

CSE7212c_R_Python_CUTe

Answer the following questions (Max. Duration: 4 hours)

Instructions:

1. You can use r script or Rmd to answer the R programming questions.
2. Use jupyter notebook to answer the python programming questions.
3. Compress your r/rmd file and ipynb file into a zip file and upload in SCT.
4. Comments are mandatory for all the questions that you answer, comments carry marks.
5. Naming conventions for all the files (mandatory):

<FirstName>_<LastName>_B56.<extension>

Example:

Chaithanya_Kumar_B56.ipynb

Chaithanya_Kumar_B56.r

Chaithanya_Kumar_B56.Rmd

Chaithanya_Kumar_B56.zip

Python (Max Marks: 25)

1. Print all the integers till 1000 which are not a multiple of 8 but are divisible by 4, also print the count of such occurrences. (2 marks)
2. Create a list of 20 integers and write code using only list comprehensions: (10 marks)
 - i. Print the sum of the list of squared values
 - ii. Print the odd values
 - iii. Print the even values
 - iv. Print the numbers that are divisible by 5 but are not divisible by 10
3. Write a function to print "Palindrome" if the input string is a palindrome, else "Not a Palindrome". (3 marks)

Eg: If input is "Rotator" the function should return "Palindrome"

If input is "Insofe" the function should return "Not a Palindrome"
4. Write a function that takes a string as an input and returns the sum of the numbers corresponding to each alphabets position from a to z. (Hint: can use a dictionary, handle cases of the input string) (5 Marks)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

Ex: Regards ----> $18 + 5 + 7 + 1 + 18 + 4 + 19 = 72$

- Return the sum of numbers from 1 to 20 raised to the power of themselves, use lambda function, list comprehension. (5 marks)

R (Max Marks: 25)

The dataset "go_tracks.csv" contains various information collected by a 'goTracks App' based on GPS information. The details of the dataset is provided in the text file "Data_Description.txt".

1. Import the 'go_tracks.csv' file into R. (1 mark)
2. Look at the structure and summary of the datasets. Observe the data and report your findings as comments in your R file. (2 marks)
3. Convert the attributes into appropriate data types. (2 marks)
4. Impute the na values, "?" should be treated as na. (2 marks)
5. Write a user-defined function which takes in a numeric attribute and returns the range of that attribute. (Do not use any inbuilt functions like range, max, min or apply functions etc).
Apply the function you made and find the range of the following attributes in this dataset - 'speed', 'time' and 'distance'. (5 marks)
6. Use any of the apply functions in R and find the mean of the attributes 'speed' and 'distance'. (2 marks)
7. Use a 'for loop' to loop through the values in the attribute 'distance' and find the maximum distance travelled. You should not use the inbuilt function for this. (2 marks)
8. Access the rows 1-10 for the columns - 'speed', 'time', 'distance'. Save the result into a new df called GPS_df. Is it a numeric or a character df? Find the sum of the columns of the newly created df and report your results. (2 marks)
9. Calculate the variance for all the numeric columns in GPS_df. (1 mark)
10. Standardize the GPS_df columns. (3 marks)
11. Find the average speed when the traffic rating given is 'bad' and the weather is 'raining'. (3 marks)