Snapchat Political Ads

This project uses political ads data from Snapchat, a popular social media app. Interesting questions to consider include:

- What are the most prevalent organizations, advertisers, and ballot candidates in the data? Do you recognize any?
- What are the characteristics of ads with a large reach, i.e., many views? What may a campaign consider when maximizing an ad's reach?
- What are the characteristics of ads with a smaller reach, i.e., less views? Aside from funding constraints, why might a campaign want to produce an ad with a smaller but more targeted reach?
- What are the characteristics of the most expensive ads? If a campaign is limited on advertising funds, what type of ad may the campaign consider?
- What groups or regions are targeted frequently? (For example, for single-gender campaigns, are men or women targeted more frequently?) What groups or regions are targeted less frequently? Why? Does this depend on the type of campaign?
- Have the characteristics of ads changed over time (e.g. over the past year)?
- When is the most common local time of day for an ad's start date? What about the most common day of week? (Make sure to account for time zones for both questions.)

Getting the Data

The data and its corresponding data dictionary is downloadable [here (https://www.snap.com/en-US/political-ads/)](https://www.snap.com/en-US/political-ads/). Download both the 2018 CSV and the 2019 CSV.

The CSVs have the same filename; rename the CSVs as needed.

Note that the CSVs have the exact same columns and the exact same data dictionaries (`readme.txt`).

Cleaning and EDA

- Concatenate the 2018 CSV and the 2019 CSV into one DataFrame so that we have data from both years.
- Clean the data.
  - Convert `StartDate` and `EndDate` into datetime. Make sure the datetimes are in the correct time zone. You can use whatever timezone (e.g. UTC) you want as long as you are consistent. However, if you want to answer a question like "When is the most common local time of day for an ad's start date," you will need to convert timezones as needed. See Hint 2 below for more information.
- Understand the data in ways relevant to your question using univariate and bivariate analysis of the data as well as aggregations.

Hint 1: What is the "Z" at the end of each timestamp?

Hint 2: `pd.to_datetime` *will be useful here.* `Series.dt.tz_convert` *will be useful if a change in time zone is needed.*

Tip: To visualize geospatial data, consider [Folium (https://python-visualization.github.io/folium/)](https://python-visualization.github.io/folium/) or another geospatial plotting library.

Assessment of Missingness

Many columns which have `NaN` values may not actually have missing data. How come? In some cases, a null or empty value corresponds to an actual, meaningful value. For example, `readme.txt` states the following about `Gender` :

> Gender - Gender targeting criteria used in the Ad. If empty, then it is targeting all genders

In this scenario, an empty `Gender` value (which is read in as `NaN` in pandas) corresponds to "all genders".

- Refer to the data dictionary to determine which columns do **not** belong to the scenario above. Assess the missingness of one of these columns.

Hypothesis Test / Permutation Test

Find a hypothesis test or permutation test to perform. You can use the questions at the top of the notebook for

Summary of Findings

Introduction

We are given two datasets from 2018 and 2019 that contain information on political ads that have been shown on Snapchat's advertising platform. The datasets share the same columns and include information on the ad's creator, cost/spending, impressions (number of views), duration, and target audience. Columns pertaining to the target audience tell us who and where the creator is directing the ad to. After concatenating the two datasets, we are left with 4268 rows and 34 columns.

The 'CreativeProperties' values give us URLs that provide us with the organization's mission statement, allowing us to categorize each ad as 'Liberal' or 'Conservative'. Some ads can be categorized using the values from the 'OrganizationName' column since they are clearly liberal or conservative (i.e. 'North Dakota Republican Party' and 'Wyoming Democratic Party'). We created a new column with these categories that we will use to answer our question.

Cleaning and EDA

In order to answer the question "From the Americans who are interested in the news, are they more likely to see conservative ads or liberal ads on snapchat?", we needed to manipulate the dataset that fit the requirements:

- Ads in United States
- Ads with interest in news
- Categorizing organizations into either Liberal or Conservative

We first grabbed all the rows with CountryCode equal to United States. We then experimented grabbing ads with interest in news by looking for the keyword FOX in that column. We then mapped each PayingAdvertiserName to either Liberal (0) and Conservative (1) by doing individual research on each of the 55 unique organizations. This was applied to a new column - Affiliation. Without much suprise, we saw that all values were conservative and this was explained through FOX News being biased. Learning from this mistake, we instead searched for the keyword NEWS in the same column - Interests - in order to expand our dataset.

We were suprised by the fact that there were nearly 3.5x amount of Liberal ads than Conservative ads. We then looked at how affiliation affects gender, spending, and impressions. We made sure to replace categorical values in gender with a numerical value in order to apply sum and mean. Plotting each of these columns grouped by affiliation showed us quite contrasting statistics. Conservative organizations spent an average of 646.93 per ad while Liberal organizations spent an average of 529.82. This might explain why conservative ads got more impressions on average which was 258492 compared to 171585 for liberal ads.

Assessment of Missingness

We believe that the column 'Regions (Included)' may be NMAR. Organizations, especially those working for political candidates, may not want to share where they are targeting their ads in order to hide political strategy from opposing candidates. Columns that are related to location such as 'Electoral Districts (Included)', 'Radius Targeting (Included)', Metros (Included)', 'Postal Codes (Included)', 'Location Categories (Included)', as well as their '(Excluded)' counterparts may also be NMAR for the same reason.
https://www.statista.com/statistics/814300/snapchat-users-in-the-united-states-by-age/
(https://www.statista.com/statistics/814300/snapchat-users-in-the-united-states-by-age/)

We focused on the "CreativeProperties" column because it had a non-trivial amount of nulls. We then sought a column where "CreativeProperties" missingness depended on, and one that it didn't depend on. For the permutation test, we used a significance level of 0.05. By using permutation test on "CreativeProperties" and "Spend" using the difference in means statistics, we were able to retrieve a p-value of 0.24 > 0.05, which allowed us to accept the hypothesis that the missingness is MCAR. This makes sense because there would be no correlation between a URL link and amount spent in an ad. We then applied a permutation test on "CreativeProperties" and "CreativeUrl" using total variance distance. We used this statistic because we wanted to find the total difference in categorical values. We discovered a p-value of 0 which proved that the observed scenario was super unique - signifiying that the missigness depended on "CreativeUrl". This exemplifies MAR.

Hypothesis Test

We stated the null and alternative hypothesis as follows:

Null Hypothesis: Snapchat ads targeted towards news watchers are equally likely to come from liberal and conservative organizations. (The distribution of liberal and conservative ads are equal)

Alternative Hypothesis: Snapchat ads targeted towards news watchers are more likely to come from liberal organizations than conservative organizations. (The distribution of liberal and conservative ads are not equal)

First, we calculated the observed test statistic. We used a 0.5 significance level. Using the total number of liberal ads as our test statistic, our calculated observed test statistic was 205. After, we simulated the data under the null hypothesis. Given a 50-50 chance, we randomly selected 'Liberal' or 'Conservative' 263 times (the length of our cleaned dataset). We simulated the data 1000 times to make sure out outcome did not happen by chance. Per each simulation, we compared the number of times the simulated test statistic was greater than or equal to the observed test statistic. We then took the average of this value to calculate our p-value which was 0.0. Since our p-value was less than the siginificance level, we reject the null hypothesis. It is valid to consider the alternative hypothesis.

Snapchat ads targeted towards news watchers are more likely to come from liberal organizations than conservative organizations.

A possible reason that Snapchat users interested in news receive more advertisements from liberal organizations compared to conservative organizations is that the advertising companies are trying to target young voters/adults. This make sense over half (53%) of Snapchat users are aged 15-25 and most young voters are liberal/democratic.

Sources: https://www.statista.com/statistics/814300/snapchat-users-in-the-united-states-by-age/ (https://www.statista.com/statistics/814300/snapchat-users-in-the-united-states-by-age/)

Code

In [135]:
```
import matplotlib.pyplot as plt
import numpy as np
import os
import pandas as pd
import seaborn as sns
import datetime
from IPython.display import HTML

%matplotlib inline
%config InlineBackend.figure_format = 'retina'  # Higher resolution figu
res
```

In [34]:
```
pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)
pd.set_option('display.width', 1000)
```

Cleaning and EDA

In [140]:
```
#concatenate two csv files
ads_2018 = pd.read_csv('2018.csv')
ads_2019 = pd.read_csv('2019.csv')
ads = pd.concat([ads_2018, ads_2019])
ads.head()
```

Out[140]:

	ADID	CreativeUrl	Currency Code	S
0	29cbbf5621975dbd4ffd3826f22e781ca4f41fe4cd61c5...	https://www.snap.com/political-ads/asset/d5926...	USD	
1	1ee35b50c5d194f4bf23196ad3645d16951439d9bfbbd2...	https://www.snap.com/political-ads/asset/fbee4...	EUR	
2	fabfccf0dfe9373fabe6723481ff2d0f3ce36b72d7a64e...	https://www.snap.com/political-ads/asset/c2d3a...	USD	
3	1c3df8a88ecfd08123d59e8378460d4973e9665e626550...	https://www.snap.com/political-ads/asset/953f2...	USD	
4	998ce79da3f259c96b928dae99b1fef80b737b25db6a18...	https://www.snap.com/political-ads/asset/e1b1d...	USD	

First lets clean the StartDate and EndDate

In [141]:
```
#helper function that converts string into datetime
def date(x):
    if pd.isnull(x):
        return np.NaN
    else:
        x=str(x)
        x = x.replace('Z','')
        dat_time = datetime.datetime.strptime(x,'%Y/%m/%d %H:%M:%S')
    return dat_time
```

In [142]:
```
#apply helper function to both startdate and enddate
ads['StartDate']= ads['StartDate'].apply(date)
ads['EndDate'] = ads['EndDate'].apply(date)
ads.head()
```

Out[142]:

	ADID	CreativeUrl	Currency Code	S
0	29cbbf5621975dbd4ffd3826f22e781ca4f41fe4cd61c5...	https://www.snap.com/political-ads/asset/d5926...	USD	
1	1ee35b50c5d194f4bf23196ad3645d16951439d9bfbbd2...	https://www.snap.com/political-ads/asset/fbee4...	EUR	
2	fabfccf0dfe9373fabe6723481ff2d0f3ce36b72d7a64e...	https://www.snap.com/political-ads/asset/c2d3a...	USD	
3	1c3df8a88ecfd08123d59e8378460d4973e9665e626550...	https://www.snap.com/political-ads/asset/953f2...	USD	
4	998ce79da3f259c96b928dae99b1fef80b737b25db6a18...	https://www.snap.com/political-ads/asset/e1b1d...	USD	

We want to answer the question: From the Americans who are interested in the news, are they more likely to see conservative ads or liberal ads on snapchat?

In order to clean the data, we first need the dataset to be filled with ads only from the United States

In [143]:
```
#only grab the ads that are in the United States
american_ads = ads[ads['CountryCode']=='united states']
american_ads.head()
```

Out[143]:

	ADID	CreativeUrl	Currency Code	S
0	29cbbf5621975dbd4ffd3826f22e781ca4f41fe4cd61c5...	https://www.snap.com/political-ads/asset/d5926...	USD	
2	fabfccf0dfe9373fabe6723481ff2d0f3ce36b72d7a64e...	https://www.snap.com/political-ads/asset/c2d3a...	USD	
3	1c3df8a88ecfd08123d59e8378460d4973e9665e626550...	https://www.snap.com/political-ads/asset/953f2...	USD	
4	998ce79da3f259c96b928dae99b1fef80b737b25db6a18...	https://www.snap.com/political-ads/asset/e1b1d...	USD	
6	69ce7aa4805de5ff99c94104b54fa68f3c3514537ec78b...	https://www.snap.com/political-ads/asset/2e4d6...	USD	

Seeing that one of the most common news channel is FOX News, lets try grabbing all the ads that target FOX News viewers.

In [144]:
```
#only grab rows whose interests contains FOX
fox_ads = american_ads[
    american_ads['Interests'].apply(lambda x: 'fox' in str(x).lower())
]
```

In order to distinguish which are Liberal and which Conservative, we look at the column 'PayingAdvertiserName' as it gives us the name of the entity that provides the funds for the Ad.

There are 55 unique entities that providing the funds. In order to label them as either 'Liberal' or 'Conservative' we went through all 55 entities online and labelled them according to their characteristics. For example, 'Trump MAGA Committee' is clearly 'Conservative' while 'Pete for America' is 'Liberal'. For others that are more unclear, we used 'CreativeProperties' column to find their mission statement.

In [145]:
```
#0 for Liberal, 1 for Conservative
group = {
    'North Dakota Republican Party':1,
    'liberteeshop':1,
    'Turning Point USA':1,
    'Proud Right Winger':1,
    'NRSC':1,
    'Republican National Committee':1,
    'Your Trump Shop':1,
    'Trump MAGA Committee':1,
    'Print Mine':1,
    'Stratos Developments Ltd.':1,
    'McCain Institute':1,
    'Our Lives Our Vote':0,
    'Change Now':0,
    'Wyoming Democratic Party':0,
    'UnRestrict Minnesota':0,
    'Knock The Vote':0,
    'Commmitte to Elect Kyle Cooper':0,
    'Melissa Moore4MNHouse Committee':1,
    'Murtaugh for Congress':1,
    'Planned Parenthood':0,
    'Everytown for Gun Safety AF':0,
    'Ammar Campa-Najjar for Congress':0,
    'Pete King for Congress':1,
    'Pete for America':0,
    'Beto for America':1,
    'Brady Campaign':0,
    'Fire45.Club':0,
    'Alliance for Gun Responsibility':0,
    'John Walsh for Colorado':0,
    'Mike Mullin for Virginia':0,
    'Warren for President':0,
    'Friends of Madeline Singas':0,
    'Make It Legal Florida':0,
    'Lightfoot for Chicago':0,
    'Kamala Harris For The People':0,
    'Congressional Leadership Fund':1,
    'ArcaMax Publishing Inc':1,
    'Only American Pride':1,
    'US Polling Research':0,
    "Ben & Jerry's":0,
    'SELC':0,
    'ACRONYM':0
}
```

In [146]:
```
#assign a new column Affiliation with the organization mapped to either
 conservative or liberal
fox_grouped = fox_ads.assign(Affiliation=fox_ads['PayingAdvertiserName']
.map(group))
fox_grouped = fox_grouped[(fox_grouped['Affiliation'] == 1)|(fox_grouped
['Affiliation'] == 0)]
fox_grouped.head()
```

Out[146]:

	ADID	CreativeUrl	Currency Code
189	ecaff54d5436d20d3a4205fdbf4f94c1efea528408d5df...	https://www.snap.com/political-ads/asset/fd847...	USD
204	b91f944c1dadb20302b97a687ed315af52645014685fe6...	https://www.snap.com/political-ads/asset/cc066...	CAD
528	f3413b23f0161c9998a6f5ebdad37e79ea00ba170838d5...	https://www.snap.com/political-ads/asset/806ff...	USD
618	21dc8fa0923d20bd11ee4afde9366ffc242a70157cb1c2...	https://www.snap.com/political-ads/asset/de229...	USD
685	3c4aae38aaeb3ea7f8c2ad9186cd0f4cc5b6c0aaa960d0...	https://www.snap.com/political-ads/asset/a0d2e...	USD

In [147]:
```
fox_grouped['Affiliation'].value_counts()
```

Out[147]:
```
1.0    40
Name: Affiliation, dtype: int64
```

From above, we can see that all the ads that are targeted towards FOX viewers are conservative (labelled as 1). We became more curious of why Republican organizations targeted FOX News viewers. After some research, we came across an article by the University of Michigan (https://guides.lib.umich.edu/fakenews (https://guides.lib.umich.edu/fakenews)). The article stated that FOX News typically has ideological bias towards the right (conservative).

In [148]:
```
s='''
<img src="sshot.jpeg">
'''
HTML(s)
```

Out[148]:

Since there is bias in fox_grouped dataset, we decided to grab all ads that targeted news viewers and not just FOX viewers. In order to do so, we can look for the keyword 'news' in the 'Interests' column.

In [149]:
```
#only grab rows with interests that contains 'news' in them
news_interest = american_ads[
    american_ads['Interests'].apply(lambda x: 'news' in str(x).lower())
]
news_interest.head()
```

Out[149]:

	ADID	CreativeUrl	Currency Code
3	1c3df8a88ecfd08123d59e8378460d4973e9665e626550...	https://www.snap.com/political-ads/asset/953f2...	USD
9	d36c2984153b2dde3fda24a487a6dbed303e46678932f9...	https://www.snap.com/political-ads/asset/ed571...	USD
11	382c2faa62726c2f915f1d7113e0fced5357518bbdc843...	https://www.snap.com/political-ads/asset/b3d0d...	USD
30	c5a0bd1dc799e2999b3e91db8cb956c767d353687349d5...	https://www.snap.com/political-ads/asset/ff200...	USD
34	48ea31497fb68f28a6fd2b5684ccec65cdb9310bcab311...	https://www.snap.com/political-ads/asset/9dfa5...	USD

We now have all the ads that are American and that target those with interest in news.

In order to distinguish which are Liberal and which Conservative, we look at the column 'PayingAdvertiserName' as it gives us the name of the entity that provides the funds for the Ad.

In [150]:
```
#assign a new column Affiliation with organization mapped to either cons
ervative or liberal
grouped = news_interest.assign(Affiliation=news_interest['PayingAdvertis
erName'].map(group))
#get rid of rows with Affiliation set to null
grouped = grouped[(grouped['Affiliation'] == 1)|(grouped['Affiliation']
== 0)]
grouped.head()
```

Out[150]:

	ADID	CreativeUrl	Currency Code
9	d36c2984153b2dde3fda24a487a6dbed303e46678932f9...	https://www.snap.com/political-ads/asset/ed571...	USD
11	382c2faa62726c2f915f1d7113e0fced5357518bbdc843...	https://www.snap.com/political-ads/asset/b3d0d...	USD
30	c5a0bd1dc799e2999b3e91db8cb956c767d353687349d5...	https://www.snap.com/political-ads/asset/ff200...	USD
34	48ea31497fb68f28a6fd2b5684ccec65cdb9310bcab311...	https://www.snap.com/political-ads/asset/9dfa5...	USD
47	feafead548693134a427f0caa7d223eaae4bb911646d89...	https://www.snap.com/political-ads/asset/10b5b...	USD

We see a lot more Liberal advertisements than Conservative advertisements.

In [237]:
```
grouped['Affiliation'].value_counts()
```

Out[237]:
```
0.0    205
1.0     58
Name: Affiliation, dtype: int64
```

Now that we have the column 'Affiliation' that determines whether it is 1 (Conservative) or 0 (Liberal), lets explore how other columns and values compare when grouped as Conservative and Liberal.

Lets see the age and gender that Conservatives and Liberals target.

In [151]:
```
grouped['Gender'].unique()
```

Out[151]:
```
array(['FEMALE', nan, 'MALE'], dtype=object)
```

We can replace MALE with 0 and FEMALE with 1 in order to easily computer and plot graphs

In [152]:
```
#convert FEMALE and MALE to numericals
grouped['Gender'] = grouped['Gender'].map({'FEMALE':1, 'MALE':0})
```

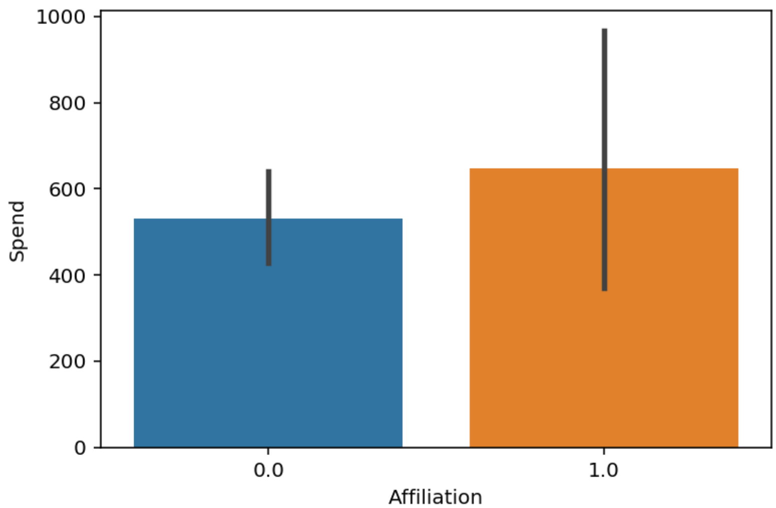
In [153]:
```
grouped.groupby('Affiliation').mean()
```

Out[153]:

Affiliation	Spend	Impressions	Gender	Electoral Districts (Excluded)	Targeting Connection Type	Targeting Carrier (ISP)
0.0	529.829268	171585.795122	1.0	NaN	NaN	NaN
1.0	646.931034	258492.379310	0.5	NaN	NaN	NaN

In [154]:
```
sns.barplot(x='Affiliation', y='Spend', data=grouped)
```

Out[154]: <matplotlib.axes._subplots.AxesSubplot at 0x134019bd0>



From the table and barplot above, we can see that conservative organizations spent more money on ads on average.

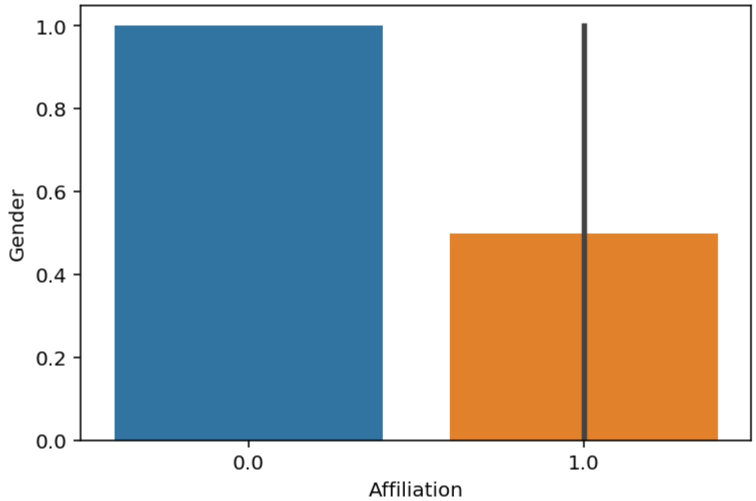
In [155]:
```
#group by Affiliation and aggregate using sum
grouped.groupby('Affiliation').sum()
```

Out[155]:

Affiliation	Spend	Impressions	Gender	Electoral Districts (Excluded)	Targeting Connection Type	Targeting Carrier (ISP)
0.0	108615	35175088	7.0	0.0	0.0	0.0
1.0	37522	14992558	1.0	0.0	0.0	0.0

In [156]:
```
sns.barplot(x='Affiliation', y='Gender', data=grouped)
```

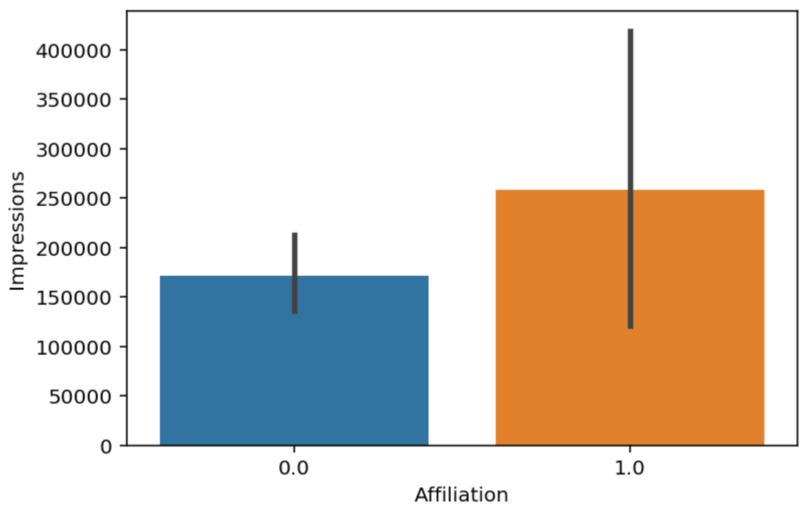
Out[156]: <matplotlib.axes._subplots.AxesSubplot at 0x130b3e410>



However, we can see that more females were targeted by liberal organizations than conservative organizations

In [157]:
```
sns.barplot(x='Affiliation', y='Impressions', data=grouped)
```

Out[157]: <matplotlib.axes._subplots.AxesSubplot at 0x12d3e7390>



In general, conservative ads got more impressions than liberal ads

Assessment of Missingness

The column that we will be observing is the 'CreativeProperties' column

In [195]:
```
def diff_in_means(data, col, group_col):
    """difference in means"""
    return data.groupby(group_col)[col].mean().diff().iloc[-1]
```

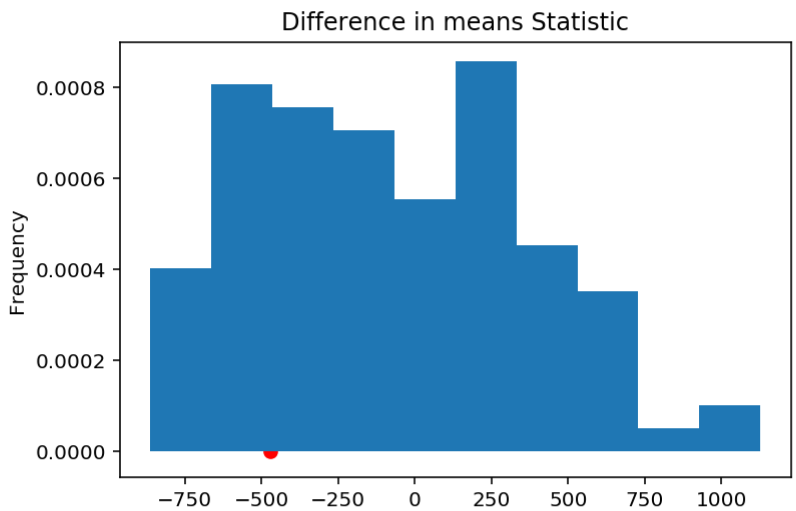
In [214]:
```
def tvd(data, col, group_col):
    """tvd of the distribution of values in col
    bewteen the two groups of group_col. col is
    assumed to be categorical."""

    tvd = (
        data
        .pivot_table(
            index=col,
            columns=group_col,
            aggfunc='size',
            fill_value=0
        )
        .apply(lambda x: x / x.sum())
        .diff(axis=1).iloc[:, -1].abs().sum() / 2
        )

    return tvd
```

In [194]:
```
def permutation_test(data, col, group_col, test_statistic, N=1000):
    """
    Return the distribution of permuted statistics and the observed stat
istic
    resulting from a permutation test.

    :param: data: DataFrame of data observations and the labels for two
 groups.
    :param: col: Column name for the column containing the data.
    :param: group_col: Column name for the column contain the labels for
the two groups.
    :param: test_statistic: The test statistic to apply to the groups (a
function).
    :param: N: The number of times N to run the permutation test.
    """

    # get the observed test statistic
    obs = test_statistic(data, col, group_col)

    # run the permutations
    shuffled_stats = []
    for _ in range(N):

        shuffled = data[group_col].sample(frac=1, replace=False).reset_i
ndex(drop=True)
        with_shuffled = data[[col]].assign(shuffled=shuffled)
        shuffled_stat = test_statistic(with_shuffled, col, 'shuffled')
        shuffled_stats.append(shuffled_stat)

    shuffled_stats = np.array(shuffled_stats)

    return shuffled_stats, obs
```

We perform a perumation test on the 'CreativeProperties' and 'Spend' column using difference in means statistic to see whether the missigness depends on 'Spend'.

In [234]:
```
#apply a permutation test
distr, obs = permutation_test(
        ads.assign(is_null=ads['CreativeProperties'].isnull()),
        'Spend',  'is_null', diff_in_means, N=100)
#create a series and plot the statistics
pd.Series(distr).plot(kind='hist', density=True, title="Difference in me
ans Statistic")
plt.scatter(obs,0,color='red',s=40);
print((distr <= obs).mean())
```

```
0.24
```

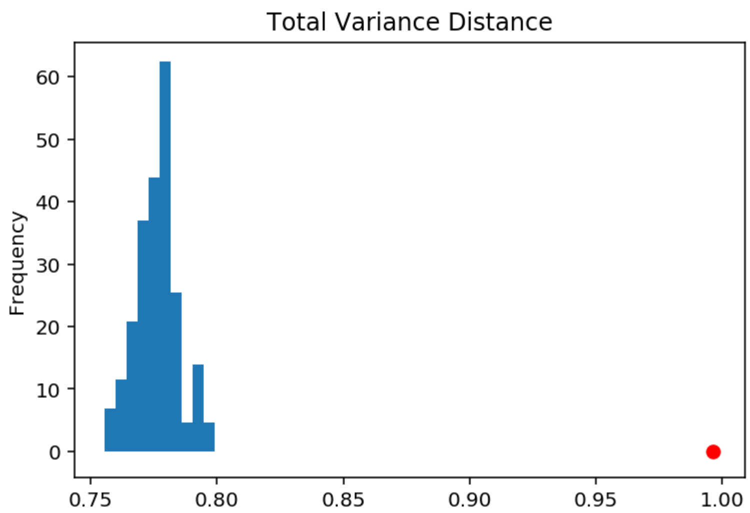


Seeing the table above, we can see a high p-value which signifies that the missingness is not dependent on Spend

We perform another permutation test on 'CreativeUrl' using total variance distance statistic since we are dealing with categorical values.

In [241]:
```
#apply a permutation test
distr, obs = permutation_test(
        ads.assign(is_null=ads['CreativeProperties'].isnull()),
        'CreativeUrl',  'is_null', tvd, N=100)
#create a series and plot the statistics
pd.Series(distr).plot(kind='hist', density=True, title="Total Variance D
istance")
plt.scatter(obs,0,color='red',s=40);
print((distr >= obs).mean())
```

```
0.0
```



We can see that the p-value of 0 proves that the observed statistic is very unique and thus there is a dependency on the 'CreativeUrl' column.

Hypothesis Test

We have our final dataset - Grouped where we can apply the hypothesis test for the following hypothesis:

Null Hypothesis: Snapchat ads targeted towards news watchers are equally likely to come from liberal and conservative organizations. (The distribution of liberal and conservative ads are equal).

Alternative Hypothesis: Snapchat ads targeted towards news watchers are more likely to come from liberal organizations than conservative organizations. (The distribution of liberal and conservative ads are not equal)

We calculated the observed statistics

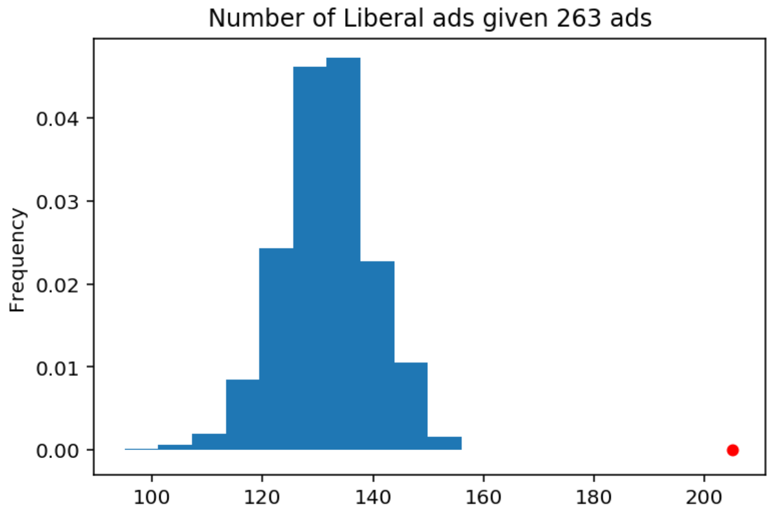
In [177]:
```
#get the sum of all rows with Affiliation set to Liberal
obs = (grouped['Affiliation'] == 0).sum()
grouped.shape[0]
```

Out[177]:
```
263
```

simulate the statistic under the null hypothesis

In [178]:
```
#apply a hypothesis test by using random choice and 50/50 probability
N = 1000
results = []
for _ in range(N):
    simulation = np.random.choice([1,0], p=[0.5,0.5], size = 263)
    results.append((simulation==0).sum())
```

In [179]:
```
#create a series and plot
pd.Series(results).plot(kind='hist', density=True, title='Number of Libe
ral ads given 263 ads');
plt.scatter([obs], [0], s=25, c='r');
```



Our significance level is 0.05

In [181]:
```
#calculate p-value with results and obs
p_value = (results>=obs).sum()/len(results)
p_value
```

Out[181]: 0.0

As we can see from the plot and p-value, there are 0 instances in the 1000 simulations where the statistic was greater or equal to the observed statistic. This allows us to reject the null hypothesis and consider the alternative hypothesis.