Assignment 3

CSE5002 Intelligent Data Analysis (Spring 2022)

Name: Jiang Yuchen   ID: 11812419

Q1 Clustering

According to the question, the number of partition is 2 and initial cluster centers are (2, 5) and (3, 3)

First, we will calculate the distances between each data and initial cluster centers, which decides the initial partition result.

Then, we will find new cluster centers according to partition result.

After that, repeat calculating the distances between each data and current cluster centers until the partition result doesn't change.

Finally, we get the k-means result: one group is [(4, 6), (5, 4), (6, 5)] and another is [(1, 2), (2, 3)]. The corresponding cluster centers are [(5.0, 5.0), (1.5, 2.5)]

(The detail iterations are show below)

Iteration	**Partition based on current cluster centers**	**Current Cluster centers**
0	[[4, 6]], [[1, 2], [2, 3], [5, 4], [6, 5]]	[[2,5], [3, 3]]
1	[[4, 6], [6, 5]], [[1, 2], [2, 3], [5, 4]]	[[5.0, 5.5], [2.6666666666666665, 3.0]]
2	[[4, 6], [5, 4], [6, 5]], [[1, 2], [2, 3]]	[[5.0, 5.0], [1.5, 2.5]]
3	[[4, 6], [5, 4], [6, 5]], [[1, 2], [2, 3]]	[[5.0, 5.0], [1.5, 2.5]]

Q2 PCA

1. According to the question and the process of PCA

   First, decentralize the dataset X. Then compute the covariance matrix of X.

   Apply eigen decomposition to covariance of X, we will get eigen values and eigen vectors of it.

   Finally, choose the maximum k(k=1 here) eigen values and their corresponding vectors. The direction is the eigen vectors.

   Thus, the direction is [-0.70710678 -0.70710678] which is a unit vector, and its eigen value is 16.

2. Apply transform matrix to original dataset X, we get the projected data points: [2.22044605e-16 -1.41421356e+00 1.41421356e+00 -2.22044605e-16]]

Q3 Learning to rank

1. Three methods are pointwise, pairwise and listwise approaches.

For pointwise approach, the ranking function f learns to assign an absolute score(categories) to each item in isolation, like regression, classification and ordinal regression.

For pairwise approach, the ranking function f learns to rank pairs of items, which compares different vectors in pair for ranking, like any binary classifier.

For listwise approach, these algorithms try to directly optimize the value of one of the evaluation measures like Discounted cumulative gain (DCG) and Normalized DCG (NDCG), averaged over all queries in the training data. This is difficult because most evaluation measures are not continuous functions with respect to ranking model's parameters, and so continuous approximations or bounds on evaluation measures have to be used.

The aim of three approaches is same, which is to give a ranking result. The objects are different which are individual data item, data items in pair and data list respectively.

2. According to the question, when having three categories A, B and C where $A \le B \le C$.

   Data transformation: Take each pair $\{(x_i, y_i), (x_j, y_j)\}$ with $y_i = C \; and \; y_j = B$ or $y_i = C \; and \; y_j = A$ or $y_i = B \; and \; y_j = A$ and make a learning example $(\hat{x}_k, \hat{y}_k)$ such that $\hat{x}_k = (x_i, x_j), \hat{y}_k = +1$.

   Function transformation: For any $f \in \mathcal{F}$, define $\hat{f} \in \hat{F}$ by $\hat{f}(\hat{x}_k) = f(x_i) - f(x_j)$, for any $\hat{x}_k = (x_i, x_j)$

   The loss function is

   $$\min_{f \in H_K}[(\frac{1}{n_C n_B} \sum_{i:y_i=C} \sum_{j:y_j=B} (1-(f(x_i)-f(x_j)))_+ + \frac{1}{n_C n_A} \sum_{i:y_i=C} \sum_{j:y_j=A} (1-(f(x_i)-f(x_j)))_+ + \frac{1}{n_B n_A} \sum_{i:y_i=B} \sum_{j:y_j=A} (1-(f(x_i)-f(x_j))))_+$$

   When predicting, input is a pair of data like (x_i, x_j), the result label will be 1 if $y_i \ge y_j$. We can evaluate it with its ranking accuracy.

3. According to the question, we need to apply different weights towards losses in Ranking SVM. Suppose each query has m_i relevant documents, in order to put higher weight on data from queries with fewer relevant documents, new loss should be divided by m_i to make those with fewer relevant documents have a higher rank than before. At the same time, those with more relevant documents will be limited by its m_i.