Practical Data Science: Assignment 2

Setup

Welcome to the second assignment for this course. You've learned some basic Python and you know how to get data into Python for analysis. The objective of this assignment is to help you become familiar data operations on Pandas data frames. Being able to filter, sort, aggregate, transform and more generally study your data is an essential prerequisite to using your data to build complex models or study statistical phenomena. Please review the material on Pandas before tackling this assignment.

Questions

There are **two** questions in this assignment and you must complete both.

Question 1 (40 points)

Our first data set contains a study made on 128 chimpanzees living in Gombe National Park, Tanzania. Here, different scientists rated the behavioral traits of the chimpanzees across 24 personality traits, which are then grouped into the six broader categories of dominance, neuroticism, openness, conscientiousness, extraversion and agreeableness. The original study involved 52 traits but the data files contain data on 24. The data can be found here: https://osf.io/s7d9d/ Make sure you download these two files:

- gombe_128.csv: We will refer to this as the **summary** file. It has a row for every chimpanzee with the summary values for all personality traits.
- gombe_460.csv: We will refer to this as the **ratings** file. Each row corresponds to the ratings given to a particular chimpanzee by a particular rater, indicated by the *ratercode*

Load the summary file into a Pandas data frame, calling it *sumdf*, and using the relevant Python code, answer the following questions:

a) What is the median impulsiveness score? HINT: Use the PDF file in the site to determine which column has which trait. ***(2 points)***
b) How many chimpanzees have 3 numerical digits in their code? ***(2 points)***
c) What is the average difference in score between the conventional and decisive traits? ***(2 points)***
d) The final six columns (*dominance* to *openness*) correspond to the broad categories. By creating any relevant data frame to support your answer, which of these six broad characteristics is most prominent among female chimpanzees in the sample, and which was most prominent among the males (assume *sex* = 0 corresponds to female)? ***(4 points)***

e) Which pair of the six broad categories are the most correlated in the sample? ***(4 points)***

f) The columns from *dom* to *innov* are the 24 personality traits. You can see what these correspond to by downloading the PDF file in the website. Create a new data frame called *sympdf* according to these rules:
   a. Select only the chimpanzees whose highest scoring personality trait was sympathetic.
   b. Use chimpcode as the index
   c. Only include the columns corresponding to the 24 personality traits mentioned above
   (4 points)

g) Create a new data frame in the long data format. Specifically, it should use the *chimpcode* column as an index, the 24 personality columns should be represented in a *traits* column and their corresponding values should appear in a *score* column like this:

chimpcode	**traits**	**score**
E131	dom	2.428571
E131	sol	3.857143
…	…	…

(4 points)

h) On the basis of the 24 personality traits, which pair of chimpanzees would you say are the most similar to each other? ***(6 points)***

Load the ratings file into a Pandas data frame, calling it *ratdf*, and using the relevant Python code, answer the following questions:

i) How many different raters are there? ***(2 points)***

j) Which rater(s) (identified by their code) rated the most chimpanzees? ***(4 points)***

k) On what year was the oldest chimpanzee of the sample born? ***(2 points)***

l) Create a data frame called *ratcntdf* in which we store how many ratings each chimpanzee received as follows:

chimpcode	**raters**
A100	3
A341	5
…	…

(2 points)

m) Join the *ratcntdf* data frame you just created with the *sumdf* data frame using the *chimpcode* column, keeping all the columns ***(2 points)***

Question 2 (30 points)

You should have downloaded the second dataset that we will use, along with the instructions to this assignment, in the file *yelp_academic_dataset_business.json*. This contains ratings made on the website Yelp (www.yelp.com) for 15,585 businesses in Arizona in the United States. You can load this into a Pandas data frame by executing the code below. Make **sure** you include this snippet in your notebook:

```
import json
yelp_file = 'yelp_academic_dataset_business.json'
yelp_rows = [json.loads(line) for line in open(yelp_file)]
yelpdf = pd.DataFrame(business_records)
```

Here are descriptions for some of the fields:

- business_id: A unique id for the business
- attributes: A dictionary with different attributes
- categories: Different categories for this business
- city: The city where the business is located in
- review_count: How many reviews have been made for this business
- latitude: The latitude of the business
- longitude: The longitude of the business
- stars: The average star rating rounded to half-stars
- full_address: The text address of the business

Ok now for some questions:

a) How many Starbucks are there in this data frame (In case you did not know, Starbucks is the name of a popular coffee shop chain)? ***(2 points)***
b) The basic US zip code format has 5 digits. You can spot these at the end of most entries in the *full_address* column. How many entries in the data frame do not have a zip code? ***(4 points)***
c) Remove all the entries in the data frame that do not have a zip code and then create a new column titled *zip_code* in the data frame by extracting this information from the text address ***(4 points)***
d) Create a multi-level index on this data frame using the *city*, *zip_code* and then *business_id* columns ***(2 points)***
e) In the first Pandas notebook in class we saw how we could plot data when given latitude and longitude to plot taxi rides. In particular, we met a function useful for doing Mercator projections as well as some code that uses Matplotlib to plot multiple data points nicely on a black and white map. Use this code to plot the businesses in your data frame using the same approach taking care to adjust the *xlim* and *ylim* parameters of your plot so that it zooms in the relevant part of the map. HINT: Examine the range of your converted *px* and *py* variables. ***(6 points)***
f) How many unique categories can we find within the categories column inside this data frame? ***(4 points)***
g) What is the most reviewed bakery ('Bakeries' category) in the city of Glendale? ***(4 points)***
h) Create a data frame that counts the number of five star businesses per city in your data frame ***(4 points)***