# Web information retrieval - 2019/2020

## Exam - November 2nd, 2020

### Time: 90 minutes

---

## Assignment 1

Assume we have the following collection of 5 documents:

- D1 = "If it walks like a pork and 'oinks' like a pork, it must be a pork."
- D2 = "Ariccia Pork is mostly prized for the thin, crispy pork skin with authentic versions of the dish serving mostly the skin."
- D3 = "Bugs' ascension to stardom also prompted the Warner animators to recast Ninetto Pork as the rabbit's rival, intensely jealous and determined to steal back the spotlight while Bugs remained indifferent to the pork's jealousy, or used it to his advantage. This turned out to be the recipe for the success of the duo."
- D4 = "6:25 PM 1/7/2007 blog entry: I found this great recipe for Rabbit Braised in Wine on cookingnice.com."
- D5 = "Last week Li has shown you how to make the Spoleto pork. Today we'll be making Italian pasta (spahetti), a popular dish that I had a chance to try last summer in Ariccia. There are many recipies for spaghetti."

1. Write a table with the TF/IDF weights of all terms in `T`, where `T = {ariccia, dish, pork, rabbit, recipe, roast}`.

   *Do not use log when computing metrics and restrict to the term set* `T = {ariccia, dish, pork, rabbit, recipe, roast}`.

2. Consider the query `Q = "Ariccia pork recipe"` and identify the two top ranked documents according to the TF/IDF rank and *using* `T` *as the dictionary and cosine similarity as a measure*. Are the top ranked documents relevant to the query? Write a table with the similarities `<doc, Q>`, where `doc` $\in$ `{D1, D2, D3, D4, D5}`.

   *Motivate your answer.*

---

## Assignment 2

You are given the following dataset:

| Confident | Studied | Sick | Result |
|---|---|---|---|
| Yes | No | No | Fail |
| Yes | No | Yes | Pass |
| No | Yes | Yes | Fail |
| No | Yes | No | Pass |
| Yes | Yes | Yes | Pass |

In this table, the fields `Confident`, `Studied` and `Sick` correspond to binary features describing the state of a person being interviewed for an internship at an important e-Commerce company. The `Result` field is a binary class describing the two possible outcome of the interview. Assume we want to predict the outcome of an applicant $X$'s interview, where $X$ is described by the vector `(Yes, Yes, No)`.

You should answer the following questions:

1. Which class (`Fail` or `Pass`) will be assigned to $X$ by a Naive Bayes, *Bernoulli* classifier?

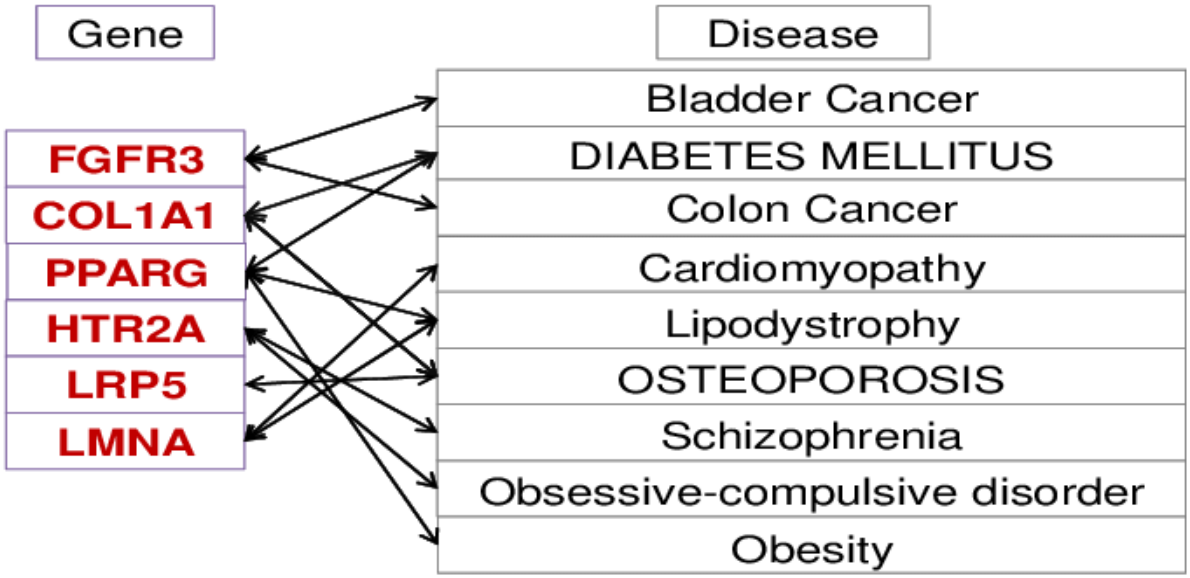
   *You should thoroughly motivate your answer, giving the details of how the probabilities of* `Fail` *and* `Pass` *are estimated for* $X$*. You should not apply smoothing*.

2. Please indicate where the conditional independence assumption is used in the derivations you did answer point 1 above.

---

## Assignment 3

Consider the network in the picture below, which is a small sample of a much large gene-disease association graph describing, as a bipartite graph, which genes are involved in which diseases.



In general, given such a graph, we are interested in finding genes that are *central*, in the sense that they are involved in *important* diseases. The problem is making the above notions of *central* and *important* quantitative and workable. This said, answer the following questions:

**3.1.** Assume in our first attempt, we simply regard the edges as undirected (or bidirectional as in the picture) and we apply a simple random walk (not Pagerank!) of the graph above to assign importance scores to genes and diseases. The underlying hypothesis here is, the more frequently we visit a node, the more important it is.

Does the Markov chain corresponding to this random walk admit a *unique stationary* distribution? Why yes? Why not? *You cannot simply answer yes or no, you should motivate your answer*.

**3.2.** After some thought, we performed a second attempt. This time, we decide to orient edges, from left (genes) to right (diseases). Propose an algorithm to quantitatively characterize *central* genes and *important* diseases in this case.

*Motivate your answer, describing the algorithm you use and how your input is mapped onto an input for your algorithm of choice*.