STAT3612 Lecture 5
Regularized Linear Models

Dr. Aijun Zhang

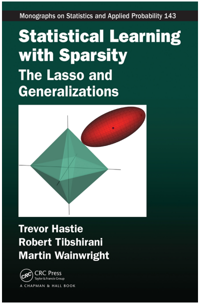
29 September 2020

Department of 統計及精算學系
Statistics & Actuarial Science

Table of Contents

Statistical Learning with Sparsity

- For big data, the number of features can be large, but the number of needed features can be small. **Sparsity** is a natural and common assumption in high-dimensional statistics, as well as for signal recovery (i.e. compressed sensing).
- A sparse model is easier to estimate and interpret.
- **Key question:** how to select the sparse features?
- There exist forward, backward and stepwise algorithms of variable selection for linear models.
- Nowadays, Lasso and Glmnet are arguably the most popular sparse regularization methods, as developed by Stanford groups led by Robert Tibshirani and Trevor Hastie.



Regularized Generalized Linear Models

- Consider the GLM with p features (including $\boldsymbol{x}_1 = \mathbf{1}$ for the intercept)

$$g[\mathbb{E}(Y)] = \boldsymbol{x}^T \boldsymbol{\beta},$$

where $\boldsymbol{x} \in \mathbb{R}^p$ can be the engineered features (e.g. expanded bases).

- When p is large, the model becomes complex with adverse effects:
  a) Parameter estimation can be unstable due to ill-posed $\boldsymbol{X}'\boldsymbol{X}$;
  b) Model interpretation can be difficult due to feature collinearity.

- In this lecture, we study three kinds of regularization methods:
  1) **Ridge regression** that controls the ℓ_2-norm of $\boldsymbol{\beta}$;
  2) **Lasso and Glmnet** that controls the ℓ_1-norm of $\boldsymbol{\beta}$ primarily;
  3) **Best subset selection** that controls the ℓ_0-norm of $\boldsymbol{\beta}$ (cardinality).

Table of Contents

Ridge Regression

- Consder the linear model $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$. The ridge regression is based on the objective with an additional ℓ_2-penalty term

$$\min_{\boldsymbol{\beta}} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_{\ell_2}^2$$

where $\lambda \geq 0$ is a tuning parameter to be determined separately.

- The closed-form ridge estimator through regularized least squares:

$$\hat{\boldsymbol{\beta}}_\lambda = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

- For training $\boldsymbol{X}$, its linear prediction is $\hat{\boldsymbol{y}} = \mathcal{S}_\lambda \boldsymbol{y}$ with the hat matrix:

$$\mathcal{S}_\lambda = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T$$

Ridge Regression: Bias-Variance Tradeoff

- As $\lambda \to 0$, $\hat{\boldsymbol{\beta}}_\lambda^{\text{Ridge}} \to \hat{\boldsymbol{\beta}}^{\text{OLS}}$ (ordinary least squares estimator)
- As $\lambda \to \infty$, $\hat{\boldsymbol{\beta}}_\lambda^{\text{Ridge}} \to \mathbf{0}$ (zero variance)
- The bias and variance of the ridge estimator:

$$\begin{aligned}
\text{Bias}(\hat{\boldsymbol{\beta}}_\lambda) &= \mathbb{E}[\hat{\boldsymbol{\beta}}_\lambda] - \boldsymbol{\beta} = -\lambda(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{\beta} \\
\text{Cov}(\hat{\boldsymbol{\beta}}_\lambda) &= \sigma^2(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}
\end{aligned}$$

- As λ increases, $\text{Bias}(\hat{\boldsymbol{\beta}}_\lambda)$ increases, while $\text{Cov}(\hat{\boldsymbol{\beta}}_\lambda)$ decreases.

Ridge Regression: Bias-Variance Tradeoff

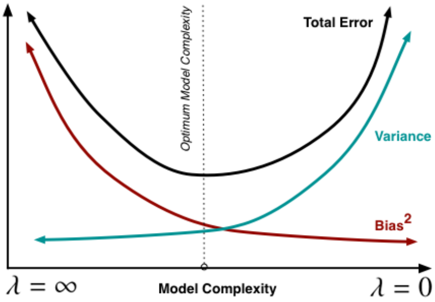
The expected squared error for a new prediction $\hat{f}_\lambda(z)$ can be decomposed:

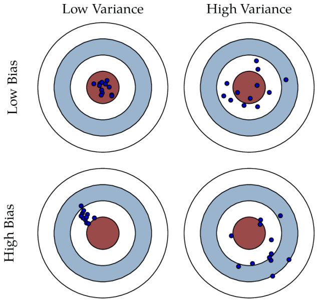
$$
\begin{aligned}
\mathrm{Err}(z) &= \mathbb{E}\left[y - \hat{f}_\lambda(z)\right]^2 = \mathbb{E}\Big[\varepsilon + f(z) - \mathbb{E}[\hat{f}_\lambda(z)] + \mathbb{E}[\hat{f}_\lambda(z)] - \hat{f}_\lambda(z)\Big]^2 \\
&= \sigma^2 + \Big[\mathbb{E}[\hat{f}_\lambda(z)] - f(z)\Big]^2 + \mathbb{E}\Big[\hat{f}_\lambda(z) - \mathbb{E}[\hat{f}_\lambda(z)]\Big]^2 \\
&= \sigma^2 + \left[\mathrm{Bias}(\hat{f}_\lambda(z))\right]^2 + \mathrm{Var}\big(\hat{f}_\lambda(z)\big)
\end{aligned}
$$

For the ridge estimator $\hat{f}_\lambda(z) = z^T\hat{\boldsymbol{\beta}}_\lambda$, the total prediction error is

$$
\mathrm{Err}(z) = \sigma^2 + \left[z^T\mathrm{Bias}(\hat{\boldsymbol{\beta}}_\lambda)\right]^2 + z^T\mathrm{Cov}(\hat{\boldsymbol{\beta}}_\lambda)z.
$$

Ridge Regression: Bias-Variance Tradeoff





Source: Internet

Table of Contents

Lasso Regression

- Lasso: least absolute shrinkage and selection operator (Tibshirani, 1996)
- The lasso method is very similar to the ridge regression with change from ℓ_2-penalty to ℓ_1-penalty:

$$\min_{\boldsymbol{\beta}} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_{\ell_1}$$

- Alternatively, it can be formulated as a constrained optimization problem:

$$\min_{\boldsymbol{\beta}} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^T (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \quad \text{subject to } \|\boldsymbol{\beta}\|_{\ell_1} \leq t$$

- Lasso turns out to enjoy the magic of automatic variable selection, due to the sparsity-inducing ℓ_1-norm constraint.

Lasso vs. Ridge Constraints

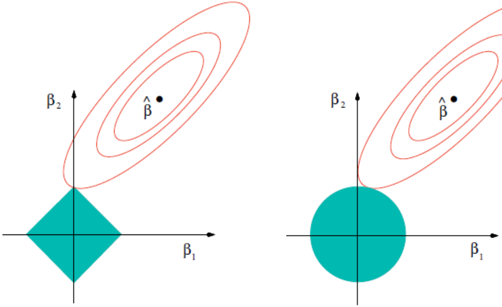
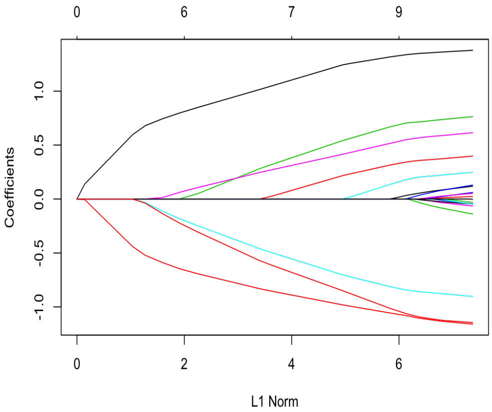


Figure 2.2 *Estimation picture for the lasso (left) and ridge regression (right). The solid blue areas are the constraint regions $|\beta_1|+|\beta_2| \leq t$ and $\beta_1^2+\beta_2^2 \leq t^2$, respectively, while the red ellipses are the contours of the residual-sum-of-squares function. The point $\widehat{\beta}$ depicts the usual (unconstrained) least-squares estimate.*

Source: Hastie, Tibshirani and Wainwright (2015)

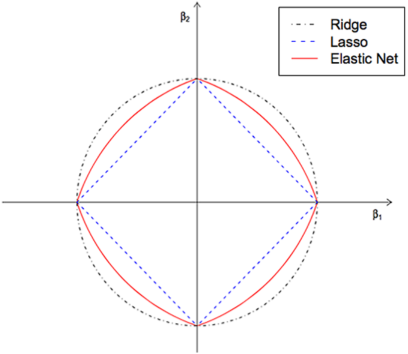
Lasso Solution Paths



Elastic Net

- The elastic net is a composite of Lasso and Ridge penalties for $\alpha \in [0,1]$:

$$\min_{\boldsymbol{\beta}}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})+\lambda\left(\alpha\|\boldsymbol{\beta}\|_{\ell_1}+(1-\alpha)\|\boldsymbol{\beta}\|_{\ell_2}^2/2\right)$$



Glmnet Package

- A practically useful and efficient package developed by Hastie's group;
- It works for the GLMs with the convex empirical loss $\sum_{i=1}^{n} L(y_i, \boldsymbol{x}_i^T \boldsymbol{\beta})$;
- It computes the regularization solution path very fast;
- It includes the cross-validation method for hyperparameter selection.
- Glmnet for R: https://web.stanford.edu/~hastie/glmnet/glmnet_alpha.html
- Python sklearn: https://scikit-learn.org/stable/modules/linear_model.html

Table of Contents

Subset Selection Methods

- Traditional "one-at-a-time" methods: forward, backward and stepwise algorithms for variable selection. See Stepwise regression in Wikipedia.
- Best subset selection by R:leaps() that performs the exhaustive search for all the subsets, using an efficient branch-and-bound algorithm. It selects the best subset model based on the following criteria:

$$C_p = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \frac{2}{n}p\hat{\sigma}_\varepsilon^2;$$

$$\text{AIC} = -2\log L + 2p;$$

$$\text{BIC} = -2\log L + p\log n$$

Best Subset Selection

- A new ℓ_0-constrained best subset selection method by Wen, et al. (2020)

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n} L(y_i, \boldsymbol{x}_i^T\boldsymbol{\beta}), \quad \text{s.t.} \quad \|\boldsymbol{\beta}\|_{\ell_0} = k$$

- It can be solved by a highly efficient primal-dual active set algorithm.
- R:BeSS Package: https://cran.r-project.org/package=BeSS

Thank You!

Q&A or Email ajzhang@hku.hk.