IEOR E4525
M. Haugh

Due Wednesday April 6^{th} 2016

Assignment 6
Answer Questions 1 to 4. Question 5 is optional.

1. *(PCA Via Optimization)*

   Let $\mathbf{x} = (x_1, \ldots, x_d)^\top$ denote a d-dimensional random vector with variance-covariance matrix, $\mathbf{\Sigma}$. Let $\boldsymbol{\gamma}_i$ be the eigen vector of $\mathbf{\Sigma}$ corresponding to the i^{th} largest eigen value, λ_i. Prove by induction that $\boldsymbol{\gamma}_i$ solves

$$\begin{array}{rl} \max\limits_{\mathbf{a}} & \mathrm{Var}(\mathbf{a}^\top \mathbf{x}) \\ \text{subject to} & \mathbf{a}^\top \mathbf{a} \; = \; 1 \\ & \mathbf{a}^\top \boldsymbol{\gamma}_j \; = \; 0, \quad j = 1, \ldots, i-1. \end{array}$$

   for $i = 1, \ldots, d$.

2. *(Missing Data Problems: Barber Section 15.5)*

   Recall the missing data formulation where we seek to solve

$$\min_{\mathbf{B},\mathbf{Z}} \sum_{i=1}^{n} \sum_{j=1}^{d} \gamma_{j,i} \left[x_{j,i} - \sum_{k=1}^{M} b_{j,k} z_{k,i} \right]^2 \tag{1}$$

   where

$$\gamma_{j,i} \; := \; \begin{cases} 1, & \text{if } x_{j,i} \text{ is available} \\ 0, & \text{otherwise.} \end{cases}$$

   The problem is not *jointly* convex in $\mathbf{B}$ and $\mathbf{Z}$ and therefore we can only expect to obtain local minima when we attempt to solve (1). Note also, however, that for a fixed $\mathbf{B}$ the objective function in (1) is convex in $\mathbf{Z}$. Similarly if $\mathbf{Z}$ is fixed then the objective is convex in $\mathbf{B}$. We will therefore use an iterative algorithm to compute local minima.

   (a) **Optimize Z for fixed B:** show that the first order conditions for solving this problem amount to solving n linear systems of equations. It may be that one or more of these systems is *under-determined*. (This occurs when the number of observations in a column of X is less than M.) Does this present any difficulty?

   (b) **Optimize B for fixed Z:** show that the first order conditions for solving this problem amount to solving d linear systems of equations.

   (c) Write a computer program that iterates (a) and (b) until convergence to within a given error tolerance, ϵ. Your code should take as input the number of basis elements, M, the matrixes $\mathbf{X}$ and $\mathbf{\Gamma}$, the error tolerance, ϵ, and the starting matrix, $\mathbf{B}_0$ say. You should also decide what “convergence” means in this problem.

Finally, your code should compute the *root-mean-squared error* (RMSE) for the local optimum you have found. The RMSE is calculated as the square-root of: the optimal objective function divided by the total number of observations in the $\mathbf{X}$ matrix. (Note that the number of observations will not be nd if some observations are missing.)

3. *(Recommender Systems for Movies)*

   (a) Run your code from Problem 2 on the movie database that you worked with in Assignment #5. In particular, you should estimate the missing elements of the ratings matrix $\mathbf{X}$. You can take $M = 5$ but feel free to try other values as well. You will also need to decide whether d should represent the number of movies (as in the slides) or the number of critics.

   You can do this in `R` or `Matlab` but the linear systems may be too big for your computer so feel free to reduce the number of users and movies to a more manageable size. But also note that the sparse matrix functionality in `Matlab` may enable you to handle larger matrices. Type **`help sparse`** at the `Matlab` prompt to see how to use this functionality. There is also sparse matrix functionality available in `R` via the *Matrix* and *SparseM* libraries, for example. They may also be useful.)

   (b) Now run your code from part (a) K times, starting from a randomly generated $d \times M$ matrix, $\hat{\mathbf{B}}$, each time.

   Compute the RMSE for each of the K runs. What do you notice about the various local minima that you have obtained? (You can choose K yourself so that the problem is manageable on your system.)

   (c) Suppose now that you have a $d \times L$ *genre* matrix $\mathbf{G}$ where

   $$\mathbf{G}_{ij} = \begin{cases} 1, & \text{if the } i^{th} \text{ movie is in genre } j \\ 0, & \text{otherwise} \end{cases}$$

   for the movies $i = 1, \ldots, d$ and genres $j = 1, \ldots, L$. Explain how you could use your code to recommend movies by genre to users.

4. *(Page-Rank)*

   (a) Write a computer program to determine the page-rank of a system of web-pages.

   (b) Run your code on Figure 14.47 from HTF with $\epsilon = .15$. Does the resulting page-rank vector make sense? (Note that the page-rank is only unique if the associated Markov chain is irreducible. This is why we must have $\epsilon > 0$.)

   (c) Rerun your code for different values of ϵ. Does the page-rank vector respond in the way you would expect it to?

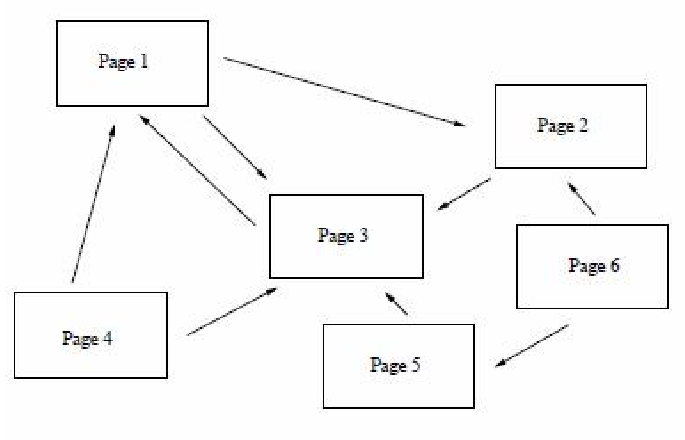

Figure 14.47 from HTF: Example of a small network

5. *(***Optional!** *Exercise 14.23 from HTF: Non-Negative Matrix Factorization)*

   A function $g(x, y)$ is said to *minorize* a function $f(x)$ if

$$g(x, y) \leq f(x), \quad g(x, x) = f(x)$$

   for all x, y in the domain. This is useful for maximizing $f(x)$ since it is easy to show that $f(x)$ is nondecreasing under the update

$$x^{s+1} = \operatorname*{argmax}_x g(x, x^s).$$

   (a) Consider maximization of the function $L(\mathbf{W}, \mathbf{H})$ (as defined in expression (18) in the slides), written here without the matrix notation

$$L(\mathbf{W}, \mathbf{H}) = \sum_{i=1}^{d} \sum_{j=1}^{n} \left[x_{ij} \log \left(\sum_{k=1}^{r} w_{ik} h_{kj} \right) - \sum_{k=1}^{r} w_{ik} h_{kj} \right].$$

   Using the concavity of $\log(x)$, show that for any set of r values $y_k \geq 0$ and $0 \leq c_k \leq 1$ with $\sum_{k=1}^{r} c_k = 1$,

$$\log \left(\sum_{k=1}^{r} y_k \right) \geq \sum_{k=1}^{r} c_k \log(y_k / c_k).$$

   Hence

$$\log \left(\sum_{k=1}^{r} w_{ik} h_{kj} \right) \geq \sum_{k=1}^{r} \frac{a_{ikj}^s}{b_{ij}^s} \log \left(\frac{b_{ij}^s}{a_{ikj}^s} w_{ik} h_{kj} \right)$$

   where $a_{ikj}^s = w_{ik}^s h_{kj}^s$ and $b_{ij}^s = \sum_{k=1}^{r} w_{ik}^s h_{kj}^s$, and s indicates the current iteration.

(b) Hence show that the function

$$
\begin{aligned}
g(\mathbf{W}, \mathbf{H} \,|\, \mathbf{W}^s, \mathbf{H}^s) \quad := \quad & \sum_{i=1}^{d} \sum_{j=1}^{n} \sum_{k=1}^{r} x_{ij} \frac{a_{ikj}^s}{b_{ij}^s} \left(\log(w_{ik}) + \log(h_{kj}) + \log\left(\frac{b_{ij}^s}{a_{ikj}^s} \right) \right) \\
& - \sum_{i=1}^{d} \sum_{j=1}^{n} \sum_{k=1}^{r} w_{ik} h_{kj}
\end{aligned}
$$

minorizes $L(\mathbf{W}, \mathbf{H})$.

(c) Set the partial derivatives of $g(\mathbf{W}, \mathbf{H} \,|\, \mathbf{W}^s, \mathbf{H}^s)$ to zero and hence derive the updating steps (19) and (20) that are also given in the slides.