Week 6 - Assessed Exercises

Data Programming with Python

This week learnt how to perform operations on Series and DataFrames, as well as covering some summary statistics. We also looked at missing data and how to replace it. In this assignment you will write functions to implement some of these ideas.

Each question asks you to write a function with a specific set of input arguments. The *.py* template defines the function name and inputs for each question, **do not** change these. Be sure you test your functions before you submit your code to make sure that they are outputting the correct answer. Unless otherwise stated, all functions must have a return value.

This week we will use the 2018 San Francisco salaries dataset to test the code. It contains salary information for over 40,000 workers in San Francisco. The file `san-francisco-2018` contains a subset of this dataset, where some of the values have been removed. Download the dataset, load it into Canopy and have a look at the first few entries to see what it looks like.The *.py* file contains a suggested tests for each function. You don't need to include the output of your tests in your PDF.

1. Write a function that will calculate the column means for a given `DataFrame` *df* and returns the `DataFrame` with the column means removed.

2. Write a function that computes summary statistics for a `DataFrame` *df*. The function should return a `DataFrame` with the following summary statistics: mean, standard deviation, minimum, maximum, the index for the first minimum and the index for the first maximum. The outputted `DataFrame` should have 6 rows, one for each piece of information listed above, labelled *mean*, *std*, *min*, *max*, *minloc*, *maxloc* (in this exact order). The columns should be the same as those in the original `DataFrame` (in the same order). You can assume that *df* does not contain categorical data

3. Write a function that will return the index of the 3 highest entries for each of the columns in a `DataFrame` *df*. The function should return a `DataFrame`, where the rows are the columns of the DataFrame df, and the columns labelled *1st*, *2nd* and *3rd* contain the index label of the 3 highest entries in the given column. Again, you can assume that df does not contain categorical data

4. In this question you need to write a function to replace all of the *NaN*s in a `DataFrame` *df* with the mean of the column for numeric data and the mode of the column for categorical data. If a column of categorical data has more than one mode you should use the first one. The function should return *df* with the missing values replaced as outlined above. This function must work for any `DataFrame`, so you cannot use the column names inside the function. You can assume that a column won't have both numerical and categorical data in it.
   Hint: You'll need to figure out how to select columns based on their data type and then use drop to make the replacements for numeric and categorical data separately.

All of your code should be written into the *.py* template. Save your filled *.py* file with the following name structure *SurnameFirstname_Week6.py* (where *Surname* and *Firstname* should be replaced with your name) and upload it to Brightspace. Additionally, you must upload a PDF of your code. Create a PDF from Canopy by selecting *File* → *Print*, and print to PDF.