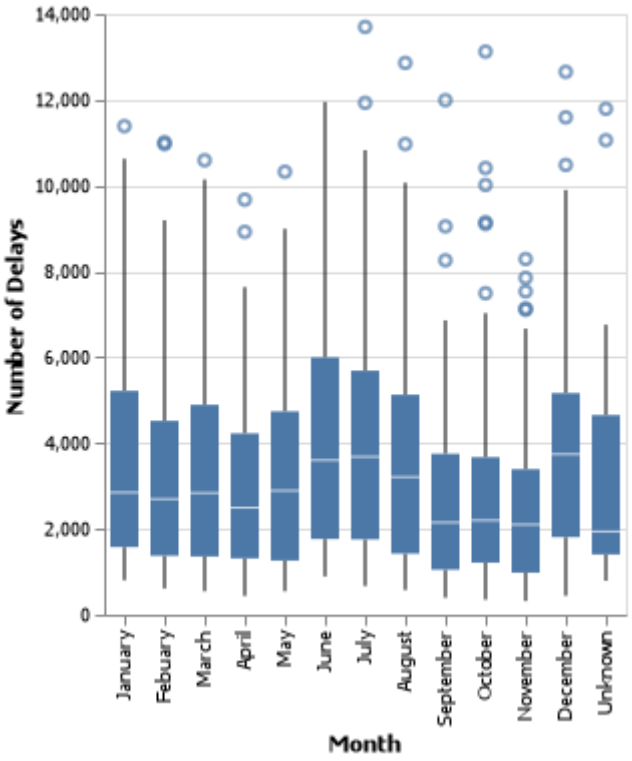
What is the worst month to fly if you want to avoid delays? Include one chart to help support your answer, with the x-axis ordered by month. You also need to explain and justify how you chose to handle the missing Month data.

July has the highest number of total delays with 319960 total delays. In the box plot, we can see that July, June, and December all have around the same average of delays but June and July have a larger spread or Standard Deviation of delays each year per month. Some month data was missing. These were replaced with "Unknown" in the dataset and is displayed in the chart.

TECHNICAL DETAILS

```
month_box = (alt.Chart(flights_clean)
    .encode(
        alt.Y('num_of_delays_total', title='Number of Delays'),
        alt.X('month:O', sort=['January', 'Febuary', 'March', 'April', 'May', 'June', 'July', 'A
            title='Month'))
    .mark_boxplot()
    )
```

insert your chart png here




```
print(worst_month
    .head(20)
    .sort_values(by="total_delays", ascending = False)
    .to_markdown(index=False))
```

month	total_delays
July	319960

month	total_delays
June	317895
December	303133
August	279699
January	265001
March	250142
Febuary	248033
October	235166
May	233494
April	231408
September	201905
November	197768

GRAND QUESTION 3

According to the BTS website the Weather category only accounts for severe weather delays. Other “mild” weather delays are included as part of the NAS category and the Late-Arriving Aircraft category. Calculate the total number of flights delayed by weather (either severe or mild).

Below is the data for flights that were delayed due to weather. For late_aircraft, I took 30% of delays to account for mild weather. For NAS, if it was April - August, I calculated 40% were mild delays and the rest was 60% mild delays. There was a total of 1080255 delays due to weather. 93298 of these was for severe weather. All missing data was changed to NaN for calculations.

TECHNICAL DETAILS

```
print(Total_weather_delays
    .head(20)
    .to_markdown(index=True, tablefmt="pretty"))
```

	Total
num_of_delays_weather	93298.0

	Total
late_aircraft_mild_weather	294134.4
nas_mild_weather	692822.65
Total	1080255.05

GRAND QUESTION 4

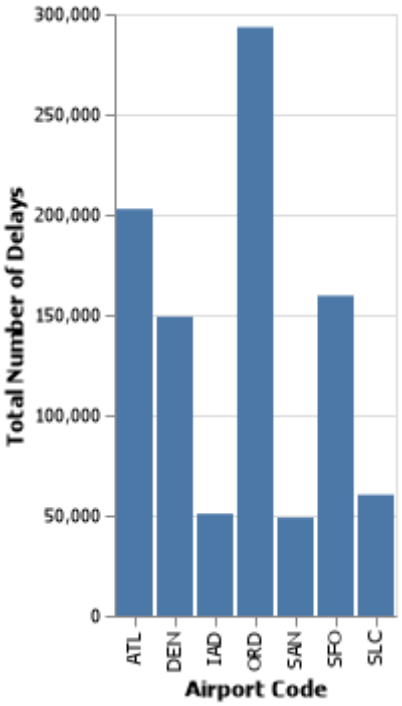
Create a barplot showing the proportion of all flights that are delayed by weather at each airport. What do you learn from this graph (Careful to handle the missing Late Aircraft data correctly)?

Chicago O'hare International Airport (ORD) had the worst delays due to weather conditions. There was a total of 293439 delays due to weather at the ORD airport.

TECHNICAL DETAILS

```
weather_chart = (alt.Chart(fixed_weather_chart)
    .encode(
        alt.X('airport_code', title = "Airport Code"),
        alt.Y('airport_weather_delay', title = "Total Number of Delays")
    )
    .mark_bar())
```

insert your chart png here



```
print(fixed_weather_chart
    .head(20)
    .to_markdown(index = False))
```

airport_code	airport_weather_delay
ATL	202803
DEN	149107
IAD	50842.7
ORD	293439
SAN	48920.6
SFO	159594
SLC	60345.8

GRAND QUESTION 5

Fix all of the varied NA types in the data to be consistent and save the file back out in the same format that was provided (this file shouldn't have the missing values replaced with a value). Include one record example from your exported JSON file that has a missing value (No imputation in this file).

type your results and analysis here

TECHNICAL DETAILS

```
#paste chart code in this snippet box
```

insert your chart png here

```
#paste your table code in this snippet box
```

replace the table below with your table

	animal
0	elk

	animal
1	pig
2	dog
3	quetzal

APPENDIX A (PYTHON CODE)

```
#%%
import pandas as pd
import numpy as np
import altair as alt
#%%
#Read in the Data into "flights"
URL = "https://github.com/byuidatascience/data4missing/raw/master/data-raw/flights_missing/fligh
flights = pd.read_json(URL)
flights
#%%
## GRAND QUESTION 1
#Creates a chart that sums up Total flights, delays, and average delay time in hours.
worst_delay = (flights.groupby('airport_code')
    .agg(
        total_delays = ('num_of_delays_total', np.sum),
        total_flights = ('num_of_flights_total', np.sum),
        average_delay_time = ('minutes_delayed_total', np.mean))
    #This creates new columns for the following agg functions
    .assign(
        average_delay_hours = lambda x: (x.average_delay_time / 60).round(2),
        proportion_delay = lambda x: (x.total_delays / x.total_flights * 100).round(2))
    .filter(['airport_code', 'total_flights', 'total_delays', 'proportion_delay', 'average_delay
    .sort_values(by = ['average_delay_hours'], ascending = False)
    .reset_index())
#%%
#A bar chart that shows the airports with the desired columns.
worst_chart = (alt.Chart(worst_delay)
    .encode(
        alt.X('airport_code', sort=alt.EncodingSortField(field='airport_code', op='count')),
        y = 'average_delay_hours')
    .mark_bar())
#%%
# Creates markdown table
print(worst_delay
    .head(20)
    .to_markdown(index=False))
#%%
#Some months had the value "n/a". These were replaced with "Unkown" and replaces
#all weird numbers with nan in late aircraft for Grand Question 3 - 4
flights_clean = (flights.replace(to_replace = 'n/a', value = 'Unknown')
    .assign(
        num_of_delays_late_aircraft = lambda x: np.where(x.num_of_delays_late_aircraft < 0, np.r
    ))
#%%
## GRAND QUESTION 2
# Finds total delays by month. Excludes unknown months
worst_month = (flights_clean.filter(['month', 'num_of_delays_total'])
    .groupby('month')
    .agg(
        total_delays = ('num_of_delays_total', np.sum)
```

```
    )
    .query('month != "Unknown"')
    .reset_index())
worst_month
#%%
# Chart showing what months have the most total delays
month_chart = (alt.Chart(worst_month)
    .encode(
        alt.Y('total_delays', title='Number of Delays'),
        alt.X('month:O', sort=['January', 'Febuary', 'March', 'April', 'May', 'June', 'July', 'A
            title='Month'))
    .mark_bar())

#%%
# Boxplot to show the averages
month_box = (alt.Chart(flights_clean)
    .encode(
        alt.Y('num_of_delays_total', title='Number of Delays'),
        alt.X('month:O', sort=['January', 'Febuary', 'March', 'April', 'May', 'June', 'July', 'A
            title='Month'))
    .mark_boxplot()
    )
#%%
# Table showing the total amount of delays per month.
print(worst_month
    .head(20)
    .sort_values(by="total_delays", ascending = False)
    .to_markdown(index=False))
#%%
## GRAND QUESTION 3
# Takes all weather, nas, and late aircraft delays. Takes 30% of late_aircraft into new column.
# and 65% during any other month.
weather_delays = (flights_clean.filter(["month","num_of_delays_weather", "num_of_delays_late_air
conditions = [(weather_delays['month'] == "April") | (weather_delays['month'] == "May") | (weath
choices = [weather_delays['num_of_delays_nas'] * .4]
weather_delays['nas_mild_weather'] = np.select(conditions, choices, default = weather_delays['nu
fixed_weather = (weather_delays.filter(['num_of_delays_weather', 'num_of_delays_late_aircraft',
    .assign(
      late_aircraft_mild_weather = lambda x: (x.num_of_delays_late_aircraft * .3)
    )
    .filter(['num_of_delays_weather', 'late_aircraft_mild_weather', 'nas_mild_weather']))
fixed_weather
#%%
# puts the sums into a table
Total_weather_delays = (fixed_weather.sum(axis=0))
Total_weather_delays.loc['Total'] = Total_weather_delays.sum(axis=0)
Total_weather_delays
#%%
# Create Table for Markdown
print(Total_weather_delays
    .head(20)
```

```
    .to_markdown(index=True, tablefmt="pretty"))
#%%
## GRAND QUESTION 4
# Does the same as above but adds a column for the totals and adds totals per aiport code.
weather_delays_chart = (flights_clean.filter(["airport_code", "month","num_of_delays_weather", "
conditions = [(weather_delays['month'] == "April") | (weather_delays['month'] == "May") | (weath
choices = [weather_delays['num_of_delays_nas'] * .4]
weather_delays_chart['nas_mild_weather'] = np.select(conditions, choices, default = weather_dela
fixed_weather_chart = (weather_delays_chart.filter(['airport_code', 'num_of_delays_weather', 'nu
    .assign(
      late_aircraft_mild_weather = lambda x: (x.num_of_delays_late_aircraft * .3),
      Totals = lambda x: (x.num_of_delays_weather + x.late_aircraft_mild_weather + x.nas_mild_we
    )
    .filter(['airport_code', 'Totals'])
    .groupby('airport_code')
    .agg(
        airport_weather_delay = ('Totals', np.sum)
    )
    .reset_index())
fixed_weather_chart
#%%
# Bar chart to display the above
weather_chart = (alt.Chart(fixed_weather_chart)
    .encode(
        alt.X('airport_code', title = "Airport Code"),
        alt.Y('airport_weather_delay', title = "Total Number of Delays")
    )
    .mark_bar())
#%%
# Print table for markdown
print(fixed_weather_chart
    .head(20)
    .to_markdown(index = False))
#%%
```