ELEN 4720: Machine Learning for Signals, Information and Data

Homework 3

Please read these instructions to ensure you receive full credit on your homework. Submit the written portion of your homework as a *single* PDF file through Courseworks (less than 5MB). In addition to your PDF write-up, submit all code written by you in their original extensions through Courseworks (e.g., .m, .r, .py, .c). Any coding language is acceptable, but do not submit notebooks. Also, do not wrap your files in .rar, .zip, .tar and do not submit your write-up in .doc or other file type. When resubmitting homeworks, please be sure to resubmit *all files*. Your grade will be based on the contents of one PDF file and the original source code. Additional files will be ignored. We will not run your code, so everything you are asked to show should be put in the PDF file. Show all work for full credit.

Problem 1 (Matrix factorization for collaborative filtering)

In this problem, you are given an $n_1 \times n_2$ matrix M with missing values. Let $\Omega = \{(i,j) : M_{ij} \text{ is measured}\}$. This matrix will come from users' ratings of movies – here, we have n_1 users and n_2 movies, and the goal is to predict users affinity for movies which they have not yet rated.

We will seek a factorization $M \approx UV$ of the matrix M. Here, $U \in \mathbb{R}^{n_1 \times r}$ is a *user matrix* – each row corresponds to the preferences of some user, while $V \in \mathbb{R}^{r \times n_2}$ is an *item matrix* – each column corresponds to some movie. Note that the matrix UV has rank at most r. Our prediction for the affinity of user i for item j is simply $[UV]_{ij}$, i.e., the dot product between the i-th row of U and the j-th column of V. We will determine U and V by solving

$$\min_{U,V} \varphi(U,V) \equiv \tfrac{1}{2}\|\mathcal{P}_\Omega[UV - M]\|_F^2 + \lambda\|U\|_F^2 + \lambda\|V\|_F^2,$$

with $U \in \mathbb{R}^{n_1 \times r}$ and $V \in \mathbb{R}^{r \times n_2}$. We will develop a method based on gradient descent for this problem, starting from a random initialization $U_0 \sim_{\text{iid}} \mathcal{N}(0,1)$ and $V_0 \sim_{\text{iid}} \mathcal{N}(0,1)$, and

$$\begin{aligned} U_{k+1} &= U_k - t\nabla_U \varphi \\ V_{k+1} &= V_k - t\nabla_V \varphi. \end{aligned}$$

- (a) What are the gradients $\nabla_U \varphi$ and $\nabla_V \varphi$? In differentiating the loss, you may find it helpful to note that

$$\tfrac{1}{2}\|\mathcal{P}_\Omega[UV - M]\|_F^2 = \sum_{ij \in \Omega} \tfrac{1}{2}\left(\sum_k U_{ik}V_{kj} - M_{ij}\right)^2.$$

- (b) Please implement gradient descent for this problem, and run your code on the user-movie ratings dataset provided on Courseworks. For your algorithm, use $r = 10$, $\lambda = 1/4$ and $t = 0.0025$. Train the model on the larger training set for 5000 iterations.[1]

  b.1) Run your code 10 times. For each run, initialize your u_i and v_j vectors as $N(0, I)$ random vectors. On a *single* plot, show the objective $\varphi(U_k, V_k)$ for iterations $k = 2$ to $5,000$ for each run. In a table, show in each row the final value of the training objective function next to the RMSE on the testing set. Sort these rows according to decreasing value of the objective function.

[1] You are welcome to experiment with other step sizes, momentum, backtracking, etc. to improve convergence behavior! You are also welcome to experiment with choosing r and λ by cross validation on a held out set.

b.2) This question will explore the movie embeddings, i.e., the columns $V = [v_1 \mid v_2 \mid \cdots \mid v_{n_2}]$ of the matrix V. For the run with the smallest objective value, pick the movies "Star Wars" "My Fair Lady" and "Goodfellas" and for each movie find the 10 closest movies according to Euclidean distance using their respective embeddings v_j. List the query movie, the five nearest movies and their distances. A mapping from index to movie is provided with the data.

Problem 2 (Nonnegative matrix factorization)

In this problem you will factorize an $n_1 \times n_2$ matrix X into a rank-r approximation WH, where W is $n_1 \times r$, H is $r \times n_2$ and all values in the matrices are nonnegative. We will solve

$$\min_{W,H} \varphi(W,H) \equiv \tfrac{1}{2}\|WH - X\|_F^2 + \mathcal{I}_{W\geq 0} + \mathcal{I}_{H\geq 0}$$

with a proximal gradient method. Recall that

$$\mathcal{I}_{W\geq 0} = \begin{cases} 0 & W \geq 0 \\ \infty & \text{else,} \end{cases}$$

and let

$$\mathcal{L}(W,H) = \tfrac{1}{2}\|WH - X\|_F^2 \tag{1}$$

denote the loss for this problem. The proximal gradient iteration takes the form

$$W_{k+1} = [W_k - t\nabla_W \mathcal{L}(W_k, H_k)]_+ \tag{2}$$
$$H_{k+1} = [H_k - t\nabla_H \mathcal{L}(W_k, H_k)]_+ \tag{3}$$

The entries of W and H can be initialized randomly to a positive number, e.g., from a Uniform(1,2) distribution.

The data to be used for this problem consists of 8447 documents from *The New York Times*. (See below for how to process the data.) The vocabulary size is 3012 words. You will need to use this data to construct the matrix X, where X_{ij} is the number of times word i appears in document j. Therefore, X is 3012×8447 and most values in X will equal zero.

a) What are the gradients $\nabla_W \mathcal{L}$ and $\nabla_H \mathcal{L}$?

b) Implement and run the NMF algorithm on this data. Set[2] the rank to 25 and run for 1,000 iterations, with $t = 10^{-6}$. This corresponds to learning 25 topics. Plot the objective as a function of iteration.

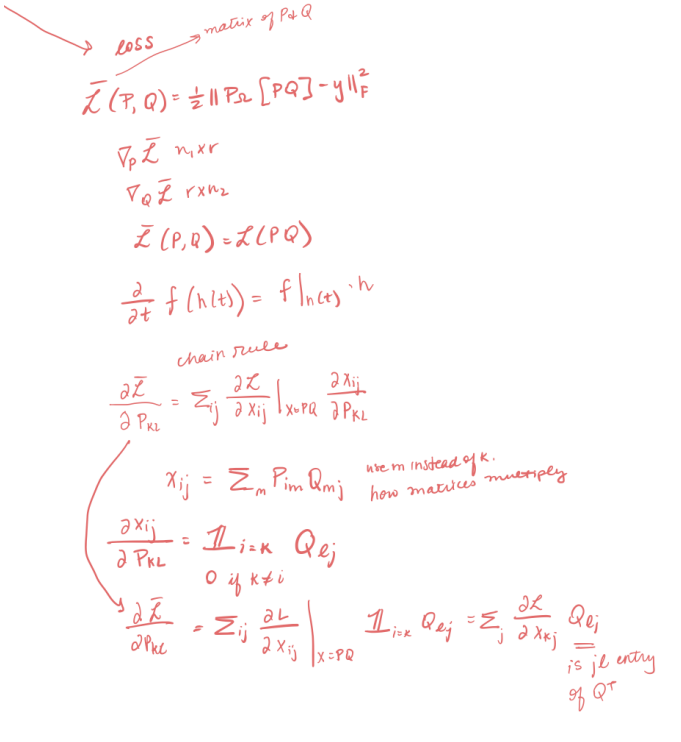
c) After running the algorithm, normalize the columns of W so they sum to one. For each column of W, list the 10 words having the largest weight and show the weight. The ith row of W corresponds to the ith word in the "dictionary" provided with the data. Organize these lists in a 5×5 table.

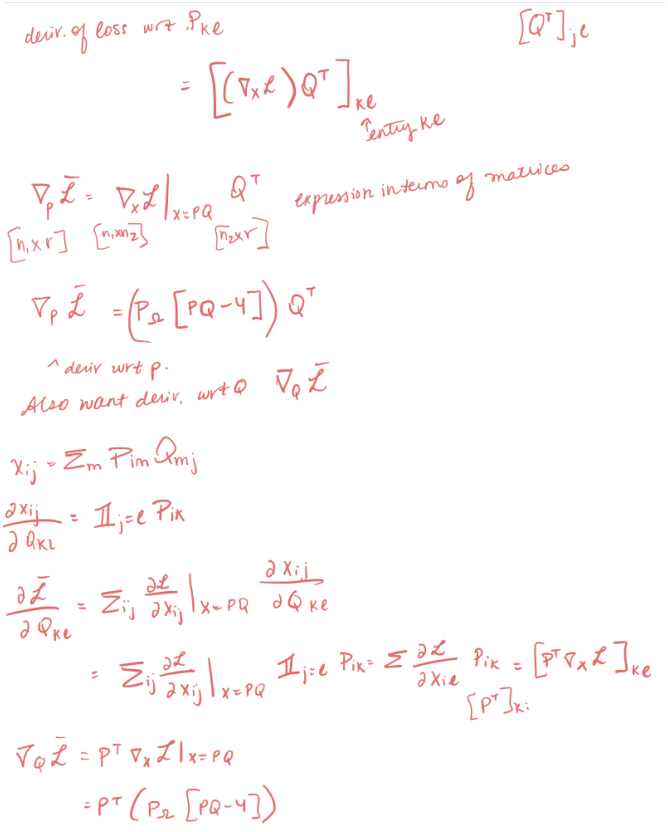
Comments about Problem 2: Each row in `nyt_data.txt` corresponds to a single document. It gives the index of words appearing in that document and the number of times they appear. It uses the format "idx:cnt" with commas separating each unique word in the document. Any index that doesn't appear in a row has a count of zero for that word. The vocabulary word corresponding to each index is given in the corresponding row of `nyt_vocab.dat`.

[2] As with the previous problem, you are welcome to modify these numbers and the algorithm.

Michelle Campoli (mec2308)
ELEN 4720: Machine Learning for Signals, Data, Info
Homework 3
Fall 2022

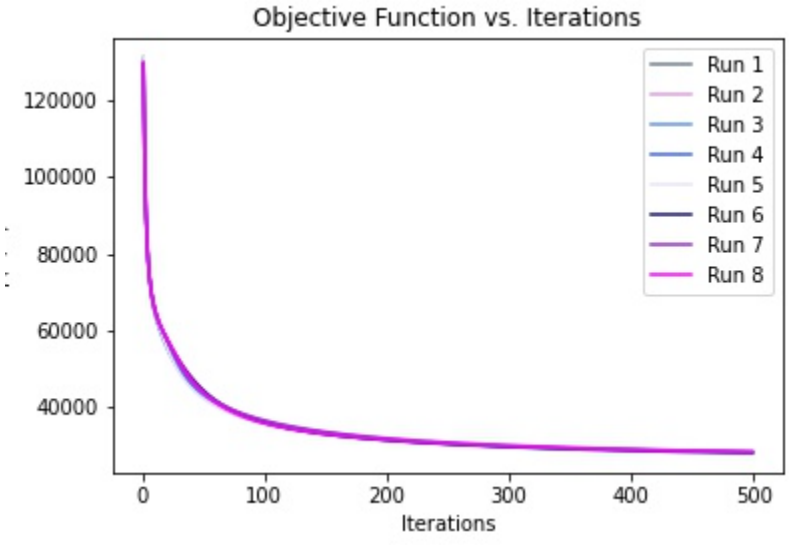
1. (a) Gradient of $\varphi(U, V)$ w.r.t. U and V.
Implemented in terms of L(P,Q) where L= φ , P = U, and Q = V.





$$\therefore \nabla_U \varphi(U,V) = (UV - M)V^T + \lambda U \;; \quad \nabla_V \varphi(U,V) = U^T(UV - M) + \lambda V$$

(b) 1. Plot the objective function over 5,000 iterations for 10 separate runs.
Running this function took a very long time and the function did not decrease substantially after 500 iterations, so I ran the code for 500 iterations 8 times. As shown below, the lines overlap so I cut the runs at 8.



Runs	Objective Function $\varphi(U,V)$	RMSE
1	28068.01720009729	3.29735450781985e-06
4	28091.29616282581	1.1273991458498974e-06
6	28294.31945933961	3.5339525628847806e-06
8	28331.902804047742	1.9219907834755354e-06
3	28377.22399727597	1.113555221657035e-06
2	28386.814016584227	1.068714924458571e-05
7	28430.588864180358	4.820205491545369e-06
5	28564.27934024715	2.434078601756626e-06

(b) 2. List of closest movies.

Run = 1 was chosen for exploring the movie embeddings V or Q. The smallest objective function value was 28068.

Star Wars (1977)

Similar Movie Title = Empire Strikes Back, The (1980)

distance = 0.425031418680749

Similar Movie Title = Return of the Jedi (1983)

distance = 0.694688187915796

Similar Movie Title = Raiders of the Lost Ark (1981)

distance = 0.9752364904402572

Similar Movie Title = Usual Suspects, The (1995)

distance = 1.0006071937937053

Similar Movie Title = Princess Bride, The (1987)

distance = 1.0772211942186167

My Fair Lady (1964)

Similar Movie Title = Oscar & Lucinda (1997)

distance = 1.2078347641315574

Similar Movie Title = King of the Hill (1993)

distance = 1.3679829815455287

Similar Movie Title = Evita (1996)

distance = 1.4706860910905786

Similar Movie Title = Beauty and the Beast (1991)

distance = 1.4849417986547604

Similar Movie Title = Cinderella (1950)

distance = 1.4868084904730532

GoodFellas (1990)

Similar Movie Title = Bonnie and Clyde (1967)

distance = 0.7480663606920528

Similar Movie Title = Godfather: Part II, The (1974)

distance = 0.8083079037881894

Similar Movie Title = Apocalypse Now (1979)

distance = 0.8226365723667162

Similar Movie Title = Quiz Show (1994)

distance = 0.9141892889357811

Similar Movie Title = Good, The Bad and The Ugly, The (1966)

distance = 1.0315513193232972

2. (a) Gradient of L(W,H) w.r.t. W and H:

hw 3 prob 2

$$\varphi(W,H) = \frac{1}{2}\|WH - X\|_F^2 + \mathbb{I}_{W\geq 0} + \mathbb{I}_{H\geq 0}$$

$$\mathcal{L}(W,H) = \frac{1}{2}\|WH - Y\|_F^2 = \mathcal{L}(WH)$$

$$\nabla_W \bar{\mathcal{L}} = \nabla_X \mathcal{L}\,|_{WH}\, H^T \quad \text{(sub Q for H from last example)}$$

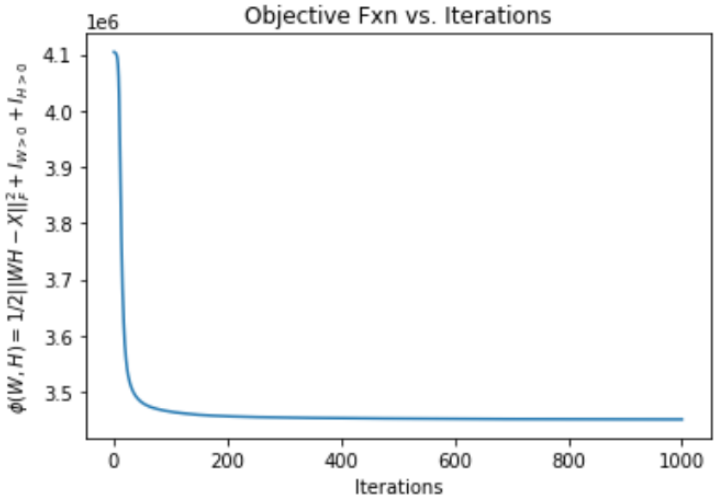
$$= (WH - Y)H^T \quad \text{right side}$$

$$\nabla_H \bar{\mathcal{L}} = W^T \nabla_X \mathcal{L}\,|_{WH}$$

$$= W^T(WH - Y) \quad \text{left side}$$

$$\therefore \nabla_W L(W,H) = (WH - X)H^T \;;\; \nabla_H L(W,H) = U^T(UV - M)$$

(b) Plot the objective as a function of iteration.



(c) For each column of W, list the 10 words having the largest weight and show the weight.

The weights when calculated and rounded were very small, so the following weights shown are x 1e-3.

sell : 36.8 company : 27.5 sale : 25.2 buy : 22.1 market : 19.0 price : 16.9 business : 16.7 store : 16.2 industry : 15.2 product : 14.9	school : 42.9 student : 29.8 class : 15.8 child : 13.7 program : 13.7 college : 13.2 education : 13.1 teacher : 11.2 parent : 10.0 group : 9.0	police : 33.0 kill : 20.9 man : 18.4 officer : 18.1 arrest : 13.8 crime : 13.8 death : 13.2 charge : 12.7 victim : 11.5 fire : 10.2	father : 64.1 son : 53.5 mrs : 50.4 daughter : 41.7 graduate : 32.0 mother : 32.0 marry : 28.4 retire : 27.0 receive : 23.3 president : 19.5	plant : 10.4 damage : 9.9 cause : 9.7 problem : 9.7 water : 8.9 official : 7.9 gas : 7.0 agency : 6.8 scientist : 6.7 system : 6.0
military : 14.2 official : 13.7 war : 11.7 government : 11.5 american : 10.5 states : 10.1 force : 9.1 leader : 8.9 peace : 7.3 meeting : 6.8	pay : 16.2 money : 15.6 state : 13.7 budget : 12.7 program : 11.2 bill : 10.8 tax : 10.7 plan : 10.4 cost : 10.3 cut : 9.9	food : 18.3 restaurant : 11.4 fresh : 11.2 serve : 11.0 taste : 10.0 eat : 8.6 cook : 8.0 dinner : 8.0 fish : 7.6 pound : 7.4	music : 20.0 play : 17.7 performance : 12.3 film : 11.7 audience : 9.9 stage : 8.9 production : 8.7 theater : 8.7 dance : 8.5 perform : 8.4	team : 33.0 game : 27.9 season : 24.29 player : 24.1 play : 19.4 coach : 15.8 baseball : 11.6 league : 11.5 football : 8.7 sport : 8.4
building : 25.7 city : 25.2 resident : 16.29 build : 15.2 area : 14.8 house : 12.6 community : 10.9 project : 10.4 site : 9.7 live : 9.6	percent : 32.4 rate : 18.8 rise : 17.5 price : 15.5 market : 15.2 fall : 11.5 increase : 11.1 low : 10.5 economy : 10.1 decline : 9.4	health : 16.0 drug : 15.7 doctor : 14.1 study : 12.8 medical : 12.4 patient : 11.4 treatment : 10.6 care : 9.5 hospital : 9.1 disease : 7.9	book : 22.3 write : 21.1 editor : 19.2 life : 15.4 writer : 11.79 article : 11.70 story : 11.0 man : 9.0 author : 8.7 read : 8.5	woman : 15.2 man : 14.5 child : 12.4 tell : 11.7 friend : 11.6 live : 11.5 young : 10.8 home : 10.6 life : 10.5 family : 9.9
art : 25.1 artist : 15.2 museum : 11.4 collection : 10.8 exhibition : 10.0 photograph : 9.7 painting : 8.3 image : 8.1 century : 7.7 gallery : 7.4	color : 9.1 wall : 7.5 white : 7.2 small : 7.1 design : 7.0 wear : 6.89 red : 6.7 fashion : 6.4 light : 5.8 foot : 5.8	country : 20.4 american : 15.2 states : 11.7 world : 11.29 political : 9.6 economic : 8.8 government : 8.7 nation : 8.5 power : 8.1 americans : 8.1	computer : 21.6 television : 19.7 network : 14.5 technology : 11.5 system : 11.4 information : 10.5 program : 10.1 internet : 9.9 site : 9.7 video : 8.8	mile : 15.9 travel : 12.4 hour : 12.20 train : 11.9 car : 11.4 road : 10.1 trip : 10.0 driver : 9.29 traffic : 8.6 fly : 8.5
company : 32.09 executive : 21.8 president : 14.6 chief : 13.7 business : 13.5 share : 12.4 chairman : 11.2 financial : 10.6 yesterday : 10.5 announce : 9.4	thing : 22.1 feel : 14.1 really : 13.0 ask : 11.9 lot : 11.6 happen : 10.7 put : 10.6 little : 10.0 question : 9.0 kind : 8.8	case : 17.9 lawyer : 15.9 court : 15.8 law : 13.6 judge : 10.9 charge : 10.2 legal : 9.7 issue : 8.0 official : 7.9 rule : 7.5	win : 28.29 second : 21.3 victory : 15.5 play : 15.4 score : 13.8 third : 13.1 game : 12.9 point : 12.6 lose : 10.9 final : 10.6	campaign : 20.1 vote : 17.7 political : 16.8 party : 13.7 election : 13.7 candidate : 13.2 republican : 13.2 leader : 11.2 democratic : 10.9 support : 9.6