#Case Study: Testing Hypothesis

#Hypothesis: Articles about Climate Change are more likely to be published by "Liberal" sources

NOTE: This case study is not complete! We are only using the first part of it to practice Graphic Analytics.

import pandas as pd

import numpy as np

import string

import re

import matplotlib.pyplot as plt

from collections import Counter

#Step 1: Load data into a dataframe

addr1 = "C://Users/mcdaniel/Documents/Temporary D Drive/DSC550/data/articles1.csv"

articles = pd.read\_csv(addr1)

#Step 2: check the dimension of the table/look at the data

print("The dimension of the table is: ", articles.shape)

#Display the data

print(articles.head(5))

#what type of variables are in the table

print("Describe Data")

print(articles.describe())

print("Summarized Data")

print(articles.describe(include=['O']))

#display length of data

print(len(articles))

#display publishers (publications)

print(articles.publication.unique())

#display min, max of years published

print(articles['year'].min())

print(articles['year'].max())

#display how many articles from each year

print(articles['year'].value\_counts())

#Step 3: Create some bar charts to show articles

#display bar chart of articles sorted by Publication Name

ax = articles['publication'].value\_counts().sort\_index().plot(kind='bar', fontsize=14, figsize=(12,10))

ax.set\_title('Article Publication\n', fontsize=20)

ax.set\_xlabel('Publication', fontsize=18)

ax.set\_ylabel('Count', fontsize=18);

plt.show()

#display bar chart of articles sorted by counts

ax = articles['publication'].value\_counts().plot(kind='bar', fontsize=14, figsize=(12,10))

ax.set\_title('Article Count - most to least\n', fontsize=20)

ax.set\_xlabel('Publication', fontsize=18)

ax.set\_ylabel('Count', fontsize=18);

plt.show()

#Step 4: clean text: no punctuation/all lowercase

def clean\_text(article):

clean1 = re.sub(r'['+string.punctuation + '’—”'+']', "", article.lower())

return re.sub(r'\W+', ' ', clean1)

articles['tokenized'] = articles['content'].map(lambda x: clean\_text(x))

print("clean text: ",articles['tokenized'].head())

#look at mean, min, max article lengths

articles['num\_wds'] = articles['tokenized'].apply(lambda x: len(x.split()))

print("Mean: ",articles['num\_wds'].mean())

print("Min: ",articles['num\_wds'].min())

print("Max: ",articles['num\_wds'].max())

#Step 5: remove articles with no words

len(articles[articles['num\_wds']==0])

articles = articles[articles['num\_wds']>0]

print("new mean: ",articles['num\_wds'].mean())

print("new min: ",articles['num\_wds'].min())

#Step 6: Check for Outliers: show bar graph of outliers

ax=articles['num\_wds'].plot(kind='hist', bins=50, fontsize=14, figsize=(12,10))

ax.set\_title('Article Length in Words\n', fontsize=20)

ax.set\_ylabel('Frequency', fontsize=18)

ax.set\_xlabel('Number of Words', fontsize=18);

plt.show()