**TEST 01 / 6.11.2019**

**Part 1 (30 Points)**

**You must take this part of the exam in order to submit Part 2 (Takehome)**

**Question 1 (Data pre-processing, 15 points)**

Consider the DataFrame object called frame, created by the following code.

# Lists with 14 entries

column1 = [10, np.nan, 11, np.nan, 9, 8, 7, np.nan, 5, np.nan, 4, 3, np.nan, 2 ]

column2 = [11, np.nan, 12, np.nan, 9, 9, 8, np.nan, 5, np.nan, 4, 5, np.nan, 8.5 ]

column3 = [10, np.nan, 11, np.nan, 9, 8, 12, np.nan, 8, np.nan, 4, 4, np.nan, 5 ]

#create DataFrame from columns

d = {'Column 1':column1, 'Column 2':column2, 'Column 3':column3}

frame = pd.DataFrame(d)

This data frame has therefore three columns with similar content and with missing values (i.e. NaN values).

Now we would like to imputate the missing values through three different approaches. Every approach will work with a few lines of code. In the table below, please write the output for each code segment. (5 points each)

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Approach** | **Code** | **Output** |
| Column 1 | Forward Fill | cols = ['Column 1']  frame[cols] = frame[cols].fillna(method='ffill')  print (round(frame['Column 1'].mean(),0)) |  |
| Column 2 | Backward Fill | cols = ['Column 2']  frame[cols] = frame[cols].fillna(method='bfill')  print (round(frame['Column 2'].mean(),0)) |  |
| Column 3 | Replace with Zeroes | cols = ['Column 3']  frame[cols] = frame[cols].fillna(0)  print (round(frame['Column 3'].mean(),0)) |  |

**Question 2 (Loading Files, 15 points)**

You have a file name dataset.txt which consists of columns of text and numbers. However, it is not exactly in comma separated values (CSV) format. Based on the content, there are some Python lines to process the file. Here is the code, with some missing part in each line. Fill out the table, showing what should be in the missing part.

|  |  |  |
| --- | --- | --- |
| **Issue** | **Code** | **What should be in the missing part?** |
| File is not in the same folder. It is in a folder named “files”. | filename = os.getcwd() + \_\_\_\_\_\_\_\_\_\_ + "\\dataset.txt" |  |
| File has some header. First 3 lines are not to be read. The separator is also a tab. | df = pd.read\_csv(filename,\_\_\_\_\_\_\_\_\_\_\_\_\_=3,sep='\t', lineterminator='\r' ) |  |
| The numeric column named Values has commas instead of dots. We should replace commas with dots and convert these text values into floats. | df["Values"] = df["Values"].str.\_\_\_\_\_\_\_\_\_\_(",","").astype(float) |  |