kaggle Search kaggle Competitions Datasets Kernels Discussion Jobs

What Do You Know?

Improve the state of the art in student evaluation by predicting whether a student will answer the next test question correctly.

$5,000 · 239 teams · 6 years ago

Overview Data Discussion Leaderboard Rules Late Submission

Competition Data

Edit

- benchmark_lmer_submi...
- solution_sorted.csv
- test_sorted.csv
- benchmark_lmer.r
- grockit_all_data.7z
- grockit_all_data.zip

benchmark_lmer.r 2.52 KB

Download

Data Description

Data description

The fields of the data are as follows (the mappings from category fields to numeric values are in category_labels.csv):

- *correct*: 0 or 1, indicating whether the student answered the question correctly. This is the field to be predicted in the test set.
- *outcome*: a numeric code representing 'correct' (1), 'incorrect' (2), 'skipped' (3), or 'timeout' (4): a more detailed indicator of the outcome. Not present in test data.
- *user_id*: an anonymized numeric identifier for the user answering the question
- *question_id*: a numeric identifier for the question being answered
- *question_type*: a numeric code representing the type of question; either 'MultipleChoiceOneCorrect' (0) for multiple choice, or 'SPR' (1) for free response questions
- *group_name*: a numeric code representing the group of the question ('act' (0). 'gmat' (1), 'sat' (2))
- *track_name*: the numeric code for the track within the test this question is associated with (mappings from category fields to numeric values are in category_labels.csv)

- *subtrack_name*: the numeric code for the subtrack within the track this question is associated with (mappings from category fields to numeric values are in category_labels.csv)
- *tag_string*: a space-separated list of tag ids this question has been tagged with (mappings from category fields to numeric values are in category_labels.csv)
- *round_started_at*: a UTC timestamp indicating when the user saw the question
- *answered_at*: a UTC timestamp indicating when the user answered the question (NULL if not answered)
- *deactivated_at*: a UTC timestamp indicating when the round finished, either because of being answered or timing out
- *answer_id*: an id for the specific answer chosen, for multiple choice questions. Not present in test data.
- *game_type*: indicates the type of game/study session (mappings from category fields to numeric values are in category_labels.csv)
- *num_players*: the number of players in the game at the time (multiple people can be viewing the same question simultaneously)
- *date_of_test: the date the user entered as their expected test date (if entered)*
- *question_set_id*: the question set for the question; most questions sets will only have one question; questions which share a question set id have a common presentation (such as a reading passage) and multiple questions based on that same information

Test data does not include answer_id or outcome (from which correctness could be determined).

Test/training generation

The data used in this competition is a sample of Grockit students (from the past three years) answering questions to prepare for the sat, gmat, or act. The test/training split is derived by finding users who answered at least 6 questions, taking one of their answers (uniformly random, from their 6th question to their last), and inserting it into the test set. Any later answers by this user are removed, and any earlier answers are included in the training set. All answers from users not in the test set are also used for the training set (as they may be useful in estimating question parameters or baseline ability distributions).

The test data distribution is thus different from training data in ways that may be significant. First, it does not include 'timeout' or 'skipped' outcomes: all test results are from the student actually answering the question. Second, it is biased towards users with more questions in the training set and biased towards their later answers. Third, it is one entry per user, so the distribution of various aspects of the data (such as correct/incorrect) is over users, not over all answered questions.

We have attempted to provide a reasonable validation split on the training data by taking the previous correct/incorrect answer for each of the students in the test set, for those users who had at least one previous answer. The results are in the additional files valid_training.csv and valid_test.csv (in the zip/7z file). Participants may find it helpful to compare potential algorithms by training on the valid_training.csv file and computing their performance on the valid_test.csv files. However, there is no guarantee that this is an optimal validation set.

Our Team Terms Privacy Contact/Support