(PSL) Coding Assignment 2

Contents

Part I: Implement Lasso

One-variable Lasso

First, write a function `one_var_lasso` that takes the following inputs:

$$\mathbf{v} = (v_1, \dots, v_n)^t, \quad \mathbf{z} = (z_1, \dots, z_n)^t, \quad \lambda > 0$$

and solves the following one-variable Lasso problem:

$$\min_b \frac{1}{2n} \sum_{i=1}^{n} (v_i - b z_i)^2 + \lambda |b| = \min_b \frac{1}{2n} \|\mathbf{v} - b \cdot \mathbf{z}\|^2 + \lambda |b|.$$

Check the [derivation] for one-variable lasso.

The CD Algorithm

Next, write your own function `MyLasso` to implement the **Coordinate Descent (CD)** algorithm by repeatedly calling `one_var_lasso`.

In the CD algorithm, at each iteration, we solve a one-variable Lasso problem for β_j while holding the other (p-1) coefficients at their current values:

$$\min_{\beta_j} \frac{1}{2n} \sum_{i=1}^{n} (y_i - \sum_{k \neq j} x_{ik}\beta_k - x_{ij}\beta_j)^2 + \lambda \sum_{k \neq j} |\beta_k| + \lambda |\beta_j|,$$

which is equivalent to solving the following one-variable Lasso problem

$$\min_{\beta_j} \frac{1}{2n} \sum_{i=1}^{n} (v_i - x_{ij}\beta_j)^2 + \lambda |\beta_j|, \quad v_i = y_i - \sum_{k \neq j} x_{ik}\beta_k.$$

Test Your Function

Download the data set `Coding2_Data0.csv`. The data set has 13 predictors, `V1` to `V13`, and one response vector `Y`.

Test your function `MyLasso` on the data set `Coding2_Data0.csv` with a specific lambda sequence. Refer to the sample code in R/Python for the specified lambda sequence.

Compare the estimated Lasso coefficients from your function with those provided in `Coding2_lasso_coefs.csv`. The maximum difference between the two coefficient matrices should be **less than 0.005**. Refer to the sample code for instructions on how to read in the coefficients from `Coding2_lasso_coefs.csv`.

The coefficients in `Coding2_lasso_coefs.csv` are Lasso coefficients returned by R with the option `standardized = TRUE`, so ensure that the X features are centered and scaled in your `MyLasso` function.

Part II: Simulation Study

Consider the following **six** procedures:

- **Full**: Fit a linear regression model using all features
- **Ridge.min** : Ridge regression using `lambda.min`
- **Lasso.min** and **Lasso.1se**: Lasso using `lambda.min` or `lambda.1se`
- **L.Refit**: Refit the model selected by Lasso using `lambda.1se`
- **PCR**: principle components regression with the number of components chosen by 10-fold cross validation

Case I

Download the data set `Coding2_Data1.csv`. The first 14 columns are the same as the data set we used in Part I with `Y` being the response variable (moved to the 1st column). The additional 78 more predictors are the quadratic and pairwise product terms of the original 13 predictors.

- [a] Conduct the following simulation exercise **50** times:
  - In each iteration, randomly split the data into two parts, **75%** for training and **25%** for testing.
  - For each of the **six** procedures, train a model using the training subset and generate predictions for the test subset. Record the **MSPE** (Mean Squared Prediction Error) based on these **test** data predictions.
- [b] Graphically summarize your findings on the MSPE using a strip chart, and consider overlaying a boxplot for additional insights.
- [c] Based on the outcomes of your simulation study, please address the following questions:
  - Which procedure or procedures yield the best performance in terms of MSPE?
  - Conversely, which procedure or procedures show the poorest performance?
  - In the context of Lasso regression, which procedure, **Lasso.min** or **Lasso.1se**, yields a better MSPE?
  - Is refitting advantageous in this case? In other words, does **L.Refit** outperform **Lasso.1se**?
  - Is variable selection or shrinkage warranted for this particular dataset? To clarify, do you find the performance of the **Full** model to be comparable to, or divergent from, the best-performing procedure among the other five?

Case II

Download the data set `Coding2_Data2.csv`. The first 92 columns are identical to those in `Coding2_Data1.csv`, with the addition of 500 columns of artificially generated **noise features**.

- Repeat [a] and [b] above for the **five** procedures **excluding** the **Full** procedure. Graphically summarize your findings on MSPE using a strip chart, and consider overlaying a boxplot for additional insights.
- [c] Address the following questions:

- Which procedure or procedures yield the best performance in terms of MSPE?
- Conversely, which procedure or procedures show the poorest performance?
- Have you observed any procedure or procedures that performed well in Case I but exhibited poorer performance in Case II, or vice versa? If so, please offer an explanation.
- Given that `Coding2_Data2.csv` includes all features found in `Coding2_Data1.csv`, one might anticipate that the best MSPE in Case II would be equal to or lower than the best MSPE in Case I. Do your simulation results corroborate this expectation? If not, please offer an explanation.

What to Submit

- A Markdown (or Notebook) file in HTML format, which contains all necessary code and the corresponding output/results.
- RMD or ipynb files are not needed.
- Set the **seed** at the beginning of part II to be the last 4-dig of your UIN. So we can get the same result if we re-run your code. If your UIN ends with "0496", for example, use "496."
- **One submission per team**. For each assignment, one and only one member submits their work on Coursera/Canvas. Please remember to include the following in your report:
  - the names and netIDs of all team members; the program (MCS-DS or campus) if the team is a mixture of students from these two;
  - a short paragraph detailing the contribution of each member.