Topic VIII: Model Selection and Regularization

Wei You

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Fall, 2023

Model Selection and the Variance-Bias Trade-Off

- In previous lectures, we are given a set of data and our goal is to find a model that fits the data well. (LRT, F statistic, R^2, nested model tests, etc.)
- This set of data is usually called the training set, and the error of the model is called the training error.
- In statistical learning, the more important task is to predict the outcome under a future scenario beyond the training set.
- To assess a model, people use a test set that is independent of the training dataset, but that follows the same probability distribution as the training dataset. The error over the test set is called testing error.

Model Selection and the Variance-Bias Trade-Off

Example: Regression models.

$$Y = f(\boldsymbol{X}) + \varepsilon$$

where ε i.i.d. with $\mathbb{E}[\varepsilon] = 0$ and $\mathrm{Var}(\varepsilon) = \sigma_\varepsilon^2$.

- From the data $(\boldsymbol{Y}, \boldsymbol{X})$, we obtain certain regression fit $\widehat{f}$.
- For a future input point $\boldsymbol{x}_0$, the squared-error loss is

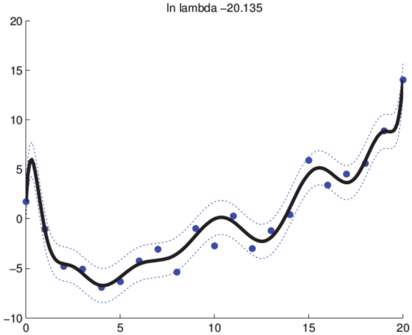
$$\mathsf{Err}(\boldsymbol{x}_0) = \mathbb{E}[(Y_0 - \widehat{f}(\boldsymbol{x}_0))^2] = \sigma_\varepsilon^2 + \mathsf{Bias}^2(\widehat{f}(\boldsymbol{x}_0)) + \mathrm{Var}(\widehat{f}(\boldsymbol{x}_0))$$

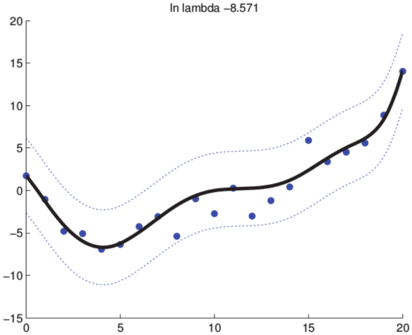
The bias-variance decomposition!

For linear regression model, OLS regression fit is unbiased. We shall see a new ridge regression that is biased, but will have smaller variance.

Model Selection and the Variance-Bias Trade-Off

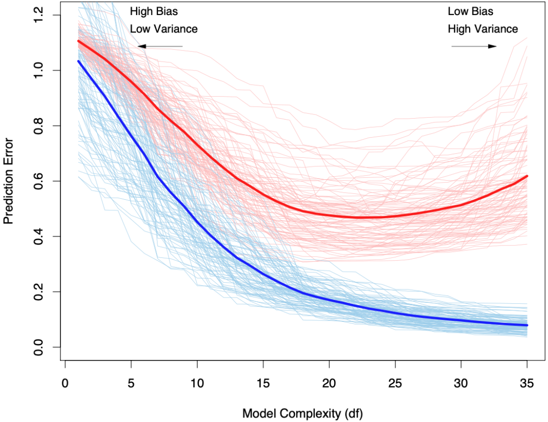
- Typically we would like to choose our model complexity to trade bias off with variance in such a way as to minimize the test error.





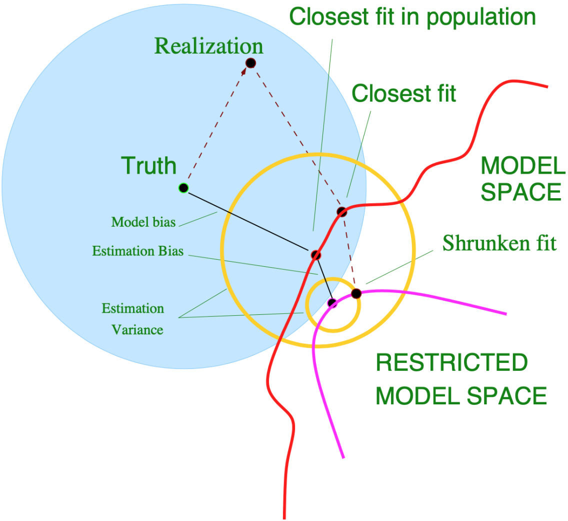
Model Selection and the Variance-Bias Trade-Off

- Training error is usually smaller than test error, and it does not properly account for model complexity. (overfitting)



Choose a model as simple as possible while keeping certain prediction accuracy.

Model Selection and the Variance-Bias Trade-Off



- Not all existing input features are important for prediction.
- Keeping redundant inputs in model can lead to poor prediction and poor interpretation.
- We consider two ways of variable/model selection:
  - Subset selection.
  - Shrinkage/regularization methods.

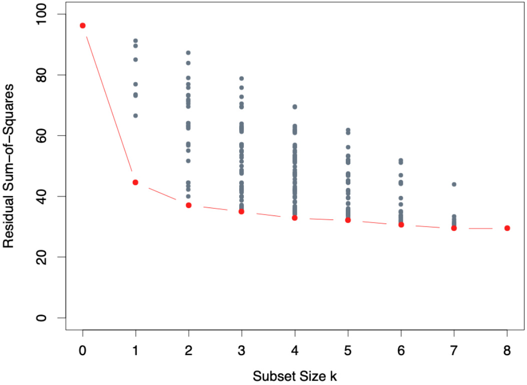
Subset Selection for Linear Models

For regression model with linear predictor $\eta_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}$.

- Nested model tests can help us to determine whether a particular simpler model is good enough.
- We have a lot of independent variables, which variables should we include in the model?

Best Subset

- A natural idea: for each $k \leq p$, find the subset of $\{x_j\}_{j=1}^p$ with size k that gives the smallest RSS!



Best Subset

How to choose the model complexity?

- Typically we choose the smallest model that minimizes an estimate of the expected prediction error.
- Cross-validation, Mellow's C_p, AIC, BIC. (More later.)

How to find the best size k subset?

- There are in total 2^{p-1} possible models. It can be very time consuming!
- The best subset selection problem is nonconvex and is known to be NP-hard.
- Efficient algorithm makes this problem feasible for p as large as 30-40.

Stepwise Method

Stepwise method: greedy algorithm that produces nested models.

- **Forward**: starting from $Y = \beta_0 + \varepsilon$. In each step, add the (one) variable x_i that most significantly improve the fit:
    - the largest F-statistic, or
    - the smallest deviance, or
    - the smallest LRT statistic.
- **Backward**: starting from the saturated model. In each step, remove the (one) variable that is least significant:
    - the smallest F-statistic, or
    - the largest deviance, or
    - the largest LRT statistic.
- Stopping rule:
    - Forward: Stop until all remaining variables are not significant.
    - Backward: Stop until all remaining variables in the model are significant.
    - Alternatively, use Cross-validation, Mellow's C_p, AIC, BIC.

Stepwise Method Cont.

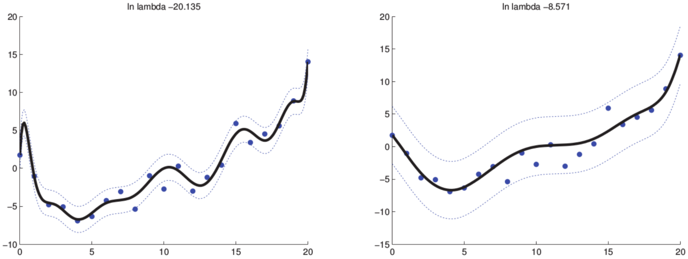
Discussion

- Ease of computation.
- The stopping rule doesn't necessarily give us the best model, because it is a greedy method.
- Lower variance but perhaps more bias.
- Forward and backward can be combined, using two significant levels and go back and forth.
- The order of removing or including variables doesn't imply the rank of importance.
- Backward and forward may give very different solutions.

Introduction – Shrinkage Methods

- MLE picks the parameter values that are the best for fitting the **training data**.
- For this reason, MLE can result in **overfitting**.
- If the data is noisy and we are using lots of parameters, we usually end up with complex functions.

Example: Overfitting.



- Fitting $n = 21$ data points using a degree $p = 14$ polynomial.
- MLE $6.560, -36.934, -109.255, 543.452, 1022.561, -3046.224, -3768.013, 8524.540, 6607.897, -12640.058, -5530.188, 9479.730, 1774.639, -2821.526$
- Many large values in the MLE.
- Small changes in data will result in huge change in the estimation (high variance).

Introduction – Shrinkage Methods

- To address overfitting, we usually encourage the parameters to be **small**.
- Methods that are developed to achieve this goal are usually called **shrinkage methods**.

Ridge Regression

Ridge regression shrinks the estimators by imposing penalization for large values.

$$\widehat{\boldsymbol{\beta}}^{\text{ridge}} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2 \right\}.$$

- l_2 norm as penalization on the $\boldsymbol{\beta}$.
- $\lambda > 0$ is a complexity parameter. The large the value of λ, the greater the amount of shrinkage.

Ridge Solution

The ridge regression is solved by

$$\widehat{\beta}^{\text{ridge}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{Y}.$$

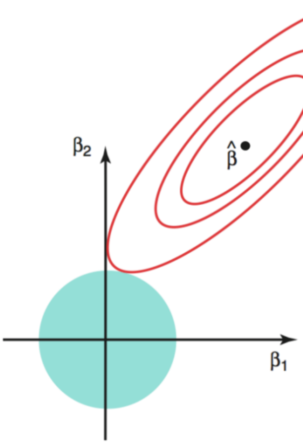
- Compare with OLS, ridge regression adds a positive constant to the diagonal of $\boldsymbol{X}^\top \boldsymbol{X}$ before inversion.
- **Ridge penalty handles collinearity**: It makes the matrix nonsingular, even if $\boldsymbol{X}^\top \boldsymbol{X}$ is not of full rank (multicollinearity).
- This is the main motivation for ridge regression when it was first introduced.

Alternative Derivation

Option 1: Ridge regression can be obtained, using Lagrange multiplier method, by

$$\widehat{\boldsymbol{\beta}}^{\text{ridge}} = \arg\min_{\boldsymbol{\beta}} \quad \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2$$

$$\text{such that} \quad \|\boldsymbol{\beta}\|_2^2 \leq t$$

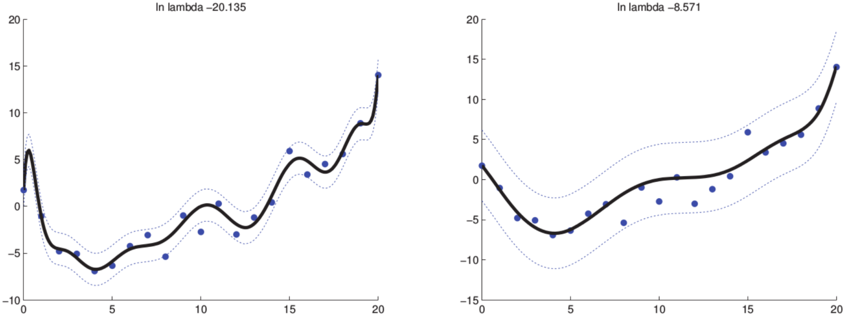


Alternative Derivation

Option 2: Ridge regression can also be obtained by a Bayesian process.

- Consider $N(0, \tau^2 \boldsymbol{I})$ as the prior distribution for $\boldsymbol{\beta}$.
- Let the likelihood of Y_i given $\boldsymbol{x}_i$ and $\boldsymbol{\beta}$ be $N(\boldsymbol{x}_i^\top \boldsymbol{\beta}, \sigma^2)$.
- Connection between λ and τ is $\lambda = \sigma^2 / \tau^2$.
- One can check that $\widehat{\boldsymbol{\beta}}^{\text{ridge}}$ maximizes the posterior density, and so it is the mode of the posterior density. It is also the posterior mean due to symmetry of normal dist.
- Such estimation is call a **maximum a posteriori (MAP) estimator**.

Ridge Regression – Bias-Variance Trade-off



- For $\lambda = 0.001$, the ridge estimation is $2.128, 0.807, 16.457, 3.704, -24.948, -10.472, -2.625, 4.360, 13.711, 10.063, 8.716, 3.966, -9.349, -9.232.$

The above observation is confirmed by the following

Bias-variance trade-off

$$\begin{aligned}\mathsf{Var}(\widehat{\boldsymbol{\beta}}_{\lambda}^{\mathsf{ridge}}) &= (\boldsymbol{X}^{\top}\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\boldsymbol{X}^{\top}\boldsymbol{X}(\boldsymbol{X}^{\top}\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\sigma^{2},\\ \mathsf{Bias}(\widehat{\boldsymbol{\beta}}_{\lambda}^{\mathsf{ridge}}) &= -\lambda(\boldsymbol{X}^{\top}\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\boldsymbol{\beta}.\end{aligned}$$

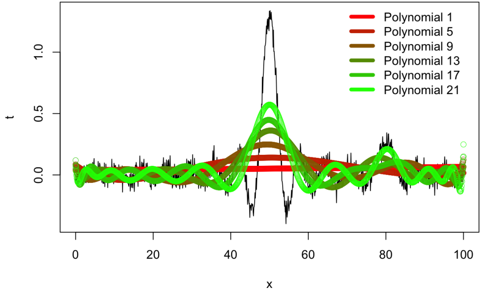
One can show that

- $\mathsf{Var}(\widehat{\boldsymbol{\beta}}_{\lambda}^{\mathsf{ridge}}) \preccurlyeq (\boldsymbol{X}^{\top}\boldsymbol{X})^{-1}\sigma^{2}$ for all $\lambda > 0$.
- $\frac{d\mathsf{MSE}(\widehat{\boldsymbol{\beta}}_{\lambda}^{\mathsf{ridge}})}{d\lambda} < 0$, so for sufficiently small λ, ridge always improve OLS.

Kernel Ridge Regression – Motivation

Example: Consider the true model

$$f(x) = 1.1\frac{\sin(x-50)}{x-50} + 0.3\frac{\sin(x-80)}{x-80}.$$



[https://rpubs.com/Saulabrm/210788]

- The true model has various degrees of smoothness, hence polynomial regression is not economic.
- We may use spline regressions, but need to work out the knots.
- Is there a method that does this automatically?

Kernel Ridge Regression

We now introduce a nonparametric generalization to ridge regression.

- It generalizes the idea of enlarging the feature space.
- In particular, it encludes the penalized polynomial regression as a special case.

Recall the ridge solution

$$\widehat{\boldsymbol{\beta}}^{\text{ridge}} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} X^\top \boldsymbol{Y}.$$

$$\widehat{\boldsymbol{\beta}}^{\text{ridge}} = \boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{X}^\top + \lambda \boldsymbol{I})^{-1} \boldsymbol{Y}, \quad \widehat{\boldsymbol{Y}}^{\text{ridge}} = \boldsymbol{X}\boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{X}^\top + \lambda \boldsymbol{I})^{-1} \boldsymbol{Y}.$$

$$\widehat{Y}^{\text{ridge}}(\boldsymbol{x}) = \boldsymbol{x}\boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{X}^\top + \lambda \boldsymbol{I})^{-1} \boldsymbol{Y}.$$

Proof. Because

$$\boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{X}^\top + \lambda \boldsymbol{I}) = \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{X}^\top + \lambda \boldsymbol{X}^\top = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}) \boldsymbol{X}^\top$$

Note that $\boldsymbol{X}\boldsymbol{X}^\top$ appears in our new expression of the ridge solution.

- The ij-th entry of $\boldsymbol{X}\boldsymbol{X}^\top \in \mathbb{R}^{n \times n}$ is $\boldsymbol{x}_i^\top \boldsymbol{x}_j = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$, the inner product between two feature vectors.
- The predicted values depends only on $\boldsymbol{x}^\top X^\top \in \mathbb{R}^n$, and the j-th entry is $\boldsymbol{x}^\top \boldsymbol{x}_j = \langle \boldsymbol{x}, \boldsymbol{x}_j \rangle$, the inner product between the new observation and the i-th training data.

Prediction by ridge regression boils down to computing the inner product between the feature vectors.

- This is the foundation of the so-called “kernel trick”.

Kernel Trick

Suppose we use another “inner product” to replace the usual one.

- Let’s replace $\langle \cdot, \cdot \rangle$ by similarity measure $K(\cdot, \cdot)$, called a <u>kernel</u>.
- A symmetric function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a positive definite kernel on $\mathcal{X}$ if

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$$

holds for any $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$ and $c_1, \ldots, c_n \in \mathbb{R}$.

Kernel Trick

- For ridge regression, we have a linear kernel

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \langle \boldsymbol{x}_1, \boldsymbol{x}_2 \rangle = \boldsymbol{x}_1^\top \boldsymbol{x}_2.$$

- Then

$$\boldsymbol{x}\boldsymbol{X}^\top \to (K(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, K(\boldsymbol{x}, \boldsymbol{x}_n)),$$

$$\boldsymbol{X}\boldsymbol{X}^\top \to \boldsymbol{K} = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{1 \le i,j \le n}.$$

- The fitted vector is now

$$\widehat{\boldsymbol{Y}} = \boldsymbol{K}(\boldsymbol{K} + \lambda \boldsymbol{I})^{-1}\boldsymbol{Y}.$$

  And the prediction for a new observation $\boldsymbol{x}$ is

$$\widehat{y} = (K(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, K(\boldsymbol{x}, \boldsymbol{x}_n))(\boldsymbol{K} + \lambda \boldsymbol{I})^{-1}\boldsymbol{Y}.$$

This is the so-called **kernel ridge regression**.

Example: Commonly used kernels

- Linear kernel:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle = \boldsymbol{x}_i^\top \boldsymbol{x}_j.$$

- Polynomial kernel of degree up to d:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i^\top \boldsymbol{x}_j)^d.$$

  - This is equivalent to enlarging the feature space to include all polynomials with degree up to d, hence polynomial regression of degree d.

- Polynomial kernel of degree exactly d:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j)^d.$$

- Gaussian (radial basis function) kernel:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2).$$

  - **Local behavior**: The kernel decreases exponentially fast in the distance of two feature vectors. Points faraway play have little effect in regression.
  - The corresponding feasture space is implicit and infinite-dimensional.

Remark

In stead of the linear model, we now consider

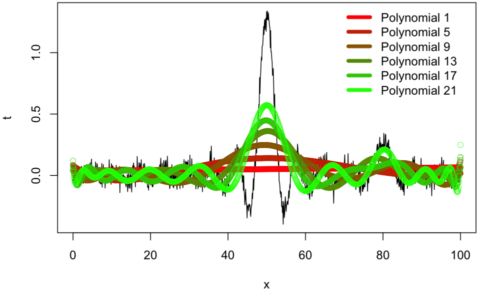
$$y = \sum_{j=1}^{n} \alpha_j K(\boldsymbol{x}, \boldsymbol{X}_j) + \varepsilon.$$

- The kernel ridge regression approximates the multivariate regression by using the kernel functions $\{K(\cdot, \boldsymbol{X}_j)\}_{j=1}^{n}$.
- For polynomial kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i^\top \boldsymbol{x}_j)^d$, this is equivalent to the polynomial regression.

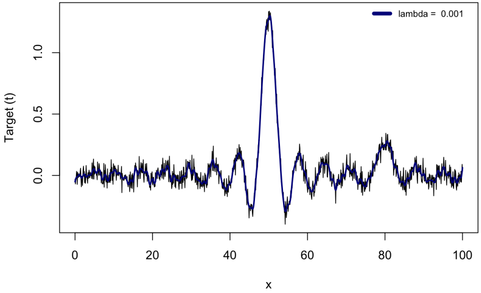
Example: Gaussian kernel versus polynomial kernal.

Consider the true model

$$f(x) = 1.1\frac{\sin(x-50)}{x-50} + 0.3\frac{\sin(x-80)}{x-80}.$$



Gaussian Kernel with sigma = 1



[https://rpubs.com/Saulabrm/210788]

Lasso Regression

Another very important shrinkage method is **least absolute selection and shrinkage operator (LASSO)**.

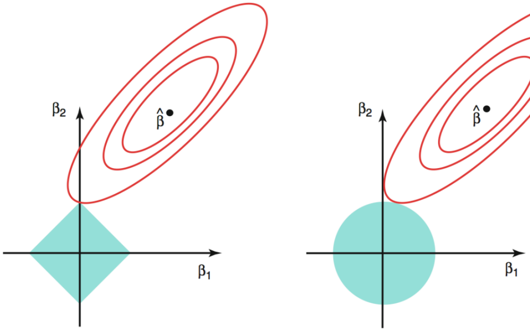
- The key difference between lasso and ridge regressions is that lasso uses l_1-norm (sum of absolute values) instead of l_2-norm (sum of squares).

Comparing Lasso with Ridge Regressions

$$\widehat{\boldsymbol{\beta}}^{\text{lasso}} = \arg\min_{\boldsymbol{\beta}} \quad \|\boldsymbol{Y} - (\beta_0 + X\boldsymbol{\beta})\|_2^2 \qquad \widehat{\boldsymbol{\beta}}^{\text{ridge}} = \arg\min_{\boldsymbol{\beta}} \quad \|\boldsymbol{Y} - (\beta_0 + X\boldsymbol{\beta})\|_2^2$$

$$\text{such that} \quad \|\boldsymbol{\beta}\|_1 \le t \qquad\qquad \text{such that} \quad \|\boldsymbol{\beta}\|_2^2 \le t$$

- There is a one-to-one correspondence between λ and t.

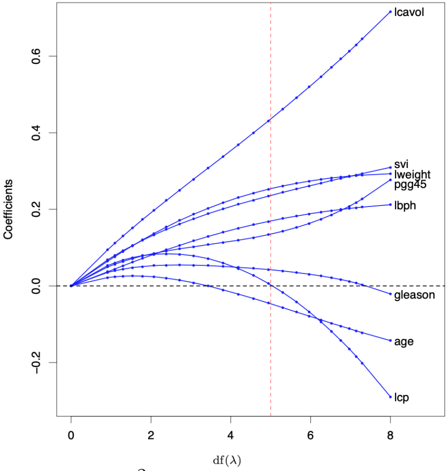


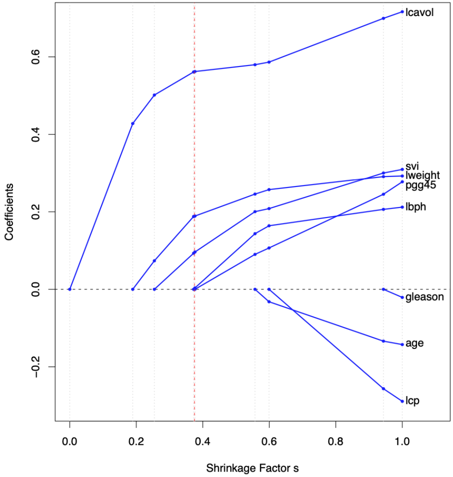
Something in Between

- Lasso results in a "sparse" solution, so that some of the estimators are exactly 0. This is a type of **model/variable selection**.
- Lasso estimates do not need to be unique if covariates are collinear. Hence it can be unstable for high dimensional data.
- Ridge Regression handles multicollinearity.
- Elastic net (Zou and Hastie, 2005) regularization uses convex combination of both:

$$\arg\min_{\boldsymbol{\beta}} \left\{ \frac{1}{n} \|\boldsymbol{Y} - X\boldsymbol{\beta}\|_2^2 + \lambda_2 \|\boldsymbol{\beta}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 \right\}$$

Discussion: Subset Selection, Ridge Regression and the Lasso





$\mathrm{df}(\lambda) = \sum_{j=1}^{p} \frac{d_j^2}{d_j^2+\lambda}$, where d_j are the singular values of X. $s = t/\sum_{j=1}^{p} |\widehat{\beta}_j|$.

Recall that

$$\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\boldsymbol{Y} - X^\top \boldsymbol{\beta}\|_2^2 + \lambda \|\beta_k\|_p^p \right\}$$

- Ridge: $p = 2$.
- Lasso: $p = 1$.
- Best subset: $p = 0$. (asumming $0^0 = 0$)

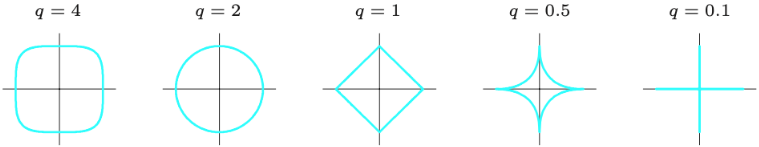
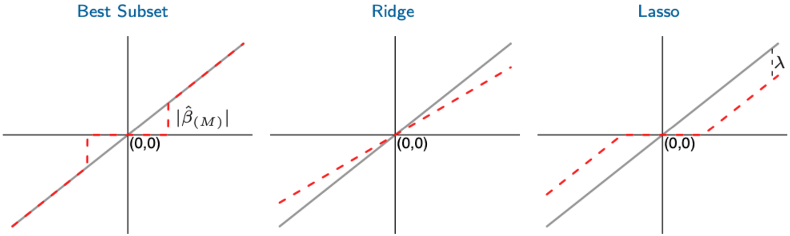


FIGURE 3.12. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for given values of* q.

Assuming that $\{x_j\}$ are orthogonal, the three procedures have explicit solution:

Estimator	Formula
Best subset (size M)	$\hat{\beta}_j \cdot I(\lvert\hat{\beta}_j\rvert \geq \lvert\hat{\beta}_{(M)}\rvert)$
Ridge	$\hat{\beta}_j/(1+\lambda)$
Lasso	$\mathrm{sign}(\hat{\beta}_j)(\lvert\hat{\beta}_j\rvert - \lambda)_+$



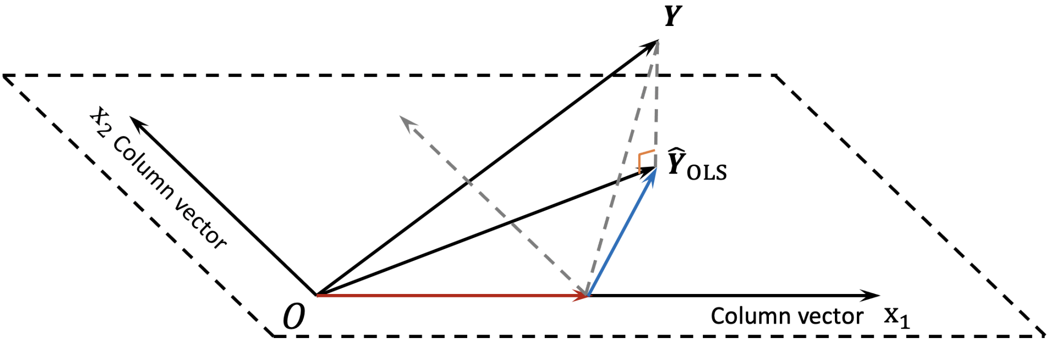
Here $\widehat{\beta}$ is the OSL estimator.

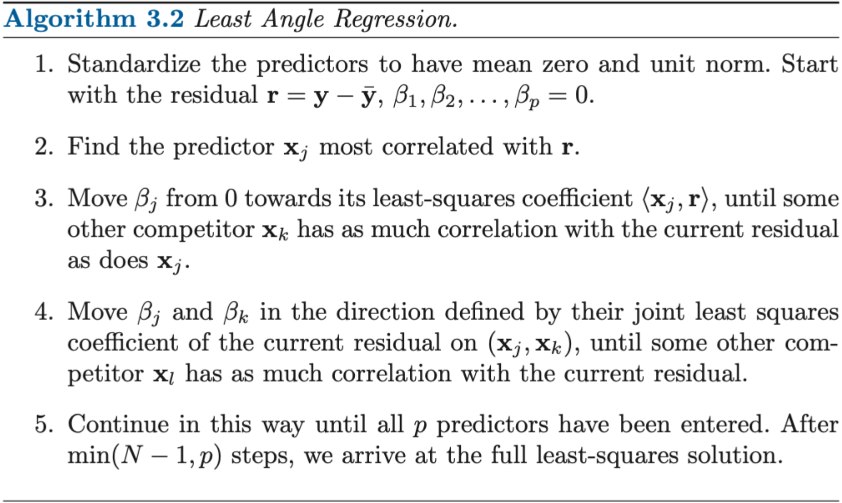
Least Angle Regression

We have not talked about how to compute the lasso solution.

- The lasso problem is formulated as a quadratic programming (QP) problem. So usual QP solvers can be applied.
- Alternatively, we now discuss an extremely efficient (and intuitive) algorithm to solve for the entire lasso path, i.e., all solutions with respect to λ.
- Efron et al. (2004), Lease angle regression.

Least Angle Regression





Algorithm 3.2 *Least Angle Regression.*

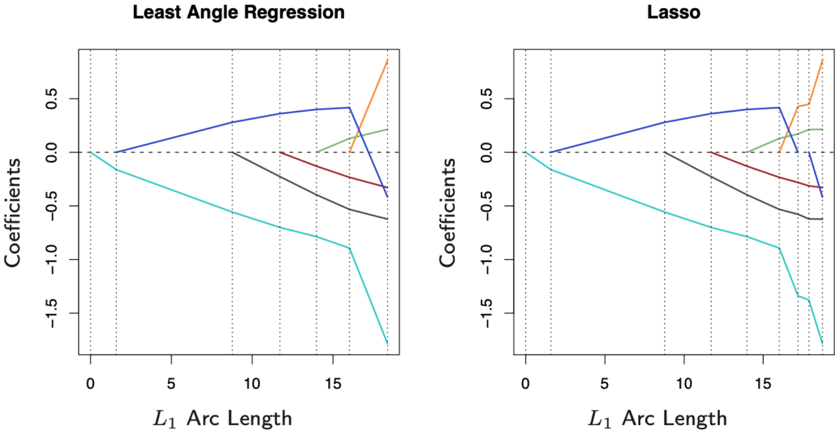
1. Standardize the predictors to have mean zero and unit norm. Start with the residual $\mathbf{r} = \mathbf{y} - \bar{\mathbf{y}}$, $\beta_1, \beta_2, \dots, \beta_p = 0$.

2. Find the predictor $\mathbf{x}_j$ most correlated with $\mathbf{r}$.

3. Move β_j from 0 towards its least-squares coefficient $\langle \mathbf{x}_j, \mathbf{r} \rangle$, until some other competitor $\mathbf{x}_k$ has as much correlation with the current residual as does $\mathbf{x}_j$.

4. Move β_j and β_k in the direction defined by their joint least squares coefficient of the current residual on $(\mathbf{x}_j, \mathbf{x}_k)$, until some other competitor $\mathbf{x}_l$ has as much correlation with the current residual.

5. Continue in this way until all p predictors have been entered. After $\min(N-1, p)$ steps, we arrive at the full least-squares solution.



Algorithm 3.2a *Least Angle Regression: Lasso Modification.*

4a. If a non-zero coefficient hits zero, drop its variable from the active set of variables and recompute the current joint least squares direction.

Model Selection Criteria

For a given data, and some candidate models, model selection criterion gives each model a score, and we can pick the model with highest score.

- Usually the criterion rewards good fit, but penalizes complexity.
- Here we look at
  - Adjusted R^2: adjust R^2, taking into account of the degrees of freedom.
  - Mallow's C_p: based on the MSE of the estimations.
  - Akaike Information Criterion (AIC): based on log-likelihood.
  - Bayesian Information Criterion (BIC): based on bayesian statistics.
- The general idea is to "estimate" the testing error by making adjustments of the training error.

Adjusted R^2

Recall the coefficient-of-determination R^2 from the linear model

$$R^2 = 1 - \frac{RSS}{SST}.$$

- R^2 reflects the training error.
- A model with larger R-squared is not necessarily better than another model with smaller R-squared when we consider test error!

Instead of directly maximizing R^2, we will maximize a penalized version of R^2.

Adjusted R-squared

The adjusted R-squared, taking into account of the degrees of freedom, is defined as

$$\text{adjusted } R^2 = 1 - \frac{RSS/(n-p-1)}{SST/(n-1)}.$$

- With more inputs, the R^2 always increase, but the adjusted R^2 could decrease since more inputs is penalized by the smaller degree of freedom of the residuals.
- Maximizing adjusted R^2 is equivalent to

$$\min\{RSS/(n-p-1)\}.$$

Mallow's C_p

Consider a regression model $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, so that $\mathbb{E}[Y_i] = \boldsymbol{x}_i^\top \boldsymbol{\beta}$ and $\mathrm{Var}(Y_i) = \sigma^2$.

Recall that our fitting minimizes RSS, but a more sensible prediction characterization of error is $\mathsf{MSE}(\widehat{Y}_i) = \mathbb{E}[(\widehat{Y}_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2]$. We have

$$(\widehat{Y}_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2 = (Y_i - \widehat{Y}_i)^2 - (Y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2 + 2(\widehat{Y}_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})(Y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})$$

Taking expectation, we have the MSE of $\widehat{Y}_i$ as estimator of the true mean $\boldsymbol{x}_i^\top \boldsymbol{\beta}$

$$\mathsf{MSE}(\widehat{Y}_i) = \mathbb{E}\left[(\widehat{Y}_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2\right] = \mathbb{E}\left[(Y_i - \widehat{Y}_i)^2\right] - \sigma^2 + 2\mathrm{Cov}(\widehat{Y}_i, Y_i)$$

Summing over i and divided by σ^2, we have

$$\sum_{i=1}^n \frac{\mathsf{MSE}(\widehat{Y}_i)}{\sigma^2} = \mathbb{E}\left[\frac{\|\widehat{\boldsymbol{Y}} - X\boldsymbol{\beta}\|_2^2}{\sigma^2}\right] = \mathbb{E}\left[\frac{\|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}\|_2^2}{\sigma^2}\right] - n + 2\sum_{i=1}^n \frac{\mathrm{Cov}(\widehat{Y}_i, Y_i)}{\sigma^2}$$

Mallow's C_p

The last term is defined as the degrees of freedom for the estimator $\widehat{\boldsymbol{Y}}$

$$\text{df}(\widehat{\boldsymbol{Y}}) = \sum_{i=1}^{n} \frac{\text{Cov}(\widehat{Y}_i, Y_i)}{\sigma^2}.$$

Now we consider multiple linear regression:

- $\text{df}(\widehat{\boldsymbol{Y}}) = p$, the number of parameters in the model. $(\text{Cov}(\widehat{Y}_i, Y_i) = \text{Cov}(e_i^\top PY, e_i^\top Y) = P_{ii}\sigma^2)$
- Mallow's C_p is an estimation of the (scaled) sum of MSE.
- For the model with p number of parameters, C_p is given by

$$C_p = \frac{\|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}\|_2^2}{S^2} - n + 2\text{df}(\widehat{\boldsymbol{Y}}) = \frac{\text{RSS(p)}}{S^2} - n + 2p$$

where S^2 is the variance estimated from the full model.

Mallow's C_p

Now in the context of model selection,

- Let $\boldsymbol{X}$ collect all possible explanatory variables. (Recall that each column correspond to one explanatory variable.)
- Let $\boldsymbol{X}_p$ be the p columns of explanatory variables that is used in the pth model.
- Assume that the true model is described by some $\boldsymbol{X}_{p0}$.
- For the pth model, recall that (projection P_p to the column space of $\boldsymbol{X}_p$) the residual is $\boldsymbol{\varepsilon}_p = (\boldsymbol{I}_n - P_p)\boldsymbol{Y}$. So

$$\text{RSS}(p) = \boldsymbol{Y}^\top(\boldsymbol{I}_n - P_p)\boldsymbol{Y} = \boldsymbol{Y}^\top(\boldsymbol{I}_n - \boldsymbol{X}_p(\boldsymbol{X}_p^\top\boldsymbol{X}_p)^{-1}\boldsymbol{X}_p^\top)\boldsymbol{Y}.$$

Mallow's C_p

Lemma

Let Σ be the covariance matrix of Y, then

$$\mathbb{E}[\boldsymbol{Y}^\top A\boldsymbol{Y}] = \mathbb{E}[\boldsymbol{Y}^\top]A\mathbb{E}[\boldsymbol{Y}] + \mathit{tr}[\Sigma A]$$

$$\begin{aligned}
\mathbb{E}[\mathsf{RSS}(p)] &= \mathbb{E}[\boldsymbol{Y}^\top(\boldsymbol{I}_n - P_p)\boldsymbol{Y}] \\
&= \boldsymbol{\beta}^\top\boldsymbol{X}^\top(\boldsymbol{I}_n - P_p)\boldsymbol{X}\boldsymbol{\beta} + \mathsf{tr}\left[\mathbf{I}_n - \boldsymbol{X}_p(\boldsymbol{X}_p^\top\boldsymbol{X}_p)^{-1}\boldsymbol{X}_p^\top\right]\sigma^2 \\
&= \boldsymbol{\beta}^\top\boldsymbol{X}^\top(\boldsymbol{I}_n - P_p)^\top(\boldsymbol{I}_n - P_p)\boldsymbol{X}\boldsymbol{\beta} + \sigma^2\left(n - \mathsf{tr}\left[(\boldsymbol{X}_p^\top\boldsymbol{X}_p)(\boldsymbol{X}_p^\top\boldsymbol{X}_p)^{-1}\right]\right) \\
&= \|(\boldsymbol{I}_n - P_p)\boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \sigma^2(n-p)
\end{aligned}$$

Mallow's C_p

So

$$\mathbb{E}[C_p] = \mathbb{E}[\mathsf{RSS}(p)/S^2] - n + 2p \approx \|(I_n - P_p)\boldsymbol{X}\boldsymbol{\beta}\|_2^2/\sigma^2 + p$$

- As p increases, $\|(I_n - P_p)\boldsymbol{X}\boldsymbol{\beta}\|_2^2$ is decreasing when the pth model is simpler than the true model.
- When the pth model is more complicated than the true model, $\|(I_n - P_p)\boldsymbol{X}\boldsymbol{\beta}\|_2^2 = 0$.
- If the pth model is true, then C_p is tend to be close to or smaller than p.
- In practice, we can look at a plot of C_p versus p to decided amongst models.

This justification works only for models fitted by OLS, because of the geometric interpretation.

Mallow's C_p: An Example

Example: Stopping criterion for least angle regression. Recall the least angle regression algorithm. In each step, a new variable enters the model. When should we stop?

One may use C_p-type criterion for stopping.

- Recall the original form of Mallow's C_p

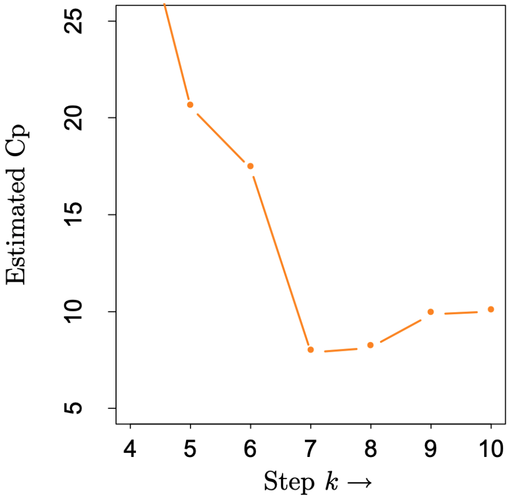
$$C_p = \frac{\|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}\|_2^2}{S^2} - n + 2\text{df}(\widehat{\boldsymbol{Y}})$$

- What is $\text{df}(\widehat{\boldsymbol{Y}})$ in least angle regression?
- Efron et al. (2004) showed that

$$\text{df}(p) \approx p.$$

- Hence, Mallow's C_p can be directly used.

Mallow's C_p: An Example



Mallow's C_p

Remark

- In some text, Mallow's C_p is defined as an estimation of the extra-sample error, i.e.,

$$\sum_{i=1}^{n} \frac{\mathbb{E}[(Y_i' - \widehat{Y}_i)^2]}{\sigma^2},$$

where $Y_i' = x_i^\top \boldsymbol{\beta} + \varepsilon_i'$ is a new data with a i.i.d. new error ε_i'.

- Then

$$\sum_{i=1}^{n} \frac{\mathbb{E}[(Y_i' - \widehat{Y}_i)^2]}{\sigma^2} = \sum_{i=1}^{n} \frac{\mathsf{MSE}(\widehat{Y}_i)}{\sigma^2} + n\sigma^2$$

and Mallow's C_p is alternatively written as

$$C_p = \frac{\|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}\|_2^2}{S^2} + 2\mathsf{df}(\widehat{\boldsymbol{Y}}) = \frac{RSS}{S^2} + 2p.$$

Akaike's Information Criterion (AIC)

Note that Mallow's C_p is only justified in regression models that is fitted by ordinary least square. We now present another criterion that can be used in more general models.

For a set of observation $(\boldsymbol{Y}, \boldsymbol{X})$, consider a family of models with parameter $\boldsymbol{\beta} \in \mathbb{R}^P$.

- Suppose the true (unknown) parameter is $\boldsymbol{\beta}^*$.
- We want to choose a model with parameter $\boldsymbol{\beta}$ that is the "nearest" to $\boldsymbol{\beta}^*$.
- The principle behind AIC is to measure the distance between $\boldsymbol{\beta}$ and $\boldsymbol{\beta}^*$ by the Kullback-Leibler divergence:

$$D_{\mathsf{KL}}(\boldsymbol{\beta} \| \boldsymbol{\beta}^*) = \int \log \left(\frac{f_Y(Y; X, \boldsymbol{\beta}^*)}{f_Y(Y; X, \boldsymbol{\beta})} \right) f_Y(Y; X, \boldsymbol{\beta}^*) \, dY,$$

where $f(Y; X, \boldsymbol{\beta})$ is the likelihood under parameter $\boldsymbol{\beta}$.

Minimizing $D_{\mathsf{KL}}(\boldsymbol{\beta} \| \boldsymbol{\beta}^*)$ is equivalent to maximizing

$$H(\boldsymbol{\beta}) = \int \log f_Y(Y; X, \boldsymbol{\beta}) f_Y(Y; X, \boldsymbol{\beta}^*)\, dY = \mathbb{E}_{\boldsymbol{\beta}^*}[\log f_Y(Y; X, \boldsymbol{\beta})]$$

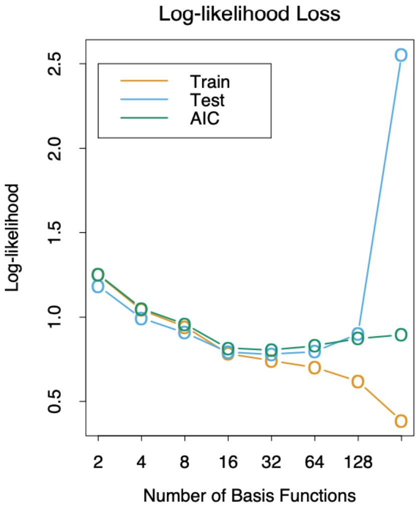
- The maximizer is the unknown $\boldsymbol{\beta}^*$.
- Let $\widehat{\boldsymbol{\beta}}_p$ be the MLE in the pth model.
- It is tempting to use the log-likelihood $l(\widehat{\boldsymbol{\beta}}_p)$ to estimate $H(\boldsymbol{\beta}^*)$.
- However, Akaike found that it is biased and proposed the bias correction

$$\mathbb{E}\left[l(\widehat{\boldsymbol{\beta}}_p) - p\right] \approx H(\boldsymbol{\beta}^*).$$

Akaike's Information Criterion (AIC)

$$\text{AIC} = -2l(\widehat{\boldsymbol{\beta}}_p) + 2p$$

- The factor 2 makes AIC equivalent to Mallow's C_p for multiple linear regression.
- Smaller AIC means smaller KL divergence (better fit!).
- AIC describe the relative (but not absolute) quality of models.
- $(\text{AIC}_{\text{min}} - \text{AIC}_i)/2$ is called the relative likelihood of model i.
- **Compare with LRT:** LRT is valid only for nested models, whereas AIC has no such restriction.



Bayesian Information Criterion (BIC)

Bayesian Information Criterion (BIC)

$$\text{BIC} = -2l(\widehat{\boldsymbol{\beta}}_p) + \log(n)p$$

- Similar to AIC, applicable to model fitted by maximization of log-likelihood.
- Motivated by Bayesian approach to model selection.
- $\exp(-\text{BIC}/2)$ approximates the posterior probability of the current model, so small BIC means large posterior probability.

Compare AIC and BIC

- No clear choice between AIC and BIC.
- 2 v.s. $\log(n)$: more conservative (more penalty to complicated models).
- BIC is asymptotically consistent: as number of observation grows, the probability that BIC will choose the true model approaches 1.
- This is not the case for AIC, which tends to choose models which are too complex.
- For finite samples, BIC often chooses models that are too simple, because of its heavy penalty on complexity.

Cross-Validation

K-fold cross-validation is a general purpose method of prediction/test error estimate, especially in a data rich scenario.

1	2	3	4	5
Train	Train	Validation	Train	Train

- Divide the data into K parts (BEFORE doing anything else).
- For each part, fit the data using data from other parts (training sets), and calculate the prediction error on the current part (validation set).

Our ultimate goal is to produce the best model with best prediction accuracy.

- Suppose in the i-th run result in a test error MSE_i on the validation set.
- Average over the above K estimates of the test errors, and obtain

$$\mathsf{CV}_{(K)} = \frac{1}{K}\sum_{i=1}^{K} \mathsf{MSE}_i$$

We then choose the model with the smallest CV error.

Leave-One-Out Cross-Validation

An interesting special case is the n-fold cross-validation, also known as the leave-one-out cross-validation (LOOCV).

- K-fold cross-validation is more biased. It estimates the performance of a model trained on a dataset of size $n\frac{k-1}{k}$. If we use a model trained on the full data, it will perform slightly better than the cross-validation estimate suggests.
- K-fold can also have higher variance if the sample size is smaller (so the training set is even smaller).
- LOOCV is approximately unbiased (because only one data is left out).
- LOOCV have higher variance because in each iteration you are using essentially the same set of data.

One advantage of using LOOCV is that it is computationally inexpensive for some models as we will see next.

Leave-One-Out Cross-Validation

Leave-one-out CV error

The leave-one-out CV error (under quadratic loss) is defined as

$$\text{CV} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}^{(-i)}(\boldsymbol{X}_i))^2,$$

where $\widehat{Y}^{(-i)}(\boldsymbol{X}_i)$ is the predicted value of $\boldsymbol{X}_i$ computed by using all the data except the i-th observation.

By definition, we need to repeat the fitting process for n times to compute this CV error. However, we can avoid such computation in many popular regression models.

Leave-One-Out CV Error

Linear smoother

A fitting method is called a linear smoother if we can write

$$\widehat{\boldsymbol{Y}} = \boldsymbol{S}\boldsymbol{Y}$$

for any dataset $(X_i, Y_i)_{i=1}^n$ where $\boldsymbol{S}$ is a $n \times n$ matrix that only depends on $\boldsymbol{X}$.

Many regression methods are linear smoothers with different $\boldsymbol{S}$ matrices.

Method	$\boldsymbol{S}$
Multiple linear regression	$\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top$
Ridge regression	$\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^\top$
Kernel ridge regression	$\boldsymbol{K}(\boldsymbol{K} + \lambda\boldsymbol{I})^{-1}$

Assume that a linear smoother is fitted on $\{(X_i, Y_i)\}_{i=1}^n$. Let $\boldsymbol{x}$ be a new covariate vector and $\widehat{f}(\boldsymbol{x})$ be its the predicted value by using the linear smoother. We then augment the dataset by including $(\boldsymbol{x}, \widehat{f}(\boldsymbol{x}))$ and refit the linear smoother on this augmented dataset.

Self-stable

The linear smoother is said to be self-stable if the fit based on the augmented dataset is identical to the fit based on the original data regardless of x.

- Multiple linear regression, ridge regression and kernel ridge regression are all self-stable.

For self-stable linear smoother, leave-one-out cross-validation is particularly appealing because in many cases we have the seemingly magical reduction.

Theorem

$$CV = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{f}^{(-i)}(\boldsymbol{X}_i))^2 = \frac{1}{n}\sum_{i=1}^{n}\frac{\left(Y_i - \widehat{Y}_i\right)^2}{(1 - S_{ii})^2}.$$

Proof. Apply the linear smoother to $\{(X_j, Y_j), j \neq i\}$ to obtain $\widehat{f}^{(-i)}(\boldsymbol{X}_i)$. Apply the linear smoother to $(\boldsymbol{X}, \tilde{\boldsymbol{Y}}) \overset{\text{def}}{=} \{(X_j, Y_j), j \neq i, (\boldsymbol{X}_i, \widehat{f}^{(-i)}(\boldsymbol{X}_i))\}$. By self-stable property,

$$\widehat{f}^{(-i)}(\boldsymbol{X}_i) = (\boldsymbol{S}\tilde{\boldsymbol{Y}})_i = S_{ii}\widehat{f}^{(-i)}(\boldsymbol{X}_i) + \sum_{j\neq i} S_{ij}Y_j.$$

Hence

$$\widehat{f}^{(-i)}(\boldsymbol{X}_i) = \frac{\sum_{j\neq i} S_{ij}Y_j}{1 - S_{ii}}.$$

On the other hand,

$$\widehat{Y}_i = (\boldsymbol{S}\boldsymbol{Y})_i = S_{ii}Y_i + \sum_{j\neq i} S_{ij}Y_j$$

Hence,

$$Y_i - \widehat{f}^{(-i)}(\boldsymbol{X}_i) = Y_i - \frac{\sum_{j\neq i} S_{ij}Y_j}{1-S_{ii}} = \frac{Y_i - (S_{ii}Y_i + \sum_{j\neq i} S_{ij}Y_j)}{1-S_{ii}} = \frac{Y_i - \widehat{Y}_i}{1-S_{ii}}$$

As a result

$$\mathsf{CV} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{f}^{(-i)}(\boldsymbol{X}_i))^2 = \frac{1}{n}\sum_{i=1}^{n}\frac{\left(Y_i - \widehat{Y}_i\right)^2}{(1-S_{ii})^2}.$$

Generalized Cross-Validation

For some smoothers $\mathrm{tr}(\boldsymbol{S})$ can be computed more easily than its diagonal elements. To take advantage of this, suppose that we approximate each diagonal elements of $\boldsymbol{S}$ by their average which equals $\mathrm{tr}(\boldsymbol{S})/n$.

Generalized cross-validation

$$\mathrm{CV} = \frac{1}{n}\sum_{i=1}^{n}\frac{\left(Y_i - \widehat{Y}_i\right)^2}{(1-S_{ii})^2} \approx \frac{1}{n}\frac{\sum_{i=1}^{n}\left(Y_i - \widehat{Y}_i\right)^2}{(1-\mathrm{tr}(\boldsymbol{S})/n)^2} =: \mathrm{GCV}.$$

Parameter Tuning Using GCV

Now we are ready to handle the tuning parameter selection issue in the linear smoother. We write $\boldsymbol{S} = \boldsymbol{S}_\lambda$ and

$$\mathsf{GCV}(\lambda) = \frac{1}{n} \frac{\boldsymbol{Y}^\top (\boldsymbol{I} - \boldsymbol{S}_\lambda)^2 \boldsymbol{Y}}{(1 - \mathsf{tr}(\boldsymbol{S})/n)^2}.$$

According to GCV, the best λ is given by

$$\lambda^{\mathsf{GCV}} = \arg\min_\lambda \mathsf{GCV}(\lambda).$$

Effective degree-of-freedom

tr$(\boldsymbol{S})$ is called the (effective) degrees of freedom.

Recall the degrees of freedom for the estimator $\widehat{\boldsymbol{Y}}$ is defined as

$$\text{df}(\widehat{\boldsymbol{Y}}) = \sum_{i=1}^{n} \frac{\text{Cov}(\widehat{Y}_i, Y_i)}{\sigma^2}.$$

For linear smoother with additive error, we have

$$\text{Cov}(\widehat{Y}_i, Y_i) = \text{Cov}(e_i^\top \boldsymbol{S} Y, e_i^\top Y) = \boldsymbol{S}_{ii}\sigma^2$$

and

$$\text{df}(\widehat{\boldsymbol{Y}}) = \sum_{i=1}^{n} \boldsymbol{S}_{ii} = \text{tr}(\boldsymbol{S}).$$

Degrees of freedom is generally a useful concept because it allows us to put two different procedures on equal footing.

Method	$\boldsymbol{S}$	$\operatorname{tr}(\boldsymbol{S})$
Multiple linear regression	$\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top$	p
Ridge regression	$\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X}+\lambda\boldsymbol{I})^{-1}\boldsymbol{X}^\top$	$\sum_{i=1}^n \frac{d_i}{d_i+\lambda}$
Kernel ridge regression	$\boldsymbol{K}(\boldsymbol{K}+\lambda\boldsymbol{I})^{-1}$	$\sum_{i=1}^n \frac{\gamma_i}{\gamma_i+\lambda}$

Table: d_j's are the singular values of $\boldsymbol{X}$ and γ_i's are the eigenvalues of $\boldsymbol{K}$.

Remarks

- N-fold CV generally preferred over single validation if computation allows.
- 5-fold or 10-fold CV generally works well.
- Except for the cases with linear smoother, leave-one-out CV requires n additional training per model, which is not always feasible in practice.

Reference

Jianqing Fan et al., “Statistical Foundation of Data Science,” https://orfe.princeton.edu/~jqfan/fan/classes/525/chapters1-3.pdf

Jerome H. Friedman, Robert Tibshirani, and Trevor Hastie, “The Elements of Statistical Learning,” Springer.

Bradley Efron, Trevor Hastie, Iain Johnstone and Robert Tibshirani, *Least Angle Regression,* https://web.stanford.edu/~hastie/Papers/LARS/LeastAngle_2002.pdf

Rafael A. Irizarry, http://rafalab.github.io/pages/754/section-09.pdf