Module 5 (Evaluation) has two assignments, this written assignment and Programming Assignment 4. The joint goal of these two assignments is for you to get experience thinking through the appropriate metrics and evaluation designs for recommender systems (WA5), and for you to get experience actually using metrics to tune and compare algorithms (PA4).

We considered how to provide meaningful grading and feedback on these assignments, and decided that objective feedback was the best fit (the alternative would be peer grading with extensive guidelines and training). Ideally, we'd like to have each student develop extensive evaluation plans, but we were concerned that the quality of peer grading in the course has not reached the level where we're confident these plans would get meaningful feedback and grading.

Accordingly, please recognize that while these questions are structured as multiple choice, they are intended to require time and thought, and that we encourage you to engage with the explanations, which are themselves educational tools. We encourage you to print out the questions in advance, and only enter the "exam" when you have all of your answers. Even if you get a correct answer, you should read through the explanations.

This assignment has 6 questions, each of which is worth 1 point. Every question has a single best answer that receives full credit, but some questions have partial credit for answers that are not the best, but have some merit. Also, a warning. Some of the answers may not make sense (e.g., they propose an evaluation for which you don't have the right data). Such answers are not the best answers.

Question #1

You are responsible for a recommender system for an e-commerce site that has three slots in which to recommend products at check-out time. In general, you are most interested in whether people buy additional products based on the recommendations, and you will be measuring sales and lift when your system is live, but you want an offline measure to help determine which potential recommenders are worth trying online. A few other details: this is a domain where people do occasionally re-purchase items, but not frequently. Most regular customers visit between once a week and once every three months. The site has complete details on all previous purchases, and has customer ratings for about 5% of the purchased items. Also, the site does a pretty good job recommending to new customers based on demographics and overall popularity, so you are focused on finding recommenders that take advantage of information learned from customers who have already purchased several items. Which of the following evaluation plans/metrics seems best?

a. RMSE-based accuracy evaluation. In this plan, we will take any customers who have at least 10 different prior ratings and measure the accuracy of predictions from the recommender using the leave-one-out method. We'll average over all predicted ratings, and will identify the best few algorithm candidates based on lowest RMSE.

b. Top-n precision evaluation. In this plan we'll use all customers who have at least 10 product purchases, and measure the top-3 precision of recommendations using a 5-fold cross-validation against a random 80% training/20% test set. We measure as a "hit" anything withheld from the test set. We will identify the best few algorithm candidates based on highest top-3 precision.

c. Spearman rank correlation evaluation. In this plan, we'll use all customers who have at least 10 product purchases, compute a recommendation list (using and 80/20 random training/test data set and five-fold cross validation), and measure the Spearman correlation between the between the withheld items and the recommedation list. We will identify the best few algorithms based on the highest Spearman correlation.

d. Diversity and Serendipity evaluation. In this plan, we'll use all customers who have at least 10 product purchases, compute a recommendation list for each customer (using their entire rating/purchase set), and measure the diversity and popularity of the top-3 recommendations. We'll take a balanced measure of diversity and inverse popularity (50% each), and identify the best few algorithms based on the top combination of high-diversity and low item popularity.

Question #2

Which of the following is a situation in which Mean Absolute Error is a reasonable choice as a metric for evaluating recommender performance?

a. You are running a streaming music site with a recommender inside that selects music to provide listeners with a personalized music listening experience (somewhat like Pandora). The basic interface is very small, it shows the name of the song and artist, and has controls that allow the user to click on a star (for favorite) or on an X (for don't play this again). It also allows users to click a forward arrow (skip to next song). Your goal is to have users enjoy the music enough to continue listening to the site (which is paid for by advertisements, which the user cannot skip).

b. You are running an online news site that presents users with an on-screen newspaper (somewhat like Google news). The site places articles into categories, the first of which is "top stories for you," and the others are traditional news categories (local news, world news, sports, business, etc.). The contents of each category are selected by the recommender, as is the order of presentation of items. Each items is displayed as a headline, and the first story in each category also has a few sentences from the start of the story. User feedback is entirely implicit -- users either read the full article or don't. Your goal is to have high site usage, as the site also contains advertisements.

c. You are running a travel-related recommendation site where users can look for hotels and restaurants (somewhat like TripAdvisor). For a given hotel or restaurant, you have a collection of data, including user ratings (lots of them), tags, written reviews and objective attributes (e.g., swimming pool, 24 hour restaurant). When displaying an item, the system shows a predicted rating (in this case personalized according to the user's profile, unlike TripAdvisor's average). Some usage is through search for specific places (e.g., Holiday Inn Hong Kong Kowloon), and search for category/location (e.g., Hotel swimming pool Paris). But about 80% of accesses go directly to a particular hotel or restaurant's page driven from search engines such as Google and Bing. The recommender is used both to produce the predicted ratings and as part of producing a ranked list of results for internal searches (where it is merged together with a search algorithm that measures quality of match to the search terms). Your site benefits both from visitors (through ads) and from booking referral payments.

d. You are running an e-commerce site for a shoe store (somewhat like Zappos). Your site has a large set of shoes, with structured product data for each (colors, materials, sizes, etc.). It also has user purchase data and user rating data. While some customers mostly visit the site to re-purchase shoes they already own that have worn out, most of the profit comes from people the large shoe collections who regularly buy new pairs of shoes. Your responsibility is a recommender system that sends out a periodic e-mail to the store's best customers suggesting items for purchase and offering a premium if the user makes a purchase within a certain time period. A typical e-mail would have pictures and descriptions of four pairs of shoes, along with a promotion such as a free travel shoe bag if placing an order for more than $100 before November 15th. Your site is a pure commerce site -- your revenue comes from sales. The goal for the e-mail program is to generate sales, though it is not considered important whether the customers buy the recommended items, or end up choosing to buy other items. All that matters is how much they buy.

Question #3

Which of the following is a situation in which it would be most useful to tune the recommender using a metric such as the receiver operating characteristic -- specifically, tuning the algorithm to find the right trade-off between true positive and false positive rates?

a. You are running a streaming music site with a recommender inside that selects music to provide listeners with a personalized music listening experience (somewhat like Pandora). The basic interface is very small, it shows the name of the song and artist, and has controls that allow the user to click on a star (for favorite) or on an X (for don't play this again). It also allows users to click a forward arrow (skip to next song). Your goal is to have users enjoy the music enough to continue listening to the site (which is paid for by advertisements, which the user cannot skip).

b. You are running an online news site that presents users with an on-screen newspaper (somewhat like Google news). The site places articles into categories, the first of which is "top stories for you," and the others are traditional news categories (local news, world news, sports, business, etc.). The contents of each category are selected by the recommender, as is the order of presentation of items. Each items is displayed as a headline, and the first story in each category also has a few sentences from the start of the story. User feedback is entirely implicit -

- users either read the full article or don't. Your goal is to have high site usage, as the site also contains advertisements.

c. You are running a travel-related recommendation site where users can look for hotels and restaurants (somewhat like TripAdvisor). For a given hotel or restaurant, you have a collection of data, including user ratings (lots of them), tags, written reviews and objective attributes (e.g., swimming pool, 24 hour restaurant). When displaying an item, the system shows a predicted rating (in this case personalized according to the user's profile, unlike TripAdvisor's average). Some usage is through search for specific places (e.g., Holiday Inn Hong Kong Kowloon), and search for category/location (e.g., Hotel swimming pool Paris). But about 80% of accesses go directly to a particular hotel or restaurant's page driven from search engines such as Google and Bing. The recommender is used both to produce the predicted ratings and as part of producing a ranked list of results for internal searches (where it is merged together with a search algorithm that measures quality of match to the search terms). Your site benefits both from visitors (through ads) and from booking referral payments.

d. You are running an e-commerce site for a shoe store (somewhat like Zappos). Your site has a large set of shoes, with structured product data for each (colors, materials, sizes, etc.). It also has user purchase data and user rating data. While some customers mostly visit the site to re-purchase shoes they already own that have worn out, most of the profit comes from people the large shoe collections who regularly buy new pairs of shoes. Your responsibility is a recommender system that sends out a periodic e-mail to the store's best customers suggesting items for purchase and offering a premium if the user makes a purchase within a certain time period. A typical e-mail would have pictures and descriptions of four pairs of shoes, along with a promotion such as a free travel shoe bag if placing an order for more than $100 before November 15th. Your site is a pure commerce site -- your revenue comes from sales. The goal for the e-mail program is to generate sales, though it is not considered important whether the customers buy the recommended items, or end up choosing to buy other items. All that matters is how much they buy.

Question #4

All of these situations are ones where it would make sense to test different recommenders empirically through A/B tests or other field tests. In which situation is it LEAST LIKELY that you could get useful data by asking users which set of outputs they prefer? In other words, in which situation are users least likely to know whether the recommender is actually achieving its goals?

a. You are running a streaming music site with a recommender inside that selects music to provide listeners with a personalized music listening experience (somewhat like Pandora). The basic interface is very small, it shows the name of the song and artist, and has controls that allow the user to click on a star (for favorite) or on an X (for don't play this again). It also allows users to click a forward arrow (skip to next song). Your goal is to have users enjoy the music enough to continue listening to the site (which is paid for by advertisements, which the user cannot skip).

b. You are running an online news site that presents users with an on-screen newspaper (somewhat like Google news). The site places articles into categories, the first of which is “top stories for you,” and the others are traditional news categories (local news, world news, sports, business, etc.). The contents of each category are selected by the recommender, as is the order of presentation of items. Each items is displayed as a headline, and the first story in each category also has a few sentences from the start of the story. User feedback is entirely implicit -- users either read the full article or don’t. Your goal is to have high site usage, as the site also contains advertisements.

c. You are running a travel-related recommendation site where users can look for hotels and restaurants (somewhat like TripAdvisor). For a given hotel or restaurant, you have a collection of data, including user ratings (lots of them), tags, written reviews and objective attributes (e.g., swimming pool, 24 hour restaurant). When displaying an item, the system shows a predicted rating (in this case personalized according to the user’s profile, unlike TripAdvisor’s average). Some usage is through search for specific places (e.g., Holiday Inn Hong Kong Kowloon), and search for category/location (e.g., Hotel swimming pool Paris). But about 80% of accesses go directly to a particular hotel or restaurant’s page driven from search engines such as Google and Bing. The recommender is used both to produce the predicted ratings and as part of producing a ranked list of results for internal searches (where it is merged together with a search algorithm that measures quality of match to the search terms). Your site benefits both from visitors (through ads) and from booking referral payments.

d. You are running an e-commerce site for a shoe store (somewhat like Zappos). Your site has a large set of shoes, with structured product data for each (colors, materials, sizes, etc.). It also has user purchase data and user rating data. While some customers mostly visit the site to re-purchase shoes they already own that have worn out, most of the profit comes from people the large shoe collections who regularly buy new pairs of shoes. Your responsibility is a recommender system that sends out a periodic e-mail to the store’s best customers suggesting items for purchase and offering a premium if the user makes a purchase within a certain time period. A typical e-mail would have pictures and descriptions of four pairs of shoes, along with a promotion such as a free travel shoe bag if placing an order for more than $100 before November 15th. Your site is a pure commerce site -- your revenue comes from sales. The goal for the e-mail program is to generate sales, though it is not considered important whether the customers buy the recommended items, or end up choosing to buy other items. All that matters is how much they buy.

Question #5

We have argued that the real proof of a recommender system is in the usage, and that offline evaluation has serious problems. At the same time, there are many situations where offline evaluation does make sense. Which of these is a valid reason for carrying out off-line metric-based evaluation rather than a live user study of a recommender?

a. The recommender may not yet exist yet, but there is data that can be used to pre-test the idea of the recommender.

b. The recommender designer wants to test a wide variety of alternative recommenders, at least to narrow them down to a few candidates that can be user tested.

c. A recommender systems researcher is trying to establish properties of recommender algorithms in general, across a wide variety of data sets.

d. All of the above.

Question #6

Which of the following is a valid objection to the validity of offline evaluation using metrics such as MAE, top-n Precision, or nDCG?

a. The offline metrics only assess the ability to “recommend” items that have already been consumed or rated. Real recommenders should usually be suggesting new items not already known to the user. Hence, something with low offline metrics might acually be better at finding new items of interest.

b. The offline metrics depend completely on the scale of the data being presented. For instance if you change a rating scale from 5 points to 100 points, the MAE will increase by a factor of 20.

c. Offline metrics come in different groups. Some measure accuracy. Some address decision correctness. Some look at rank. That means that we can’t tell if any metric is relevant for any particular recommender system.

d. All of the above objections are valid.