Week 3, HW 1: ML Modeling

Matt Goldman
mattgold@microsoft.com

Brian Quistorff
Brian.Quistorff@microsoft.com

June 26, 2017

1. Let's *simulate* some data, so that we can practice forecasting with LASSO & Ridge.

   (a) First, let's simulate a design matrix (X) with $N = 100$ data points, each with $K = 100$ normally distributed covariates (look up the matrix and rnorm functions).

   (b) Second, let's simulate a $K \times 1$ matrix of coefficients (β) and a $N \times 1$ matrix of error terms (e) each from a normal distribution.

   (c) Now let us calculate our observed outcomes as: $Y = X * \beta + e$.

   (d) This is our *training data*, use $cv.glmnet()$ to train a Ridge, LASSO and OLS model on this data.

   (e) Repeat this process to generate a corresponding set of *test data*. Use your estimated OLS, Ridge and LASSO models to generate prediction $\hat{Y}$) on both the training and test data. Then the *mean squared error* of your forecasts on each dataset ($MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2$).

   (f) Write a loop to repeat this process for 200 replications and compute the average mean squared error of each estimator across these replications. Which model seems to be working best on the test data? Which models had the biggest change in performance from the training to the test data?

   (g) Finally, let us repeat steps (a-e) but this time, instead of drawing β from a normal distribution, let us fix the value of $\beta = (5, 5, 5, 5, 0, ...0)$. That is, set the first four elements of β to be 5 and all remaining elements of β to be zero. Which model yielded the lowest out of sample MSE in this second set of simulations? Can you give any reason as to why various estimators might have

performed differently when the true data was generated in this way?

2. Trees!

   (a) Let's see how a tree operates with simple functions. Construct a dataset of N=100 where $x_i = i$ and $e_i = N(0, 1)$.

      i. Let $y = \beta x + e$ where $\beta = 1$. Use rpart to build a model of $y = f(x)$. Look at the splits. Look at how the final group average changes when moving up the domain of x. Does it roughly recover $\beta = 1$ (explain)? When a split occurs how are the rough sizes between the two groups? Visualize the splits.

      ii. Add the variable $y2 = \beta x^2 + e$. Build a model of $y2 = f_2(x)$. How are the sizes roughly split now? Explain (either why the difference or why the same as above). Visualize the splits.

   (b) Now let's see how it does with discontinuities. Construct $y3 = 2 * 1\{x > 50\} + e$.

      i. Estimate $y3 = f_3(x)$ using rpart. How does it do? What is the in-sample MSE?

      ii. Estimate f_3 by LASSO allowing it to use x, x^2, and x^3. How does it do? What is the MSE?

      iii. What guidance can you draw from this exercise?