Data Analysis Problem

Mohamad Lakkis

March, 2024

This PDF is to explain some of the coding part in the *Data Analysis Problem*

Introduction

Since the goal of this problem is to replicate the figures in the book (ESL), we will do the same things as they did. And by that I mean we will be using the Xtrain scaled everytime. this is not always a good practice, but since the book did that we will be doing the same. So What we will do is that we will scale the Xtrain and use it in all of the code and analysis (i.e. we will forget that we have unscaled Xtrain and we will be using the scaled one as our new Xtrain) We knew that they are using the scaled training set from table 3.3 by looking at the OLS coefficients, and also form looking at the F-statistic!

How to get the unscaled coefficients after we have the scaled ones?

This is in general, we won't be using these formulas in the code (as mentioned in the Introduction!) We know that for linear models, especially penalized ones it is better to scale the features (i.e. the X_{Train},and NOT, Y_{Train}), but once we fit the model using the scaled training data, we will get scaled $\hat{\beta}_{sc}$, so every time we want to make predictions or even try to understand how powerful each predictor is we should scale the new data. So the question is how to get the unscaled coefficients from the scaled ones?
We have with X_i' means that it is scaled and $\hat{\beta}_{i,sc}$ means the ith predictor scaled,

$$Y = \hat{\beta}_{0,sc} + \hat{\beta}_{1,sc}X_1' + ... + \hat{\beta}_{p,sc}X_p'$$

We know that, $X_i' = \frac{X_i - \mu_i}{\sigma_i}$, so plugging in we get,

$$Y = \hat{\beta}_{0,sc} + \hat{\beta}_{1,sc}\frac{X_1 - \mu_1}{\sigma_i} + ... + \hat{\beta}_{p,sc}\frac{X_p - \mu_p}{\sigma_p}$$

And so, by reducing the terms, we get for the intercept

$$\hat{\beta}_0 = \hat{\beta}_{0,sc} - \sum_{i=1}^{p}(\frac{\beta_i - \mu_i}{\sigma_i})$$

with μ_i and σ_i are the mean and std of the X_i in the training data.

Now for the rest of the coefficients we get,

$$\hat{\beta}_i = \frac{\hat{\beta}_{i,sc}}{\sigma_i}$$

And now we have a proper formula to get the unscaled coefficients !

How did we find the correct λ to use for each *df* in the ridge regression (i.e. the λ array in Ridge Regression)?

We know that, it is better to use the SVD decomposition since $X^T X$ might not be of full rank,

$$df_{RR}(\hat{y}) = tr(S) = \sum_{i=1}^{r}(\frac{d_i^2}{d_i^2 + \lambda})$$

With, $S = X(X^T X + \lambda I)^{-1} X$.
The way we get the eigenvalues d_i for the matrix in python is by the use of U, d, VT = np.linalg.svd(Xtrain), with d is an array of the eigenvalues!
Now in order to get the correct λ for a particular *df* we will try different λ and calculate the *df* using the formula above until we get something very close to our desired *df*. (Reminder that by Xtrain we mean the scaled one as mentioned in the Introduction!)

How did we find the correct λ to use for each shrinking factor s (i.e. what is the relation between s and λ)?

So we noticed that s $\approx \frac{\|\hat{\beta}_\lambda^L\|_1}{\|\hat{\beta}\|_1}$, with $\|\hat{\beta}_\lambda^L\|_1$ is the ℓ_1 norm of the vector of the the Lasso coefficients for a given value of λ and $\hat{\beta}$ is the ℓ_1 norm of the vector for the OLS coefficients(i.e. Lasso with $\lambda = 0$).
Why is that the case?
Since both of them measures how much the ℓ_1 norm of the coefficient vector has shrunk due to the penalty as compared to the ℓ_1 norm of the OLS coefficient vector(or lasso with $\lambda = 0$). As the ratio decreases from 1 to 0, the Lasso penalty becomes more dominant, shrinking the coefficients towards zero !

Now, in order to get the particular λ that will get us the correct s, we will try different values of λ and get the lasso coefficients and calculate the s. Notice that the denominator is constant (relative to the Xtrain of course !)

How can we get the main features's coefficients in PCR?

Say we have the training data X of dimension n x p, with n number of independent observations and p is the number of features !
Once we do a pca with k nb of components we will get k features, and each feature is a linear combination of the main p features. So the components matrix(say M) is of dimension k x p, with element M[i][j] is the coefficient of the j'th main feature in the i'th component. (Simple Linear combination)
Now once we apply the Linear Model to these pca components we will get a matrix (say S) of coefficients of dimension k x 1 with S[i][1] represents the coefficient of i'th component which is simply a linear combination from the main features (and it is represented in the M matrix). So with simple linear algebra we find that in order to get the main coefficients after applying the pcr is simply M^T x S, which will give us a result matrix of p x 1, which represents the coefficients of the initial features !

Final Notes

Notice that if we compare figure 3.8 and figure 3.10 we can see that in the lasso model the coefficients can shrink to exactly 0 (figure 3.10), but in the ridge regression model the coefficients get very close to 0 but not exactly 0 (figure 3.8 see PLOTS AND TABLES section). In addition to that, we can see that when s = 1, and when df(λ) = 8, which in both cases this means that $\lambda = 0$, they become the OLS coefficients, which is shown in both figures as having the same coefficients! And keep in mind that these coefficients that we are getting are in terms of the scaled X, so in order to get the coefficients in terms of the unscaled X we will have to do what we discussed earlier in the first point!

Note that in fig 3.7 there is two, one with the ranges and one without them

Notice that we could have chosen the best parameter for each model based on the lowest CV estimate of error(or when they become very close in values take the model with the lowest complexity to avoid overfitting), but in this case we chose to take the best model based on the criterion of the lowest test error using the test set !
Notice also that for both PCR and Best Subset Selection, the same null model is used (i.e. at # of components = 0, or k = 0), we applied cross-validation on it to get the range since witout cross validation we will just get a point !

PLOTS AND TABLES

Table 1: Table 3.3

Term	OLS	Best Subset	RIDGE	LASSO	PCR
Intercept	2.452345	2.452345	2.452345	2.452345	2.452345
lcavol	0.711041	0.774017	0.517054	0.569856	0.566307
lweight	0.290450	0.349274	0.271865	0.226034	0.320860
age	-0.141482	0.000000	-0.078536	-0.000000	-0.152568
lbph	0.210420	0.000000	0.186844	0.098608	0.214382
svi	0.307300	0.000000	0.259546	0.166214	0.319707
lcp	-0.286841	0.000000	-0.068079	0.000000	-0.050024
gleason	-0.020757	0.000000	0.030533	0.000000	0.226861
pgg45	0.275268	0.000000	0.162426	0.061197	-0.063150
Test Error	0.521274	0.492482	0.487273	0.452334	0.448309
Std Error	0.131771	0.130864	0.126833	0.116512	0.146816

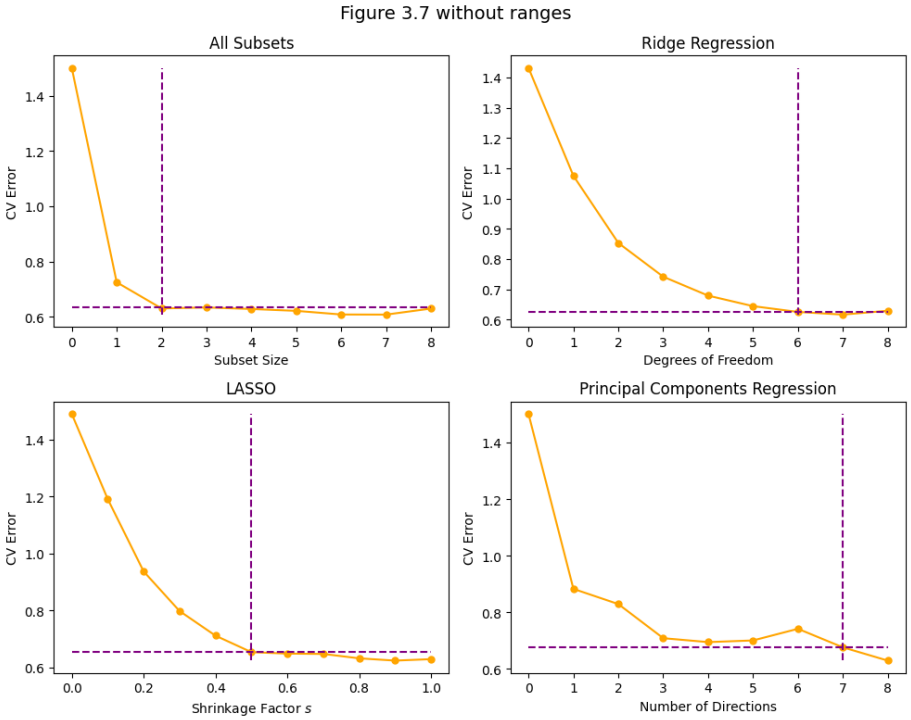

Figure 1: Figure 3.7: Estimated prediction error curves and their standard errors for various selection and shrinkage methods(without ranges).

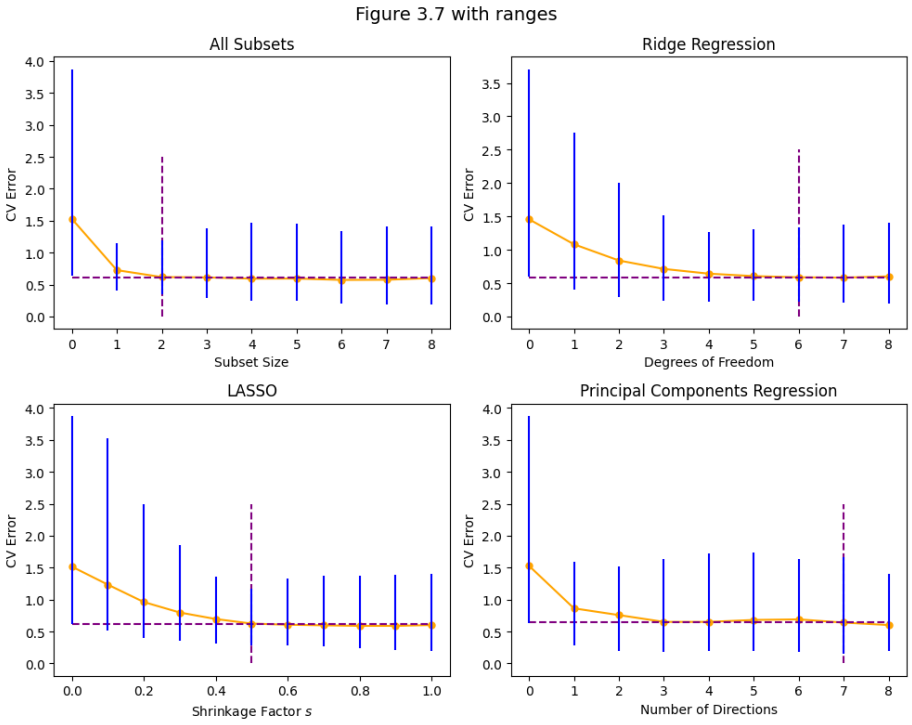

Figure 2: Figure 3.7: Estimated prediction error curves and their standard errors for various selection and shrinkage methods (with ranges).

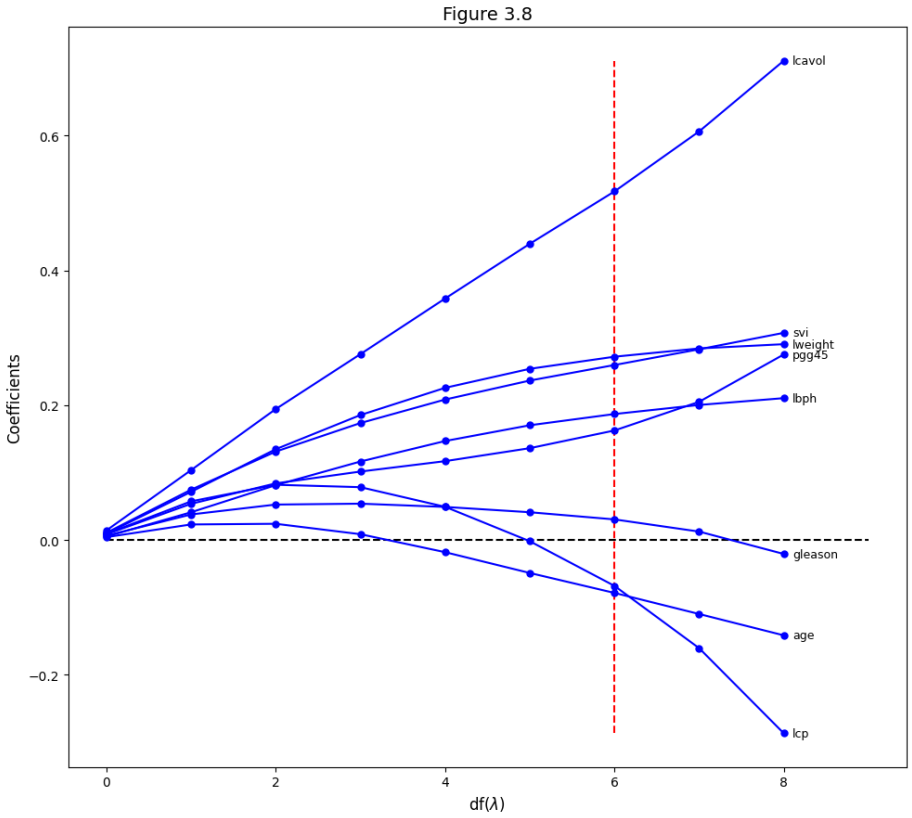

Figure 3: Figure 3.8: Coefficients of different ridge regression model are plotted versus df(λ). A vertical line is drawn at df $= 6$, the value chosen by the smallest test error.

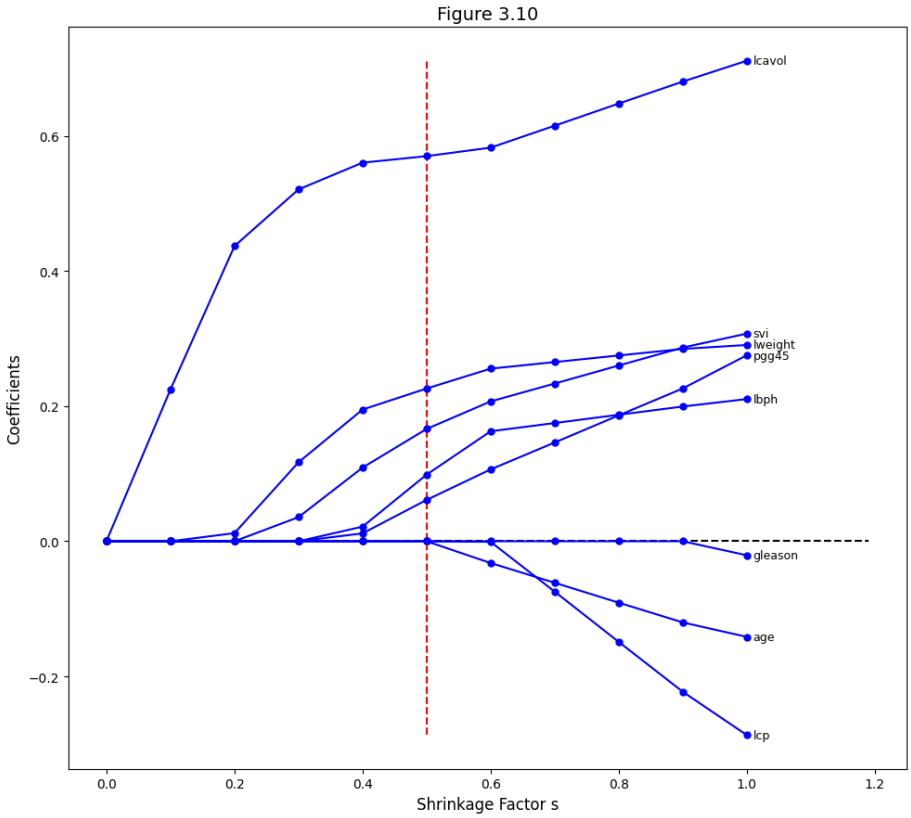

Figure 4: Figure 3.10: Coefficients of different lasso models are plotted versus s. A vertical line is drawn at s = 0.5 the value chosen by the smallest test error.