STAT406- Methods of Statistical Learning Lecture 6

Matias Salibian-Barrera

UBC - Sep / Dec 2018

“Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise.”

John Tukey. The future of data analysis. Annals of Mathematical Statistics, 33(1), (1962), p. 13.

Progress report?

- Piazza - course content discussions
- Coming to “lectures” isn’t enough: read reference texts, dissect / break code on Github, discuss w/peers
- Google is not your friend

Effective degrees of freedom

- How many "effective" parameters are we using?
- In linear regression, we have p parameters
- A more general definition is as follows. For a fitting method producing $\hat{y}_1$, $\hat{y}_2$, $\ldots$, $\hat{y}_n$,

$$\text{edf} = \frac{1}{\sigma^2} \sum_{i=1}^{n} \text{cov}\left(\hat{y}_i, y_i\right)$$

Efron, B. (1986). How biased is the apparent error rate of a prediction rule? Journal of the American Statistical Association, 81(394):461-470.

Effective degrees of freedom

- It is easy to see that for least squares predictors, we have

$$\hat{\mathbf{y}} = \mathbf{H}\,\mathbf{y}$$

with

$$\mathbf{H} = \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'$$

and

$$\mathrm{edf} = \frac{1}{\sigma^2}\sum_{i=1}^{n}\mathrm{cov}\left(\hat{y}_i, y_i\right) = \mathrm{trace}\left(\mathbf{H}\right) = p$$

Effective degrees of freedom

- More in general, for any linear predictor

$$\hat{\mathbf{y}} = \mathbf{S}\,\mathbf{y}$$

we have

$$\text{edf} = \text{trace}\,(\mathbf{S}) = \sum_{i=1}^{n} \mathbf{S}_{i,i}$$

Effective degrees of freedom

- The ridge regression fit satisfies

$$\hat{\mathbf{y}} = \mathbf{S}_\lambda \, \mathbf{y}$$

where

$$\mathbf{S}_\lambda = \mathbf{X} \left(\mathbf{X}'\mathbf{X} + \lambda \, \mathbf{I}_p \right)^{-1} \mathbf{X}'$$

$$\text{trace}\left(\mathbf{S} \right) = ?$$

Effective degrees of freedom

- Using the singular value decomposition (SVD) of $\mathbf{X}$

$$\mathbf{X} = \mathbf{U}\,\boldsymbol{\Lambda}\,\mathbf{V}'$$

where $\mathbf{U} \in \mathbb{R}^{n\times p}$, $\mathbf{V} \in \mathbb{R}^{p\times p}$ with

$$\mathbf{U}'\mathbf{U} = \mathbf{I}_p = \mathbf{V}'\,\mathbf{V}$$

and

$$\boldsymbol{\Lambda} = \text{diag}\left(d_1, \ldots, d_p\right) ,$$

we have

$$\text{trace}\left(\mathbf{S}\right) = \sum_{i=1}^{p} \left(\frac{d_i^2}{d_i^2 + \lambda} \right)$$

Effective degrees of freedom

- For example, in the Air Pollution data example, if we use

$$\lambda = \exp(6)$$

we get

$$\text{edf} = 9.9$$

Model / feature selection - LASSO

- Another regularized method is given by LASSO

$$\min_{\alpha,\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \alpha - \boldsymbol{\beta}' \mathbf{x}_i \right)^2 + \lambda \sum_{j=1}^{p} \left| \beta_j \right|$$

$$\min_{\alpha,\boldsymbol{\beta}} \sum_{i=1}^{n} \left(y_i - \alpha - \boldsymbol{\beta}' \mathbf{x}_i \right)^2 + \lambda \left\| \boldsymbol{\beta} \right\|_1$$

for some $\lambda > 0$

Model / feature selection - LASSO

- The above is equivalent to

$$\min_{\alpha,\beta} \sum_{i=1}^{n} \left(y_i - \alpha - \beta' \mathbf{x}_i \right)^2$$

subject to

$$\sum_{j=1}^{p} \left| \beta_j \right| \leq K$$

for some $K > 0$

LASSO

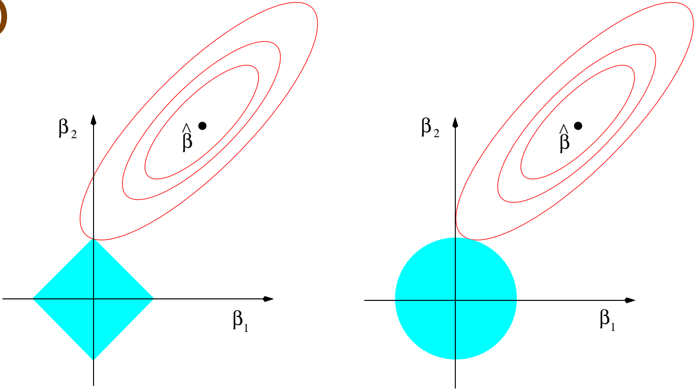
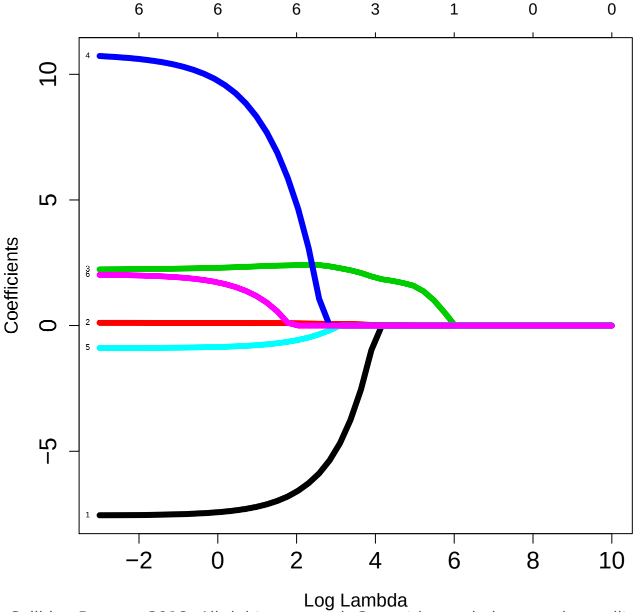


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions* $|\beta_1| + |\beta_2| \leq t$ *and* $\beta_1^2 + \beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the least squares error function.*

Credit data - glmnet output



Credit data - glmnet output

```
a <- glmnet(x=xm, y=yc, lambda=lambdas,
    family='gaussian', alpha=1, intercept=FALSE)

> coef(a, s=1)
7 x 1 sparse Matrix of class "dgCMatrix"
                     1
(Intercept)   .
Income      -7.4285710
Limit        0.1078894
Rating       2.3006418
Cards        9.7499618
Age         -0.8515917
Education    1.7182477
```

Credit data - glmnet output

```
> coef(a, s=exp(4))
7 x 1 sparse Matrix of class "dgCMatrix"
                        1
(Intercept)   .
Income       -0.63094341
Limit         0.02749778
Rating        1.91772580
Cards         .
Age           .
Education     .
```

Credit data - another implementation

```
> library(lars)
> b <- lars(x=xm, y=yc, type='lasso', intercept=FALSE)
> coef(b)
        Income      Limit   Rating     Cards        Age Education
[1,]  0.000000 0.00000000 0.000000  0.000000  0.0000000  0.000000
[2,]  0.000000 0.00000000 1.835963  0.000000  0.0000000  0.000000
[3,]  0.000000 0.01226464 2.018929  0.000000  0.0000000  0.000000
[4,] -4.703898 0.05638653 2.433088  0.000000  0.0000000  0.000000
[5,] -5.802948 0.06600083 2.545810  0.000000 -0.3234748  0.000000
[6,] -6.772905 0.10049065 2.257218  6.369873 -0.6349138  0.000000
[7,] -7.558037 0.12585115 2.063101 11.591558 -0.8923978  1.998283
> b

Call:
lars(x = xm, y = yc, type = "lasso", intercept = FALSE)
R-squared: 0.878
Sequence of LASSO moves:
     Rating Limit Income Age Cards Education
Var       3     2      1   5     4         6
Step      1     2      3   4     5         6
```