Final Project

Should This Loan be Approved or Denied?

Nov. 13, 2020

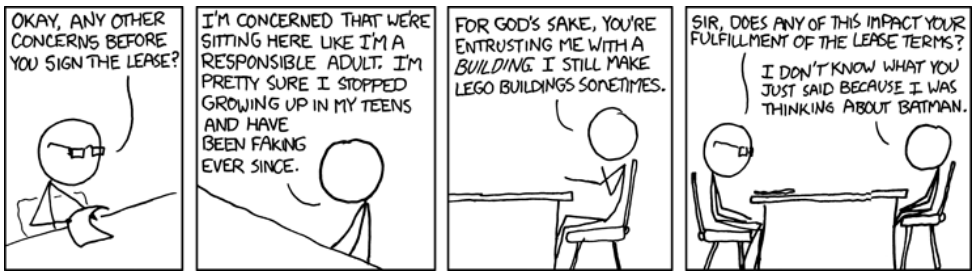

Figure 1: Credit: `https://xkcd.com/616/`

The goal of this project is to give you experience in a real-life classification task, which is one of the most practical and commonly encountered type of data-related task. It will give you experience applying approaches such as exploratory data analysis, visualization, preprocessing, and how to select and evaluate various classification approaches.

The task is to predict whether a loan will be "charged-off" (i.e. written off by the lender, when the borrower fails to pay it). This data set is from the U.S. Small Business Administration (SBA) and provides historical data from 1987 through 2014. A full description of the dataset and evaluation metrics, can be found at the competition page: `https://www.kaggle.com/t/dad03cf936a24c6cae008d166a500f42`

Submitting your Predictions on Kaggle

The Kaggle page contains training data `Xtrain.csv`, and the associated labels in `Ytrain.csv`, in the form of the "ChargeOff" variable. The test data `Xtest.csv` contains the test set which your model should predict. The predictions you submit should be a csv file with 2 columns: the "Id" variable, which correspond to the IDs in the test data, and your binary predictions for the "ChargeOff" variable; for reference, see the sample predictions file `sample_output.csv`, which has been included in the dataset package, and shows what your submission should look like.

To prevent overfitting to the leaderboard, you are allowed to make at most 5 submissions to the leaderboard per day. Note that you are expected to write your own procedures for evaluating how well your method is performing using the given datasets: for example, this can be done by cross-validation, splitting the data into a training and a validation set, or similar approaches. Do not rely solely on the public leaderboard to evaluate your model: this could potentially overfit to the public leaderboard, resulting in poor performance on the private leaderboard, and the submission limit would hinder your model development process. During evaluation, we will place greater consideration on the private rather than the public leaderboard scores.

Final Report

One member of your team should submit a zipped folder to LumiNUS containing your final report (in PDF format) describing your data preprocessing steps, exploratory data analysis, the steps you have done in model fitting and evaluation, and the evaluation results (include both your public leaderboard score, as well as any evaluation metrics you have computed on your own to test and compare different approaches.) The report should be at most 10 pages, including figures. The zipped folder should also contain the code you use in your approach, or include a link to your github repository in your final report. Note that 10% of your grade will be based on the reproducibility of your approach, which includes the organization and readability of your code. Your report should include the names and student IDs (e.g. A123456) of all your team members, and your Kaggle team name. While the exact format of the report is up to you, you can structure it based on the following:

- **Preprocessing and Exploratory Data Analysis**: Explain how you tried to understand the data, e.g. through plotting or exploratory data analysis. Explain all preprocessing steps taken; e.g. what features did you use, and did you perform any feature transformations, and why were these transformations done?
- **Data Mining**: Clearly describe how you ran your classification model(s). You do not need to explain the model itself (e.g. what a random forest is), but should explain all other relevant details for running it. Which approaches did you try? What values did you use for its hyperparameters (or other choices involved in the model)? How did you select these hyper parameters?
- **Evaluation and Interpretation**: How did you compare between different approaches? Which approaches performed the best, and why?

An example of a good report (from a previous semester of the class) can be found in the LumiNUS folder for this final project. Note that this was based on a different dataset, with its own characteristics, so the steps involved may not necessarily be appropriate for the current dataset and task. Also, there is no need to follow the formatting of this report: formatting is flexible, as long as you have described your approach clearly and fully.

Grading

Grading will be primarily based on the quality of the methodological approach and the final report i.e. whether your approach is methodologically sound, whether you can understand and clearly describe each step of the data mining process, and whether you are able to build an effective classification model (and clearly explain the steps you have taken to do so).

Your scores on the public and private leaderboards will be considered only used as part of the whole picture, i.e. as one indication of the quality of your methodology (which will also be assessed based on your report). Scoring well on the leaderboard is not a requirement for getting very good scores - it is possible to get full credit as long as your overall approach as detailed in the report is methodologically sound and of high quality, and meets a reasonable standard of prediction accuracy, even if it does not get top positions on the leaderboard. Even so, attaining top positions in the leaderboard is a significant

achievement, so the top 5-10 scoring methods (taking into account both public and private leaderboards) will receive very good methodology scores, unless there are significant deficiencies in the approach as outlined in your report. Grading will be based on:

- **Methodological quality: 60%**
  a. **Preprocessing**: appropriate preprocessing methods are chosen and correctly implemented; e.g. missing values and categorical variables are handled appropriately.
  b. **Visualization**: appropriate and informative use of visualization and plots, leading to good data understanding.
  c. **Methods**: methods are well motivated and correctly implemented, in a methodologically sound manner.
  d. **Evaluation**: methods are compared or evaluated in an effective manner, with the use of appropriate metrics, and with appropriate experimental setups (e.g. cross-validation, splitting into training and testing sets, or other approaches).
- **Quality of report: 30%**
  Report explains the results in a clear and comprehensive manner, demonstrating and communicating correct understanding of the various steps you have done
- **Reproducibility: 10%**
  Code is included, and is sufficiently well-organized and readable so as to be usable by an outsider

Helpful Resources

Some useful resources include the following:

- **Getting Started Guide on Kaggle**: `https://www.kaggle.com/getting-started/45113`
- **Exploratory Data Analysis**: `https://www.kaggle.com/kashnitsky/topic-1-exploratory-d`
- **Evaluation using Cross Validation**: `https://www.kaggle.com/dansbecker/cross-validation` Cross validation (or similar tools) are important in evaluating your approach to see which of various approaches works better. Note that you should not rely solely on the public leaderboard for this, as you are only allowed 2 submissions per day (and also, this may overfit to the public leaderboard, which would affect your score on the private leaderboard).
- **Avoiding Data Leakage**: `https://www.kaggle.com/dansbecker/data-leakage/` Data leakage is a commonly encountered problem on Kaggle (and similar settings). Informally, for test accuracy to be meaningful, the algorithm being tested should generally not be trained on information from the test data. Data leakage means that there is "leakage of information" from your test set to your training set, which makes test accuracy an unreliable metric. This can happen in subtle ways, e.g. when the training and test set columns are preprocessed together, such as through normalization.

- **Additional Learning Materials on Kaggle**: see `https://www.kaggle.com/dansbecker/learning-materials-on-kaggle` for a comprehensive and useful list of resources, e.g. on handling categorical data.