DATA PREP & EDA SHOWCASE

Business Intelligence Analyst

Applicant: **Nick Belgau**

Chemical Engineer at Marathon Petroleum Co.

1

1

OVERVIEW

- This report was a voluntary submission to the FleetPride hiring team to showcase my technical skills when working with large datasets.
- A real dataset was queried from the FleetPride website and merged with data scrapped from Google Reviews to gain insight into what the public is saying about each FleetPride location.
- There are two main sections
  - Data Extraction & Preparation
  - Exploratory Data Analysis
- Procedure
  - The FleetPride dataset was copied from an online PDF into excel and cleaned using VBA script and other techniques.
  - Each FleetPride location had Google Reviews scraped from the Google Places API, and concatenated with the excel file.
  - The final dataset was uploaded to a Jupyter Notebook (Python 3) for analysis using Seaborn and Pandas.
- Disclaimer: The figures, slides, and annotations are brief notes to convey my cognition when beginning to analyze a dataset. In no way are these "report-ready" for a professional audience. Throughout this slide pack, several "next-steps" and "follow-up Q's" are outlined. An additional next-step would certainly be to import these tables in a BI suite such as PowerBI or Tableau.

2

2

DATA CLEANING

3

3

DATA CLEANING

1. Copied table from FleetPride website to Excel (note how messy it was)

AL Birmingham (205) 322-5621 • • AL Decatur (256) 353-7977 • 750-8232 • • AR Conway (501) 358-6929 • • • • • • • • • • • • • • • • • • AR Texarkana (870) 772-9545 • • • • • • • • • • • • • • • • •

			TYPE				TRUCK REPAIR												
State	City	Phone	Service Center	Driveline	Fluid Power	Tank	Full Service Repair	A/C Service	Alignments	Clutch Removal & Replacement	DOT Inspections	Exhaust System, including DPF/DOC Service	Electrical System	Emergency Road Service	Engine Code Diagnostics	External Engine Repair	Internal Engine Repair	Hydraulic Cylinder Removal & Replacement	
AL	Birmingham	(205) 322-5621		•															
AL	Decatur	(256) 353-7977		•															
AL	Dothan	(334) 793-6444		•															
AL	Mobile	(251) 438-2489	•	•			•	•	•	•	•	•	•		•	•		•	
AL	Tuscaloosa	(205) 750-8232		•															
AR	Conway	(501) 358-6929	•				•	•	•	•	•	•	•	•	•	•			

https://www.fleetpride.com/wp-content/uploads/2020/04/1825.LineCard-LocationsServices040120.pdf

4

4

DATA CLEANING

1. Copied table from FleetPride website to Excel

AL Birmingham (205) 322-5621 • • AL Decatur (256) 353-7977 • 750-8232 • • AR Conway (501) 358-6929 • • • • • • • • • • • • • • • • • AR Texarkana (870) 772-9545 • • • • • • • • • • • • • • • • •

2. SUBSTITUTE() formula to remove special chars & insert comma delimiter
3. VBA script to remove numbers
4. Built-in 'Text to Columns' feature and transpose
5. TRIM() formula to remove trailing & leading spaces
6. Manually matched Service Center as Boolean classifier
   - If list was longer, I would've automated this step
   - Allowed for detail combing to check each entry for errors

FleetPride_data.xlsm - Module1 (Code)

(General)

```
Function RemoveNumbers(Txt As String) As String
With CreateObject("VBScript.RegExp")
.Global = True
.Pattern = "[0-9]"
RemoveNumbers = .Replace(Txt, "")
End With
End Function
```

24	FL Tampa
25	GA Albany GA Ellenwood
26	GA Valdosta
27	ID Idaho Falls

	A	B
1	locations	service_center
2	AL Birmingham	1
3	AL Decatur	
4	AL Dothan	
5	AL Mobile	1
6	AL Tuscaloosa	
7	AR Conway	1
8	AR Little Rock	1

5

5

WEB SCRAPING

- Objective: Use GCP Places API with python to gather valuable data from public reviews on FleetPride locations

- Procedure
  - Imported required libraries
  - Define googlemaps object
  - **Confirm retrieval for one location**
  - Identify fields of interest
  - Loop over FleetPride data locations that are in the excel file
  - Concatenate the dataframes to a final dataset & export

Note: the detailed reviews were limited to 5 results by CGP, so unfortunately this was omitted from the data set.

```
import googlemaps
import pandas as pd
gmaps = googlemaps.Client(key=APIkey)
```

```
#This shows the meta data available from the API for each location

place_name = 'Fleetpride' + ' ' + 'AL Decatur'
places_result = gmaps.places(place_name)

full_text = places_result['results'][0]
full_text
```

```
{'business_status': 'OPERATIONAL',
 'formatted_address': '1101 McEntire Ln NW, Decatur, AL 35601, Unitec
 'geometry': {'location': {'lat': 34.6189116, 'lng': -87.0339113},
  'viewport': {'northeast': {'lat': 34.62026487989272,
    'lng': -87.03274217010728},
   'southwest': {'lat': 34.61756522010728, 'lng': -87.03544182989272]
 'icon': 'https://maps.gstatic.com/mapfiles/place_api/icons/v1/png_7
 'icon_background_color': '#7B9EB0',
 'icon_mask_base_uri': 'https://maps.gstatic.com/mapfiles/place_api/i
 'name': 'FleetPride',
 'opening_hours': {'open_now': True},
```

6

6

WEB SCRAPING

- Objective: Use GCP Places API with python to gather valuable data from public reviews on FleetPride locations

- Procedure
  - Imported required libraries
  - Define googlemaps object
  - Confirm retrieval for one location
  - **Identify fields of interest**
  - Loop over FleetPride data locations that are in the excel file
  - Concatenate the dataframes to a final dataset & export

Note: the detailed reviews were limited to 5 results by CGP, so unfortunately this was omitted from the data set.

```
import googlemaps
import pandas as pd
gmaps = googlemaps.Client(key=APIkey)

place_name = 'Fleetpride' + ' ' + 'AL Decatur'
places_result = gmaps.places(place_name)
```

```
place_id = places_result['results'][0]['place_id']
address = places_result['results'][0]['formatted_address']
rating = places_result['results'][0]['rating']
price_level = places_result['results'][0]['price_level']
review_count = places_result['results'][0]['user_ratings_total']

loc = [place_id, address, price_level, rating, review_count]
loc
```

```
['ChIJ8dRAsxGFYogRGOTNT0AsFJk',
 '1101 McEntire Ln NW, Decatur, AL 35601, United States',
 2,
 4.8,
 24]
```

7

7

WEB SCRAPING

- Procedure
  - Imported required libraries
  - Defined googlemaps object
  - Confirm retrieval for one location
  - Identify fields of interest
  - **Loop over FleetPride data locations that are in the excel file**

```
xls = pd.ExcelFile('C:\\Users\\Belgau\\Desktop\\FleetPride_data.xlsm')
df = pd.read_excel(xls, 'data')

scrapped_data = []

for location in df['locations']:
    location = 'FleetPride ' + location
    places_result = gmaps.places(location)

    business_status = places_result['results'][0]['business_status']
    address = places_result['results'][0]['formatted_address']
    rating = places_result['results'][0]['rating']
    try:
        price_level = places_result['results'][0]['price_level']
    except:
        price_level = ''

    review_count = places_result['results'][0]['user_ratings_total']

    scrapped_data.append({
        'business_status': business_status,
        'address': address,
        'rating': rating,
        'price_level': price_level,
        'review_count': review_count})

df_scrap = pd.DataFrame(scrapped_data)
df_scrap
```

	business_status	address	rating	price_level	review_count
0	OPERATIONAL	2403 21st St N, Birmingham, AL 35234, United S...	4.7	2	60
1	OPERATIONAL	1101 McEntire Ln NW, Decatur, AL 35601, United...	4.8	2	24
2	OPERATIONAL	2308 N Range St, Dothan, AL 36303, United States	4.8	2	28
3	OPERATIONAL	5245 Halls Mill Rd, Mobile, AL 36619, United S...	4.5	2	48
4	OPERATIONAL	5947 Old Montgomery Hwy, Tuscaloosa, AL 35405,...	4.7	2	26
...	...	...	...	...	...
106	OPERATIONAL	1790 Velp Ave, Green Bay, WI 54303, United States	4.6	2	22
107	OPERATIONAL	5201 Heffron Ct Bldg B, Stevens Point, WI 5448...	4.7	2	26
108	OPERATIONAL	490 Ragland Rd, Beckley, WV 25801, United States	4.9	2	8
109	OPERATIONAL	60 Columbia Blvd, Clarksburg, WV 26301, United...	4.3	2	36
110	OPERATIONAL	3204 MacCorkle Ave SW, South Charleston, WV 25...	4.4	2	23

111 rows × 5 columns

8

8

WEB SCRAPING

- Procedure
  - Imported required libraries
  - Defined googlemaps object
  - Confirm retrieval for one location
  - Identify fields of interest
  - Loop over FleetPride data locations that are in the excel file
  - **Concatenate the dataframes to a final dataset & export**

```
ds = pd.concat([df, df_scrap], axis=1)
ds.to_csv('FP_dataset.csv')
ds.head()
```

	locations	service_center	business_status	address	rating	price_level	review_count
0	AL Birmingham	1.0	OPERATIONAL	2403 21st St N, Birmingham, AL 35234, United S...	4.7	2	60
1	AL Decatur	NaN	OPERATIONAL	1101 McEntire Ln NW, Decatur, AL 35601, United...	4.8	2	24
2	AL Dothan	NaN	OPERATIONAL	2308 N Range St, Dothan, AL 36303, United States	4.8	2	28
3	AL Mobile	1.0	OPERATIONAL	5245 Halls Mill Rd, Mobile, AL 36619, United S...	4.5	2	48
4	AL Tuscaloosa	NaN	OPERATIONAL	5947 Old Montgomery Hwy, Tuscaloosa, AL 35405,...	4.7	2	26

9

9

EDA WITH PANDAS

10

10

EDA WITH PANDAS

- Import libraries and view data

```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
```

```
filepath = 'C:\\Users\\Belgau\\Desktop\\FP_dataset.csv'
df = pd.read_csv (filepath)
df = df.iloc[: , 1:] #drop first col (numbered index from web scrap)
df.head()
```

	locations	service_center	business_status	address	rating	price_level	review_count
0	AL Birmingham	1.0	OPERATIONAL	2403 21st St N, Birmingham, AL 35234, United S...	4.7	2.0	60
1	AL Decatur	NaN	OPERATIONAL	1101 McEntire Ln NW, Decatur, AL 35601, United...	4.8	2.0	24
2	AL Dothan	NaN	OPERATIONAL	2308 N Range St, Dothan, AL 36303, United States	4.8	2.0	28
3	AL Mobile	1.0	OPERATIONAL	5245 Halls Mill Rd, Mobile, AL 36619, United S...	4.5	2.0	48
4	AL Tuscaloosa	NaN	OPERATIONAL	5947 Old Montgomery Hwy, Tuscaloosa, AL 35405,...	4.7	2.0	26

11

FleetPride TRUCK & TRAILER PARTS

11

EDA WITH PANDAS

- Meta data & preliminary statistics

```
In [3]: df.shape
Out[3]: (111, 7)
```

```
In [4]: for i,col in enumerate(df.col
            print(i,col)

0 locations
1 service_center
2 business_status
3 address
4 rating
5 price_level
6 review_count
```

```
In [5]: df.dtypes
Out[5]: locations          object
        service_center     float64
        business_status     object
        address             object
        rating             float64
        price_level        float64
        review_count         int64
        dtype: object
```

```
df.describe(include='all')
```

Found bad data

	locations	service_center	business_status	address	rating	price_level	review_count
count	111	55.0	111	111	111.000000	97.0	111.000000
unique	111	NaN	2	109	NaN	NaN	NaN
top	MN St. Cloud	NaN	OPERATIONAL	17000 S Main St, Gardena, CA 90248, United States	NaN	NaN	NaN
freq	1	NaN	110	2	NaN	NaN	NaN
mean	NaN	1.0	NaN	NaN	4.441441	2.0	39.441441
std	NaN	0.0	NaN	NaN	0.305545	0.0	26.934832
min	NaN	1.0	NaN	NaN	3.600000	2.0	2.000000
25%	NaN	1.0	NaN	NaN	4.200000	2.0	22.500000
50%	NaN	1.0	NaN	NaN	4.400000	2.0	34.000000
75%	NaN	1.0	NaN	NaN	4.700000	2.0	51.500000
max	NaN	1.0	NaN	NaN	5.000000	2.0	165.000000

12

FleetPride TRUCK & TRAILER PARTS

12

EDA WITH PANDAS

- Found some bad data & corrected the web scraper

```
In [7]: #there were some duplicate addresses
        pd.concat(g for _, g in df.groupby("address") if len(g) > 1)

        #Looks like the scrapper or data cleaning process messed up 2 of these 4 entries.
        #Note: Longview has zero google reviews...which may have made it challenging for the API to pick up?
```

Out[7]:

	locations	service_center	business_status	address	rating	price_level	review_count
13	CA Ontario	NaN	OPERATIONAL	17000 S Main St, Gardena, CA 90248, United States	4.2	2.0	37
46	ME Scarborough	1.0	OPERATIONAL	17000 S Main St, Gardena, CA 90248, United States	4.2	2.0	37
97	TX Kilgore	NaN	OPERATIONAL	502 TX-135, Kilgore, TX 75662, United States	4.4	2.0	63
98	TX Longview	1.0	OPERATIONAL	502 TX-135, Kilgore, TX 75662, United States	4.4	2.0	63

- Reupload new data

13

FleetPride TRUCK & TRAILER PARTS

13

EDA WITH PANDAS

- Separate the STATE from LOCATIONS into a new col for grouping

```
In [9]: #separate state from location
        df['state'] = df['locations'].str[:2]
        df['locations'] = df['locations'].str[3:]
        df.head()
```

Out[9]:

	locations	service_center	business_status	address	rating	price_level	review_count	state
0	Birmingham	1.0	OPERATIONAL	2403 21st St N, Birmingham, AL 35234, United S...	4.7	2.0	60	AL
1	Decatur	NaN	OPERATIONAL	1101 McEntire Ln NW, Decatur, AL 35601, United...	4.8	2.0	24	AL
2	Dothan	NaN	OPERATIONAL	2308 N Range St, Dothan, AL 36303, United States	4.8	2.0	28	AL
3	Mobile	1.0	OPERATIONAL	5245 Halls Mill Rd, Mobile, AL 36619, United S...	4.5	2.0	48	AL
4	Tuscaloosa	NaN	OPERATIONAL	5947 Old Montgomery Hwy, Tuscaloosa, AL 35405,...	4.7	2.0	26	AL

14

FleetPride TRUCK & TRAILER PARTS

14

EDA WITH PANDAS

- Dealing with missing & null values

```
In [131]: #remove BLANKS and NULLS for plotting 'rating' vs 'review_count'
          df_noblanks = df.replace("", "NaN")
          df_noblanks.dropna(subset = ['rating', 'review_count'], inplace=True) #Drop NULLS
          df.shape, df_noblanks.shape #one row was removed

          #df[df.locations.str.contains('longview',case=False)] #this row was longview, as no longer in df

Out[131]: ((110, 8), (109, 8))
```

15

FleetPride TRUCK & TRAILER PARTS

15

EDA WITH PANDAS

- Remove locations no longer in operation, according to Google

```
In [132]: df[df.business_status.str.contains('OPERATIONAL')== False]
```

Out[132]:

locations	service_center	business_status	address	rating	price_level	review_count	state

```
In [133]: df = df[df.business_status.str.contains('OPERATIONAL')== True]
          df.shape
          #Drop Milford from dataset

Out[133]: (110, 8)
```

16

FleetPride TRUCK & TRAILER PARTS

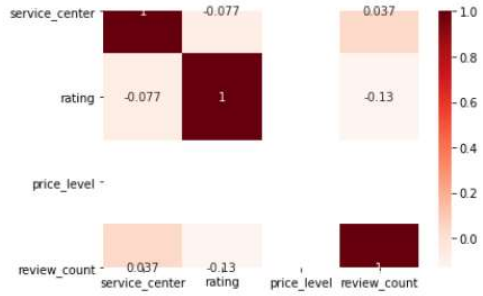
16

EDA WITH PANDAS

- Examine distributions & correlations of numerical data

```
corrMatrix = df.corr() #find corr of all var relationships (only works with float and int64 dtypes)
sns.heatmap(corrMatrix, annot=True, cmap=plt.cm.Reds) #-1 is weak, 1 is strong
plt.show()

#weak corr between rating & review_count
```



17

17

EDA WITH PANDAS

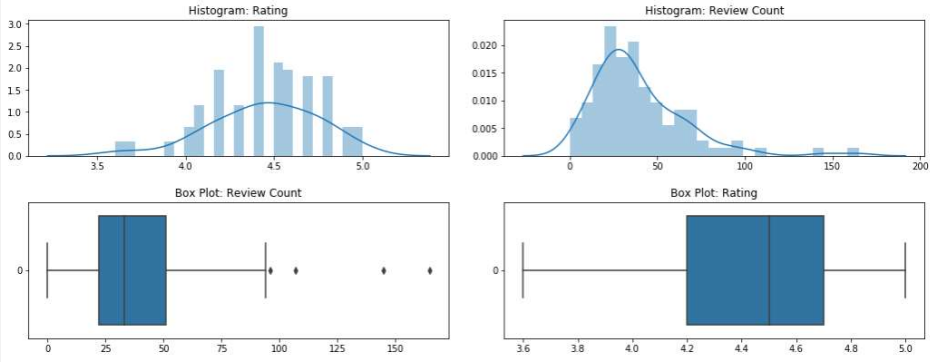
- How do the distributions look?

```
# How do the distributions look?
fig, axes = plt.subplots(2, 2, figsize=(15,6))
plt.tight_layout(pad=3.0)

sns.distplot(df['rating'], bins=25, ax=axes[0,0]).set(title='Histog
sns.distplot( df['review_count'],  bins=25, ax=axes[0,1]).set(title

sns.boxplot(ax=axes[1,0], data=df['review_count'], orient="h").set(
sns.boxplot(ax=axes[1,1], data=df['rating'], orient="h").set(title=

#min, max, median, IQR
#some outliers potentially
```



18

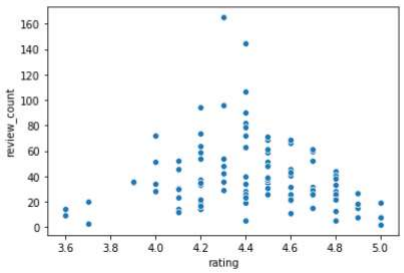
18

EDA WITH PANDAS

- Let's examine the relationship between review_count and rating more since the heat map showed some corr

```
In [140]: sns.scatterplot(x='rating', y='review_count', data=df)
          #there might be an interesting relationship here
          #use spline to smooth this out, but need unique x-axis values in array format
          #create an average value for each unique 'rating'
```

```
Out[140]: <matplotlib.axes._subplots.AxesSubplot at 0x1f913ff8448>
```



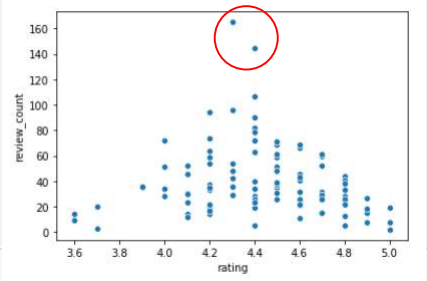
19

FleetPride TRUCK & TRAILER PARTS

19

EDA WITH PANDAS

- Let's examine the relationship between review_count and rating more…




```
# What stores are top two for review_count?
# they appear to be performing well
df.sort_values(['review_count'], ascending=False, inplace=False).head(2)
```

	locations	service_center	business_status	address	rating	price_level	review_count	state
22	Tampa	1.0	OPERATIONAL	3517 N 40th St, Tampa, FL 33605, United States	4.3	2.0	165	FL
9	Phoenix	0.0	OPERATIONAL	1801 N Black Canyon Hwy, Phoenix, AZ 85009, Un...	4.4	2.0	145	AZ

```
# How do these locations compare to the set?
df['review_count'].mean(), df['rating'].mean()

#rating a bit lower, but within reasonable margin (not likely to be underperforming)
```

```
(38.93636363636364, 4.445871559633027)
```

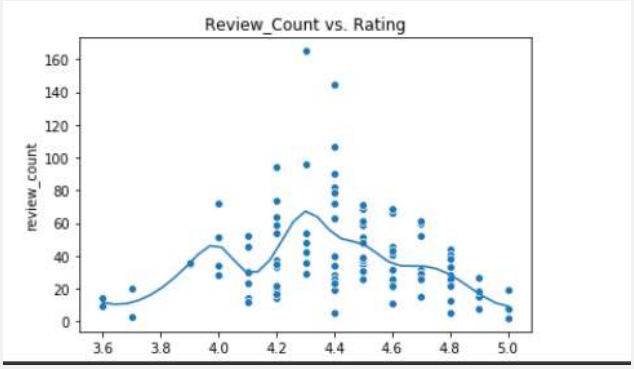
20

FleetPride TRUCK & TRAILER PARTS

20

EDA WITH PANDAS

- Smooth out this trend




```
sns.scatterplot(x='rating', y='review_count', data=df)

from scipy.interpolate._bsplines import make_interp_spline

x = df2['rating'].to_numpy()
y = df2['review_count'].astype(float).to_numpy()

X_Y_Spline = make_interp_spline(x, y)

# Returns evenly spaced numbers over a specified interval.
X_ = np.linspace(x.min(), x.max(), 35)
Y_ = X_Y_Spline(X_)

sns.lineplot(x=X_, y=Y_)

plt.title("Review_Count vs. Rating")
plt.xlabel("rating")
plt.ylabel("review_count")
```

- Next steps: can we fit an equation to this curve to identify which stores are performing above/below avg?

21

FleetPride TRUCK & TRAILER PARTS

21

EDA WITH PANDAS

- Group by State
- Develop new metrics

```
#aggregate functions by state
df_state_loccount = df.groupby(['state']).size() #finds how many locations in ea state
df_state = df.groupby(['state']).sum()

df_state = pd.concat([df_state,df_state_loccount], 1)

#formatting
df_state.drop(['price_level'], 1, inplace=True)
df_state.rename(columns={df_state.columns[3]: "num_of_locs"}, inplace = True)
df_state.rename(columns={'service_center': "num_of_servicecenters"}, inplace = True)
df_state.reset_index(level=0, inplace=True) #turn state index into a new col

#additional metrics
df_state['reviews_per_loc'] = df_state['review_count'] / df_state['num_of_locs']
df_state['avg_rating_loc_weighted'] = df_state['rating'] / df_state['num_of_locs'] #does not account for total reviews
df_state.drop(['rating'], 1, inplace=True)
df_state['percent_servicecenter'] = df_state['num_of_servicecenters'] / df_state['num_of_locs'] *100

df_state.head()
```

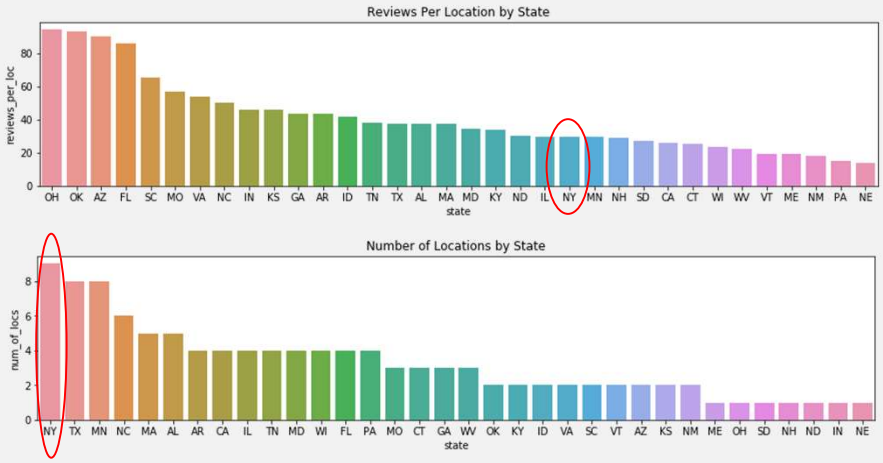
	state	num_of_servicecenters	review_count	num_of_locs	reviews_per_loc	avg_rating_loc_weighted	percent_servicecenter
0	AL	2.0	186	5	37.2	4.700	40.0
1	AR	3.0	174	4	43.5	4.300	75.0
2	AZ	0.0	180	2	90.0	4.450	0.0
3	CA	0.0	104	4	26.0	4.275	0.0
4	CT	0.0	75	3	25.0	4.200	0.0

22

FleetPride TRUCK & TRAILER PARTS

22

EDA WITH PANDAS



23

FleetPride TRUCK & TRAILER PARTS

23

EDA WITH PANDAS

- Why does NY have so many locations, but low review/location ratio?
- Is it being suppressed by one location?

```
#why does NY have so many locations but low review/loc ratio (low num of reviews)?
df_NY = df[df['state'].str.contains('NY')]
df_NY.loc[(df_NY['locations'] == 'Pleasant Valley')]

#maybe it's being brought down by the Pleasant Valley location?
```

	locations	service_center	business_status	address	rating	price_level	review_count	state
73	Pleasant Valley	1.0	OPERATIONAL	1931 US-44, Pleasant Valley, NY 12569, United ...	5.0	NaN	2	NY

- How do these stats change with this loc removed?

```
tot_review_avg: 38.93636363636364
NY_review_avg: 29.666666666666668
```

→

```
tot_review_avg: 38.93636363636364
NY_review_avg: 33.125
```

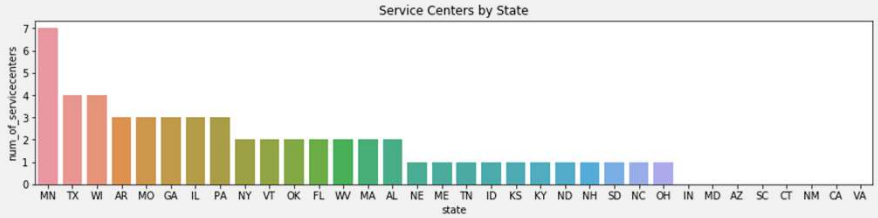
- Maybe the number of service centers is correlated to review count?

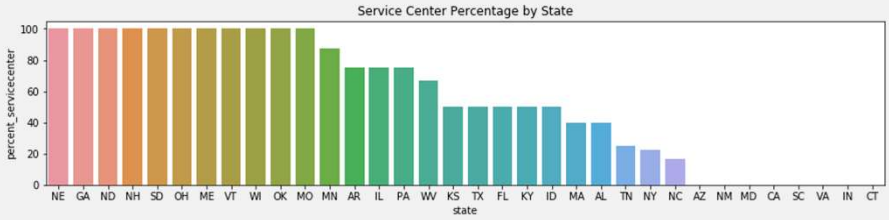
24

FleetPride TRUCK & TRAILER PARTS

24

EDA WITH PANDAS





25

EDA WITH PANDAS

- Does being a service center impact number of reviews or rating?

```
df['service_center'] = df['service_center'].fillna(0) #turn all nulls to zero

df_ss_count = df.groupby(['service_center']).size() #count how many locations are service centers
df_ss1 = df.groupby(['service_center']).mean()

df_ss1 = pd.concat([
    df_ss1,
    df_ss_count], 1)

df_ss1.rename(columns={df_ss1.columns[3]: "num_of_locs"}, inplace = True)
df_ss1.rename(columns={df_ss1.columns[2]: "avg_num_of_reviews"}, inplace = True)
df_ss1.reset_index(level=0, inplace=True) #turn service_center index into a new col (boolean)
df_ss1.drop(['price_level'], 1, inplace=True)

df_ss1
```

	service_center	rating	avg_num_of_reviews	num_of_locs
0	0.0	4.469091	37.927273	55
1	1.0	4.422222	39.945455	55

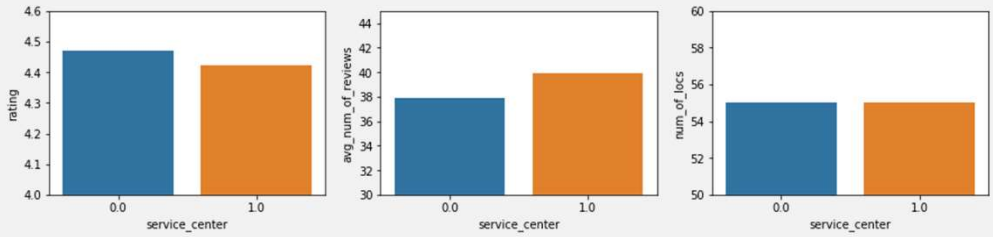
EDA WITH PANDAS

- Does being a service center impact number of reviews or rating?

```
fig, axes = plt.subplots(1, 3, figsize=(15,3))
sns.barplot(ax=axes[0], x='service_center', y='rating', data=df_ss1).set(ylim=(4,4.6))
sns.barplot(ax=axes[1], x='service_center', y='avg_num_of_reviews', data=df_ss1).set(ylim=(30,45))
sns.barplot(ax=axes[2], x='service_center', y='num_of_locs', data=df_ss1).set(ylim=(50,60))

#these figures aren't very pretty, but communicates that there isn't a big difference between service_center qualifier
#the ylim scaling impacts interpretation, so we would probably want to just show these as numerics rather than figures.

#can do ANOVA test (scipy.stats.f_oneway) to analyze f & p values to confirm no difference of rating & review_count
#if p>0.05 then there is not a sig diff
```



We can do an ANOVA test, to provide statistical evidence

27

27