582631 Introduction to Machine Learning, Autumn 2013
Exercise set 2 (based on first week of lectures)

To be discussed in the exercise session 8 November.

Credit is given based on the written solutions turned in to the course assistant. Extra credit (0.5 points per problem) is given for willingness to present the solution to the pen-and-paper problems at the exercise session.

The deadline for turning in your solutions is **9:00am on Wednesday, 6 November.**

Send your solutions to the course assistant (Yuan.Zou@cs.helsinki.fi) by e-mail. You should send one PDF file including your solutions to the pen-and-paper problems and the report for the programming problem, and a single compressed file including the code for the programming problem.

Pen-and-paper problems

Problem 1 (3 points)

Consider a document-term matrix, where tf_{ij} is the number of times that the i^{th} word (term) appears in the j^{th} document, and let m be the total number of documents in the collection. Consider the variable transformation that is defined by

$$tf'_{ij} = tf_{ij} \log \frac{m}{df_i}, \tag{1}$$

where df_i is the number of documents in which the i^{th} term appears, which is known as the *document frequency* of the term. This transformation is known as the *inverse document frequency* transformation.

(a) What is the effect of this transformation if a term occurs in only one document? In every document?

(b) What is the overall effect and what might be the purpose of this transformation?

(c) Can you think of other (non-document) data in which this transformation might be useful?

Problem 2 (3 points)

In this exercise we explore the relationships between the cosine and correlation similarity measures and Euclidean distance for data vectors in R^n.

(a) What is the range of values that are possible for the cosine measure?

(b) If two objects have a cosine measure of 1, are they necessarily identical? Explain.

(c) What is the relationship of the cosine measure to correlation, if any? (Hint: Look at statistical measures such as mean and standard deviation in cases where cosine and correlation are the same and different.)

(d) Derive the mathematical relationship between cosine similarity and Euclidean distance when each data object has an L_2 length (norm) of 1.

(e) Derive the mathematical relationship between correlation and Euclidean distance when each data point has been standardized by subtracting its mean and dividing by its standard deviation.

Problem 3 (3 points)

Proximity is typically defined between a pair of objects.

(a) Give two ways in which you might define the 'proximity' among a set of (more than two) objects (i.e. a single measure of how similar an arbitrary number of items are all to one another)

(b) How might you define the distance between two sets of points in Euclidean space?

(c) How might you define the proximity between two sets of data objects? (Make no assumptions about the data objects, except that a proximity measure is defined between any pair of objects.)

See the next page for the the programming problem!

Programming problem

General instructions:

- Return a brief written report (as PDF, included in the same file as your pen-and-paper solutions) and one directory (as a zip or tar.gz file) including all your code.
- Do not include any of the data files in your solution file.
- Always mention your name and student ID in the report.
- We use the report as the main basis for grading: All your results should be in the report. We also look at the code, but we won't however go fishing for results in your code.
- The code needs to be submitted as a runnable file or set of files (command to run it given in the report).
- In your report, the results will be mostly either in the form of a figure or program output. In both cases, add some sentences which explain what you are showing and why the results are the answer to the question.
- If we ask you to test whether an algorithm works, always give a few examples showing that the algorithm indeed works

Problem 4 (15 points)

In this problem we will consider similarity measures for movies on the Movielens dataset.

(a) Download the Movielens data from the course web page. In addition to the data, the file also contains some functions for easily loading the data into Matlab/Octave/R and some example code that you can use if you wish. See the README files for details.

(b) We will now construct a similarity measure over the movies. For simplicity, let us first consider a simple measure that does not use the explicit (numerical) ratings given by the users, nor the time stamps of the ratings, but only whether or not a given movie was rated by a given user. Create a function that, given two different movie IDs as input, outputs the Jaccard coefficient: the number of users who rated both movies divided by the number of users who rated at least one of the movies. For example, for the movies 'Toy Story' and 'GoldenEye' the coefficient should be 0.217. What is the Jaccard coefficient between 'Three Colors: Red' and 'Three Colors: Blue'? What are the 5 movies with highest Jaccard coefficient to 'Taxi Driver'? Select a movie of your own choosing (which you are familiar with), what are the 5 movies with highest Jaccard coefficient to that movie? Do they make sense?

(c) Now let's try a similarity measure that uses the explicit ratings. Create a second function that, given two different movie IDs as input, outputs the correlation coefficient of the ratings given to those two movies by all users which have rated both movies. (Note, the function may need to return 0 when the number of users who have rated both is so low that one cannot compute a correlation coefficient.) What is now the similarity between 'Toy Story' and 'GoldenEye'? How about 'Three Colors: Red' and 'Three Colors: Blue'? What are the 5 movies with highest similarity to 'Taxi Driver'? Again, select a movie of your own choosing and list the 5 movies with highest similarity.

(d) Provide some brief thoughts on which similarity measure seems to work 'better', in the sense that the computed similarity matches your intuitive sense of similarity. Why do you think this is? Explain.