

Week 9-2: Problem Formulation

1. Suppose you run a bookstore and have ratings (1 to 5 stars) of books. Your collaborative filtering algorithm has learned a parameter vector $\theta^{(j)}$ for user j and a feature vector $x^{(i)}$ for each book. You would like to compute the “training error”, meaning the average squared error of your system’s predictions on all the ratings that you have gotten from your users. Which of these are correct ways of doing so (check all that apply)?

For this problem, let m be the total number of ratings you have gotten from your users. (Another way of saying this is that $m = \sum_{i=1}^{n_m} \sum_{j=1}^{n_u} r(i, j)$ [Hint: Two of the four options below are correct.]

$$\begin{aligned} \text{(a)} \quad & \frac{1}{m} \sum_{j=1}^{n_u} \sum_{i:r(i,j)=1} [(\theta^{(j)})_i x_j^{(i)} - y^{(i,j)}]^2 \\ \text{(b)} \quad & \frac{1}{m} \sum_{j=1}^{n_u} \sum_{i:r(i,j)=1} \left[\sum_{k=1}^n (\theta^{(j)})_k x_k^{(i)} - y^{(i,j)} \right]^2 \end{aligned}$$

2. In which of the following situations will a collaborative filtering system be the most appropriate learning algorithm (compared to linear or logistic regression)?

- (a) You run an online news aggregator and for every user, you know some subset of articles that the user likes and some different subset that the user dislikes. You’d want to use this to find other articles that the user likes.
- (b) You manage an online bookstore and you have the book ratings from many users. For each user, you want to recommend other books she will enjoy, based on her own ratings and ratings of other users.

3. You run a movie empire, and want to build a movie recommendation system based on collaborative filtering. There were three popular review websites (which we’ll call A, B and C) which users go to rate movies, and you have just acquired all three companies that run these websites. You’d like to merge the three companies’ datasets together to build a single/unified system. On website A, users rank a movie as having 1 through 5 stars. On website B, users rank on a scale of 1 - 10, and decimal values (e.g., 7.5) are allowed. On website C, the ratings are from 1 to 100. You also have enough information to identify users/movies on one website with users/movies on a different website. Which of the following statements is true?

You can merge the three datasets into one, but you should first normalize each dataset’s ratings (say rescale each dataset’s ratings to a 1-100 range).

4. Which of the following are true of collaborative filtering systems? Check all that apply.

- (a) If you have a dataset of users ratings' on some products, you can use these to predict one user's preferences on products he has not rated.
- (b) Recall that the cost function for the content-based recommendation is

$$J(\theta) = \frac{1}{2} \sum_{j=1}^{n_u} \sum_{i:r(i,j)=1} [(\theta^{(j)})^T - y^{(i,j)}]^2 + \frac{\lambda}{2} \sum_{j=1}^{n_u} \sum_{k=1}^n (\theta_k^{(j)})^2$$

Suppose there is only one user and he as rated every movie in the training set. This implies that $n_u = 1$ and $r(i, j) = 1$ for ever i, j . In this case, the cost function $J(\theta)$ is equivalent to the one used for regularized linear regression.

5. Suppose you have two matrices A and B , where A is 5x3 and B is 3x5. Their product is $C = AB$, a 5x5 matrix. Furthermore, you have a 5x5 matrix R where every entry is 0 or 1. You want to find the sum of all elements $C(i, j)$ for which the corresponding $R(i, j)$ is 1, and ignore all elements $C(i, j)$ where $R(i, j) = 0$ One way to do so is the following code:

```
C = A * B;
total = 0;
for i = 1:5
    for j = 1:5
        if (R(i, j) == 1)
            total = total + C(i, j);
        end
    end
end
```

Which of the following pieces of Octave code will also correctly compute this total? Check all that apply. Assume all options are in code.

The key fact here is that both C and R are 5x5 matrices. Since we only want to sum where $R = 1$, we can multiply the two matrices element wise, then sum.

- (a) `total = sum(sum((A * B) .* R))`
- (b) `C = (A * B) .* R; total = sum(C(:))`