ECS 189G Term Project

Due Date

Scheduled final exam day (no written final), 11:59 pm. Tip: Act as if the due date is one day before the real one.

Overview

All quarter, we've been predicting item ratings, with output like "Our predicted rating for user 88 and movie 220 is 3.72." But it would be nice to report probabilities instead, making statements like, "For user 88 and movie 220, the probabilities of ratings 1, 2, 3, 4 and 5 are 0.102, 0.468, 0.057, 0.111 and 0.262, respectively." We'll do that here!

Details

- You will write a function with call form

  ```
  ratingProbsFit(dataIn,maxRating,predMethod,embedMeans,specialArgs)
  ```

  where the arguments are as follows:

  - **dataIn:** Data frame, with first 2 columns being R factors for user ID and item ID, and then a numerical rating. The 3 columns must be named 'userID', 'itemID' and 'rating'.

  - **maxRating:** Top rating. Scores are assumed to range from 1, consecutively through this value.

  - **predMethod:** One of 'logit', 'NMF', 'kNN' and 'CART'.

  - **embedMeans:** If TRUE, replace user and item IDs by means, as in our revised book. Not valid if **predMethod** is 'NMF', but mandatory if it is 'CART'.

  - **specialArgs:** An R list, with named elements, specifying method-specific arguments to be used, if any, e.g. rank for NMF.

  The return value will be of class **recProbs**, S3. It will contain whatever information is needed for **predict.recProbs()**.

- You will write a function **predict.recProbs()**with call form

  ```
  predict(probsFitOut,newXs)
  ```

  ```
  where the arguments are as follows:
  ```

  - **probsFitOut:** Output of **ratingProbsFit()**.
  - **newXs:** A data frame of new (user,item) combinations to be predicted. No new users or items will be allowed. Columns must be named 'userID' and 'itemID'.

  Note that this will be a case of R's *generic* functions.

- The return value will be a matrix, with **maxRating** columns and **nrow(newXs)** rows.

- Other than the above specs, this project is open-ended. For any given **predMethod**, there are various possible ways you might estimate the rating probabilities.

- Note, though, that the fundamental relation that will probably guide your thinking is $E(I_A) = P(A)$, where I_A is a random variable equal to 1 or 0, depending on whether the event A occurs or not. In sample terms, the average of a dummy variable is the proportion of time it is equal to 1.

- You will compare the accuracies of the various types of **predMethod** on two datasets, which you will choose from this list:

  - Czech Dating Agency
  - Harvard/MIT MOOCs (maybe take grade as the "rating")
  - InstEval
  - Song List
  - Stanford Network Datasets treat an edge as s "Like" (warning: very large)

- Your report must explain in detail your design decisions, and how you implemented them.

Important Rules

PLEASE FOLLOW THESE RULES 100%!

- You must use LaTeX and R throughout.
- Submit your report, including all files, i.e. **.tex**, **.pdf**, R code (see below), any image files, etc., to my **handin** site on CSIF (NOT the TAs' sites), directory **189gproject**.
- Your LaTeX file must be named **ProjectReport.tex**, and the corresponding PDF **ProjectReport.pdf**. The grading script will look for these; don't disappoint the grading script!
- The name of your submitted file must be of the form **email1.email2....tar** , where each **emaili** is the UCD e-mail address of group member i, e.g. **bclinton.gwbush.bobama.dtrump.tar**. Note the periods separating fields. Don't get the address wrong! Otherwise the grading script may not give someone credit.
- Your **.tar** file must contain only regular files, NO SUBDIRECTORIES!!!! And **.tar** does NOT mean **.tar.gz** or **.tar.bz2** (or for that matter **.rar**, which one student used once). The grading script will execute

  ```
  tar xf youraddresses.tar
  ls ProjectReport.tex
  xpdf ProjectReport.pdf
  ```

  Note that it will NOT do **cd**.
- Place all your code in a file **TermProject.R**, *as well as* in an Appendix to your report (LaTeX **\appendix \section{}**). I may execute your code, so make sure it is runnable.
- Absolutely NO late reports will be accepted. **As you near the deadline, keep submitting what you have** (each one will overwrite the last), so that at least you will get a lot of credit even if you don't finish.
- Include a section listing each team member's contribution to the project -- who did what. If a member did not participate, do not include him/her in this section, nor in the **.tar** file name. **Don't forget this section!**
- **DOUBLE-CHECK THAT YOU ARE MEETING ALL SPECS!** Whoever does the actual turning in of your submission, impress upon him/her that this is a

huge responsiblity that can affect everyone's grade.

Grading

General commnets:

- Groups that put in a reasonable amount of time -- and thought! -- almost always receive at least a B+ grade on the project, typically better. Groups that do not complete the project usually get a D grade. **PLEASE START EARLY!**

- As explained in class, groups that do good work on the project receive an extra bonus in their course grades, beyond what your quiz and homework grades are. The boost is usually at least one notch (e.g. B to B+) and often two notches (e.g. B to A-). E.g. a student could have strictly B work in the homework and quizzes and yet still get an A- in the course.

- A+ grades are very possible, and can have a significant impact on your course grade, letters of recommendation, knighthoods, marriage prospects, coronations, etc.

Criteria:

- Technical content of the work (correctness, thoroughness etc.).

- Adherence to instructions.

- Professional quality of the work: Clear, engaging writing, using correct grammar; it need not (should not) be pretentious, but avoid being too colloquial ("the mean was kinda low"). Presentation need not be fancy, but graphs and tables should be used when helpful.