CSE 8803 Homework 4

Jingyi Feng

December 10, 2023

1 Solution

$$min_{W,H} \|X - WH\|_F \qquad X \in \mathbb{R}^{m\times n}, W \in \mathbb{R}^{m\times k}, H \in \mathbb{R}^{k\times n}, k < min(m,n) \tag{1}$$

The above nonconvex optimization problem has a global minimum based on SVD. According to *Eckart–Young–Mirsky theorem*, the best k-rank approximation of X, X_k, is obtained from truncated SVD, where $X_k = \sum_{i=1}^{k} \sigma_i u_i v_i^*$. u_i, v_i are the ith column vectors of U and V in $X = U\Sigma V^*$ correspondingly, and σ_i is the ith diagonal entry of Σ.

Proof: By the definition of Frobenius norm, we know

$$\|X - X_k\|_F^2 = \| \sum_{i=k+1}^{n} \sigma_i u_i v_i^* \|_F^2 = \sum_{i=k+1}^{n} \sigma_i^2 \tag{2}$$

Let C_k be any k-rank $m \times n$ matrix, $A = X - C_k$ and $B = C_k$. We know $A + B = X$. Let A_i and B_i be the i-rank approximation of A and B. Following the triangle inequality of norms, we know for spectral norm, $\|A+B\|_\sigma \leq \|A\|_\sigma + \|B\|_\sigma$. Hence, we can know

$$\begin{aligned} \sigma_i(A)+\sigma_j(B) &= \|A-A_{i-1}\|_\sigma + \|B-B_{j-1}\|_\sigma \\ &\geq \|A+B-(A_{i-1}+B_{j-1})\|_\sigma \\ &\geq \|X-X_{i+j-2})\|_\sigma \\ &= \sigma_{i+j-1}(X) \end{aligned} \tag{3}$$

Let $j=k+1$. C_k has rank k, hence $\sigma_{k+1}(B)=0$. Therefore we have $\sigma_i(X-C_k) \geq \sigma_{k+i}(X)$. Take sum on i, finally we have

$$\|X-C_k\|_F^2 = \sum_{i=1}^{n} \sigma_i(X-C_k)^2 \geq \sum_{i=k+1}^{n} \sigma_i(X)^2 = \|X-Xk\|_F^2 \quad \blacksquare \tag{4}$$

Because $W \in \mathbb{R}^{m\times k}, H \in \mathbb{R}^{k\times n}$, WH is a rank k matrix. Hence $\|X-WH\|_F$ reaches global minimum when $WH = X_k$.

1.1 Algorithm

Compute the economic SVD decomposition of matrix X. We get $X = \hat{U}\hat{\Sigma}\hat{V}^*$. Notice that $\hat{U} \in \mathbb{R}^{m\times k}, \hat{\Sigma} \in \mathbb{R}^{k\times k}, \hat{V}^* \in \mathbb{R}^{k\times n}$. Hence we can let $W = \hat{U}$ and $H = \hat{\Sigma}\hat{V}^*$, or $W = \hat{U}\hat{\Sigma}^{\frac{1}{2}}$ and $H = \hat{\Sigma}^{\frac{1}{2}}\hat{V}^*$, or $W = \hat{U}\hat{\Sigma}$ and $H = \hat{V}^*$, based on our choices.

2 Solution

We first compute the least square problem without non-negative constraint, then show the solution is exactly same with the solution with non-negative constraint.

For the NLS problem

$$min_{x\geq 0} \|Ax-b\|_2 \qquad A \in \mathbb{R}_+^{m\times 1}, b \in \mathbb{R}_+^{m\times 1}, x \in \mathbb{R}_+ \tag{5}$$

we firstly consider the regular LS problem

$$min_x \|Ax-b\|_2 \qquad A \in \mathbb{R}_+^{m\times 1}, b \in \mathbb{R}_+^{m\times 1}, x \in \mathbb{R} \tag{6}$$

To solve LS problems, we can use normal equation $A^TAx = A^Tb$, which gives the solution $x = \frac{A^Tb}{A^TA}$. Now we add the non-negative constraint, the inner product $A^Tb = A_1b_1 + A_2b_2 + ... + A_mb_m \geq 0$ because $A \in \mathbb{R}_+^{m\times 1}$ and $b \in \mathbb{R}_+^{m\times 1}$. Easy to see $\|A\|^2 > 0$. Therefore, $x \geq 0$, which gives us $argmin_{x\geq 0} \|Ax-b\|_2$. Hence the closed form solution of the above NLS problem is exactly the general solution $x = \frac{A^Tb}{A^TA}$ for regular LS problem.

2.1 Algorithm

See algorithm 1.

Algorithm 1 NLS

Require: $A \in \mathbb{R}_+^{m\times 1}, b \in \mathbb{R}_+^{m\times 1}$

$p, q \leftarrow 0$

$p = A^TA$

$q = A^Tb$

$x \leftarrow q/p$

return x

3 Solution

For the special case, k=2, in the following NLS problem,

$$min_{x\geq 0} \|Ax-b\|_2 \qquad A \in \mathbb{R}_+^{m\times 2}, b \in \mathbb{R}_+^{m\times 1}, x \in \mathbb{R}_+^{2\times 1} \tag{7}$$

we can write it in the the form of

$$min_{x_1\geq 0, x_2\geq 0} \|A_1x_1 + A_2x_2 - b\|_2 \qquad A \in \mathbb{R}_+^{m\times 2}, b \in \mathbb{R}_+^{m\times 1}, x_1, x_2 \in \mathbb{R}_+ \tag{8}$$

We can apply the block principal pivoting method to search for the optimal active set on the index set $V = \{1, 2\}$ to get corresponding optimization problem and its solution. Active set methods is solving an unconstrained least squares problem for the nonzero variables x_P and setting the rest variables x_A to zeros, where $A \cup P = V$ and $A \cap P = \varnothing$. However, we can enumerate the active sets and passive sets because $|V| = 2$.

A	P	loss $f(x)$	optimization problem
$\{1, 2\}$	$\varnothing$	$\|b\|_2$	$min_{x \geq 0} \|b\|_2$
$\{1\}$	$\{2\}$	$\|A_2 x_2 - b\|_2$	$min_{x_2 \geq 0} \|A_2 x_2 - b\|_2$
$\{2\}$	$\{1\}$	$\|A_1 x_1 - b\|_2$	$min_{x_1 \geq 0} \|A_1 x_1 - b\|_2$
$\varnothing$	$\{1, 2\}$	$\|A_1 x_1 + A_2 x_2 - b\|_2$	$min_{x_1 \geq 0, x_2 \geq 0} \|A_1 x_1 + A_2 x_2 - b\|_2$

For each possible passive set P, we have a corresponding optimization problem. Hence, we need to search through all of those problems to get solutions for $x \geq 0$, and choose the x with the minimal $f(x)$, respectively, to be the optimal solution.

According to problem 2, we can always have the optimal solution $x \geq 0$ by solving $min_{x_2 \geq 0} \|A_2 x_2 - b\|_2$ or $min_{x_1 \geq 0} \|A_1 x_1 - b\|_2$ with unique optimal solution $x_1 = \frac{A_1^T b}{A_1^T A_1}$ or $x_2 = \frac{A_2^T b}{A_2^T A_2}$. If $P = \{1, 2\}$, $x_1 > 0$ and $x_1 > 0$. Hence the solution $x = [x_1, x_2]^T$ is not the solution of either $min_{x_2 \geq 0} \|A_2 x_2 - b\|_2$ nor $min_{x_1 \geq 0} \|A_1 x_1 - b\|_2$. If $P = \varnothing$, consider the solution $argmin_{x_i \geq 0} \|A_i x_i - b\|_2 = \frac{A_i^T b}{A_i^T A_i}$, where $A_i x_i$ is exactly the projection of b onto the span of A_i. Therefore, we know

$$\|A_i x_i - b\|_2^2 = \|Proj_{A_i} b\|^2 = \|b\|^2 - \|Proj_{A_i^\perp} b\|^2 \leq \|b\|^2 \tag{9}$$

Because $\|A_i x_i - b\|_2 \leq \|b\|$, we know that $min_{x \geq 0} \|b\|_2$ is a special case of $min_{x_i \geq 0} \|A_i x_i - b\|_2$ when $b \perp A_i x_i$, i.e. $P = \varnothing$ can be solved by solving either $P = \{1\}$ or $P = \{2\}$. WLOG, we assume it is included in $P = \{1\}$.

In conclusion, we need to brute-force the following problems.

$$\begin{aligned} &min_{x_1 \geq 0, x_2 \geq 0} \|A_1 x_1 + A_2 x_2 - b\|_2 \\ &min_{x_1 \geq 0} \|A_1 x_1 - b\|_2 \\ &min_{x_2 \geq 0} \|A_2 x_2 - b\|_2 \end{aligned} \tag{10}$$

Consider the following relationship between those problems represented in Figure 1. If we

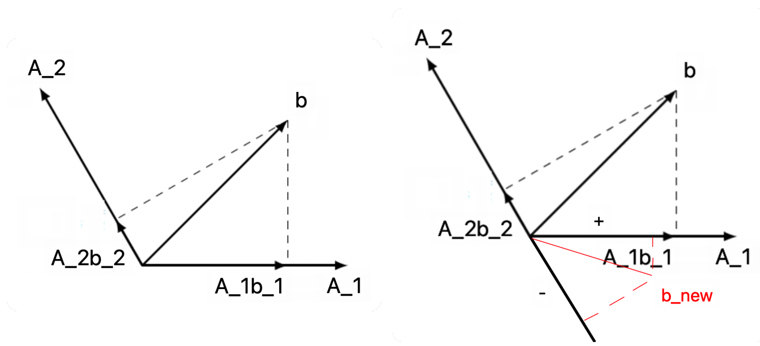

Figure 1: Projection onto A_1 and A_2

want to project b_{new} onto both the direction of A_1 and A_2 to solve $min_{x_1 \geq 0, x_2 \geq 0} \|A_1 x_1 + A_2 x_2 - b\|_2$, we can always find the closed solution x such that $x = [x_1, 0]^T$ or $x = [0, x_2]^T$ where $x_1, x_2 \in \mathbb{R}^+$ to approximate $x = [x_1, x_2]^T$ where $x_1 \cdot x_2 \in \mathbb{R}^-$. To put it bluntly, we can solve $min_{x_1 \geq 0, x_2 \geq 0} \|A_1 x_1 + A_2 x_2 - b\|_2$ by ignoring the negative x_1 or x_2, set it to 0, and transform the original optimization problem $min_{x_1 \geq 0, x_2 \geq 0} \|A_1 x_1 + A_2 x_2 - b\|_2$ into the optimization problem $min_{x_1 \geq 0} \|A_1 x_1 - b\|_2$ or $min_{x_2 \geq 0} \|A_2 x_2 - b\|_2$. By the proof of active set method, we know the error is convergent in iterations. Based on above explanation, we can solve for $x \in \mathbb{R}_+^{2 \times 1}$ using the following algorithm.

3.1 Algorithm

See algorithm 2.

Algorithm 2 Rank2NLS

Require: $A = [A_1, A_2] \in \mathbb{R}_+^{m\times 2}, b \in \mathbb{R}_+^{m\times 1}$
  $x = [x_1, x_2]^T \in \mathbb{R}_+^{2\times 1} \leftarrow argmin_x \|Ax - b\|_2$
  if $x_1 > 0$ and $x_2 > 0$ **then**
    return x
  else
    $x_1 \leftarrow (A_1^T b)/(A_1^T A_1)$
    $x_2 \leftarrow (A_2^T b)/(A_2^T A_2)$
    if $x_1 \cdot \|A_1\| \geq x_2 \cdot \|A_2\|$ **then**
      $x \leftarrow [x_1, 0]^T$
    else
      $x \leftarrow [0, x_2]^T$
    end if
  end if
  return x

4 Solution

Consider the NMF problem with matrix A and we want to find rank-3 approximation of A. The optimization problem is

$$min_{W\geq 0, H\geq 0} \|A - WH\|_F \qquad A \in \mathbb{R}_+^{m\times n}, W \in \mathbb{R}_+^{m\times 3}, H \in \mathbb{R}_+^{3\times n} \tag{11}$$

which is equivalent in Frobenius norm to

$$min_{H\geq 0, W\geq 0} \|W^T H^T - A^T\|_F \qquad A \in \mathbb{R}_+^{m\times n}, W \in \mathbb{R}_+^{m\times 3}, H \in \mathbb{R}_+^{3\times n} \tag{12}$$

General idea We are going to apply alternating NLS algorithm in Problem 2 to solve this NMF problem by solving W and H^T column by column based on (13), where w_i is the ith column of W, h_i^T is the ith row of H or the ith column of H^T.

$$\|WH - A\|_F = \|\sum_{i=1}^{k} w_i h_i^T - A\|_F = \sum_{i=1}^{k}\sum_{j=1}^{n} \|w_i h_{ij} - A_j\|_2 \tag{13}$$

In each iteration on i, we update one pair of columns in W and H^T by minimizing the Frobenius norm of the difference $\|A - \sum_{j=1}^{i-1} w_j h_j^T - w_i h_i^T\|_F$, where $\sum_{j=1}^{i-1} w_j h_j^T$ comes from the column pairs from previous iteration. As iteration number goes large, the Frobenius norm of difference $\|A - WH\|_F$ goes down, until reaching a given tolerance. By reducing the Frobenius norm of difference each time, finally we can converge to a local minimum Frobenius norm(not global minimum).

Steps We randomly initialize w_1. Based on (9), we can update h_1^T entry by entry, i.e. $h_{11}, h_{12}, ..., h_{1n}$ based on the algorithm in problem 2 First, with w_1, h_{11} and A_1, form the

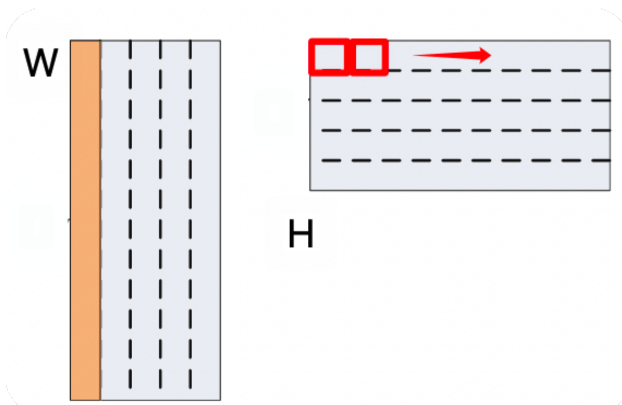

Figure 2: Update order of h_1: entry-wise

NLS problem, where A_i is the ith column of A

$$min_{x \geq 0} \|w_1 h_{11} - A_1\|_2 \tag{14}$$

According to Problem 2, we can get a unique minimizer of this problem, hence we have h_{11}. Next, with w_1, h_{12} and A_2, form the NLS problem

$$min_{x \geq 0} \|w_1 h_{12} - A_2\|_2 \tag{15}$$

So on and so forth, we have $h_{11}, h_{12}, ..., h_{1n}$, which is h_1^T. The solution we got is $h_{1i} = \frac{<w_1, A_i>}{\|w_1\|^2}$. To improve computational efficiency, we can solve h_1^T by solving the NLS problems all at once. Hence, we can solve $h_1^T = \frac{Aw_1}{\|w_1\|^2}$

Now, we move to the second iteration. Because $A = \sum_{i=1}^k w_i h_i^T$ and we already know $w_1 h_1^T$, let $R_{(1)} = A - w_1 h_1^T \in \mathbb{R}^{m\times n}$. We want to get w_2 based on $R_{(1)}$ and h_1^T. Consider (8), and we can form NLS problems similar to (14) and (15), where $R_{(1)i}^T$ is the ith column of $R_{(1)}^T$.

$$\begin{aligned} &min_{x\geq 0}\, \|h_1 w_{21} - R_{(1)1}^T\|_2 \\ &min_{x\geq 0}\, \|h_1 w_{22} - R_{(1)2}^T\|_2 \\ &\quad ... \\ &min_{x\geq 0}\, \|h_1 w_{2m} - R_{(1)m}^T\|_2 \end{aligned} \tag{16}$$

By solving above NLS problems, we can solve $w_{21}, w_{22}, ..., w_{2m}$, which is w_2. Writing in vector form, $w_2 = \frac{Ah_1^T}{\|h_1^T\|^2}$.

Repeat the process, we can solve h_2^T, w_3, h_3^T. When all the columns of W and H^T are updated and we have not reached the tolerance, we can start over by solve a new w_1 based on h_3^T from last iteration. So on and so forth, we stop when we reach the tolerance.

4.1 Algorithm

See algorithm 3.

5 Solution

We can easily derive the algorithm for this problem using the combination of the methods in both Problem 3 and Problem 4. Consider the NMF problem with matrix A and we want to find rank-4 approximation of A. The optimization problem is

$$min_{W\geq 0, H\geq 0}\, \|A - WH\|_F \qquad A \in \mathbb{R}_+^{m\times n}, W \in \mathbb{R}_+^{m\times 4}, H \in \mathbb{R}_+^{4\times n} \tag{17}$$

Algorithm 3 BCD

Require: $A \in \mathbb{R}_+^{m\times n}, \epsilon$
  $W \in \mathbb{R}_+^{m\times 3} \leftarrow \infty, H \in \mathbb{R}_+^{3\times n} \leftarrow \infty$
  Initial random $W[1]$
  $i \leftarrow 1$
  while true do
    $H^T[i] \leftarrow A \cdot W[i]/\|W[i]\|^2$
    if $i == 3$ **then**
      $W[1] \leftarrow A \cdot H^T[i]/\|H^T[i]\|^2$
    else
      $W[i+1] \leftarrow A \cdot H^T[i]/\|H^T[i]\|^2$
    end if
    $i \leftarrow i+1$
    if $i > 3$ **then**
      $i \leftarrow 1$
    end if
    if $\|A - WF\| \leq \epsilon$ **then**
      break
    end if
  end while
  return W, H

which is equivalent in Frobenius norm to

$$min_{H\geq 0, W\geq 0} \|W^T H^T - A^T\|_F \qquad A \in \mathbb{R}_+^{m\times n}, W \in \mathbb{R}_+^{m\times 4}, H \in \mathbb{R}_+^{4\times n} \tag{18}$$

General idea We are going to apply alternating NLS algorithm in Problem 3 to solve this NMF problem by solving W and H^T columns by columns based on (19), where W_i is the ith block of 2 columns of W, H_i^T is the ith block of 2 rows of H or the ith block of 2 columns of H^T, $(H_i^T)_j$ and A_j are the jth column of A and H_i^T, respectively.

$$\|WH - A\|_F = \|\sum_{i=1}^{k/2} W_i H_i^T - A\|_F = \sum_{i=1}^{k/2}\sum_{j=1}^{n} \|W_i (H_i^T)_j - A_j\|_2 \tag{19}$$

In each iteration on i, we update two pair of columns in W and H^T by minimizing the Frobenius norm of the difference $\|A - \sum_{j=1}^{i-1} W_j H_j^T - W_i H_i^T\|_F$, where $\sum_{j=1}^{i-1} W_j H_j^T$ comes from the column pairs from previous iteration. As iteration number goes large, the Frobenius norm of difference $\|A - WH\|_F$ goes down, until reaching a given tolerance. By reducing the Frobenius norm of difference each time, finally we can converge to a local minimum Frobenius norm(not global minimum).

Steps Steps are similar with steps in problem 4, but we use Rank2NLS algorithm instead of NLS to solve for each sub-problem

$$\begin{gathered}
min_{x\geq 0}\,\|W_1(H_1)_1 - A_1\|_2 \\
min_{x\geq 0}\,\|W_1(H_1)_2 - A_2\|_2 \\
\ldots \\
min_{x\geq 0}\,\|W_1(H_1)_n - A_n\|_2 \\
min_{x\geq 0}\,\|H_1(W_2)_1 - R^T_{(1)1}\|_2 \\
min_{x\geq 0}\,\|H_1(W_2)_2 - R^T_{(1)2}\|_2 \\
\ldots \\
min_{x\geq 0}\,\|H_1(W_2)_m - R^T_{(1)m}\|_2 \\
\ldots
\end{gathered} \tag{20}$$

With randomly initialized W_1, to improve computational efficiency, we solve each of H_1, W_2, H_2 all at once. Repeating the above procedure, we stop when we reach the tolerance ϵ.

5.1 Algorithm

Algorithm 4 BlockRank2NLS

Require: $A = [A_1, A_2] \in \mathbb{R}_+^{m\times 2}, B \in \mathbb{R}_+^{m\times n}$
  $x = [x_1, x_2, ..., x_n] \in \mathbb{R}_+^{2\times n} \leftarrow argmin_x \|Ax - b\|_2$
  $p \leftarrow (BA_1)/(A_1^T A_1)$
  $q \leftarrow (BA_2)/(A_2^T A_2)$
  for $i = 1, 2, ..., n$ **do**
    if $x_{i1} > 0$ and $x_{i2} > 0$ **then**
      return $x_i = x_i$
    else
      if $p_i \cdot \|A_1\| \geq q_i \cdot \|A_2\|$ **then**
        $x_i \leftarrow [p_i, 0]^T$
      else
        $x_i \leftarrow [0, q_i]^T$
      end if
    end if
  end for
  return x

Algorithm 5 Rank2BCD

Require: $A \in \mathbb{R}_+^{m\times n}, \epsilon$
  $W \in \mathbb{R}_+^{m\times 4} \leftarrow \infty, H \in \mathbb{R}_+^{4\times n} \leftarrow \infty$
  Initial random $W[1:2]$
  $i \leftarrow 1$
  while true do
    $H^T[i : i+1] \leftarrow$ BlockRank2NLS($W[1:2], A$)
    if $i == 3$ **then**
      $W[1:2] \leftarrow$ BlockRank2NLS($H^T[i : i+1], A$)
    else
      $W[i+2 : i+3] \leftarrow$ BlockRank2NLS($H^T[i : i+1], A$)
    end if
    $i \leftarrow i + 2$
    if $i > 3$ **then**
      $i \leftarrow 1$
    end if
    if $\|A - WF\| \leq \epsilon$ **then**
      break
    end if
  end while
  return W, H