IMC 466
Professor Edward Malthouse

Homework 1

Due: Dec 5, Start of class

Submit your answers on Canvas. I will crate a "quiz."

You may work in self-selected groups of at most four students. You may use any software you like to complete this exercise. The data are available on Canvas under Files/SFC.

This homework will use the SFC data again from the file `modeldataNoCensor.txt`. Two important uses of CRM data are to predict the customer that will churn (churn model) and best-customer clones (clone model). The variable `churn` indicates if a customer stops paying for the SFC and will be called a "hard churn." A problem with hard churn is that it is often too late, and the customer sends signals before the hard churn that the relationship will soon end. If an organization can accurately predict which customers are likely to churn in the near future, it can intervene with a contact point and potentially avert the hard churn.

The variable `readingDayNum`, hereafter called *regularity*, is the number of days in a given month that some subscriber (`SubscriptionId`) reads at least some content. Other analyses have shown that regularity is the most important predictor of hard churn, and is therefore an early warning variable that something is wrong in the relationship. Define the following

- Soft churn: when a person reads 2 or fewer days per month
- Rock star: when a person reads at least 20 or more days per month

The goal of this homework is to build predictive models for soft churn and rock star status in the next month (use reading behaviors this month to predict churn/star status next month). The `month` variable indicates the month number of the customer's life. For example, predict churn in month $t+1$ from reading behaviors in month t. See here for hints on how to lag variables in dplyr.

Q1 How accurately can we predict next-period hard churn? (report AUC on test set) What variables are power predictors? Which models do best?

Q2 How accurately can we predict next-period rock starness? (report AUC on test set) What variables are power predictors? Which models do best?

I have another question: do we need two models? Instead of building two binary classification models, we could build a single regression model predicting regularity (or log(regularity+1)). Then those with small predicted values are likely to churn, and those with large predicted values are rock stars.

Q3 How accurately can we predict next-period log-regularity? (report R^2 on test set) What variables are power predictors? Which models do best?

Q4 Do we need two binary classification models, or just one regression model? Justify your answer.

The file `idtrain.csv` gives a list of unique subscribers and whether they are assigned to training or test sets. Do not use the test set for any model selection. Pick your final model of a type, then apply it to the test set. It is up to you to set aside a test set or use k-fold CV on the training data to do your model selection. Do not use the first 10 columns in your models, or the `churn` variable. Be sure use next-period values for the dependent variable.

Here are some thoughts and requirements:

1. All teams must fit a neural network model for each task. You should probably error on the side of having too many hidden nodes and then using regularization. You have over 100 predictor variables, and to avoid melting your computer, I suggest that you reduce the number of features before estimating the neural network. Here are some ideas: (1) find principal components and give the first 10–20 PCs to the NN; (2) fit a RF model and use only the 10–20 variables with the most importance; (3) fit a lasso model and only give the variables that enter to the NN.

2. Since we've spent a lot of time on NNs, I have another question

   Q5 What NN architecture and training to you suggest? How many hidden notes? By-pass connectors? Regularization?

   If you are feeling ambitious, you could also try deep learning with a four-layer networks and testing other activation functions such as tanh, ReLU or SmoothReLU, but I'm not expecting this.

3. All predictors are counts or amounts and are probably right skewed. I suspect logging will help.

4. You cannot use NNs to answer the question about power predictors. I'd use lasso/ridge and/or RF/GBM for this task.

5. Don't get stuck pursuing the perfect model. Quickly get to some working models and improve on them. One you have the lags figured out, you should be able to fit a lasso/ridge model in minutes.