

exam prep

2022

question: Recall the notion of F1-score of a recommender system. Given a precision of 0.27 and a recall of 0.46, what is the F1-score of the recommender?

answer (number):

- $F_1 = 2 \cdot \frac{P \cdot R}{P+R} = \frac{2}{\frac{1}{P} + \frac{1}{R}}$
 - $F1 = 2 \cdot 0.25 \cdot 0.46 / (0.25 + 0.46) = 0.323943662$
-

question: Recall the notion of F1-score of a recommender system. Given a precision of 0.62 and a recall of 0.14, what is the F1-score of the recommender?

answer (number):

- $F1 = 2 \cdot 0.62 \cdot 0.14 / (0.62 + 0.14) = 0.263030303$
-

question: Recall the notion of recall of a recommender system. Let us assume that in a movie catalogue there are 16 movies that are relevant for user u. What is the recall of the top-10 recommendations if 2 are relevant for user u?

answer (number):

- $\text{recall}@k = \# \text{ relevant and recommended in top } k / \# \text{ relevant}$
 - $\text{recall}@10 = 2/16 = 0.125$
-

question: Recall the notion of recall of a recommender system. Let us assume that in a movie catalogue there are 20 movies that are relevant for user u. What is the recall of the top-18 recommendations if 6 are relevant for user u?

answer (number):

- $\text{recall}@10 = 6/18 = 0,33$
-

question: Recall the notion of precision of a recommender system. Let us assume that in a movie catalogue there are 20 movies that are relevant for user u. What is the precision of the top-18 recommendations if 6 are relevant for user u?

answer (number):

- $\text{precision}@k = \# \text{ relevant and recommended in top } k / k$
 - $\text{precision}@18 = 6/18 = 0.33$
-

question: Recall the notion of Mean Absolute Error MAE. Compute MAE for a test set of the following predictions where the true rating is always 5 but the system predicts in 72 cases 5, in 26 cases 4, and in 11 cases 3.

answer (number):

- $\text{MAE} = \frac{1}{|R|} \sum_{r_{ui} \in R} |r_{ui} - \hat{r}_{ui}|$
 - $\text{MAE} = (1/(72 + 26 + 11)) \cdot (72 \cdot (5-5) + 26 \cdot (5-4) + 11 \cdot (5-3)) = 0.44$
-

question: Recall the notion of Mean Absolute Error MAE. Compute MAE for a test set of the following predictions where the true rating is always 5 but the system predicts in 96 cases 5, in 27 cases 4, and in 23 cases 3.

answer (number):

- $MAE = \frac{1}{|R|} \sum_{r_{ui} \in R} |r_{ui} - \hat{r}_{ui}|$
- $MAE = (1/(96 + 27 + 23)) \cdot (96 \cdot (5-5) + 27 \cdot (5-4) + 23 \cdot (5-3)) = 0.5$

question: Calculate the pearson correlation as given in the lecture:

- $I_u = \{i \in I \mid r_{ui} \in R\}$
- $w_{uv} = \frac{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u) \cdot (r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u} (r_{ui} - \bar{r}_u)^2} \cdot \sqrt{\sum_{i \in I_v} (r_{vi} - \bar{r}_v)^2}}$
- user vector $\mathbf{u} = (5, 4, 3)$
- user vector $\mathbf{v} = (4, 4, 5)$

answer (number):

- $\bar{r}_u = 4$
- $\bar{r}_v = 4.3333333333$
- $\sqrt{\sum_{i \in I_u} (r_{ui} - \bar{r}_u)^2} = \sqrt{(5-4)^2 + (4-4)^2 + (3-4)^2} = 1.4142135624$
- $\sqrt{\sum_{i \in I_v} (r_{vi} - \bar{r}_v)^2} = \sqrt{(4-4.33)^2 + (4-4.33)^2 + (5-4.33)^2} = 0.8165169931$
- $\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u) \cdot (r_{vi} - \bar{r}_v) = (5-4) \cdot (4-4.33) + (4-4) \cdot (4-4.33) + (3-4) \cdot (5-4.33) = -1$
- $w_{uv} = \frac{-1}{1.414 \cdot 0.817} = -0.8660037539$

question: If two users have rated the same items, and one gives consistently half a star higher ratings than the other, then their similarity based on Pearson's correlation coefficient is 1.

answer (boolean):

- their ratings would still be perfectly aligned in a linear manner, resulting in a coefficient of 1
- proof:
 - ratings: X_i, Y_i
 - mean ratings: \bar{X}, \bar{Y}
 - $Y_i - \bar{Y} = (X_i + 0.5) - (\bar{X} + 0.5) = X_i - \bar{X}$
 - $r = \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}} = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (X_i - \bar{X})^2}} = 1$

question: A more advanced user-user collaborative filtering formula is:

$$s(u, i) = \bar{r}_u + \frac{\sum_{v \in U} w_{uv} \cdot (r_{vi} - \bar{r}_v)}{\sum_{v \in U} |w_{uv}|}$$

What is the purpose of the \bar{r}_u and \bar{r}_v terms in this version of the formula?

answer (single choice):

- a) These terms weight the recommendations so closer neighbors count more than distant neighbors.
- b) These terms normalize the computation to adjust for different users' rating scales. ☒
- c) These terms limit the number of neighbors used in the computation.
- d) These terms specify that we're combining the ratings of lots of other users together.

question: For any rank k, if $NDCG@k=1$ then it must hold that $P@k=1$.

answer (boolean):

- True
- a perfect nDCG score implies that the top document must be relevant so $P@1$ must also be 1

(incomplete) question: Calculate parts of item-item similarity (according to the formula from the lecture)

answer (number):

- pearson correlation / mean-centered cosine-similarity (but the question is missing the arguments for the computation)

question: Which of the following types of users have been the source of data for making recommendations in recommender systems?

answer (multiple choice):

- a) People with similar tastes to the target user ☒
 - in user-user collaborative filtering systems
- b) All system users who have expressed opinions ☒
 - in popularity-based systems
- c) Experts whose opinions were solicited for the site ☒
 - maybe in hybrid systems - but they are not the primary source of data

question: In an environment where all you have is prior purchase data (e.g., supermarket data), what's wrong with just recommending the most popular items?

answer (single choice):

- a) Popular items don't make a lot of profit for businesses.
- b) The recommendations are likely to be obvious (like bananas) and are probably items the customer will buy anyway even without the recommender. ☒
- c) Popular items probably don't go well with whatever else the customer has in her shopping cart.
- d) Most customers probably don't like popular items.

question: Why would you use a different metric for evaluating prediction vs. top-N recommendations?

answer (single choice):

- a) Because predictions are mostly about accuracy and error within a particular item, while top-N is mostly about ranking and comparisons between items. ☒
- b) Because you need different algorithms to compute predictions vs. top-N recommendations.
 - this is technically also true - but it doesn't reason why
- c) Because predictions are a harder problem; recommendations are just suggestions and can never be wrong.
- d) Because prediction metrics are usually on a 1 to 5 scale, and you need a larger scale for top-N metrics.

question: A smaller neighborhood in U-U CF, implies a worse accuracy of the recommendations.

answer (boolean):

- False
- not necessarily. you might have less data to work with, but if the users within that neighborhood are very similar, the accuracy of recommendations could still stay high.
- a neighborhood is a subset of users with some level of similarity to reduce noise in rating.

question: The term collaborative in collaborative filtering refers to combining multiple recommendation techniques to determine recommendations.

answer (boolean):

- False
- we only combine techniques in hybrid systems

question: Cold-start problems only appear when collaborative filtering techniques are used.

answer (boolean):

- False
- they appear whenever we have a new system / item / user without insufficient data to make good recommendations

(incomplete) question: We've discussed the Netflix Competition. Which of the following statements about the competition and the winning solution is most correct?

answer (multiple choice):

- The winning algorithm involved a complex hybrid algorithm that used statistical/machine learning techniques to mix together a variety of general-purpose and special-purpose algorithms, in the end resulting in a significantly improved prediction performance for the competition data. ☒
 - other options unknown
-

question: In basic matrix factorization, the matrix being factorized is the ratings matrix.

answer (boolean):

- True
 - to be more precise, we're learning the truncated-SVD of the rating-matrix without the $\hat{\Sigma}$ matrix: $\hat{R} = P^T \cdot Q$
-

question: On a travel platform, a user rates a hotel with 5 out of 5 stars. What kind of recommendation input is this? (Check all that apply)

answer (multiple choice):

- a) ?
 - b) Indication that the user likes the hotel ☒
 - we're assuming a five-point rating scale with 1 being the lowest and 5 being the highest score
 - c) Explicit Feedback ☒
 - d) ?
-

question: The goal of regularization is to avoid the problem of overfitting.

answer (boolean):

- True
 - you can't really avoid overfitting because of the bias-variance trade-off, but regularization can help to improve the generalizability of our model
-

question: The goal of regularization is to ensure stochastic gradient descent converges to a local minimum.

answer (boolean):

- False
 - it's to avoid overfitting
-

question: Collaborative filtering requires explicit ratings

answer (boolean):

- False
 - it can also work with implicit-feedback if we modify our model
-

question: What is the purpose of classification accuracy metrics such as precision?

answer (single choice):

- a) They're designed to measure what percentage of items in the product set users actually like ☒
 - they don't consider the ranking order of recommendations (although we can measure precision at a specified cutoff)
 - b) They're designed to measure whether the top-recommended items are indeed the best items available
-

question: Why is item-item more amenable to pre-computation than user-user?

answer (single choice):

- a) When there are many fewer items than users that means there are many fewer correlations to pre-compute
 - b) Because item-item tends to exhibit lower serendipity, and therefore less popular items don't matter much
 - c) Because items don't really have correlations; all the correlations are made through the user
 - d) When there are many more users than items, item similarities are fairly stable, while user similarities can change rapidly as the user interacts with the system ☒
-

question: What is the main point of hybrid algorithms?

answer (single choice):

- a) To take advantage of situations where no single algorithm provides the best recommendations by combining different algorithms together to achieve a better result. ☒
 - b) To speed up the computation of recommendations which too often are slow using non-hybrid algorithms. Hybrid algorithms also are easier to optimize for parallel execution.
 - c) To handle cases where a recommender is trying to balance objectives, e.g., to recommend good products for individual users, but also to make sure that each product gets recommended to enough different users to get sold.
 - that could be a benefit but not the main point
 - d) None of the others.
-

question: Which of the following is a problem with using Pearson correlation (as opposed to other similarity metrics) for computing user similarities in user-user collaborative filtering?

answer (single choice):

- a) The user may not know any other users in the system.
 - b) If users have only rated a small number of the same items, their correlation may be too high. ☒
 - users who have just 1 common item, are understood as identical. there are ways to fix this.
 - c) Users may use different portions of the rating scale.
 - d) Users may not have rated any of the same items.
-

question: What factors do we consider when deciding whether to assign weights to the item vectors being incorporated into a user's profile?

answer (single choice):

- a) Whether we believe more recently consumed (or rated) items are more reflective of a user's actual tastes.
 - b) Whether we have rating data that identifies "greater" or "lesser" liking.
 - c) We should consider all of these factors. ☒
 - d) Whether we have rating data that distinguishes dislike from like.
-

question: There are typically multiple stakeholders involved in a recommender system. In particular (check all that apply)

answer (multiple choice):

- a) Content providers ☒
 - b) None of the others
 - c) System owners ☒
 - d) Users ☒
-

question: In a precision-recall curve, precision may increase as recall increases

answer (boolean):

- True
 - in practice precision and recall are inverses of one another: improving recall (completeness) typically comes at the cost of reduced precision (correctness), because you're likelier to make more mistakes as you retrieve more data.
 - but it's still possible for precision to temporarily increase with recall.
-

question: In a precision-recall curve, the lowest precision appears at the highest recall level.

answer (boolean):

- False
 - precision and recall are inverses of one another
-

question: When is "term-frequency" most useful as part of a content-filtering recommender?

answer (single choice):

- a) When certain items are much more popular than other items.
 - popularity is separate from term-frequency
 - b) When the attributes of the items can apply in different degrees to different items. ☒
 - most useful in a content-based recommender systems when the attributes of items can be found in different degrees to different items / documents it helps to identify the importance of each term within each item
 - c) When users are unlikely to have experienced many of the items in the system.
 - term-frequency in content-based recommenders can help with user-cold-starts but that's not their most useful benefit
-

question: In order to deliver recommendations, popularity-based Recommender Systems require an accurate model of the target

answer (boolean):

- False
-

question: Which of the following would most indicate a situation where user-user collaborative filtering would be strongly preferable to content-based filtering (i.e., filtering based on user preferences of keywords or attributes)?

answer (single choice):

- a) Only implicit ratings are available; users won't provide explicit ratings.
 - collaborative-filtering can work with both kinds of ratings, although explicit ratings are preferred
 - b) The items being recommended don't have good attributes or keywords to describe them (e.g., user-submitted children's drawings without tags). ☒
 - content-based recommenders rely on content (meta)data, collaborative-filtering doesn't
 - c) There are lots of items to recommend, and relatively few users.
 - collaborative-filtering can work with few users
 - d) Most users have rated a core set of popular items, though they have different tastes on that core set.
 - user taste doesn't matter
-

question: Content-based Recommender Systems (check all that apply).

answer (multiple choice):

- a) have in general no cold-start problem for items. ☒
 - b) are known for delivering surprising recommendations.
 - the opposite is true. users can land in a filter-bubble and get repetitive recommendations
 - c) capture the similarity of items based on their content. ☒
 - d) exploit the history of user-item interactions. ☒
 - you can create user-embeddings out of their history that you then compare with items via cosine-similarity
-

question: Either Pearson correlation or cosine similarity are often used to compute the weights in user-user collaborative filtering. What are these metrics trying to measure?

answer (single choice):

- a) These are measures of the number of ratings users have in common.
 - number of ratings != similarity of ratings
 - b) These are measures of the similarity of the recommendations and the user's preferences.
 - c) These are measures of the similarity of ratings history between users. ☒
 - d) These are measures of how much the target user likes popular items.
-

question: A user views the first 13 seconds of a 5 minute video on YouTube, then browses away. What kind of recommendation input is this?

answer (single choice):

- a) A recommendation
- b) A rating
- c) Implicit feedback ☒

- d) Indication that the user likes the video

question: Group Recommender Systems (check all that apply)

answer (multiple choice):

- can incorporate all kinds of additional attributes of the group members including demographics and social relations. ✓
- always elicit the individual preferences prior to the group decision making process. ✓
- cannot deal with random groups.
- need to come up with a group profile that combines the individual preferences. ✓

question: A false positive is a non-relevant item recommended to the user.

answer (boolean):

- True

question: Which of the following is a true statement about why someone might prefer to use RMSE (Root Mean Squared Error) instead of MAE (Mean Absolute Error)?

answer (single choice):

- a) None of the others.
- b) Unlike MAE, RMSE aims to capture how close the predicted ratings are to the actual rating.
- c) MAE penalizes all errors the same, regardless of size, while RMSE penalizes large errors more than small ones. ✓
- d) RMSE can handle different types of ratings, whereas MAE is optimized for 5 star ratings.

question: Matrix Factorization (check all that apply)

answer (multiple choice):

- a) can only handle implicit feedback.
- b) became widely known during the Netflix Price competition. ✓
- c) is a model-based Collaborative Filtering Method. ✓
 - because it utilizes a mathematical model to represent users and items in a lower-dimensional latent space
- d) aims to represent users and items in a lower dimensional latent space. ✓

question: A switching hybrid selects only one recommender algorithm for each situation. Which of the following is the best example of a situation where a switching hybrid algorithm would be most useful?

answer (single choice):

- a) When we have several different algorithms, but don't really like the results from any of them.
- b) When we have several algorithms each of which does a great job with the top 4-5 recommendations, but is worse deeper down in the list. ✓
- c) When we have different algorithms that are better for recommending to users with few ratings vs. to users with many ratings.
- d) None of the others.


question: When using item-item CF with binary data, we usually just sum the similarities between the item and its neighbors, rather than computing a weighted average. Why?

answer (single choice):

- a) Because summation is significantly faster computationally than computing a weighted average, and a major benefit of item-item is faster performance.
 - b) Because weighted averages cannot be pre-computed, but sums can be easily cached and reused for future computations.
 - c) Since there are no ratings, the weighted average is effectively an average of a set of 1s, which is always 1. Summing similarities creates a meaningful score. ✓
 - d) Because sums help adjust for the fact that we don't really know whether the non-ratings represent items that are disliked or just never consumed.
-

question: You've learned about many techniques for evaluation. We also pointed out that most evaluation techniques do not address the question of whether the items recommended are actually useful recommendations. Instead, those evaluations focus on whether the recommender is successful at retrieving "covered up" old ratings. Which of the following evaluation metrics successfully focuses on whether the recommender can produce recommendations for new items that haven't already been experienced by the user?

answer (single choice):

- a) Accuracy metrics such as RMSE
- b) Classification metrics such as top-N precision
- c) Rank metrics such as NCDG
- d) None of the others 

question: Please fill in the missing values (use decimals instead of fractions) calculated by the Borda Count Strategy presented in the lecture (Ties - tournament style):

	A	B	C	D	E
Jane	10	4	5	6	2
John	5	7	8	7	5
Mary	8	8	7	6	2
	A	B	C	D	E
Jane					
John					
Mary					
Group					

answer (table):

	A	B	C	D	E
Jane	4	1	2	3	0
John	0.5	2.5	4	2.5	0.5
Mary	3.5	3.5	2	1	0
Group	8	7	8	6.5	0.5

- we turn the rater's ratings into rankings on a discrete scale 0;4
- we take the average of two ranks if it's a split
- we get the sum for each candidate A-E

question: Please calculate and fill in the missing values based on the "Least Misery Strategy" shown in the lecture!

	A	B	C	D	E	F	G	H	I
Jane	10	4	5	6	2	3	4	5	7
John	5	7	8	7	5	7	9	1	6
Mary	8	8	7	6	2	6	10	2	9
Group									

answer (table):

	A	B	C	D	E	F	G	H	I
Group	5	4	5	6	2	3	4	1	6

question: Please calculate and fill in the missing values based on the "Multiplicative Strategy" shown in the lecture!

	A	B	C	D	E	F	G	H	I
Jane	10	4	5	6	2	3	4	5	7
John	5	7	8	7	5	7	9	1	6
Mary	8	8	7	6	2	6	10	2	9
Group									

answer (table):

	A	B	C	D	E	F	G	H	I
Group	400	225	280	252	20	126	360	10	379

question: Compute the prediction for $s(u_x, 10)$ in three cases:

- Non-personalized User-User CF
- Personalized User-User CF
- User-User CF taking into account personal bias

we know the following:

- u_x = user for whom the prediction needs to be computed
- Rating scale: 1 to 5 stars
- Users' rating for item 10:
 - $r_{x,10}$ = not rated
 - $r_{5,10} = 4$
 - $r_{8,10} = 3.5$
 - $r_{9,10} = 3$
- Users' similarities:
 - $W_{x5} = 0.5$
 - $W_{x8} = 0.5$
 - $W_{x9} = 0.8$
- Previous ratings of users:
 - $u_5 = [1,5,3]$
 - $u_8 = [2,1,4]$
 - $u_9 = [3,2,2]$
 - $u_x = [1,3,4]$

answer (3 numbers):

- Non-personalized User-User CF:
 - $U_i = \{u \in U | r_{ui} \in R\}$ = all users that have rated the target-item
 - $s(u, i) = \frac{\sum_{v \in U_i} r_{vi}}{|U_i|}$
 - $|U_i| = 3$
 - $s(u_x, 10) = (4+3.5+3)/3 = 3.5$
- Personalized User-User CF:
 - $s(u, i) = \frac{\sum_{v \in U} w_{uv} r_{vi}}{\sum_{v \in U} w_{uv}}$
 - $s(u_x, 10) = (0.5 \cdot 4 + 0.5 \cdot 3.5 + 0.8 \cdot 3) / (0.5 + 0.5 + 0.8) = 3.4166$
- User-User CF taking into account personal bias:
 - $s(u, i) = \bar{r}_u + \frac{\sum_{v \in U} w_{uv} (r_{vi} - \bar{r}_v)}{\sum_{v \in U} w_{uv}}$
 - $\bar{r}_5 = (1+5+3)/3 = 3$
 - $\bar{r}_8 = (2+1+4)/3 = 2.3333$
 - $\bar{r}_9 = (3+2+2)/3 = 2.3333$
 - $\bar{r}_x = (1+3+4)/3 = 2.6666$
 - $s(u_x, 10) = 2.66 + (0.5 \cdot (4 - 3) + 0.5 \cdot (3.5 - 2.33) + 0.8 \cdot (3 - 2.33)) / (0.5 + 0.5 + 0.8) = 3.5605$

question: Assume we want to compute the item-to-item CF rating prediction for product 1 for user n:

- Baseline Item-Item CF
- Advanced formula that incorporates additional ideas (as shown on the slides)

In this exercise, the neighbourhood and weighted average are already given. You need to perform the normalization (mean-centring) of the item ratings based on the given information. This exercise refers to the advanced formula that incorporates additional ideas as shown on the lecture slides.

- neighborhood $N(i)$: We must compute the similarity of product 1 with the products bought by user n but we can do that only with products: 11, 14, and 20 (items that have at least 2 users that rated it and prod 1).
- weighted average:
 - $w(1, 11) = 0.4$
 - $w(1, 14) = 0.3$
 - $w(1, 20) = 0.3$
- normalize (mean-center) item ratings:

- What is the final prediction for product 1?

data (from a super blurry graph i had difficulties reading):

- Previous ratings of users:
 - $u_n = [1, 4, 5, 4, 5, 1, 5, 4, 4, 4, 3]$
- Previous ratings of items:
 - $r_{n,1} = \text{unknown}$
 - $r_{n,11} = 5$
 - $r_{n,14} = 5$
 - $r_{n,20} = 4$
 - $\bar{r}_1 = (2+5+4)/3 = 3.6666$
 - $\bar{r}_{11} = (3+5+5+5+5+5+3)/8 = 4.25$
 - $\bar{r}_{14} = (4+5+5+5+5+5+5+4)/9 = 4.77777$
 - $\bar{r}_{20} = (3+4+1+4+4+4+5+4+4+1+5)/11 = 3.545$

answer (two numbers):

- $N(i_1) = \{i_{11}, i_{14}, i_{20}\}$
- Baseline Item-Item CF:
 - $I_u = \{i \in I \mid r_{ui} \in R\}$ = all items that have been rated by the target-user
 - $s(u, i) = \frac{\sum_{j \in I_u} r_{uj}}{|I_u|}$
 - $|I_u| = 12$
 - $s(u_n, i_1) = (1+4+5+4+5+1+5+4+4+4+4+3)/12 = 3.666$
- Advanced formula that incorporates additional ideas (as shown in the slides):
 - $s(u, i) = \bar{r}_i + \frac{\sum_{j \in N(i)} w_{ij} (r_{uj} - \bar{r}_j)}{\sum_{j \in N(i)} |w_{ij}|}$
 - $s(u_n, i_1) = 3.666 + (0.4 \cdot (5 - 4.25) + 0.3 \cdot (5 - 4.7777) + 0.3 \cdot (4 - 3.5454)) / (0.4 + 0.3 + 0.3) = 3.666 + 0.50307 = 4.16907$

2020

question: Write a reasonable formula that computes the similarity between two songs based on the number of plays by users. For notation, you may use i and j to denote the two songs, and r_{ui} to denote the number of plays of song i by user u .

answer (formula):

- this is just item-item collaborative-filtering but with implicit feedback instead of ratings
- intuition: two songs are more similar if they have been played by the same users
- formula: cosine-similarity, pearson-correlation

question: In I-I CF, recommendations can be obvious because recommended items are similar to the items the user has already rated. Can you explain why a non-similar item cannot be recommended?

answer (open question):

- in this case item similarity is defined by the pearson-correlation: items are similar, if they get rated by users similarly

question: Cold-start problems only appear when collaborative filtering techniques are used.

answer (boolean):

- False
- A new system, item or user all create cold-starts that we have to deal with

question: Collaborative filtering requires explicit feedback (e.g., ratings) from users on items.

answer (boolean):

- False
 - collaborative filtering also works with implicit feedback
-

question: A larger neighborhood in U-U CF, implies a better accuracy of the recommendations.

answer (boolean):

- False
 - we use neighborhoods to reduce noise - see above
-

question: In the matrix factorization model that uses baseline estimates, what is the number of model parameters that need to be learned? Assume, n items, m users and latent feature dimensionality of k.

answer (formula):

$$\hat{r}_{ui} = \underbrace{\mu + b_u + b_i}_{\text{baseline estimate}} + \underbrace{q_i^T}_{\text{item model}} \cdot \left(\underbrace{p_u}_{\text{user model}} + \underbrace{|N(u)|^{-\frac{1}{2}} \sum_{j \in N(u)} y_j}_{\text{implicit feedback}} \right)$$

- params to be learned:
 - global bias (μ): 1 parameter
 - user biases (b_u): m parameters
 - item biases (b_i): n parameters
 - item latent factor vectors (q_i): n \times k parameters
 - user latent factor vectors (p_u): m \times k parameters
 - therefore, the total number of model parameters is: $1 + m + n + n \times k + m \times k = 1 + (1 + k) \times (m + n)$
-

question: Matrix factorization is another term for singular value decomposition.

answer (boolean):

- False
 - matrix factorization = truncated-SVD without the $\hat{\Sigma}$ matrix
 - matrix factorization and singular value decomposition (SVD) are not the same thing
-

question: The goal of regularization is to ensure stochastic gradient descent converges to a local minimum.

answer (boolean):

- False
 - see above for explanation
-

question: In content-based recommender systems, what are the prototype vectors in the Relevance Feedback approach (Rocchio's method)?

answer (open question):

- the prototype vectors are are emeddings of the user profile. they're usually based on the user-rating-history
-

question: Explain in one sentence the main difference between ranking accuracy and rating prediction accuracy metrics.

answer (open question):

- Ranking metrics evaluate how well a model can rank items relative to each other, while rating metrics evaluate how well a model can predict the absolute rating or score for an item.
-

question: A system recommends 12 items, among which only 4 are relevant. In total, there exist 8 relevant items. What is the F1 measure of this system?

answer (number):

- $TP = 4$ of recommendations were relevant

- $FP = 12 - 4 = 8$ of recommendations were not relevant
 - $FN = 8 - 4 = 4$ of relevant candidates weren't recommended
 - $P = TP / (TP + FP) = 4 / (4 + 8) = 0.3333$
 - $R = TP / (TP + FN) = 4 / (4 + 4) = 0.5$
 - $F_1 = 2 \cdot \frac{P \cdot R}{P + R} = \frac{2}{\frac{1}{P} + \frac{1}{R}} = 2 \cdot (0.3333 \cdot 0.5) / (0.3333 + 0.5) = 0.4$
-

question: The Twitter RecSys Challenge 2020 is about achieving good rating prediction accuracy.

answer (boolean):

- True
 - irrelevant, not part of curriculum anymore
-

question: In a precision-recall curve, the highest precision appears at the lowest recall level.

answer (boolean):

- True
- see above for explanation