Problem Set 4: Linear Regression Models

Stat 154, Fall 2017, Prof. Sanchez

Due date: Fr Oct-20 (before midnight)

Instructions

Do not give raw computer output as your main answer to any question. Remember that providing a clear and reasonable justification of your answers is at least as important as getting the right answer.

Problem 1 (10 pts)

Multicollinearity is easy to detect in the two-predictor case; we need only look at the value of $r_{12} = cor(X_1, X_2)$. When there are more than two regressors, however, inspection of the r_{ij} is not sufficient.

For example, assume that we have four predictors X_1, X_2, X_3 and X_4, and correlation coefficients, r_{ij} are $r_{12} = r_{13} = r_{23} = 0$, with variances $\sigma_1^2 = \sigma_2^2 = \sigma_3^2$, and $X_4 = X_1 + X_2 + X_3$.

Show that $r_{14} = r_{24} = r_{34} = 0.577$

That is, three of the pairwise correlations are zero and the other three are not especially large, yet we have the most extreme multicollinearity problem possible in that there is an exact linear combination between the four regressors.

Recall that:

$$r_{ij} = cor(X_i, X_j) = \frac{cov(X_i, X_j)}{\sqrt{var(X_i)}\sqrt{var(X_j)}}$$

Problem 2 (10 pts)

In Partial Least Squares Regression, we obtain uncorrelated components $\mathbf{z_1}, \mathbf{z_2}, \ldots, \mathbf{z_h}$ that summarize variability in predictors $\mathbf{X}$ as well as capture variation in the response $\mathbf{y}$.

One of the properties of the original PLS regression algorithm is that it produces ofthogonal components. Show that any two components $\mathbf{z_h}$ and $\mathbf{z_l}$ $(h \neq l)$ are indeed orthogonal, that is:

$$\mathbf{z_h^T z_l} = 0 \quad \text{for} \quad h \neq l$$

Hint: The demonstration is done by recursivity.

Problem 3 (100 pts)

In this problem, you will compare various (linear) regression models using the data set `prostate`. The overall goal is to replicate (as much as possible) the *Prostate Cancer* case study described in Chapter 3 of the book *The Elements of Statistical Learning* (ESL)—not to confuse with ISL—by Hastie, Tibshirani, and Friedman. You can find the pdf version of the 2nd edition at:

https://web.stanford.edu/~hastie/ElemStatLearn/

The data file `prostate.data` is available at:

https://web.stanford.edu/~hastie/ElemStatLearn/datasets/prostate.data

(or you can also find it in the R package `"ElemStatLearn"`).

Unfortunately, the authors of ESL don't describe all the details of the extensive analysis done in chapter 3. And it may be possible that there are a couple of discrepancies (errors?) in some of their displayed results, although nothing has been reported in the Errata section of the book's website (let's see if we can shed some light about such discrepancies).

Data set

The example *Prostate Cancer* is introduced in section 3.2.1, page 49 of ESL. The data is from a study by Stamey et al. (1989) in which the level of prostate-specific antigen and a number of clinical measures in men who were about to receive a radical prostatectomy. The variables are:

- `lcavol`: log cancer volume
- `lweight`: log prostate weight
- `age`: age of patient
- `lbph`: log of the amount of benign prostatic hyperplasia
- `svi`: seminal vesicle invasion
- `lcp`: log of capsular penetration
- `gleason`: Gleason score
- `pgg45`: percent of Gleason scores 4 or 5
- `lpsa`: log of prostate-specific antigen (response variable)

Models to be fitted

You will apply the following methods—via the associated function—to predict `lpsa` using the rest of the variables as predictors:

- Ordinary Leat Squares regression (OLS), with function `lm()`.
- Best subset regression, with function `regsubsets()` in `"leaps"`.
- Principal Components regression (PCR), with function `pcr()` in `"pls"`.
- Partial Least Squares regression (PLSR), with function `pls()` in `"pls"`.

- Ridge regression (RR), with function `glmnet()` and `cv.glmnet()` in `"glmnet"`.
- Lasso regression (lasso), with function `glmnet()` and `cv.glmnet()` in `"glmnet"`.

Take a look at chapter 6 in ISL, especially the computer lab section 6.5, to learn about the listed functions (and packages). You may also want to look at the following vignettes:

- https://cran.r-project.org/web/packages/pls/vignettes/pls-manual.pdf
- https://web.stanford.edu/~hastie/glmnet/glmnet_alpha.html

After completing the model building cycle, the primary purpose is to obtain a table of results similar to table 3.3 of ESL, in page 63 (see screenshot below). You probably won't be able to reproduce the results of the book, but the idea is to do something similar.

TABLE 3.3. *Estimated coefficients and test error results, for different subset and shrinkage methods applied to the prostate data. The blank entries correspond to variables omitted.*

Term	LS	Best Subset	Ridge	Lasso	PCR	PLS
`Intercept`	2.465	2.477	2.452	2.468	2.497	2.452
`lcavol`	0.680	0.740	0.420	0.533	0.543	0.419
`lweight`	0.263	0.316	0.238	0.169	0.289	0.344
`age`	−0.141		−0.046		−0.152	−0.026
`lbph`	0.210		0.162	0.002	0.214	0.220
`svi`	0.305		0.227	0.094	0.315	0.243
`lcp`	−0.288		0.000		−0.051	0.079
`gleason`	−0.021		0.040		0.232	0.011
`pgg45`	0.267		0.133		−0.056	0.084
Test Error	0.521	0.492	0.492	0.479	0.449	0.528
Std Error	0.179	0.143	0.165	0.164	0.105	0.152

Training and Test sets

The data table contains an additional column `train` (logical values) indicating which observations form the *training* set (`TRUE`), and which observations form the *test* set (`FALSE`).

- Use the column `train` to split the data into a training set and a test set. There should be 67 training observations, and 30 test observations.
- You will use the training set to fit all the models, and perform a *model assessment* stage for each applied method.
- You will use the test set in the *model selection* stage to determine which method provides the best predictive performance.

Correlations of predictors, and some preprocessing (10 pts)

Obtain the matrix of correlations of predictors. These correlations are like those displayed in table 3.1 (page 50):

TABLE 3.1. *Correlations of predictors in the prostate cancer data.*

	lcavol	lweight	age	lbph	svi	lcp	gleason
lweight	0.300						
age	0.286	0.317					
lbph	0.063	0.437	0.287				
svi	0.593	0.181	0.129	−0.139			
lcp	0.692	0.157	0.173	−0.089	0.671		
gleason	0.426	0.024	0.366	0.033	0.307	0.476	
pgg45	0.483	0.074	0.276	−0.030	0.481	0.663	0.757

Once you've split the data and selected the training set, you need to standardize the predictors (mean = 0, variance = 1). Confirm you get the following `summary()` statistics

- summary statistics for lcavol, lweight, age

```
    lcavol           lweight              age          
Min.   :-2.1411   Min.   :-2.62526   Min.   :-3.16524  
1st Qu.:-0.6641   1st Qu.:-0.62054   1st Qu.:-0.49935  
Median : 0.1242   Median :-0.05755   Median : 0.03382  
Mean   : 0.0000   Mean   : 0.00000   Mean   : 0.00000  
3rd Qu.: 0.8334   3rd Qu.: 0.54029   3rd Qu.: 0.56700  
Max.   : 2.0180   Max.   : 2.42189   Max.   : 1.89994  
```

- summary statistics for lbph, svi, lcp

```
     lbph                svi                lcp          
Min.   :-0.99595   Min.   :-0.5331   Min.   :-0.8368  
1st Qu.:-0.99595   1st Qu.:-0.5331   1st Qu.:-0.8368  
Median :-0.08385   Median :-0.5331   Median :-0.4171  
Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000  
3rd Qu.: 1.00848   3rd Qu.:-0.5331   3rd Qu.: 0.8631  
Max.   : 1.54057   Max.   : 1.8480   Max.   : 2.0496  
```

- summary statistics for gleason, pgg45

```
   gleason            pgg45        
Min.   :-1.032   Min.   :-0.8965  
1st Qu.:-1.032   1st Qu.:-0.8965  
Median : 0.379   Median :-0.3846  
Mean   : 0.000   Mean   : 0.0000  
3rd Qu.: 0.379   3rd Qu.: 0.8099  
Max.   : 3.200   Max.   : 2.5163  
```

Least Squares Model (10 pts)

The first linear model you will fit is an ordinary least squares regression. Regress the response `lpsa` on the standardized predictors (using the training data). See if you can reproduce the table 3.2 in ESL, page 48.

TABLE 3.2. *Linear model fit to the prostate cancer data. The Z score is the coefficient divided by its standard error (3.12). Roughly a Z score larger than two in absolute value is significantly nonzero at the $p = 0.05$ level.*

Term	Coefficient	Std. Error	Z Score
`Intercept`	2.46	0.09	27.60
`lcavol`	0.68	0.13	5.37
`lweight`	0.26	0.10	2.75
`age`	−0.14	0.10	−1.40
`lbph`	0.21	0.10	2.06
`svi`	0.31	0.12	2.47
`lcp`	−0.29	0.15	−1.87
`gleason`	−0.02	0.15	−0.15
`pgg45`	0.27	0.15	1.74

Note: I have the suspicion that the first three coefficients reported in the book's table may be wrong. Why? Because if you divide the coefficient by the standard error, you don't get the Z score that is displayed on the table 3.2.

Best Subset Regression (10 pts)

Use the function `regsubsets()`, from the package `"leaps"`, to find the best subset regression.

PCR and PLSR (40 pts)

Use the functions `pcr()` and `plsr()`, from the packages `"pls"`, to fit PCR and PLSR models with the training data, respectively, using ten-fold cross validation. Since the predictors are already standardized, you don't really need to set the argument `scale = TRUE`. Make sure to set a random seed so you can reproduce all results.

Report the tuning parameter (i.e. number of components), and the associated coefficients, of the model fits with the smallest CV-MSE.

Make a plot of *Profiles of Coefficients* for each method; the x-axis corresponds to the number of components, and the y-axis corresponds to the coefficients.

Also, make a plot of the CV-MSE for each method. An example of these types of plots is in figure 6.20 of ISL, page 236 (see screen-capture below).

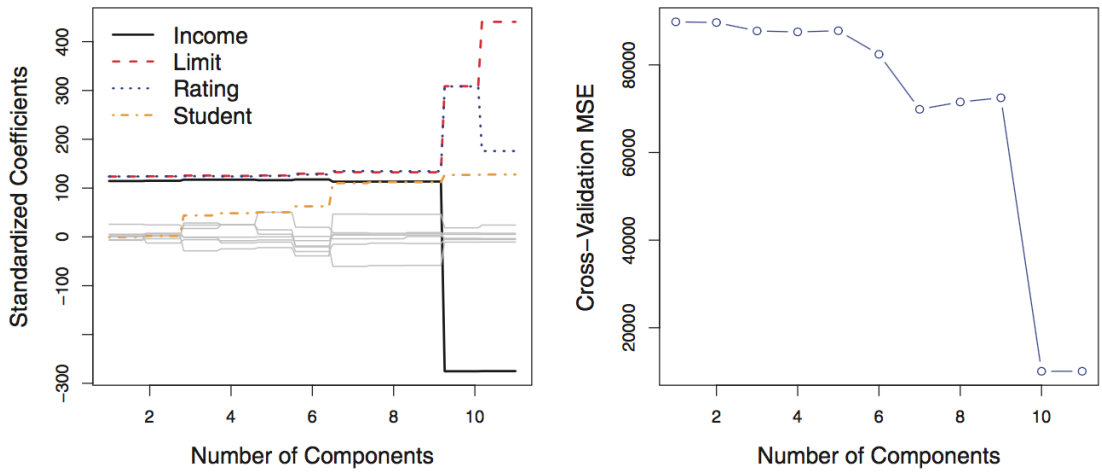


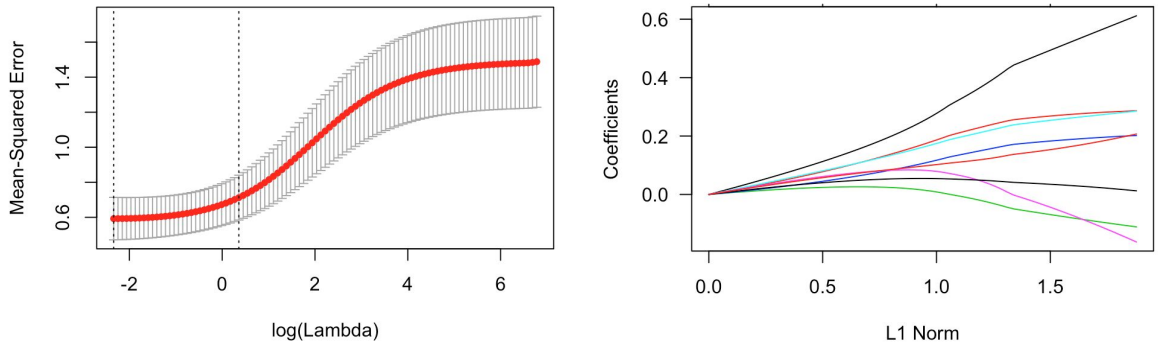
FIGURE 6.20. Left: *PCR standardized coefficient estimates on the* `Credit` *data set for different values of* M. Right: *The ten-fold cross validation MSE obtained using PCR, as a function of* M.

RR and Lasso (40 pts)

Use the function `cv.glmnet()`, from the package `"glmnet"`, to fit Ridge Regression and Lasso models with the training data, respectively, using ten-fold cross validation. Make sure to set a random seed so you can reproduce all results. Report the tuning parameter (i.e. λ).

Use the `glmnet()` function to refit a model with the chosen minimum λ. This will allow you to recover the associated coefficients of the chosen model.

See the `plot()` methods for `plot.cv.glmnet` and `plot.glmnet` to obtain a plot of the CV-MSE, and a plot of coefficients profiles. An example of these types of plots (for ridge regression) is in the screen-capture below.



Model Selection (20 pts)

The final stage of the predictive modeling cycle consists of selecting the best model among the candidates obtained in the model assessment stages. That is, you should have six candidate models: OLS, best subset, PCR, PLSR, RR, and Lasso.

The next step is to determine the best model using the test set. This means computing *test MSE* for each candidate model, and selecting the one with the smallest test MSE. (10 pts)

You should then form a table similar to ESL's table 3.3; no need to report the standard error, just the test error i.e. test MSE (10 pts):

TABLE 3.3. *Estimated coefficients and test error results, for different subset and shrinkage methods applied to the prostate data. The blank entries correspond to variables omitted.*

Term	LS	Best Subset	Ridge	Lasso	PCR	PLS
`Intercept`	2.465	2.477	2.452	2.468	2.497	2.452
`lcavol`	0.680	0.740	0.420	0.533	0.543	0.419
`lweight`	0.263	0.316	0.238	0.169	0.289	0.344
`age`	−0.141		−0.046		−0.152	−0.026
`lbph`	0.210		0.162	0.002	0.214	0.220
`svi`	0.305		0.227	0.094	0.315	0.243
`lcp`	−0.288		0.000		−0.051	0.079
`gleason`	−0.021		0.040		0.232	0.011
`pgg45`	0.267		0.133		−0.056	0.084
Test Error	0.521	0.492	0.492	0.479	0.449	0.528
Std Error	0.179	0.143	0.165	0.164	0.105	0.152

Comment on the results of the table you obtained. And compare them with the table reported in ESL.