CS 304 Homework Assignment 1

Due: 11:59pm, Thursday, February 1st

This assignment is scored out of 50. It consists of 5 questions. When you submit, you are required to create a folder with your name (Last name first, then First name), CS422, HW1, e.g., LastName_FirstName_CS422_HW1. Type your answers into a text file (**only `.txt`, `.doc`, and `.pdf` file formats are accepted**) and save it in this folder. Put all your Java programs (`*.java`) as well as output files in the same folder. Zip this folder, and submit it as one file to Desire2Learn. Do not hand in any printouts. Triple check your assignment before you submit. **If you submit multiple times, only your latest version will be graded and its timestamp will be used to determine whether a late penalty should be applied.**

Short Answers

P1. (6pts) You are given a set of m objects that is divided into k groups, where the ith group is of size m_i. If the goal is to obtain a sample of size $n < m$, what is the difference between the following two sampling schemes? (Assume sampling with replacement.)

(a) We randomly select $n \times m_i/m$ elements from each group.

(b) We randomly select n elements from the data set, without regard for the group to which an object belongs.

P2. (5pts) In text mining, the TF-IDF (term frequency – inverse document frequency) statistic is often used to quantify how important a word is to a document in a collection or corpus. For a term-document matrix (each row corresponds to a word and each column corresponds to a document), we use tf_{ij} (also referred to the **term frequency**) to represent the frequency of the ith word in the jth document, and df_i (also referred to the **document frequency**) to represent the number of documents in which the ith word appears. $idf_i = \log\frac{m}{df_i}$ is a transformation on the document frequency, which is also known as the **inverse document frequency**. The following formula shows the TF-IDF value for the ith word in the jth document:

$$tfidf_{ij} = tf_{ij} \times idf_i = tf_{ij} \times \log\frac{m}{df_i}$$

where m is the number of documents in the corpus.

Explain the purpose of the inverse document frequency in the formula and how it affects the importance of a word in a particular document.

P3. (6pts) Suppose you are going to calculate the similarity between two vehicle records, whose attributes are $\{type, size, color, displacement, horsepower, weight, acceleration\}$. For example, car $A = \{sedan, mid\ size, white, 2.5, 184, 3300, 8.4\}$, and car $B = \{SUV, full\ size, red, 3.0, 280, 4500, 9.0\}$. What are the issues you might encounter in the calculation? How would you resolve these issues?

P4. (6pts) Given the following two vectors:

$$A = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \qquad B = \begin{bmatrix} 8 \\ -15 \\ 2 \\ 4 \end{bmatrix}$$

(a) Are they orthogonal? Explain your answer.

(b) Are they orthonormal? If not, can you make them orthonormal? **Show all your work.**

Programming Questions

P6. (27pts)

a. Completing the `Homework1` class

You are provided with two files "**`Homework1.java`**" and "**`TestHomework1.java`**". You are required complete the following four methods in the former:

(Note: you should NOT change the contents of `A` and `B` in any of the following methods!)

`int[][] matrixMultiplication(int[][] A, int[][] B)`
This method takes as parameters two 2-dimensional **`int`** array **`A`** and **`B`**, which are two integer matrices. The method calculates the product of **`A`** and **`B`** (i.e., **`A`**×**`B`**) and returns the result as a 2-dimensional array. You can assume that the matrices passed into the method have correct dimensions, i.e., the column dimension of **`A`** is equal to the row dimension of **`B`**.

`double jaccard(int[] A, int[] B)`
This method takes as parameters two **`int`** arrays **`A`** and **`B`**, which are two integer vectors. The method calculates and returns the Jaccard coefficient between the two vectors as a **`double`**.

`double cosineSimilarity(int[] A, int[] B)`
This method takes as parameters two **`int`** arrays **`A`** and **`B`**, which are two integer vectors. The method calculates the cosine similarity between the two vectors, and returns the angle in degrees between the two vectors. For example, the angle between vectors (1, 1) and (1, 0) is 45°, whereas the angle between vectors (0, 1) and (1, 0) is 90° because they are perpendicular to each other.

`double euclidean(int[] A, int[] B)`
This method takes as parameters two **`int`** arrays **`A`** and **`B`**, which are two integer vectors. The method calculates and returns the Euclidean distance between the two vectors.

Note that you are only supposed to touch the above methods. You are NOT allowed to create any other methods, instance variables, or make any changes to methods other than the above methods or files other than "`Homework1.java`". Points will be taken off if you fail to follow this rule.

b. Code Testing

You are provided with a test driver implemented by "**`TestHomework1.java`**" (**Do not make any changes to this file!**) so there is no need to write your own.

Once you have completed the above method, you can run the test. You should create a plain text file named "**`output.txt`**", copy and paste the output (if your code crashes or does not compile, copy and paste the error messages) to this file and save it.

Grading Rubrics:

Code does not compile: -10
Code compiles but crashes when executed: -5
Changes were made to things other than the required methods: -5
Has output file: 5
`matrixMultiplication` changes the content of the array parameters: -5
`jaccard` changes the content of the array parameters: -5
`cosineSimilarity` changes the content of the array parameters: -5
`euclidean` changes the content of the array parameters: -5
Code passes 11 test cases: 22 (each test case worth 2 points)

Sample Output:

```
Test 1: matrixMultiplication - [Passed]
 A =
     1     2     3
     4     5     6
     7     8     9

 B =
    10    20    30
    40    50    60
    70    80    90
 --------------
 Expected:
   300   360   420
   660   810   960
  1020  1260  1500

 Yours:
   300   360   420
   660   810   960
  1020  1260  1500

Test 2: matrixMultiplication - [Passed]
 A =
     7     8     9

 B =
    10    20    30
    40    50    60
    70    80    90
 --------------
 Expected:
  1020  1260  1500
```

```
 Yours:
  1020  1260  1500

...

Test 4: jaccard - [Passed]
 A = [0, 1, 1, 0, 0, 0, 1, 1, 0]
 B = [0, 1, 1, 0, 0, 0, 1, 1, 0]
 --------------
 Expected: 1.0
 Yours: 1.0

Test 5: jaccard - [Passed]
 A = [0, 1, 1, 0, 0, 0, 0, 1]
 B = [0, 0, 0, 0, 0, 0, 1, 1]
 --------------
 Expected: 0.25
 Yours: 0.25

...

Test 10: euclidean - [Passed]
 A = [3, 1]
 B = [5, 1]
 --------------
 Expected: 2.0
 Yours: 2.0

Test 11: euclidean - [Passed]
 A = [3, 2, 0, 5, 0, 0, 0, 2, 0, 0]
 B = [1, 0, 0, 0, 0, 0, 0, 1, 0, 2]
 --------------
 Expected: 6.2
 Yours: 6.2

Total test cases: 11
Correct: 11
Wrong: 0
```