DS 303 Homework 6
Due: Oct. 09, 2023 on Canvas by 11:59 pm (CT)

Instructions: Homework is to be submitted on Canvas by the deadline stated above. Please clearly print your name and student ID number on your HW.

Show your work (including calculations) to receive full credit. Please work hard to make your submission as readable as you possibly can - **this means no raw `R` output or code** (unless it is asked for specifically or needed for clarity).

Code should be submitted with your homework as a separate file (for example, a .R file, text file, Word file, or .Rmd are all acceptable). You should mark sections of the code that correspond to different homework problems using comments (e.g. `##### Problem 1 #####`).

Problem 1: Follow-up to in-class activity

Let's recap what we learned from our in-class activity from 9/27/23 (see the post on Ed Discussion if you need a refresher).

(a) Clearly the models in Part 2 and Part 3 are not equivalent. Explain again in plain language (using plots if you deem it helpful) why these two approaches will not give the same results.

(b) Load in the `Insurance` dataset again. Show how you can modify the model in Part 2 so that it is **exactly equivalent** to the models in Part 3. Present your code and results to show that the fitted models for males only and females only are exactly the same in Part 2 and Part 3 (after making your modification).

(c) Other than domain knowledge, how might you determine empirically (in a data-driven way) that an interaction term is needed in the model? Justify and then implement your approach here on the `Insurance` dataset.

Problem 2: Predictions in the presence of multicollinearity

(a) Is multicollinearity a problem for making accurate predictions? Make an educated guess based on what we have learned in class.

(b) Let's carry out a simulation study to answer this. We will simulate data with and without multicollinearity. This is our true model:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \epsilon_i$$

where $i = 1, \ldots, 100$, $\epsilon \sim N(\mu = 0, \sigma^2 = 4)$, $\beta_0 = 3$, $\beta_1 = 2$, and $\beta_2 = 4$.
To generate your predictors **with** multicollinearity use the following code:

```
set.seed(42)
x1 = runif(100)
x2 = 0.8*x1 + rnorm(100,0,0.1)
```

Using these predictors, generate your `Y` values in `R`. Check the correlation between `x1` and `x2` using the `cor()` function. Report that value here.

(c) Split your data into a training set and test set. Train your model on the training set.

```
lm(Y ~ x1+x2, data = train)
```

Report the test MSE for this model.

(d) Repeat this process 2,500 times. Each time you'll need to generate a random set of Y's, fit a model on your training set, and then obtain the test MSE for that model. We do not need to generate new predictor values (again, think about why). **Remember to store the test MSE for each iteration and do not set seed**. What is the mean test MSE in this setting when the predictors are highly correlated? Plot a histogram of your 2,500 test MSEs and comment on what you observe.

(e) Now generate predictors **without** multicollinearity using the following code:

```
set.seed(24)
x1 = runif(100)
x2 = rnorm(100,0,1)
```

Using these predictors, generate your `Y` values in `R`. Check the correlation between `x1` and `x2`. Report that value here.

(f) Again run 2,500 simulations to obtain the test MSE of our model when the predictors are not correlated. What is the mean test MSE in this setting when the predictors are not correlated? Plot a histogram of your 2,500 test MSEs and comment on what you see.

(g) Based on our simulation study, is multicollinearity a problem for making accurate predictions? Comment on your findings.

Problem 3: Regularized Regression

For this problem, we will use the `College` data set in the `ISLR2` R package. Our aim is to predict the number of applications (`Apps`) received using the other variables in the dataset.

(a) Split the data set into a training and a test set. Please `set.seed(12)` so that we can all have the same results.

(b) Fit a ridge regression model (using all predictors) on the training set. The function `glmnet`, by default, internally scales the predictor variables so that they will have standard deviation 1. Explain why this scaling is necessary when implementing regularized models.

(c) Find an optimal λ for the ridge regression model on the training set by using 5-fold cross-validation. Report the optimal λ here.

(d) What is the value of the l_2 norm of the estimated regression coefficients (excluding the intercept) associated with the optimal λ for ridge regression? You can evaluate this from the training set.

(e) Using that optimal λ, evaluate your trained ridge regression model on the test set. Report the test MSE obtained.

(f) Find an optimal λ for the lasso regression model on the training set by using 5-fold cross-validation. Report the optimal λ here.

(g) What is the value of the l_1 norm of the estimated regression coefficients (excluding the intercept) associated with the optimal λ for lasso regression? You can evaluate this from the training set.

(h) Using that optimal λ, evaluate your trained lasso regression model on the test set. Report the test MSE obtained.

(i) Comment on your results. How accurately can we predict the number of college applications received? Is there much difference among the test errors resulting from these two approaches?