Algorithmic Methods of Data Mining

Homework 1

Due: 22/10/2017, 23:59.

Instructions

You must hand in the homeworks electronically and before the due date and time.

The first homework has to be done by each **person individually**.

Handing in: You must hand in the homeworks by the due date and time by an email to `fazzone@diag.uniroma1.it` that will contain as attachment **(not links to some file-uploading server!)** a .zip file with your answers. The filename of the attachment should be
`AMD_Homework_1__StudentID_StudentName_StudentSurname.zip`;
for example:
`AMD_Homework_1__1235711_Robert_Anthony_De_Niro.zip`.
The email subject should be
`[Algorithmic Methods for Data Mining] Homework_1 StudentID StudentName StudentSurname`;
For example:
`[Algorithmic Methods for Data Mining] Homework_1 1235711 Robert Anthony De Niro`.
After you submit, you will receive an acknowledgement email that your project has been received and at what date and time. If you have not received an acknowledgement email within 2 days after you submit then contact Adriano.

The solutions for the theoretical exercises must contain your answers either typed up or hand written clearly and scanned.

For information about collaboration, and about being late check the web page.

Problem 1. We are given a collection of n documents, represented as vectors in an m-dimensional array, the ith document $\mathbf{d^i}$ represented as

$$\mathbf{d^i} = (d^i_1, d^i_2, \ldots, d^i_m).$$

Provide an algorithm that finds the pair of documents that are closest to each other, using cosine similarity. What is its time complexity (as a function of n and m)? Explain why.

Problem 2. The goal of the first assignment focuses on building up your skills on Python. The assignment is done electronically using the HACKERRANK online service available from the following URL:

`https://www.hackerrank.com`

You must create an account and complete as many challenges as you can from the list included below. When you are finished and want to submit your assignment, from the `hackerrank` site, go to the *submissions* page under your profile and produce a PDF printout. This is available from the following url:

`https://www.hackerrank.com/submissions/all`

Your solutions must include as attachment:

- The pdf file that you created.
- The Python code of your solutions, clearly organized.

Don't forget to check the collaboration policy at the course web page. To summarize, you can discuss with each other, but the writing at the end should be yours. In addition, you should preferably not look at the solutions provided by HACKERRANK; if you do so, you should explicitly mention it in the corresponding response.

Let's go to the homework. The Python challenges that you need to complete are the following:

- Introduction (all – total: 7 - max points: 75)
  `https://www.hackerrank.com/domains/python/py-introduction`

- Data types (all – total: 6 - max points: 60)
  `https://www.hackerrank.com/domains/python/py-basic-data-types`

- Strings (all – total: 14 - max points: 220)
  `https://www.hackerrank.com/domains/python/py-strings`
  `https://www.hackerrank.com/domains/python/py-strings/2`

- Sets (all – total: 13 - max points: 170)
  `https://www.hackerrank.com/domains/python/py-sets`
  `https://www.hackerrank.com/domains/python/py-sets/2`

- Collections (all – total: 8 - max points: 220)
  `https://www.hackerrank.com/domains/python/py-collections`

- Date and Time (all – total: 2 - max points: 40)
  `https://www.hackerrank.com/domains/python/py-date-time`

- Exceptions (only 1 - max points: 10)
  `https://www.hackerrank.com/challenges/exceptions`

- Built-ins (only 3 - max points: 80)
  `https://www.hackerrank.com/challenges/zipped`
  `https://www.hackerrank.com/challenges/python-sort-sort`
  `https://www.hackerrank.com/challenges/ginorts`

- Python Functionals (only 1 - max points: 20)
  `https://www.hackerrank.com/challenges/map-and-lambda-expression`

- Regex and Parsing challenges (all – total: 17 - max points: 560)
  `https://www.hackerrank.com/domains/python/py-regex`
  `https://www.hackerrank.com/domains/python/py-regex/2`

- XML (all – total: 2 - max points: 40)
  `https://www.hackerrank.com/domains/python/xml`

- Closures and Decorations (all – total: 2 - max points: 60)
  `https://www.hackerrank.com/domains/python/closures-and-decorators`

  Numpy (all – total: 15 - max points: 300)
  `https://www.hackerrank.com/domains/python/numpy`
  `https://www.hackerrank.com/domains/python/numpy/2`