Problem Set 1

You can provide either two documents (a pdf with the results and the .R script) or (preferably) an interactive presentation using rmarkdown/jupyter notebook. Write clean code and don't cheat. Some students will be asked to present their results. ***Due date: Thursday 22, March.***

Exercise 1

In this exercise, we will generate simulated data, and will then use this data to perform best subset selection.

(a) Generate a predictor *X* of length *n* = 100 distributed uniformly, as well as a noise vector *ε* of length *n* = 100 distributed normally.

(b) Generate a response vector *Y* of length *n* = 100 according to the model

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_4 X^4 + \varepsilon$$

where β_0, β_1, β_2, and β_4 are constants of your choice.

(c) Use the regsubsets() function to perform best subset selection in order to choose the best model containing the predictors $X, X^2, \dots, X^{10}$. What is the best model obtained according to C_p, BIC, and adjusted R^2? Show some plots to provide evidence for your answer, and report the coefficients of the best model obtained.

(d) Repeat (c), using forward stepwise selection and also using backwards stepwise selection. How does your answer compare to the results in (c)?

(e) Now fit a lasso model to the simulated data, again using $X, X^2, \dots, X^{10}$ as predictors. Use cross-validation to select the optimal value of λ. Create plots of the cross-validation error as a function of λ. Report the resulting coefficient estimates, and discuss the results obtained.

(f) Now generate a response vector *Y* according to the model

$$Y = \beta_0 + \beta_7 X^7 + \varepsilon$$

and perform best subset selection and the lasso. Discuss the results obtained.

Exercise 2

In this exercise, we will predict the times that some student raises his hand in class ("raisedhands" variable) using the other features in the College data set. You can find information about the dataset in the Attributes.txt file. The College.csv file contains the data. Use the read.csv() function to read the data set.

(a) Check whether there are any missing values. Clean the data set, while trying to maintain as much information as possible.

(b) Report the dimensions of your clean data set.

(c) Split the data set into a training set and a test set with a 70%-30% ratio.

(d) Fit a linear model using least squares on the training set, and report the test error obtained.
(e) Fit a 3rd degree polynomial that predicts raisedhands using only the VisitedResources feature on the training test and report the test error obtained. Repeat the process using a cubic spline. Create a scatter plot and then add lines for the two models. What do you observe? Did the spline improve the prediction error?
(f) Fit a ridge regression model on the training set, with λ chosen by cross-validation. Report the test error obtained.
(g) Fit a lasso model on the training set, with λ chosen by cross-validation. Report the test error obtained, along with the number of non-zero coefficient estimates.
(h) Fit a model of your choice and provide arguments for that choice. Some criteria could be interpretability, predictive accuracy, cost or domain knowledge.

Exercise 3

Using the data you generated in Exercise 1.b create a dataframe that contains the design matrix and the response variable. You should avoid using ready functions for this exercise.

1) Define a function that takes as inputs some dataframe and the column index of the dataframe that belongs to the response variable y and computes the analytical solution of the least squares minimization problem using matrices.
2) Repeat step (a), for the ridge solution. In this case the new function should take an additional input, the lambda parameter.
3) Define a function that takes as inputs a vector that contains the beta estimates and the design matrix and returns the predicted values for the response variable.
4) Define a function that takes as inputs the response and the predicted values for that response and computes the mean squared error.
5) Define a function that takes as inputs a dataframe, the number of CV folds k and using the above functions performs k-fold cross-validation, where in each iteration:
   a) computes the ols estimates and the rigde estimates
   b) predicts the y values for the two cases
   c) computes the mean squared error for the two cases and stores it in a matrix

   After finishing the CV process, the function prints the average MSE for the two cases.