Data Science Honors Thesis: Reproducibility Plan

Rodrigo Palmaka
University of California, Berkeley, California, USA.

1 Code/Data Documentation

Data documentation will primarily take place in the Datasets folder of the GitHub repository. I will include a Readme file explaining the contents of each dataset (features, dimensions, date ranges). Datasets will come in two types: covariates – financial ratios and other variables used for prediction, ratings – the labels we seek to predict, specifically the S&P long term issuer credit rating.

Code documentation will involve a two-pronged approach. I will include a Readme providing a summary of the contents of each Jupyter notebook and python files that define helper functions and custom model classes. Python files will mostly serve as a black box for functions used to process data or assist in training models – function descriptions will be included as docstrings in the .py files. Jupyter notebooks will include model specific notebooks, EDA, and a high level notebook that can run any of the models and produce results while maintaining a level of abstraction. Model and EDA notebooks will serve to document the model training process and model assumption testing as well as data exploration. The written contents of these notebooks will elaborate on the paper component of the thesis. They will be self contained and explore specific models.

2 Reproducibility

As detailed in the above section, individual EDA and model notebooks will converge to one high-level notebook where a user can apply any of the studied models and output results on given datasets. Users will have the option of choosing different data processing approaches as well (train/test split, SMOTE, CCA, etc.). This way anyone studying this project will have a simple way of conferring the results of my thesis without necessarily having to understand the code or implement it themselves. This high-level notebook will serve as an index for exploring the model-specific code. For example, if a user is exploring the results of my Ordinal SVM model, I will be able to refer them to the file that defines that model in this notebook. If time permits, I will pursue using MyBinder to further modularize the execution of my project.