DSC640-T301
Chris Kellogg
Weeks 9 & 10 Exercise

Audience
My audience is the National Association of Entitled Persons (NAEP). The NAEP is a group of like-minded individuals who think that they're entitled to whatever they want and that policies exist for other people, not for them. They can tell you with absolute confidence that you *do* work here, you *can* take their expired coupons, the rules don't apply to their situation, and if you won't do what they want, then you need to get the manager. *Right now*.

Purpose
I want the NAEP members to know which US airports are the easiest pickings for complaints and where to go to challenge themselves and make their own personal mark.

Medium
I'm drafting a sort of a short travel guide for NAEP members, explaining which airports are best for practice, since they take high volumes of complaints. I'm also explaining which airports take so few complaints that they'll never see the NAEP coming, and they'll be talking about that visit for weeks to come.

Design
I used the choropleths (US maps) with yellow for low numbers and bolder colors for high numbers. I like the way that the color scheme de-emphasized the lower numbers and really made it easy to find the top end values of the scale. I love the heat map and the way it emphasizes the percentages above 20 with not only warm colors, but also black text; those two values really jump out of the graphic, which is appropriate because they're significantly higher than the others. The box plot was a little bit underwhelming for looking at the distribution of complaint numbers, but it think it kind of gets rescued by using the option to plot the individual values, really emphasizing the density at the low end and the tremendous right skew. Finally, I turned to a couple simple text tables because I think they're perfect for the simple, straight-forward data they're presenting.

Ethical Considerations
I don't see any ethical issues with using any of the data, since it's publicly available and not sensitive. I'll note that I did drop records with missing values whenever I was categorizing complaints, simply because a complaint that didn't have a category or an airport doesn't have any value in the graphics that I was making.

National Association of Entitled Persons (NAEP)

Vacation Planning Tips and Information

Welcome new members

This publication is distributed each year in late fall, before the heavy winter holiday travel season, and again in late spring before summer vacations. You'll find useful information about the best places to practice and hone your skills and where you're most likely to meet other members to exchange advice and techniques. We also maintain the Low-Complaint List (which our lawyers will no longer allow us to call the Bounty List) for people who feel like they're ready to make history. Enjoy the info; you know you deserve it!

Easy pickin's

If you're just learning to ask for the manager, you may want to start out with this map as you plan your upcoming travel. New York is a popular NAEP destination because it boasts two of the most-complained-about (MCA) airports in the United States!

Florida is a pretty reasonable destination as well, boasting several airports within driving distance, many of which are more in the middle range of complaints. Florida is generally a little bit more relaxed, so if you just want an easy environment to complain in, but you want to be the only one griping, Florida has a *lot* of opportunity.

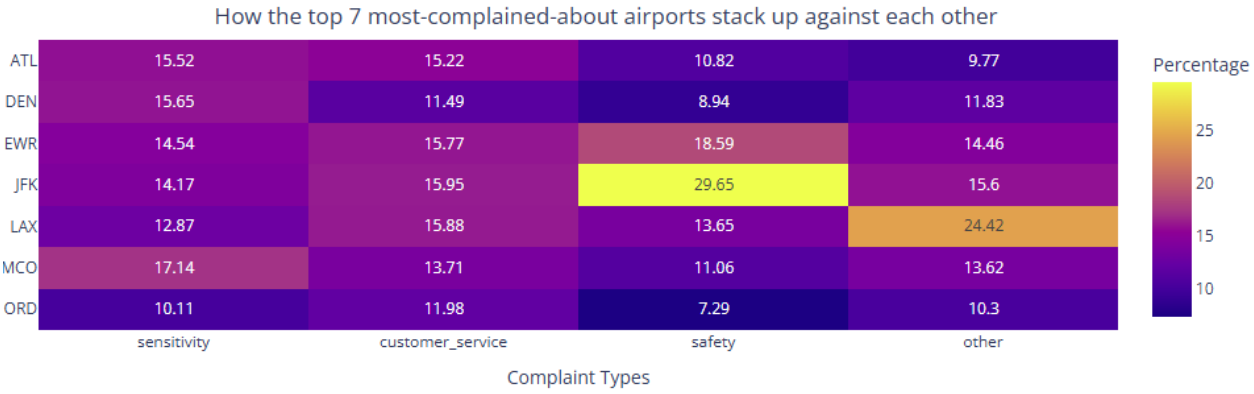
The easiest complaints to make

Remember, the easiest and most basic way to practice your entitlement is to complain that people aren't going out of their way enough for you, that they're insensitive to your particular, personal, individual situation, which you can pretty much adapt on the fly! You can easily complain that TSA agents took too long, or that they weren't thorough enough. You can bump into them and complain that they touched you. You can even carry something like a water bottle (that everybody knows is

not allowed) and complain that it's not a big deal, or just refuse to surrender it. Sensitivity complaints make up a pretty steady percentage of all complaints, as show below.

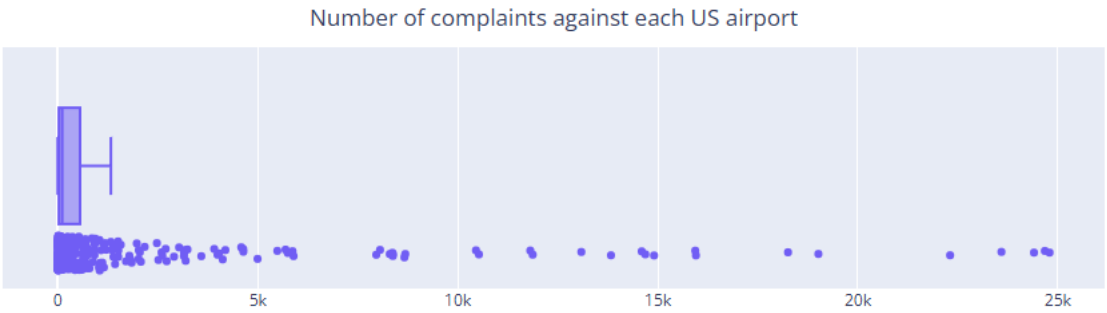
Showing some more range

If you don't want to keep making sensitivity complaints like a rookie, you should head for one of the two largest metro areas in the country. New York's JFK airport is an absolute *haven* for NAEP members who want to try making more substantiated complaints, particularly when it comes to safety concerns. Out of the top 7 MCA airports, JFK is responsible for nearly *30 percent* of the safety complaints made. Avoid Chicago's O'Hare and Denver International, though, because they're almost laughably safe in comparison. If you prefer the sunny west coast, LAX is the best high-volume complaint receiver when it comes to the wide-ranging "other" category with almost a quarter of those miscellaneous annoyances.



Does it really matter?

It might seem like the choice is airport is pretty arbitrary; after all, NAEP members really should be able to find something to complain about anywhere they go. Well, over the last 10 years or so, most airports have fewer that 2000 complaints total! As shown on the next page, there are probably fewer than 40 airports with over 5000 complaints in that time frame.



Those chosen few above the 18000 mark are the ones that are the best place to observe expert complainers and learn what separates average NAEP members from the Hall of Famers. The hotbeds of NAEP activity are listed here. You can't really go wrong with any of these airports.

US Airports with the most TSA complaints

Complaints	Airport	State
24794	Los Angeles International Airport	California
24677	John F. Kennedy International Airport	New York
24405	Newark Liberty International Airport	New Jersey
23591	Hartsfield-Jackson Atlanta International Airport	Georgia
22308	Orlando International Airport	Florida
19015	Denver International Airport	Colorado
18256	Chicago O'Hare International Airport	Illinois

Low-Complaint List (formerly the NAEP Bounty List)

As noted above, our Legal Department has insisted that we change the name of this list, but we're not prevented from continuing to share it! These 7 airports each have fewer than 2 complaints in the last 10 years, and Worland Municipal Airport in Wyoming has never had a TSA complaint. If you're looking to really make a name for yourself, think of the notoriety of being the only person ever to register a TSA complaint there. It's not for the faint of heart. It has had no scheduled airline service since 2016, it houses only 15 aircraft (including 8 helicopters), and it serves an average of 60 people per week. If you can figure out how to get there, you could make history!

US Airports with the fewest TSA complaints

Complaints	Airport	State
0	Worland Municipal Airport	Wyoming
1	Athens Ben Epps Airport	Georgia
1	Cold Bay Airport	Alaska
1	Dawson Community Airport	Montana
1	Kingman Airport	Arizona
1	McNary Field	Oregon
1	Tupelo Regional Airport	Mississippi

Kellogg640Week9and10b

November 5, 2024

[1]:
```
## Chris Kellogg
## DSC640-T301
## Weeks 9 & 10
## ##############################################################################
## Weeks 9 & 10 Exercises: TSA Complaints
## ##############################################################################
##
```

[2]:
```
##
## load necessary packages
##

# import and alias Pandas
import pandas as pd

# import and alias NumPy
import numpy as np

# import the warnings packages and suppress filter warnings
import warnings
warnings.filterwarnings("ignore")

# import and alias the plotting packages
import matplotlib.pyplot as plt
import plotly.graph_objects as go
import plotly.express as px
```

[3]:
```
##
## define a function to read and summarize a CSV
##

# define the CSV directory
path = 'TSAcomplaints/'

# function to read, summarize, and return a CSV
def read_and_summarize (name):

    # print the path name to the CSV
```

```
    print ('loading ' + path + name + '.csv\n')

    # read the CSV into a DataFrame
    df = pd.read_csv(path + name + '.csv')

    # print the number of records in the DataFrame
    print ('length:', len(df), 'records\n')

    # show any missing data
    print ('missing data:\n', df.isna().sum(), '\n', sep='')

    # preview the DataFrame
    print ('preview:\n', df.head(), sep='')

    # return the DataFrame
    return df
```

[4]:
```
##
## read and summarize the TSA complaint CSV files
##

df_airport = read_and_summarize('complaints-by-airport')
df_category = read_and_summarize('complaints-by-category')
df_subcategory = read_and_summarize('complaints-by-subcategory')
df_icao = read_and_summarize('iata-icao')
```

```
loading TSAcomplaints/complaints-by-airport.csv

length: 41721 records

missing data:
pdf_report_date      0
airport            109
year_month           0
count                0
dtype: int64

preview:
  pdf_report_date airport year_month  count
0         2019-02     ABE    2015-01      0
1         2019-02     ABE    2015-02      0
2         2019-02     ABE    2015-03      0
3         2019-02     ABE    2015-04      0
4         2019-02     ABE    2015-05      2
loading TSAcomplaints/complaints-by-category.csv

length: 241588 records
```

```
missing data:
pdf_report_date        0
airport             4042
category               0
year_month             0
count                  0
clean_cat              0
clean_cat_status       0
dtype: int64

preview:
  pdf_report_date airport                          category year_month  \
0         2019-02     ABE         Hazardous Materials Safety    2015-01
1         2019-02     ABE  Mishandling of Passenger Property    2015-01
2         2019-02     ABE         Hazardous Materials Safety    2015-02
3         2019-02     ABE  Mishandling of Passenger Property    2015-02
4         2019-02     ABE         Hazardous Materials Safety    2015-03

   count                          clean_cat clean_cat_status
0      0         Hazardous Materials Safety         original
1      0  Mishandling of Passenger Property         original
2      0         Hazardous Materials Safety         original
3      0  Mishandling of Passenger Property         original
4      0         Hazardous Materials Safety         original
loading TSAcomplaints/complaints-by-subcategory.csv

length: 504512 records

missing data:
pdf_report_date                  0
airport                      13464
category                         0
subcategory                      0
year_month                       0
count                            0
clean_cat                        0
clean_subcat                     0
clean_cat_status                 0
clean_subcat_status              0
is_category_prefix_removed       0
dtype: int64

preview:
  pdf_report_date airport                          category  \
0         2019-02     ABE         Hazardous Materials Safety
1         2019-02     ABE  Mishandling of Passenger Property
2         2019-02     ABE         Hazardous Materials Safety
3         2019-02     ABE  Mishandling of Passenger Property
```

```
4         2019-02     ABE         Hazardous Materials Safety

                              subcategory year_month  count  \
0                                 General    2015-01      0
1  Damaged/Missing Items--Checked Baggage    2015-01      0
2                                 General    2015-02      0
3  Damaged/Missing Items--Checked Baggage    2015-02      0
4                                 General    2015-03      0

                           clean_cat                              clean_subcat  \
0         Hazardous Materials Safety                                   General
1  Mishandling of Passenger Property  *Damaged/Missing Items--Checked Baggage
2         Hazardous Materials Safety                                   General
3  Mishandling of Passenger Property  *Damaged/Missing Items--Checked Baggage
4         Hazardous Materials Safety                                   General

  clean_cat_status clean_subcat_status  is_category_prefix_removed
0         original            original                       False
1         original            original                       False
2         original            original                       False
3         original            original                       False
4         original            original                       False
loading TSAcomplaints/iata-icao.csv

length: 8937 records

missing data:
country_code      32
region_name        0
iata               0
icao            1143
airport            0
latitude           0
longitude          0
dtype: int64

preview:
  country_code region_name iata  icao                          airport  \
0           AE    Abu Zaby  AAN  OMAL     Al Ain International Airport
1           AE    Abu Zaby  AUH  OMAA  Abu Dhabi International Airport
2           AE    Abu Zaby  AYM   NaN         Yas Island Seaplane Base
3           AE    Abu Zaby  AZI  OMAD      Al Bateen Executive Airport
4           AE    Abu Zaby  DHF  OMAM               Al Dhafra Air Base

   latitude  longitude
0   24.2617    55.6092
1   24.4330    54.6511
2   24.4670    54.6103
```

```
3   24.4283    54.4581
4   24.2482    54.5477
```

[5]:
```
##
## create functions to return flags for sensitivity, safety, and
## customer service based on categories and subcategories of complaints
##

def is_sensitivity_complaint(row):
    result = 0
    if row.clean_cat in [
        'Civil Rights',
        'Persons w/ Disabilities (PWD)',
        'Request for Assistance',
        'Screening',
        'Animals'
    ] or row.clean_subcat in [
        'Wheelchair Assistance',
        'Children',
        'Breast Milk/Baby Formula/Breast Pump',
        'Medication/Prescription/Medically Necessary Items',
        'Remains',
        'Non-Persons w/ Disabilities (PWD)',
        'Wounded Warrior (MSIJSOC)'
    ]:
        result = 1
    return result

def is_customer_service_complaint(row):
    result = 0
    if row.clean_cat in [
        'Mishandling of Passenger Property',
        'Customer Service',
        'Call Disconnect/Dropped/Wrong Number',
        'Expedited Passenger Screening Program',
        'Identification (ID) Requirements',
        'Notice of Violation/Civil Penalties',
        'Property - Special Handling'
    ] or row.clean_subcat in [
        'Missing or Damaged Lock'
    ]:
        result = 1
    return result

def is_safety_complaint(row):
    result = 0
    if row.clean_cat in [
```

```
        'Hazardous Materials Safety', \
        'Flying Armed', \
        'HAZMAT Endorsement', \
        'Prohibited & Permitted Items'
    ]:
        result = 1
    return result
```

[6]:
```
##
## create 1-hot columns for sensitivity, safety, and customer service
## based on the categories and subcategories of the complaints
##

df_subcategory['sensitivity'] = df_subcategory.apply(is_sensitivity_complaint,␣
 ↪axis=1)
df_subcategory['customer_service'] = df_subcategory.
 ↪apply(is_customer_service_complaint, axis=1)
df_subcategory['safety'] = df_subcategory.apply(is_safety_complaint, axis=1)
```

[29]:
```
##
## define a function to draw a map of the complaints in the US
##

def draw_complaint_map(data, title):

    # merge the complaint count with the airport info
    df = data \
        .dropna() \
        .groupby('airport') \
        .agg(
            count = ('count', 'sum')
        ) \
        .merge(
            df_icao,
            left_on='airport',
            right_on='iata',
            how = 'inner'
        )

    # create a hover legend for the points
    df['text'] = df.iata + ' (' + df['count'].astype(str) + ' complaints)'

    # build a scatterplot of the airport locations,
    # colored according to complaint count
    fig = go.Figure(
        data = go.Scattergeo(
            lon = df.longitude,
```

```
            lat = df.latitude,
            text = df.text,
            mode = 'markers',
            marker = dict(
                size = 8,
                opacity = 0.8,
                reversescale = True,
                color = df['count'],
                cmax = df['count'].max(),
                colorbar_title = "Complaints"
            )
        ),
        layout = dict(
            width = 1000,
            height = 500
        )
    )

    # show the scale
    fig.update_layout(
            title = title,
            title_x = 0.6,
            title_y = 0.85,
            geo_scope = 'usa',
        )

    # draw the visual
    fig.show()
```

[31]:
```
##
## draw a map of the total number of TSA complaints in the US
##

draw_complaint_map(
    df_airport,
    'Total Number of TSA Complaints for US Airports (Jan 2015-Jan 2024)'
)
```

[33]:
```
##
## draw a map of the number of sensitivity TSA complaints in the US
##

draw_complaint_map(
    df_subcategory.query('sensitivity == 1'),
    'Number of Sensitivity Complaints for US Airports (Jan 2015-Jan 2024)')
```

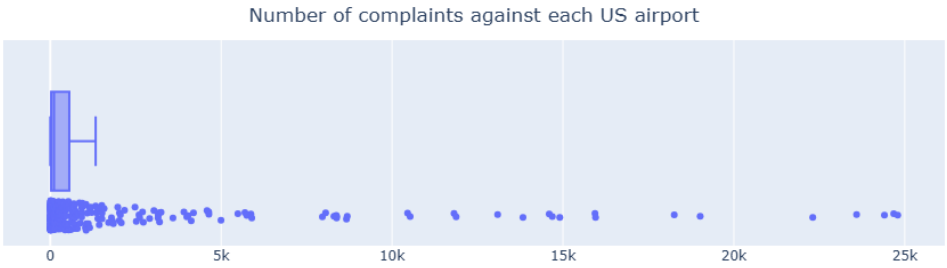
[10]:
```
##
```

```
## draw a heat map of how the most-complained-about airports in the US stack up␣
 ↪against each other
##

# get the list of airports with more than 18000 total complaints
airport_list = df_airport \
    .dropna() \
    .groupby('airport') \
    .agg(
        count = ('count', 'sum')
    ) \
    .query('count > 18000') \
    .index \
    .to_list()

# build a new list of complaint counts
df_items = []

# for each airport in the top list
for airport in airport_list:

    # add a row with complaint counts to the new list
    df_items.append([
        airport,
        df_subcategory.query(f'airport == "{airport}" and sensitivity ==␣
 ↪1')['count'].sum(),
        df_subcategory.query(f'airport == "{airport}" and customer_service ==␣
 ↪1')['count'].sum(),
        df_subcategory.query(f'airport == "{airport}" and safety ==␣
 ↪1')['count'].sum(),
        df_subcategory.query(f'airport == "{airport}" and␣
 ↪sensitivity+customer_service+safety == 0')['count'].sum()
    ])

# build a data frame from the list
df = pd.DataFrame(df_items)
df.columns = ['airport', 'sensitivity', 'customer_service', 'safety', 'other']
df.set_index('airport', inplace=True)

# calculate the total of each complaint type
total_sensitivity = df.sensitivity.sum()
total_customer_service = df.customer_service.sum()
total_safety = df.safety.sum()
total_other = df.other.sum()

# calulate percentages
df.sensitivity = df.sensitivity / total_sensitivity * 100.0
```

```
df.customer_service = df.customer_service / total_customer_service * 100.0
df.safety = df.safety / total_safety * 100.0
df.other = df.other / total_other * 100.0

# round the ppercentages to 2 decimals
df = df.round({'sensitivity': 2, 'customer_service': 2, 'safety': 2, 'other':␣
 ↪2})

# create the heatmap
fig = px.imshow(
    df,
    text_auto = True,
    aspect = 'auto',
    labels = dict(
        x='Complaint Types',
        y='',
        color='Percentage'
    )
)

# add the title
fig.update_layout(
    title = 'How the top 7 most-complained-about airports stack up against each␣
 ↪other',
    title_x = 0.5,
    title_y = 0.9
)

# show the plot
fig.show()
```

[11]:
```
##
## draw a boxplot of the number of TSA complaints at each US airport
##

## build a dataset of complaints at each US airport
df = df_category \
    .dropna() \
    .groupby('airport') \
    .agg(
        complaints = ('count', 'sum')
    ) \
    .merge(
        df_icao,
        left_on='airport',
        right_on='iata',
        how = 'inner'
```

```
    ) \
    .query('country_code == "US"') \
    [['region_name', 'airport', 'complaints']]

# create a boxplot of the distribution, includiong the points
fig = go.Figure()
fig.add_trace(
    go.Box(
        x = df.complaints,
        boxpoints = 'all',
        name = ''
    )
)

# add the titles
fig.update_layout(
    title = 'Number of complaints against each US airport',
    title_x = 0.5,
    title_y = 0.8,
    xaxis_title = '',
    yaxis_title = ''
)

# show the plot
fig.show()
```



[12]:
```
##
## define a function to generate a text table
##

def generate_text_table(df, title, ascending):
```

```
    # sort the table values
    df.sort_values('complaints', ascending = ascending, inplace = True)

    # creat a table from the data
    fig = go.Figure(data=[go.Table(
        header=dict(values=['Complaints', 'Airport', 'State']),
        cells=dict(values=[
            df.complaints,
            df.airport,
            df.region_name
        ])
    )])

    # add the title
    fig.update_layout(
        title = title,
        title_x = 0.5,
        title_y = 0.8
    )

    # display the table
    fig.show()
```

[13]:
```
##
## draw a text table of the US airports with the fewest TSA complaints
##

generate_text_table(
    df.query('complaints < 2'),
    title = 'US Airports with the fewest TSA complaints',
    ascending = True
)
```

US Airports with the fewest TSA complaints

Complaints	Airport	State
0	Worland Municipal Airport	Wyoming
1	Athens Ben Epps Airport	Georgia
1	Cold Bay Airport	Alaska
1	Dawson Community Airport	Montana
1	Kingman Airport	Arizona
1	McNary Field	Oregon
1	Tupelo Regional Airport	Mississippi

[14]:
```
##
## draw a text table of the US airports with the most TSA complaints
##

generate_text_table(
    df.query('complaints > 18000'),
    title = 'US Airports with the most TSA complaints',
    ascending = False
)
```

US Airports with the most TSA complaints

Complaints	Airport	State
24794	Los Angeles International Airport	California
24677	John F. Kennedy International Airport	New York
24405	Newark Liberty International Airport	New Jersey
23591	Hartsfield-Jackson Atlanta International Airport	Georgia
22308	Orlando International Airport	Florida
19015	Denver International Airport	Colorado