# of Data Mining

Homework 1

Due: 21/10/2018, 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 Leonardo (`martini.1722989@studenti.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
`DM_Homework_1__StudentID_StudentName_StudentLastname.zip`;
for example:
`DM_Homework_1__1235711_Robert_Anthony_De_Niro.zip`.
The email subject should be
`[Data Mining] Homework_1 StudentID StudentName StudentLastname`;
For example:
`[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 Leonardo.

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 shuffle a standard deck of cards, obtaining a permutation that is uniform over all 52! possible permutations.

1. Define a proper probability space Ω for the above random process. What is the probability of each element in Ω?
2. Find the probability of the following events:
   (a) The first two cards include at least one ace.
   (b) The first five cards include at least one ace.
   (c) The first two cards are a pair of the same rank (they are the same number or both are J, or both are Q, etc.)
   (d) The first five cards are all diamonds.
   (e) The first five cards form a full house (three of one rank and two of another rank).
3. (Optional) Develop some small programs in Python to perform simulations to check your answers.

Problem 2. A group of n men and m women go to a Chinese restaurant and sit in a round table, such that each person has two other persons next to him/her.

1. Describe a sample space that describes the random process.

2. Find the expected number of men who will be sitted next to at least one woman.

Problem 3. The Erdős-Rényi $G_{n,p}$ random-graph model, is a mathematical model for creating random graphs. Fix a positive integer n and a value $p \in [0,1]$. Then a graph created according to the $G_{n,p}$ model, has n nodes, and each pair of distinct nodes is connected with an edge with probability p; all pairs being independent from each other.

1. Define an appropriate probability space Ω to describe the $G_{n,p}$ model. What is its size?
2. What is the probability of each element of Ω?
3. What is the probability that a graph created according to $G_{n,p}$ contains exactly two cycles of size $n/2$ and no other edges (assume that n is even for this)? A cycle is a sequence of vertices $v_1, v_2, \ldots, v_\ell, v_1$ ($\ell \geq 3$), with $v_i \neq v_j$ for $i,j \in \{1, \ldots, \ell\}$, such that each edge (v_i, v_{i+1}) exists in the graph (as well as (v_ℓ, v_1)).
4. What is the probability that a graph created according to $G_{n,p}$ contains exactly two cycles of any size with no nodes in common, and no other edges?
5. In a graph created by the $G_{n,p}$ model, what is the expected degree of a node?
6. In a graph created by the $G_{n,p}$ model, what is the expected number of edges?
7. We define a *house subgraph* to be a subgraph of 5 nodes, $v_1, \ldots, v_5$ in which the edges $\{v_1,v_2\}$, $\{v_1,v_3\}$, $\{v_2,v_3\}$, $\{v_2,v_4\}$, $\{v_3,v_5\}$, $\{v_4,v_5\}$ exist, and no other edge between them exists (see Figure 1). What is the expected number of house subgraphs in a random graph according to $G_{n,p}$?

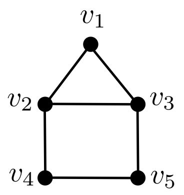

Figure 1: The *house subgraph*.

Problem 4. Quite often you can analyze your data just by using simple unix tools. Some usefull commands are the `grep`, `sort`, `uniq`, `cut`, `sed`, `awk`, `join`, `head`, `tail`, `wget`, `curl`. You can find more information using the `man` command or by checking the web. Shell scripting can help you even more.

As a simple exercise do a simple analysis of the reviews in
`http://aris.me/contents/teaching/data-mining-2018/protected/beers.txt`.
After you download and unzip the file, use some of the commands above to find the 10 beers with the highest number of reviews. (**Hint:** You can do it with a single command line, by chaining commands through pipes!)

Problem 5. We will now go one step further and start practicing with Python. Write a Python program to find the top-10 beers with the highest average overall score among the beers that have had at least 100 reviews. (You may need to preprocess the file first.)

Problem 6.

Often, to obtain data, we need to crawl a website. Here we will design a simple crawler for downloading and parsing job advertisements from the kijiji web site.

For downloading the web pages you may use the package `Requests` or the package `urllib2`. To parse the page you can either use regular expressions through the package `re` (it is anyway a good idea become familiar with regular expressions), or, probably better, use an HTML/XML parser. The `Beautiful Soup` package is a good one but it loads the whole file in memory. This is fine for this problem, since the pages to parse are small, but be careful if you want to use it on large XML files; for those ones check the `lxml` library and the tutorial at

`http://www.ibm.com/developerworks/xml/library/x-hiperfparse/`

Write a program that will download from `http://www.kijiji.it` and parse all the job positions in Italy about *Informatica/Grafica/Web.* Download regular and top announcements, but not sponsored ads. Save in a tab-separated value (TSV) file, for every job (one line per job), the *title*, *short description* (from the job summary page), *the location*, the *publication date* of the job announcement, and the *URL link* to its web page. **Because you will make a lot of calls to the kijiji site, make sure that you have a delay (use: `sys.sleep()`) between different downloads of kijiji pages, to avoid being blocked.**

Optional, extra credit: Download and store also the *full text* of the ads. (You will need to visit the ad pages for that.)