Non-Negative Matrix Factorization

We have the following ratings on 5 movies by 4 users:

user	*Titanic*	*Tiffany*	*Terminator*	*Star Trek*	*Star Wars*
Ada	5	4	1	1	-
Bob	3	2	1	-	1
Steve	-	-	-	-	5
Margaret	1	1	5	4	4

We will decompose the matrix manually to get a grip on how NMF works.

Step 1: Create a movie-genre matrix

First, divide the movies into two genres. Assign positive coefficients to each movie.
Use numbers from 0-3:

	Titanic	*Tiffany*	*Terminator*	*Star Trek*	*Star Wars*
genre 1	3	2	0	1	1
genre 2	1	0	3	2	2

Step 2: Create a user-genre matrix

Next, assign the users' preference for genres. Assign positive coefficients to each user.
Use numbers from 0-2:

	Ada	*Bob*	*Steve*	*Margaret*
genre 1	1.8	1	0	0
genre 2	0	0	1	2

Hint: Use your intuition! Don't try to come up with a super-accurate assignment of the numbers.

Step 3: Recompose the matrix

Now calculate the dot product of the two matrices.

Example:

> *Titanic* belongs to genre 1 with strength 2.0 and to genre 2 with 0.5
>
> *Ada* likes genre 1 with strength 2.0 and genre 2 with 1.0
>
> The recomposed value for Titanic/Ada is:
>
> 2.0 * 2.0 + 0.5 * 1.0 = 4.5

Fill the matrix below. It contains the original numbers for comparison.

user	*Titanic*	*Tiffany*	*Terminator*	*Star Trek*	*Star Wars*
Ada	5.4 *5*	4 *4*	0 *1*	1 *1*	2
Bob	3 *3*	1 *2*	0 *1*	1	1 *1*
Steve	1	0	3	2	2 *5*
Margaret	1 *1*	0 *1*	6 *5*	4 *4*	4 *4*

See how close you get to the original numbers.

Step 4: Reflection

- What movie recommendations could you generate for Steve?
- How could you make the reconstructed matrix more similar to the original?
- Would it help to have more genres?
- Are the genres created by the procedure really genres? What other properties of movies or users could thes ***hidden features*** represent?
- Would the method suffer if some of your users are *"grumpy"*?
  (i.e. they always give lower ratings)
- Would the method suffer if the data is very sparse?
  (e.g. each user gives only 1-2 reviews)

Yes, the method does suffer. It will perform better the more data we have. Additionally, if we have a new user, we usually only have very little data on the new user. —> cold start problem