CSE 258, Winter 2017: Homework 3

Instructions

Please submit your solution **by the beginning of the week 7 lecture (Feb 20).** Submissions should be made on **gradescope**. Please complete homework **individually**.

These homework exercises are intended to help you get started on potential solutions to Assignment 1. We'll work directly with the Assignment 1 dataset to complete them, which is available here:
`http://jmcauley.ucsd.edu/data/assignment1.tar.gz`

Executing the code requires a working install of Python 2.7 or Python 3.

You'll probably want to implement your solution by modifying the baseline code provided.

Note that you should be able to join the competitions using a UCSD e-mail. The competition pages can be found here:
`https://inclass.kaggle.com/c/cse158-258-helpfulness-prediction`
`https://inclass.kaggle.com/c/cse158-categorization`
`https://inclass.kaggle.com/c/cse258-rating-prediction`

Please include the code of (the important parts of) your solutions.

Tasks (Helpfulness prediction)

First, since the data is quite large, when prototyping solutions it may be too time-consuming to work with all of the training examples. Also, since we don't have access to the test labels, we'll need to simulate validation/test sets of our own.

So, let's split the training data ('train.json.gz') as follows:
(1) Reviews 1-100,000 for training
(2) Reviews 100,001-200,000 for validation
(3) Upload to Kaggle for testing only when you have a good model on the validation set. This will save you time (since Kaggle can take several minutes to return results), and also will stop us from crashing their website...

1. Fitting the 'nHelpful' variable directly may not make sense, since its scale depends on the total number of votes received. Instead, let's try to fit $\frac{\text{nHelpful}}{\text{outOf}}$ (which ranges between 0 and 1). Start by fitting a simple model of the form

   $$\frac{\text{nHelpful}}{\text{outOf}} \simeq \alpha.$$

   What is the value of α (1 mark)?

2. What is the performance of this trivial predictor on the validation set? Recall that this should be measured in terms of the *mean absolute error* (`https://www.kaggle.com/wiki/AbsoluteError`) (1 mark).

3. To fit the same quantity, train a predictor of the form

   $$\frac{\text{nHelpful}}{\text{outOf}} \simeq \alpha + \beta_1(\#\text{ words in review}) + \beta_2(\text{review's rating in stars}).$$

   Report the fitted parameters and the MAE on the validation set (1 mark).

4. To run our model on the *test* set, we'll have to use the files 'pairs_Helpful.txt' to find the userID/itemID pairs about which we have to make predictions, and 'helpful.json.gz' to get the review data for those pairs. Using that data, run the above model and upload your solution to Kaggle. Tell us your Kaggle user name (1 mark). If you've already uploaded a better solution to Kaggle, that's fine too!

Tasks (Rating prediction)

Let's start by building our training/validation sets in the same way as we did for the helpfulness task.

5. What is the performance of a trivial predictor

   $$\text{rating}(\text{user}, \text{item}) = \alpha$$

   on the validation set, and what is the value of α (1 mark)?

6. Fit a predictor of the form

$$\text{rating}(\text{user}, \text{item}) \simeq \alpha + \beta_{\text{user}} + \beta_{\text{item}},$$

   by fitting the mean and the two bias terms as described in the lecture notes. Use a regularization parameter of $\lambda = 1$. Report the MSE on the validation set (1 mark).

7. Report the user and item IDs that have the largest and smallest values of β (1 mark).

8. Find a better value of λ using your validation set. Report the value you chose, its MSE, and upload your solution to Kaggle by running it on the test data (1 mark).