

IC152: Assignment 6

Correlation, Dictionary, and Writing CSVs

2 questions

PLAG CHECK WILL BE DONE FROM THIS ASSIGNMENT. MAKE SURE YOUR CODE SUBMITTED ON LMS IS NOT COPIED FROM OTHER GROUPS

You must keep filenames as mentioned in the assignment.

Important: If you copy the assignment or any of its parts from others or share with others, our plagiarism softwares will catch it and you will be awarded 0 marks for the whole assignment or F grade for the course.

If you are solving this assignment in the A11's PC Lab: Keep Fn + F9 pressed during the start of your machine (do not repeatedly press, keep it continuously pressed), and then select the second option with "ubuntu". Please check if you are able to login to moodle, else shift to another machine.

Problem 0: Command Line Arguments on Terminal and Writing csv.

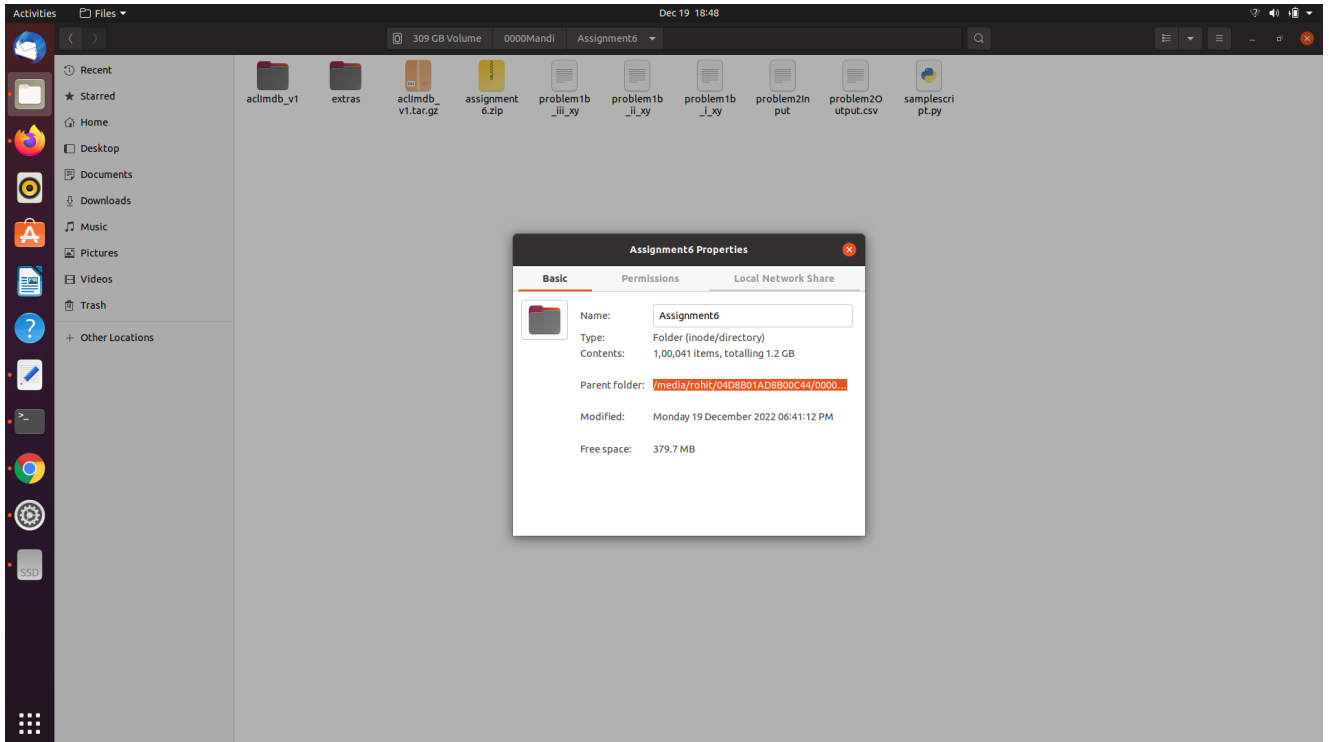
- Open samplescript.py and try to understand its contents.
- In python, you can 'import sys' and then use 'fname1 = sys.argv[1]' for the first input file name, 'fname1 = sys.argv[2]' for the second input file name and so on. You will need this in problem 1 and problem 2 both.
- The remaining part of the code writes a list of dictionaries to a csv with name 'sampleScriptOutput.csv', which you will need in problem

2. Follow the next 5 steps to run “samplescript.py” on the terminal with command line arguments.

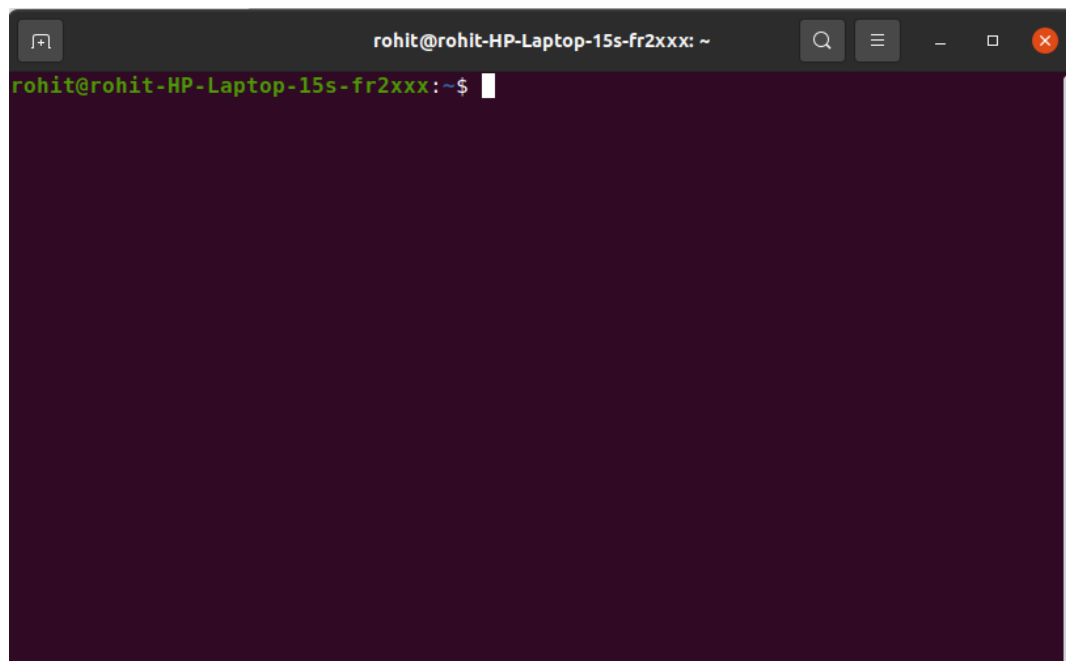
1. Click on properties where your assignment folder is located:



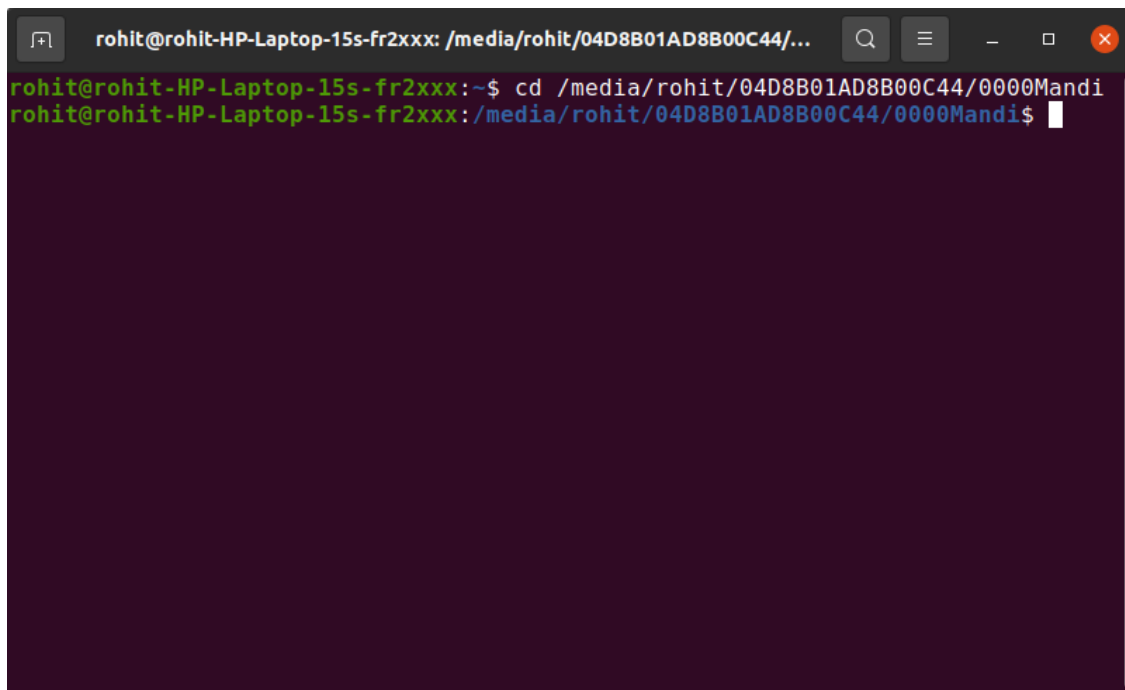
2. Copy the path under attribute “Parent folder:”:



3. Click on “Activities” in top right of Ubuntu (Linux) and type terminal. Following tab should appear:-

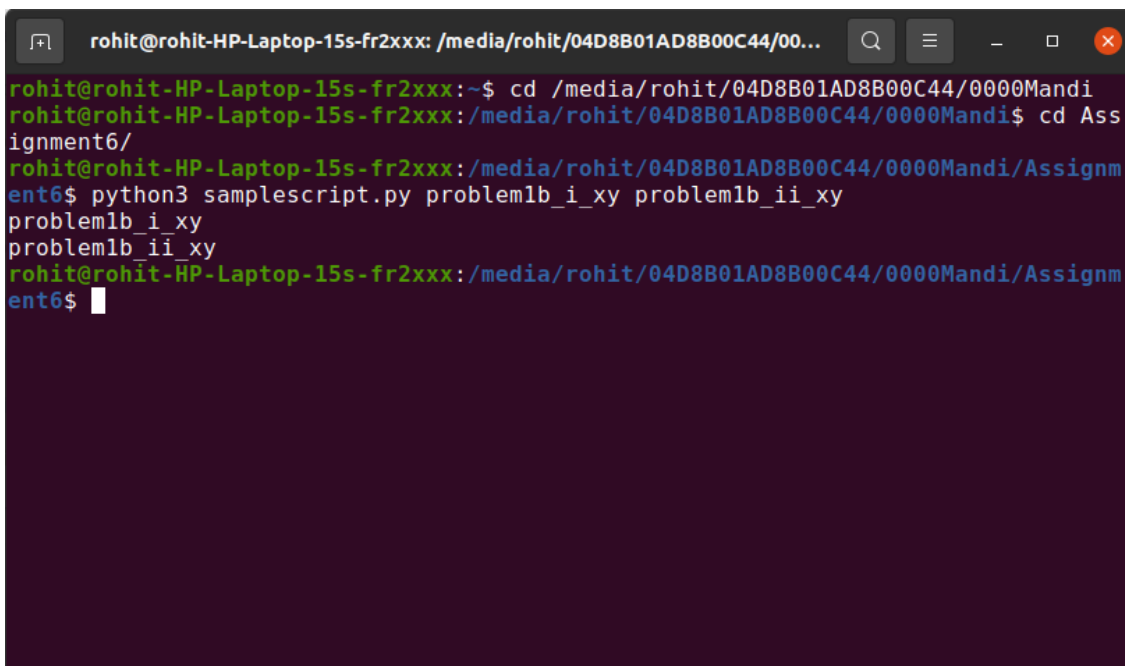


4. write “cd ” (cd followed by a space) and then paste the copied path using “Shift + Ctrl + V” keys:



```
rohit@rohit-HP-Laptop-15s-fr2xxx: /media/rohit/04D8B01AD8B00C44/...  
rohit@rohit-HP-Laptop-15s-fr2xxx:~$ cd /media/rohit/04D8B01AD8B00C44/0000Mandi  
rohit@rohit-HP-Laptop-15s-fr2xxx:/media/rohit/04D8B01AD8B00C44/0000Mandi$
```

5. Now you can “cd Assignment6/” and then try “python3 samplescript.py problem1b_i_xy problem1b_ii_xy” and see what it does (note: sampleScriptOutput.csv will be created in Assignment6/):



```
rohit@rohit-HP-Laptop-15s-fr2xxx: /media/rohit/04D8B01AD8B00C44/00...  
rohit@rohit-HP-Laptop-15s-fr2xxx:~$ cd /media/rohit/04D8B01AD8B00C44/0000Mandi  
rohit@rohit-HP-Laptop-15s-fr2xxx:/media/rohit/04D8B01AD8B00C44/0000Mandi$ cd Assignment6/  
rohit@rohit-HP-Laptop-15s-fr2xxx:/media/rohit/04D8B01AD8B00C44/0000Mandi/Assignment6$ python3 samplescript.py problem1b_i_xy problem1b_ii_xy  
problem1b_i_xy  
problem1b_ii_xy  
rohit@rohit-HP-Laptop-15s-fr2xxx:/media/rohit/04D8B01AD8B00C44/0000Mandi/Assignment6$
```

Problem 1: Correlation

Consider two vectors: $X = [x_1 - \mu_x, x_2 - \mu_x, \dots, x_n - \mu_x]$ and $Y = [y_1 - \mu_y, y_2 - \mu_y, \dots, y_n - \mu_y]$. μ_x is the mean of x_i 's, and μ_y of y_i 's. The cosine of the angle between these two vectors can be given by ratio of dot product of these two vectors with their magnitude, i.e. ratio of $X \cdot Y$ and $|X||Y|$.

Interestingly, correlation is a statistical method that measures the similarity of the variation between two random vectors. The correlation coefficient (value in between -1 to +1 similar to cosine) in between two vectors can be calculated with the help of the given formula:

$$r = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\left[n \sum x_i^2 - \left(\sum x_i \right)^2 \right]^{\frac{1}{2}} \left[n \sum y_i^2 - \left(\sum y_i \right)^2 \right]^{\frac{1}{2}}}$$

Where, n = sample size, x_i and y_i are the sample points with index i .

- Prove that the ratio of $X \cdot Y$ and $|X||Y|$ is equal to r given in above equation (X and Y are vectors defined before) . Show your proof to the lab instructor/TAs for evaluation. **5 marks**
- When one variable increases as the other increases the correlation (r) is positive. If one decreases as the other increases r is negative. Complete absence of correlation is represented by $r = 0$. Plot the variables (x and y) as scatter plots given in files: **15 marks**
 - problem1b_i_xy

ii. problem1b_ii_xy

iii. problem1b_iii_xy

- Each file has two lines. The first line has the character 'x' followed by the values of x_i 's. Similarly the second line has the character 'y' followed by values of y_i 's. You should ignore x and y written in the file and use remaining values in each line for plotting.
- Although there are a fixed number of points in the above mentioned files, the code should be generic to work for any number of points in the text file.
- The code should also handle corner cases, e.g. when a file has characters instead of numbers.
- Save the file as problem1b.py. It should take input files as three arguments, exact usage:

```
#####  
python3 problem1b.py problem1b_i_xy problem1b_ii_xy problem1b_iii_xy  
#####
```

- In python, you can 'import sys' and then use 'file_name1 = sys.argv[1]' for the first input file name, 'file_name1 = sys.argv[2]' for second input file name and so on.
- Executing the above command (between #s) in the linux terminal must save the images with the same names as input files but with the .png extension. E.g., problem1b_i_xy.png, problem1b_ii_xy.png, and problem1b_iii_xy.png for inputs mentioned in the terminal command above.
- Your python file/code should work for any number of input files.
- The python code/file should prompt the user with the usage instruction if the user forgets to provide any input file.

- c. Write the code to find correlation between the different cases of two variables (x and y) as given in part b and use the equation for r (mentioned above). **10 marks**

- The code file name must be problem1c.py.
- Executing problem1c.py with following usage in linux terminal, should write different values of r separated by a space in a line of a file.

```
#####  
python3 problem1b.py problem1b_i_xy problem1b_ii_xy problem1b_iii_xy  
#####
```

- The output file name must be Output1c.txt
- Although there are a fixed number of points in the given input files, the code should be generic to work for any number of points in the text files.
- The code should also handle corner cases, e.g. when a file has characters instead of numbers.
- Your python file/code should work for any number of input files.
- The python code/file should prompt the user with the usage instruction if the user forgets to provide any input file.
- Analyze the numerical value and scatter plots of the variables (i.e. if correlation is positive, y should increase with increase in x). Tell your observations to the instructor/TAs.

Problem 2: Dictionary and Writing a csv file

Language models are used to complete the sentences and correct the recognized text in different AI applications. This question forms the basis

for language models, where not just words in the language but their context and frequency is also important. **20 marks**

- Read the problem2Input^[1] in your python code and find the frequency of each word in the file using a dictionary. This text is from [Stanford's large movie review dataset v1.0](#).
- Sort the keys (words) and values (frequencies) in the dictionary in descending order of the values/frequencies.
- Write the words (in descending order of frequencies) in the first column of the csv file. Write the corresponding frequencies in the second column.
- You can use the following code to write a dictionary to file:

```
#####code starts here#####  
# dict format required for csv  
myDict = [{'word': 'a', 'frequency': 1000}, {'word':  
'the', 'frequency': 700}, {'word': 'me', 'frequency':  
20}]  
# code to write above dict to csv  
import csv  
with open('problem2Output.csv', 'w') as csvop:  
    # creating dictionary writer object  
    writerObj = csv.DictWriter(csvop, fieldnames =  
['word', 'frequency'])  
  
    # write fieldnames  
    writerObj.writeheader()  
    writerObj.writerows(myDict)  
#####code ends here#####
```

- You will need to convert your dictionary to a list of dictionaries as required by the above code.

- The program should run for any type of input text file, if it is an empty file the program should prompt the user to give a file with text.
- 'problem2Output.csv' must be the name of the output file.
- The program should show the usage to the user if no input is given.
- The name for file should be problem2.py, and should take the input file as an argument similar to previous question:

```
#####
python3 problem2.py problem2Input
#####
```

After the csv file is created, open the csv file and observe the words with top 10 frequencies. Would you have guessed them without solving this problem? Share your observations with the instructor/TAs.

Extra Problem: Try problem 2 code with the text from a wikipedia article in the language you know. Try to guess the top 5 words in the language before you start coding or open the csv file after coding (save csv as extraProblemOutput.csv).

Create the folder having your python files, with name having your roll number followed by “_assignment6” (don’t use inverted commas in folder name), compress the folder with .zip extension and submit it on moodle.

Make sure that you delete all your files from the lab PC/Laptop, and shut it down before you leave.

References:

[1] Maas, Andrew, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. "Learning word vectors for sentiment

analysis." In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 142-150. 2011.