Exercise 10: Non-negative Matrix Factorization

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 14, 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

What does the W matrix represent in a non-negative matrix factorization? (1 point)
What does the H matrix represent in a non-negative matrix factorization? (1 point)

Task 2

Implement non-negative matrix factorization without using ready-made packages. (3 points)
Using your implementation, factorize

$$\begin{bmatrix} 1 & 2 & 0 \\ 0 & 3 & 0 \\ 2 & -4 & 2 \end{bmatrix}$$

into two three-dimensional subspaces. Report a $W \cdot H$ for a good random initialization.
(1 point)

Task 3

Using your implementation, factorize the term-document matrix into two two-dimensional subspaces. (1 point)
Using the decomposition, calculate the term-term similarity matrix. (1 point)
Using the decomposition, calculate the document-document similarity matrix. (1 point)
How do these matrices compare with those produced by SVD with two singular values? (1 point)