Math 583 HW 7 2023 Due Wednesday, Oct. 18. Problems A)–E). Two pages.

In the Math lab, the computers in the back, 10)–25), tend to have the *R* packages `glmnet` and `pls`. If you have *R* on your personal computer, you may need to install packages the first time you use a given computer.

install.packages("glmnet")
install.packages("pls")

See the near the top of the (http://parker.ad.siu.edu/Olive/hdrhw.txt) homework file. SIU computers probably will not allow you to install packages. Do not forget the two source commands from near the top of this file.

A) Consider the MLR model $\boldsymbol{Z} = \boldsymbol{W}\boldsymbol{\eta} + \boldsymbol{e}$. Give the formulas for a) $\hat{\boldsymbol{\eta}}_{OLS}$, b) $\hat{\boldsymbol{\eta}}_{OPLS}$, c) $\hat{\boldsymbol{\eta}}_{MMLE}$, and d) $\hat{\boldsymbol{\eta}}_{R}$.

(Note: if variable selection picks model I, then in the above formulas, replace $\boldsymbol{W}$ by $\boldsymbol{W}_I$. You may assume that $\boldsymbol{W}$ has the standardized predictors $\boldsymbol{w}_i$ for the MMLE.)

B) Consider choosing $\hat{\boldsymbol{\eta}}$ to minimize the elastic net criterion

$$Q(\boldsymbol{\eta}) = RSS(\boldsymbol{\eta}) + \lambda_1 \|\boldsymbol{\eta}\|_2^2 + \lambda_2 \|\boldsymbol{\eta}\|_1$$

where $\lambda_i \geq 0$ for $i = 1, 2$.

a) Which values of λ_1 and λ_2 correspond to ridge regression? (For example, both are zero, λ_1 is zero, or λ_2 is zero.)

b) Which values of λ_1 and λ_2 correspond to the OLS full model?

C) Let $Y = \alpha + \boldsymbol{x}^T\boldsymbol{\beta} + e$. The k-component estimator

$$\hat{\boldsymbol{\beta}}_{kE} = \hat{\boldsymbol{A}}_{k,n}^T (\hat{\boldsymbol{A}}_{k,n} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}} \hat{\boldsymbol{A}}_{k,n}^T)^{-1} \hat{\boldsymbol{A}}_{k,n} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x},Y}.$$

Suppose $k = p$ and $\hat{\boldsymbol{A}}_{p,n}^{-1}$ exists. Show that $\hat{\boldsymbol{\beta}}_{pE} = \hat{\boldsymbol{\beta}}_{OLS}$.

D) There are several ways to compute k-component PLS estimators. The simplest way is to do the OLS regression on $W_1, ..., W_k$ where $W_j = \hat{\boldsymbol{\eta}}_j^T \boldsymbol{x}$ and $\hat{\boldsymbol{\eta}}_j = \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}}^{j-1} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}Y}$, and $k < n - 1$. Then the one component PLS estimator is OPLS while the 2-component PLS estimator regresses Y on $W_1 = \hat{\boldsymbol{\eta}}_1^T \boldsymbol{x} = \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}Y}^T \boldsymbol{x}$ and $W_2 = \hat{\boldsymbol{\eta}}_2^T \boldsymbol{x} = [\hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{x}Y}]^T \boldsymbol{x}$.

The *slpack* function `tpls` computes the 2-component PLS estimator in this way, and also uses the function `plsr` from the package *pls* to compute the two component PLS estimator. For example

```
#ch3 two component PLS
tpls(belx,bely)
$b2pls
        X
0.5041478
$b2  #from plsr function using library(pls)
        X
0.5041478
```

a) Copy and paste the commands for this part to get the two component PLS estimator from regressing Y = brain weight on several predictors. Copy and paste the output into *Word*.

b) Are the two estimators b2pls and b2 the same (equal to $\hat{\boldsymbol{\beta}}_{2PLS}$)?

E) The following simulation will compare lasso and lasso variable selection (OLS applied to the variables with nonzero lasso $\hat{\boldsymbol{\beta}}_i$) using all n cases and a sequential method of data splitting that attempts to use fewer than $n/2$ cases for $H = H_d$. The output gives the number of nonzero lasso coefficients $\hat{\boldsymbol{\beta}}_i$, including the constant, for lasso applied to the n_d cases in H_d. The program computed large sample 95% PIs for lasso applied to all n cases (*lsapi*), lasso variable selection applied to all n cases (*LVSpi*), lasso applied to V_d (*lsplitpi*), and the model selected using H_d applied to V_d (*splitpi*). The second and fourth models used OLS applied to the n cases or the cases in V_d. The coverage and average length of the prediction intervals was given. A value of *noundfit* greater than 4500 indicates that, in over 90% of the 5000 runs, the lasso model I did not underfit: $S \subseteq I$. The program uses $\boldsymbol{\beta} = (\beta_1, 1, .., 1, 0, ..., 0)^T$ with constant β_1, k ones and $p-k-1$ zeros.

Table 1: mlrsplit, J=5, type=3

n	p/k	psi= ψ	mnnd/mnad	lsapi	LVSpi	lsplitpi	splitpi	noundfit
100	4	0.8000	33.3354	0.9676	0.9672	0.9768	0.9764	4306
	1		2.6874	4.0502	4.0545	4.6570	4.6614	

In the above table, $n = 100$, $p = 4$, $k = 1$, $n_d = n_h$ averaged 33.33 in the 5000 runs, a_d averaged 2.69 (2 for the nonzero β_1 and β_2 would be ideal), and in 4306 out of 5000 runs $S = 1, 2 \subseteq I$, the model selected by lasso. All models I contained a constant, so in 694 runs, the predictor x_2 was not selected by lasso. The lasso PI using all n cases had coverage 0.9676 (the percentage of runs where Y_f was in the PI) with average PI length $= 4.05$ (the asymptotically optimal length is 2.99, but $n = 100$ is small).

For the homework, we will use 100 runs instead of 5000 runs, but the simulation still takes a few minutes. With 100 runs, PI coverage between 0.89 and 1.0 gives no reason to believe that the actual coverage is not close to the nominal coverage of 0.95.

a) Copy and paste the commands for this part into *R*. Then make a table similar to the above table. Here $n = 100, p = 100, k = 1$, and N(0,1) errors are used.

b) Copy and paste the commands for this part into *R*. Then make a table similar to the above table. Here $n = 100, p = 100, k = 10$, and N(0,1) errors are used. Now there is much more underfitting, but lasso picks models good for prediction, so some of the PIs have adequate coverage.