MCE5901 Homework Set 2

October 1, 2023

NOTICE: The homework is due on Oct. 13 (Friday) 11:59pm. **Please draft your solution with steps and provide the codes that you use for the applied problems.** You will need to submit two files to the blackboard: a PDF file including the solutions and results for all problems, and a file containing the codes for the applied problems. You are allowed, and even encouraged, to discuss the homeworks with your classmates. However, you must **write up the solutions on your own**. Plagiarism and other anti-scholarly behavior will be dealt with severely.

Problem 4 is optional. You will receive 20 points bonus if you solve it and the maximum score of the assignment is 100 points.

Problem 1. Ridge Regression and the Lasso

In order to obtain a better intuition about the behavior of ridge regression and the lasso, consider a simple special case with $n = p$, i.e., the number of data points is equal to the number of features. The traning data set is $\{(x_1, y_1), \ldots, (x_n, y_n)\}$. The design matrix $\mathbf{X}$ is a diagonal matrix with 1's on the diagonal and 0's in all off-diagonal elements. To simplify the problem further, assume also that we are performing regression without an intercept, that is,

$$\hat{Y} = \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p.$$

With these assumptions, let us derive the estimates for $\beta_1, \ldots, \beta_p$ under different methods.

(a) Ordinary least squares. Write out the optimization problem used to estimate the coefficents. Derive the optimal $\beta_1^O, \ldots, \beta_p^O$ (as a function of $y_i, i = 1, \ldots, n$).

(b) Ridge regression. Write out the optimization problem used to estimate the coefficents. Derive the optimal $\beta_1^R, \ldots, \beta_p^R$ (as a function of $y_i, i = 1, \ldots, n$ and tuning parameter λ).

(c) The lasso. Write out the optimization problem used to estimate the coefficents. Derive the optimal $\beta_1^L, \ldots, \beta_p^L$ (as a function of $y_i, i = 1, \ldots, n$ and tuning parameter λ).

(d) Compare β^O, β^R and β^L, what do you find?

Problem 2. Applied problem: Ridge Regression and Lasso

In this exercise, you will get familiar with the scikit-learn library for machine learning in Python. You will implement ridge regression, Lasso and get to see how they work on data. The file "Credit.csv" contains the dataset, which we have seen in class. We are interested in predicting the credit card balance based on the other information.

(a) Check out the data. How many samples are included in the data? What are the features?

(b) For the categorical variables, create dummy variables. You may use "pandas.get_dummies" method.

(c) Standardize the data. You may use "StandardScaler".

(d) Fit the data using "Ridge" function. Set the tuning parameter "alphas = 10**np.linspace(6,-2,100)". Plot the coefficient estimates versus α. What do you find?

(e) Fit the data using "Lasso" function. Set the tuning parameter "alphas = 10**np.linspace(6,-2,100)". Plot the coefficient estimates versus α. What do you find?

(f) We now use cross-validation to choose the tuning parameter. We can do this using the cross-validated ridge regression and Lasso function, "RidgeCV" and "LassoCV". Report the optimal α under the two methods and the corresponding validation MSE. Is there much difference between these two methods?

Reference links: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html

http://www.science.smith.edu/~jcrouser/SDS293/labs/lab10-py.html

Problem 3. Applied problem: Logistic regression

In this exercise, we work on the data set "Smarket.csv". It consists of the percentage returns for the S&P 500 stock index over 1,250 days, from the beginning of 2001 until the end of 2005. For each date, there are the following pieces of information: (1)Lag1 through Lag5: the percentage returns for each of the five previous trading days. (2)Volume: the number of shares traded on the previous day, in billions. (3)Direction: whether the market was Up (positive percentage return) or Down (negative percentage return) on this date. We will fit a logistic regression model in order to predict Direction using Lag1 through Lag5 and Volume.

(a) Split the data into a training set including the samples from 2001 through 2004 and a test set including the samples from 2005.

(b) Produce a scatterplot matrix of all the variables in the training set. To visualize the difference between "Up" days and "Down" days, use the Direction column to determine the hue. Use "import seaborn as sns" and "sns.pairplot(Smarket, hue = "Direction")". Comment on the results.

(c) Use scikit-learn's LogisticRegression class to fit the logistic regression model on the training set. Report the coefficient estimates and the probabilities of each class label for the first ten training observations.

(d) Test the model using the held-out test set. Report the test error rate.

Reference link: https://www.kaggle.com/code/suugaku/islr-lab-3-python

Problem 4. Logistic regression (optional)

Implement Newton's method to fit a logistic regression model to the Smarket data in Problem 3. Write your own version, and do not call a built-in library function. Compare the results you obtain with those obtained in Problem 3.