Homework 3 (due 2/16)

DS-210 @ Boston University

Spring 2022

Before you start...

Collaboration policy: You may verbally collaborate on required homework problems. However, you must write your solutions independently without showing them to other students. If you choose to collaborate on a problem, you are allowed to discuss it with at most 2 other students currently enrolled in the class.

The header of each assignment you submit must include the field "Collaborators:" with the names of the students with whom you have had discussions concerning your solutions. If you didn't collaborate with anyone, write "Collaborators: none." A failure to list collaborators may result in a credit deduction.

You may use external resources such as software documentation, textbooks, lecture notes, and videos to supplement your general understanding of the course topics. You may use references such as books and online resources for well known facts. However, you must always cite the source.

You may **not** look up answers to a homework assignment in the published literature or on the web. You may **not** share written work with anyone else.

Submitting: Solutions should be submitted via Gradescope. The entry code is 3Y85PZ. Please submit a solution to this homework as one or two IPython notebooks (`.ipynb`) as much as this is possible.

Grading: Whenever we ask for a solution, you may receive partial credit if your solution is not sufficiently efficient or close to optimal. For instance, if we ask you to solve a specific problem that has a polynomial–time algorithm that is easy to implement, but the solution you provide is exponentially slower, you are likely to receive partial credit.

Late submission policy: No extensions, except for extraordinary circumstances. We accept submissions submitted up to one day late, but we may deduct 10% of points.

Questions

1. **(20 points)** Start by reading the following:

   - https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html
   - https://pandas.pydata.org/pandas-docs/stable/user_guide/dsintro.html
   - https://pandas.pydata.org/pandas-docs/stable/user_guide/basics.html

   Then:

(a) Find out how to sample t elements from a list of $k \geq t$ elements without replacement (i.e., a name cannot appear twice on the resulting list). Give an example.

Hint: You can use `random.sample` or `numpy.random.choice`. You can also write a function for this yourself from scratch.

(b) Create an arbitrary list of 18 different names of cars. (It can be as simple as "`A`", "`B`", ..., "`R`", but you are invited to be more creative. It could, for instance, start from "`Fiat 500`" and "`Ferrari Testarossa`".) Then create two data frames, with 10 rows each, as follows.

- The first one—say, `df1`—should have two labelled columns: `Name` and `MPG` (i.e., miles per gallon). The names should be selected at random from the list of names without replacement. The values in the other column should be independent random real numbers from 5 to 50.
- The second one—say, `df2`—should have two labelled columns: `Name` and `SecondsTo60` (i.e., time needed to reach 60 mph). Names should be selected in the same way as in `df1`, but independently of the selection in `df1`. The values in the other column should be independent random real numbers from 1 to 20.

(c) Combine `df1` and `df2` to create a new data frame, `df`, with three columns: `Name`, `MPG`, and `SecondsTo60`. The resulting data frame should have at most one row for each car. A car should appear in `df` if and only if it appears in at least one of `df1` and `df2`. If the value of `MPG` or `SecondsTo60` for a given car appears in `df1` or `df2`, respectively, then it should be included in the new data frame. Otherwise, use `NaN`.

(d) Fix all occurrences of `NaN` in `df`. First, replace all `NaN`'s in the `MPG` column with the mean of values in this column for all cars. Second, replace all `NaN`'s in the `SecondsTo60` column by the maximum of values in this column for all cars.

(e) Add "`Awesome `" as a prefix to the name of each car in `df`.

(f) Create a new data frame, `df_eco`, from `df` by selecting rows for which `MPG` is strictly greater than the median of values for all cars.

General note about Question 1: Lower credit may be assigned for less elegant solutions that do not use the full power of pandas.

2. **(20 points)** You own a very popular bakery that always sells everything they make. One morning you wake up with a brilliant idea. You check how much yeast, flour, and sugar you have (you know you have a sufficient amount of all other ingredients) and try to maximize your profits for the day. For each type of pastry, you know how much you can sell it for, and how much yeast, flour, and sugar, producing it requires. What is the maximum profit you can generate and how much of each type of pastry should you make?

(a) Read currently available amounts of ingredients from `ingredients.csv`. The file should have the following format, with only the numbers being possibly different:

```
Yeast,Flour,Sugar
57.5,112.5,245
```

You can assume that all the numbers are non-negative.

(b) Read the list of goods the bakery can produce from `pastries.csv`. Each item on the list has a name and price and also requires a certain amount of yeast, flour, and sugar to make. The file should have the following format, which should be easy to understand:

```
Name,Price,Yeast,Flour,Sugar
"Apple Pie",2.99,0,0.75,1.50
Croissant,2.50,0.5,1.5,0.25
"Poppy seed roll",5.99,1.15,1.5,0.75
```

You can assume that all numbers are non-negative and all prices are strictly positive.

(c) Find the best solution, given your circumstances. Output the total profit and how much of each pastry you have to make.

A sample possible output for the input above is

```
The max profit is 449.00 and can be achieved by producing
    * Apple Pie: 50.00 pieces
    * Poppy seed roll: 50.00 pieces
```

Note 1: Describe at least briefly how you find your solution, especially if the code is not straightforward.

Note 2: If the best solution you can find requires making fractions of some pastries, leave it this way and do not try to fix this.

(d) (Optional, no credit) Share your favorite pastry recipe on Piazza.

3. (Optional, no credit) How much time did you spend on this homework? The answer will have no impact on the credit you receive, but it may help us adjust the difficulty of future homework assignments.