MATH 829: Introduction to Data Mining and Analysis
Penalizing the coefficients

Dominique Guillot

Departments of Mathematical Sciences
University of Delaware

February 22, 2016

Shrinkage methods

Penalizing the coefficients:

- Suppose we want to restrict the number or the size of the regression coefficients.
- Add a penalty (or "price to pay") for including a nonzero coefficient.

Examples: Let $\lambda > 0$ be a parameter.

1.
$$\hat{\beta}^0 = \operatorname*{argmin}_{\beta\in\mathbb{R}^p} \left(\|y - X\beta\|_2^2 + \lambda \sum_{i=1}^p \mathbf{1}_{\beta_i \neq 0}\right).$$

   - Pay a fixed price λ for including a given variable into the model.
   - Variables that do not significantly contribute to reducing the error are excluded from the model (i.e., $\beta_i = 0$).
   - Problem: difficult to solve (combinatorial optimization). Cannot be solved efficiently for a large number of variables.

Shrinkage methods (cont.)

Relaxations of the previous approach:

2. Ridge regression/Tikhonov regularization:

$$\hat{\beta}^{\text{ridge}} = \operatorname*{argmin}_{\beta\in\mathbb{R}^p} \left(\|y - X\beta\|_2^2 + \lambda \sum_{i=1}^p \beta_i^2\right).$$

   - Shrinks the regression coefficients by imposing a penalty on their size.
   - $\text{Penalty} = \lambda \cdot \|\beta\|_2^2$.
   - Problem equivalent to $\hat{\beta}^{\text{ridge}} = \operatorname{argmin}_{\beta\in\mathbb{R}^p} \|y - X\beta\|_2^2 \text{ subject to } \sum_{i=1}^p \beta_i^2 \leq t$.
   - Penalty is a smooth function.
   - Easy to solve (solution can be written in closed form).
   - Generally does not set any coefficient to zero (no model selection).
   - Can be used to "regularize" a rank deficient problem ($n < p$).

Ridge regression: closed form solution

We have

$$\begin{aligned}\frac{\partial}{\partial\beta}\left(\|y - X\beta\|_2^2 + \lambda \sum_{i=1}^p \beta_i^2\right) &= 2(X^TX\beta - X^Ty) + 2\lambda\sum_{i=1}^p \beta_i \\ &= 2\left((X^TX + \lambda I)\beta - X^Ty\right).\end{aligned}$$

Therefore, the critical points satisfy

$$(X^TX + \lambda I)\beta = X^Ty.$$

Note: $(X^TX + \lambda I)$ is positive definite, and therefore invertible.

Therefore, the system has a **unique** solution. Can check using the Hessian that the solution is a minimum. Thus,

$$\beta^{\text{ridge}} = (X^TX + \lambda I)^{-1}X^Ty.$$

Remarks:

- When $\lambda > 0$, the estimator is defined even when $n < p$.
- When $\lambda = 0$ and $n > p$, we recover the usual least squares solution.
- Makes rigorous "adding a multiple of the identity" to X^TX.

The Lasso

3. The Lasso (Least Absolute Shrinkage and Selection Operator):

$$\hat{\beta}^{\text{lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left(\|y - X\beta\|_2^2 + \lambda \sum_{i=1}^{p} |\beta_i| \right).$$

- Introduced in 1996 by Robert Tibshirani.
- Equivalent to $\hat{\beta}^{\text{lasso}} = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2$ subject to $\|\beta\|_1 = \sum_{i=1}^p |\beta_i| \leq t$.
- Both sets coefficients to zero (model selection) and shrinks coefficients.
- More "global" approach to selecting variables compared to previously discussed greedy approaches.
- Can be seen as a convex relaxation of the $\hat{\beta}^0$ problem.
- No closed form solution, but can solved efficiently using convex optimization methods.
- Performs well in practice.
- Very popular. Active area of research.

Important model selection property

$\hat{\beta}^{\text{lasso}} = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2$
subject to $\|\beta\|_1 = \sum_{i=1}^p |\beta_i| \leq t$

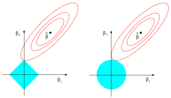


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions $|\beta_1| + |\beta_2| \leq t$ and $\beta_1^2 + \beta_2^2 \leq t^2$, respectively, while the red ellipses are the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

ESL, Fig. 3.11.

Solutions are the intersection of the ellipses with the $\|\cdot\|_1$ or $\|\cdot\|_2$ balls. Corners of the $\|\cdot\|_1$ have zero coefficients.

We will explore the Lasso (computation, properties, etc.) in the next lecture.

Python

Scikit-learn has an object to compute Lasso solution.

Note: the package solves a slightly different (but equivalent) problem than discussed above:

$$\underset{w \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2n} \|y - Xw\|_2^2 + \alpha \|w\|_1.$$

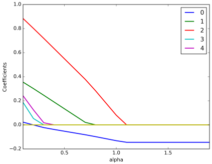
```
from sklearn.linear_model import Lasso

clf = linear_model.Lasso(alpha=0.1)
clf.fit(X,y)
print(clf.coef_)
print(clf.intercept_)
```


Python (cont.)

A simple example with simulated data

```
import numpy as np
from sklearn.linear_model import Lasso
import matplotlib.pyplot as plt

# Generate random data
n = 100
p = 5

X = np.random.randn(n,p)
epsilon = np.random.randn(n,1)
beta = np.random.rand(p)
y = X.dot(beta) + epsilon

alphas = np.arange(0.1,2,0.1)  # 0.1 to 2, step = 0.1
N = len(alphas) # Number of lasso parameters

betas = np.zeros((N,p+1))  # p+1 because of intercept

for i in range(N):
    clf = Lasso(alphas[i])
    clf.fit(X,y)
    betas[i,0] = clf.intercept_
    betas[i,1:] = clf.coef_

plt.plot(alphas,betas,linewidth=2)
plt.legend(range(p))
plt.xlabel('alpha')
plt.ylabel('Coefficients')
plt.xlim(min(alphas),max(alphas))
plt.show()
```

Python (cont.)



Elastic net

Elastic net (Zou and Hastie, 2005)

$$\hat{\beta}^{\text{e-net}} \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda_2 \|\beta\|_2^2 + \lambda_1 \|\beta\|_1.$$

- Benefits from both ℓ_1 (model selection) and ℓ_2 regularization.
- Downside: Two parameters to choose instead of one (can increase the computational burden quite a lot in large experiments).