D5: Databases and Information Systems
Data Mining and Matrices, SS 2015
Homework #4: NMF & CUR
Due: **16 June 2015** at 12:15 CEST

You can discuss these problems with other students, but everybody must hand in their own answers. You can use computers etc. to perform the algebraic operations, but you must show the intermediate steps (and "computer said so" is never a valid answer). You can return either computer-typeset solutions by email (but no scanned or photographed solutions are accepted), or legibly hand-written or computer-typeset solutions personally to the lecture. Notice that the DL is strict. *Remember to write your name and matriculation number to every answer sheet! If you want to discuss the solutions with the tutor, the tutorial meeting is the time to do that. If you cannot attend the tutorial meeting, you must schedule a meeting with the tutor via email.*

Problem 1 (NMF as k-means). The k-means algorithm tries to optimize the function

$$\sum_{j=1}^{k} \sum_{i \in C_j} \left\| \boldsymbol{a}_i - \boldsymbol{\mu}_j \right\|_2^2 \ , \tag{1.1}$$

where $\boldsymbol{a}_i \in \mathbb{R}^d, i = 1, \ldots, n$ are the input (row) vectors, $C_j \subset \{1, 2, \ldots, n\}, j = 1, \ldots, k, C_i \cap C_j = \emptyset$ if $i \neq j$, and $\cup_j C_j = \{1, 2, \ldots, n\}$ define the k clusters of $\boldsymbol{a}_i$, and $\boldsymbol{\mu}_j \in \mathbb{R}^d, j = 1, \ldots, k$, are the centroids for the clusters. Given a clustering, the centroid $\boldsymbol{\mu}_j$ is computed as the element-wise average, $\boldsymbol{\mu}_j = \frac{1}{|C_j|} \sum_{i \in C_j} \boldsymbol{a}_i$ (summation and division are element-wise).

Show that if all $\boldsymbol{a}_i$ are non-negative, we can write (1.1) as a special type of semi-orthogonal NMF

$$\|\boldsymbol{A} - \boldsymbol{G}\boldsymbol{M}\|_F^2 \ , \quad \boldsymbol{G}^T \boldsymbol{G} = \boldsymbol{I} \ . \tag{1.2}$$

That is, show how to transform (1.1) into (1.2) and verify that all matrices stay non-negative and that $\boldsymbol{G}$ is column-orthogonal.

Problem 2 (Pseudo-inverse for full column rank matrices). Let $n > m$ and let $\boldsymbol{X} \in \mathbb{R}^{n \times m}$ be such that $\operatorname{rank}(\boldsymbol{X}) = m$. Show that in this case the Moore–Penrose pseudo-inverse of $\boldsymbol{X}$ can be computed as

$$\boldsymbol{X}^+ = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \ . \tag{2.1}$$

Remember to make sure that $(\boldsymbol{X}^T \boldsymbol{X})^{-1}$ exists.

Problem 3 (NMF and pLSA). In the lectures the pLSA was presented as NMF optimizing the generalized KL divergence. In this problem we aim at proving why GKL is used instead of the Frobenius norm.

Recall that in pLSA, the joint probability of a document d_i and term t_j using K topics $(z_k)_{k=1}^K$ is defined as

$$\Pr[d_i, t_j] = \sum_{k=1}^{K} \Pr[z_k] \Pr[d_i \mid z_k] \Pr[t_j \mid z_k] \ . \tag{3.1}$$

Using the NMF formulation with document-term matrix $\boldsymbol{A}$ that is normalized to sum to unity and NMF factor matrices $\boldsymbol{W}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{H}$, where columns of $\boldsymbol{W}$, diagonal of $\boldsymbol{\Sigma}$, and rows of $\boldsymbol{H}$ sum to unity, we can write (3.1) as

$$\Pr[d_i, w_j] = \sum_{k=1}^{K} \sigma_{kk} w_{ik} h_{kj} = (\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{H})_{ij} \ . \tag{3.2}$$

Now, the likelihood of observing $\boldsymbol{A}$ when drawing the data from the distribution (3.2) is proportional to

$$L = L(\boldsymbol{A} \mid \boldsymbol{W}, \boldsymbol{\Sigma}, \boldsymbol{H}) = \prod_i \prod_j \Pr[d_i, w_j]^{\boldsymbol{A}_{ij}} \ . \tag{3.3}$$

Show that NMF with GKL divergence as the error metric is maximizing the likelihood L by showing that maximizing the log-likelihood $\log L(\boldsymbol{A} \mid \boldsymbol{W}, \boldsymbol{\Sigma}, \boldsymbol{H})$ is equivalent to minimizing the (generalized) KL divergence

$$D_{GKL}(\boldsymbol{A} \| \boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{H}) = \sum_i \sum_j \left(\boldsymbol{A}_{ij} \ln \frac{\boldsymbol{A}_{ij}}{(\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{H})_{ij}} - \boldsymbol{A}_{ij} + (\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{H})_{ij} \right) \ . \tag{3.4}$$

Problem 4 (CX and RRQR). Recall that an RRQR decomposition of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ is of form

$$\boldsymbol{A}\boldsymbol{\Pi} = \boldsymbol{Q}\boldsymbol{R} = \boldsymbol{Q} \begin{pmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} \\ 0 & \boldsymbol{R}_{22} \end{pmatrix} \ , \tag{4.1}$$

where $\boldsymbol{\Pi} \in \{0,1\}^{m\times m}$ is a permutation matrix, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ is an orthogonal matrix, $\boldsymbol{R}_{11} \in \mathbb{R}^{k\times k}$ is upper-triangular with positive values in diagonal, and $\boldsymbol{R}_{12} \in \mathbb{R}^{k\times(m-k)}$ and $\boldsymbol{R}_{22} \in \mathbb{R}^{(n-k)\times(m-k)}$ are arbitrary.

Let $\boldsymbol{\Pi}_k\{0,1\}^{n\times k}$ be the first k columns of $\boldsymbol{\Pi}$ and set $\boldsymbol{C} = \boldsymbol{A}\boldsymbol{\Pi}_k \in \mathbb{R}^{n\times k}$. Show that

$$\left\| \boldsymbol{A} - \boldsymbol{C}\boldsymbol{C}^+\boldsymbol{A} \right\|_\xi = \|\boldsymbol{R}_{22}\|_\xi \ , \tag{4.2}$$

where ξ is either F or 2 (i.e. we compute either the Frobenius or spectral norm).

Hint: Use the fact that $\boldsymbol{R}_{11}$ is guaranteed to be invertible and that both of the studied norms are orthogonally invariant.

Problem 5 (CX and RRQR again). Let $\boldsymbol{A}\boldsymbol{\Pi} = \boldsymbol{Q}\boldsymbol{R}$ be the RRQR factorization of $\boldsymbol{A}$ as above. Assume the factorization admits the following inequalities for some polynomials p_1 and p_2 over k and m:

$$\frac{\sigma_k(\boldsymbol{A})}{p_1(k,m)} \le \sigma_{\min}(\boldsymbol{R}_{11}) \le \sigma_k(\boldsymbol{A}) \tag{5.1}$$

$$\sigma_{k+1}(\boldsymbol{A}) \le \sigma_{\max}(\boldsymbol{R}_{22}) \le p_2(k,m)\sigma_{k+1}(\boldsymbol{A}) \ . \tag{5.2}$$

Using (4.2) from Problem 4 and the above inequalities, show that

$$\left\| \boldsymbol{A} - \boldsymbol{C}\boldsymbol{C}^+\boldsymbol{A} \right\|_2 \le p_2(k,m) \left\| \boldsymbol{A} - \boldsymbol{A}_k \right\|_2 \ , \tag{5.3}$$

where $\boldsymbol{A}_k = \boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^T$ is the rank-k truncated SVD of $\boldsymbol{A}$.

Problem 6 (Generating CUR data). A standard practise when validating that a proposed matrix factorization algorithm works in practice is to generate random data that has the kind of structure the factorization aims at finding, add some random, structure-less noise, and use the resulting matrix as an input for the algorithm. For example, for NMF, we would first choose some n, m, and k, then we would generate random matrices $\boldsymbol{W} \in \mathbb{R}_+^{n\times k}$ and $\boldsymbol{H} \in \mathbb{R}_+^{k\times m}$, multiply them to obtain $\boldsymbol{A} = \boldsymbol{W}\boldsymbol{H}$, and add some noise to $\boldsymbol{A}$.

Design a method that creates random synthetic matrices for CUR decomposition. That is, explain how to generate matrices $\boldsymbol{C} \in \mathbb{R}^{n\times k}$, $\boldsymbol{U} \in \mathbb{R}^{k\times k}$, and $\boldsymbol{R} \in \mathbb{R}^{k\times m}$ $(k < n, m)$ such that matrix $\boldsymbol{A} = \boldsymbol{C}\boldsymbol{U}\boldsymbol{R}$ has k columns that are exactly the columns of $\boldsymbol{C}$ and k rows that are exactly the rows of $\boldsymbol{R}$. The factor matrices cannot be completely random, but try to have as much randomness as possible.