Homework Assignment #5

(Visualizing and Analyzing IMDB and Twitter Data)

Submission instructions

- Due to EEE dropbox by 2pm Monday Feb 12th
- Please write up your solution in the form of a Jupyter notebook. For portions of the homework where you are asked to provide comments, put these in markdown cells.
- Upload both
    - your .ipynb notebook (remove any personal information such as your database password!)
    - and a .html file (created from File -> Download as -> HTML in Jupyter) containing all the graphics, comments, etc

The goal of this assignment is to give you practice at generating plots of various types, using Python to grab data from the backend PostgreSQL IMDB and Twitter databases and then doing visual exploration and analysis of various aspects of the data

Get Ready...

Before starting this assignment you should make sure your IMDB and Twitter PostgreSQL databases are up and running and that you can connect to and issue SQL queries to them from Python (in Jupyter). Use Pandas for accessing/storing data once you bring it into Python from the database and use Seaborn for generating plots. In your plots please label the x-axis (and the y-axis if it makes sense to do so). You should import all the same packages as in the Jupyter notebook that we worked with in class this week.

Go...!

Part 1: How long do people live?

- create a dataframe called names by reading in the namebasics table from your postgreSQL IMDB database for individuals who died after 1990 (1991 onwards). Drop any rows in the dataframe that contain NaNs.
- define a variable called "age" for the age at death (define it as deathyear - birthyear) of these individuals and add it as a column to your dataframe
- print out the number of unique values, the min, max, median, standard deviation for “age”
- generate a plot of age values using both a histogram (with K=20 bins) and a kernel density plot and write a sentence or two commenting on what you see in the plot.

- now regenerate the histogram and turn off the kernel density part of the plot. Look at the values on the y-axis in both plots. Explain why you think they are different (be as precise as you can in your explanation).
- now regenerate a histogram with 100 bins. You should see a bunch of "spikes". See if you can explain why they show up. Is it possible that they are "real" in the sense that people are much more likely to die at these ages than at ages that are slightly above or below these ages? or are they just artifacts of the way the bins were selected by the function? Explain how you went about answering this question (you may need to do a little experimentation and digging). The value_counts() function in Pandas may be helpful

Part 2: Age Distributions for People born in the 1700's and 1800's

- now create a new dataframe called names2 by loading in from your IMDB database all people born in the years >=1700 or < 1880. Drop any rows with NaNs in your dataframe. You should get 7918 names.
- as you did before, in part 1, add an age variable (defined as before) to the dataframe
- find the rows (people) who lived longer than 120 years. Notice anything unusual here? (there's 1 value that is probably a data entry error - we should really remove it, but lets keep going).
- generate 2 density (kde) plots, both overlaid on the same figure (e.g., one red, one blue, you can use shading if you like), one density for the age variable from Part 1 and one for the age values from Part 2
- it should look like there is a significant difference between the 2 densities. We would ideally like to quantify this. We can do this by performing a statistical test to see if the 2 categorical distributions are the same (the null hypothesis). For simplicity, I recommend that you group the data into 11 bins of ages: [0-10], [11-20], ..., [91-100],[>100]. This will give you 2 sets of counts for the 2 data sets. Use the Chi-square test to test if the 2 distributions are the same. To use Chisquare you can treat the frequencies from Part 1 as the expected frequencies and the frequencies in Part 1 as the observed frequencies. You can use the default value for degrees of freedom. Compute a p-value and write a sentence or two about your conclusion from this analysis. For reference see
https://docs.scipy.org/doc/scipy-0.19.0/reference/generated/scipy.stats.chisquare.html and
https://en.wikipedia.org/wiki/Pearson%27s_chi-squared_test

[Note that binning the values into 11 bins is not really the right way to analyze this data but it's fine for this homework. There are also other ways to compare distributions to see if they are statistically similar, e.g., the Kolmogorov-Smirnoff 2-sided test. If you are interested in the statistical aspect of this problem there is plenty to follow up on and read about!]

Part 3: Do Actors live longer than Writers?

Redo the density plots and statistical calculation from Part 2, but now just focus only on the "names" dataframe (people who died since 1990) and explore whether there are either visual or statistically significant differences of ages (at death) *between any pairs of professions*. Maybe writers live longer than actors? actresses longer than actors? This part of the assignment is a bit

more open-ended - pick at least one pair of professions, analyze it, and write up a brief description of your analysis and conclusions(with 2 density plots, a p-value from a chi-square analysis as before (e.g., with 10 bins)). If you wish to do more than 1 pair of professions, feel free to do this and to also write this up (doing more than 1 pair is optional, just for fun).

Part 4: Plotting Information over Time in Twitter

As in HW4 use the Twitter API and to gather 10,000 tweets and store them in your Twitter database. Your query should gather tweets related to the Super Bowl, e.g., use “Super Bowl” as your query and/or related hastags. The time-period for the tweets should span at least 2 or 3 days before and after the SuperBowl itself (ideally you should run gather this data this week). Generate the following time-series plots in Python:

(1) Number of tweets per hour over a time period that spans a few days before and after the Super Bowl event
(2) Number of “created at” events per month, for users in your database, plotted every month going back to Jan 1 2010.

Add a few comments about what you see in the plots. Be sure to clearly put labels on your plots for the x and y axes.

Part 5: (This is purely Optional, will not be graded, here if you want to explore this) IMDB again: How has Diversity in Movies changed over the Years?

There is a lot of current discussion about diversity in movies, TV, music, etc. Use the IMDB database to extract, for each year from 1910 to 2017, the total number of movies per year and plot a time-series of the number of movies per year. Now extract the total number of actor roles and actress roles per year, where an "actor role" is defined as every time a specific actor appears in a movie (so if John Wayne appeared in 5 different movies in 1950 then we would have a count of 5 for the actor John Wayne in 1950). Calculate the proportion of actress roles (compared to total number of actor + actress roles), per year, and generate a time-series plot of this proportion from 1910 to 2017. Write a few lines of commentary on what you can infer from these plots. Note that this section of the homework will require some manipulation of fields and tables within IMDB - you should do all this from Python, but can use SQL commands to do as much of the work as you like.