Assignment 3

Data Preparation Techniques (COMP 3400)
Fall 2024

Important notes:

1. You are required to submit your assignment in *one* file in *IPython Notebook* format (due date: Nov 3). Please note that:
   - You may use *Markdown* in your IPython Notebook.
   - How you develop your IPython Notebook is your choice. You may use *jupyter.org* or *Google Colab*, or a locally installed Jupyter platform on your machine.
2. For some of the problems you may have to refer to the `pandas` API. In particular, you will need to use `value_counts()` which returns a Series containing counts of unique values in a given Series.
3. You are not allowed to use loops, in any of the problems unless otherwise stated (or you'll get a mark of 0 for that problem).
4. Slides covered in this assignment: *3-01* to *3-08*.

Problem 1 (40 pts). For the following sub-problems you will need to use `athlete_events` dataframe. For the last part you will also need to use `worldcities` dataframe. Both dataframes are available in the *data* folder in the Course Shell as `csv` files. Recall that you may use the `value_counts()` method of `pandas`.

Here is the guide for the columns in `athlete_events`. ID: Unique number for each athlete; Name: Athlete's name; Sex: M or F; Age: Integer; Height: In centimeters; Weight: In kilograms; Team: Team name; NOC: National Olympic Committee 3-letter code; Games: Year and season; Year: Integer; Season: Summer or Winter; City: Host city; Sport: Sport; Event: Event; Medal: Gold, Silver, Bronze, or NA. A guide for the columns in `worldcities` can be found in `https://simplemaps.com/data/world-cities`.

a. Create a dataframe containing data instances belonging solely to Summer Olympics [2pts].

b. Find the first 10 countries who have the most number of athletes participating in all the Olympics events [4pts].

c. Determine he counts of gold, silver, and bronze medals won by Canada in 1996 [4pts].

d. Has there ever been any Canadian athlete 45 years old or above who has won a gold medal? Extract the name and the sport of the athlete, along with the year and the city in which the event happened [5pts].

e. Find the 5 countries whose athletes with age equal to or greater than 45 years have won the most medals. Present the result in descending order [5pts].

f. Produce a table which presents the average age of female and male athletes in each and every Olympics. Your table consists of three columns: the first column signifies the year of the Olympics in ascending order, the second column signifies the average age of the female athletes, and the third column presents the average age of the male athletes [5pts].

g. Find the 10 most successful sports for team Canada in terms of the total number of medals won (the sports names along with the number of medals won) [7pts].

h. If we call all the matches which lead to distribution of three medals a competition, what are the five countries which have held the most competitions. Your result should indicate the year, country, and the number of competitions in three different columns [8pts].

Problem 2 (15pts). Create a dataframe with 3 columns containing 20 data instances. The values of the dataframe should range from 0 to 1 come from the *uniform* distribution. The 3rd column should contain five `NaN` values *uniformly* distributed [2pts]. Impute the `NaN` values by `a * 0.5`, where `a` belongs to the first column and is the corresponding value (same row) of a given `NaN` value. Implement your solution in three different ways using:

1. `fillna()` [2pts]

2. The `loc` indexer [3pts]

3. `apply()` [8pts]

Problem 4 (15pts). Create a a dataframe of positive random integers with two columns `A` and `B` each containing ten values. Create a column called `GCD` which contains $gcd(a, b)$ where a and b are corresponding values in `A` and `B`, and *gcd* stands for the *greatest common divisor*. Use the values in the GCD column to create a column called `COPRIME` which contains Boolean values indicating if the a and b are *coprimes* (or relative primes). You must use `apply()` function to produce GCD and COPRIME.

Problem 3 (30pts). In this problem you practice with the concept of *outlier* in the context of the athlete_events data. An outlier is a data instance which is not *similar* to the rest of the data instances in some respect. Here we use the *Body Mass Index* or BMI to detect the outliers. BMI is calculated by squaring the height in meters divided by the weight in kilograms. Accordingly, in athlete_events, we define a data instance whose BMI is less than the 1st percentile or greater than the 99th percentile as an outlier (where NaN values are filled with the average BMI). With reference to this definition, remove all the data instances from athlete_events except for the outliers; and then find the three most frequent sports where athletes' BMI is less than the 1st percentile, and find the three most frequent sports where athletes' BMI is greater than the 99th percentile.