Feedback — XVI. Recommender Systems

Help

You submitted this quiz on **Mon 6 Jan 2014 9:13 PM PST**. You got a score of **5.00** out of **5.00**.

Question 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.]

Your Answer		Score	Explanation
☑ $\frac{1}{m} \sum_{i=1}^{n_m} \sum_{j:r(i,j)=1} (\sum_{k=1}^{n} (\theta^{(j)})_k x_k^{(i)} - y^{(i,j)})^2$	✔	0.25	This correctly sums over all ratings and computes the predicted rating with the explicit sum $\sum_{k=1}^{n} \theta^{(j)})_k x_k^{(i)}$.
☑ $\frac{1}{m} \sum_{(i,j):r(i,j)=1} (\sum_{k=1}^{n} (\theta^{(j)})_k x_k^{(i)} - y^{(i,j)})^2$	✔	0.25	This correctly sums over all ratings and computes the predicted rating with the explicit sum $\sum_{k=1}^{n} \theta^{(j)})_k x_k^{(i)}$.
☐ $\frac{1}{m} \sum_{j=1}^{n_u} \sum_{i:r(i,j)=1} ((\theta^{(j)})_i x_j^{(i)} - y^{(i,j)})^2$	✔	0.25	This incorrectly indexes into $\theta^{(j)}$ and $x^{(i)}$.
☐ $\frac{1}{m} \sum_{j=1}^{n_u} \sum_{i:r(i,j)=1} (\sum_{k=1}^{n} (\theta^{(k)})_j x_i^{(k)} - y^{(i,j)})^2$	✔	0.25	This incorrectly indexes into $\theta^{(j)}$ and $x^{(i)}$.
Total		1.00 / 1.00	

Question 2

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

Your Answer		Score	Explanation
☐ You manage an online bookstore and you have the book ratings from many users. You want to learn to predict the expected sales volume (number of books sold) as a function of the average rating of a book.	✔	0.25	This is a regression problem of predicting sales volume from ratings data, so collaborative filtering is not applicable.
☐ You're an artist and hand-paint portraits for your clients. Each client gets a different portrait (of themselves) and gives you 1-5 star rating feedback, and each client purchases at most 1 portrait. You'd like to predict what rating your next customer will give you.	✔	0.25	Since there is no overlap in the items reviewed by different clients, you cannot get good results using collaborative filtering.
☑ You run an online bookstore and collect the ratings of many users. You want to use this to identify what books are "similar" to each other (i.e., if one user likes a certain book, what are other books that she might	✔	0.25	You can find "similar" books by learning feature values using collaborative filtering.

also like?)			
☑ 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.	✔	0.25	This is a good application of collaborative filtering, as you can use the like / dislike as a rating to learn features for the articles and recommend articles similar to those each user likes.
Total		1.00 / 1.00	

Question 3

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.

Your Answer		Score	Explanation
☑ C = (A * B) .*	✔	0.25	This sums up all elements of `(A * B) .* R`, where the .* operator performs element-wise multiplication, setting the

R; total = sum(C(:));			elements of `A * B` to zero that correspond to zero entries in `R`.
☑ total = sum(sum((A * B) .* R));	✔	0.25	This sums up all elements of `(A * B) .* R`, where the .* operator performs element-wise multiplication, setting the elements of `A * B` to zero that correspond to zero entries in `R`.
☐ C = (A * B) * R; total = sum(C(:));	✔	0.25	Multiplying `(A * B) * R` will perform regular matrix multiplication and won't "mask out" entries.
☐ total = sum(sum(A(R == 1) * B(R == 1));	✔	0.25	You cannot use R to perform logical indexing into A and B, since R does not have the same dimension as those two matrices.
Total		1.00 / 1.00	

Question 4

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 to 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?

Your Answer	Score	Explanation
◯ It is not possible to combine these websites' data. You must build three separate recommendation systems.		
◯ Assuming that there is at		

least one movie/user in one database that doesn't also appear in a second database, there is no sound way to merge the datasets, because of the missing data.			
You can combine all three training sets into one as long as your perform mean normalization and feature scaling **after** you merge the data.			
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).	✔	1.00	By normalizing each dataset, you ensure that all ratings are on the same scale, so they are comparable during training.
Total		1.00 / 1.00	

Question 5

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

Your Answer		Score	Explanation
Even if you each user has rated only a small fraction of all of your products (so $r(i,j) = 0$ for the vast majority of (i,j) pairs), you can still build a recommender system by using collaborative filtering.	✔	0.25	The collaborative algorithm can still leverage the ratings that are present to build a reasonable recommender system.
For collaborative filtering, it is possible to use one of the advanced optimization	✔	0.25	You can compute the cost function and gradient, so any of these algorithms will work fine.

algoirthms (L-BFGS/conjugate gradient/etc.) to solve for both the $x^{(i)}$'s and $\theta^{(j)}$'s simultaneously.			
☐ Suppose you are writing a recommender system to predict a user's book preferences. In order to build such a system, you need that user to rate all the other books in your training set.	✔	0.25	Collaborative filtering can still work with missing data, such as a user who has not rated every book.
☐ When using gradient descent to train a collaborative filtering system, it is okay to initialize all the parameters ($x^{(i)}$ and $\theta^{(j)}$) to zero.	✔	0.25	You need to initialize them to different values so that you learn different features and parameters (i.e., perform symmetry breaking).
Total		1.00 / 1.00	