Exercise 9: Latent Semantic Analysis

You can earn up to 10 points on this exercise. 5 points is the lowest passing score.
You may submit individually or as a group of up to 3 people.
You may use any programming language you wish, but any submission that I cannot run on my computer without installing things must be presented to the class. (I like Python).

Please email your solution to `claytong@coli.uni-saarland.de` *by* 23:59 CEST on **July 7, 2016**. *Your name(s) should be present when I print the files you send!*

This exercise is adapted from data in:
Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes, 25*(2-3), 259–284.

Consider the following 5 documents on human-(c)omputer interaction and 4 documents on (m)athematical graph theory. We will focus on the *italicized* words.

Example of text data: Titles of Some Technical Memos

c1:	*Human* machine *interface* for ABC *computer* applications
c2:	A *survey* of *user* opinion of *computer system response time*
c3:	The *EPS user interface* management *system*
c4:	*System* and *human system* engineering testing of *EPS*
c5:	Relation of *user* perceived *response time* to error measurement
m1:	The generation of random, binary, ordered *trees*
m2:	The intersection *graph* of paths in *trees*
m3:	*Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4:	*Graph minors*: A *survey*

Ignoring case, this gives a term-document matrix of:

	c1	**c2**	**c3**	**c4**	**c5**	**m1**	**m2**	**m3**	**m4**
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minor	0	0	0	0	0	0	0	1	1

Task 1

Compute the document-document similarity matrix, A^tA. (1 point)
Compute the term-term similarity matrix, AA^t. (1 point)
In words, what do these matrices mean? Qualitatively evaluate the numbers. (1 point)

Task 2

Perform singular value decomposition (SVD) on the term-document matrix[1]. (3 points)
Using your decomposition, compute the document-document similarity matrix, $(SD^t)^t(SD^t)$, and the term-term similarity matrix, $(TS)(TS)^t$. (1 point)
Compare the quality of these matrices to that of the matrices obtained in Task 1. (1 point)

Task 3

Remove all but the first two singular values from your decomposition. Hence, $\hat{T}$ will be a 12×2 matrix, $\hat{S}$ will be a 2×2 matrix, and $\hat{D}$ will be a 9×2 matrix. Using these, compute the document-document similarity matrix, $(\hat{S}\hat{D}^t)^t(\hat{S}\hat{D}^t)$, and the term-term similarity matrix, $(\hat{T}\hat{S})(\hat{T}\hat{S})^t$. (1 point)
Compare the quality of these matrices to that of the matrices obtained in Task 1 and Task 2. (1 point)

[1] For python, you may use the SVD package within `NumPy` described here: `http://docs.scipy.org/doc/numpy/reference/generated/numpy.linalg.svd.html`.