Penalizing the coefficients

Dominique Guillot

Departments of Mathematical Sciences
University of Delaware

February 20, 2017
Recall: least-squares regression:

$$\hat{\beta}^{LS} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2.$$
Shrinkage methods

Recall: least-squares regression:

$$\hat{\beta}_{LS} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X\beta \|^2_2.$$ 

Penalizing the coefficients:

- Want to restrict the number or the size of the regression coefficients.
- Add a penalty (or “price to pay”) for including a nonzero coefficient.
**Shrinkage methods**

**Recall:** least-squares regression:

$$\hat{\beta}^{LS} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2.$$ 

**Penalizing the coefficients:**

- 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.

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

Recall: least-squares regression:

\[ \hat{\beta}^{LS} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X\beta \|^2_2. \]

Penalizing the coefficients:

- 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.

\[ \hat{\beta}^0 = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X\beta \|^2_2 + \lambda \sum_{i=1}^{p} 1_{\beta_i \neq 0} \right). \]

- Pay a fixed price \( \lambda \) for including a given variable into the model.
Recall: least-squares regression:

\[ \hat{\beta}_{LS} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2. \]

**Penalizing the coefficients:**

- 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.

\[ \hat{\beta}^0 = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X \beta \|_2^2 + \lambda \sum_{i=1}^{p} 1_{\beta_i \neq 0} \right). \]

- Pay a fixed price \( \lambda \) 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., \( \beta_i = 0 \)).
Shrinkage methods

Recall: least-squares regression:

\[ \hat{\beta}_{LS} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2. \]

Penalizing the coefficients:
- 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.

\[ \hat{\beta}^0 = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X \beta \|_2^2 + \lambda \sum_{i=1}^p 1_{\beta_i \neq 0} \right). \]
- Pay a fixed price \( \lambda \) 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.
Relaxations of the previous approach:

2 Ridge regression/Tikhonov regularization:

\[ \hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \|y - X\beta\|_2^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right). \]
Shrinkage methods (cont.)

Relaxations of the previous approach:

- **Ridge regression/Tikhonov regularization:**

  \[
  \hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X\beta \|_2^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right).
  \]

  - Shrinks the coefficients by imposing a penalty on their size.
  - Penalty = $\lambda \cdot \| \beta \|_2^2$. 
Relaxations of the previous approach:

2. **Ridge regression/Tikhonov regularization:**

\[
\hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \|y - X\beta\|_2^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right).
\]

- Shrinks the coefficients by imposing a penalty on their size.
- Penalty = \(\lambda \cdot \|\beta\|_2^2\).
- Problem equivalent to

\[
\hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 \text{ subject to } \sum_{i=1}^{p} \beta_i^2 \leq t.
\]
Relaxations of the previous approach:

2 Ridge regression/Tikhonov regularization:

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

- Shrinks the coefficients by imposing a penalty on their size.
- Penalty = $\lambda \cdot \|\beta\|^2_2$.
- Problem equivalent to

$$\hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|^2_2 \text{ subject to } \sum_{i=1}^{p} \beta_i^2 \leq t.$$

- Penalty is a smooth function.
- Easy to solve (solution can be written in closed form).
Shrinkage methods (cont.)

Relaxations of the previous approach:

2 Ridge regression/Tikhonov regularization:

\[
\hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X\beta \|_2^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right).
\]

- Shrinks the coefficients by imposing a penalty on their size.
- Penalty = \( \lambda \cdot \| \beta \|_2^2 \).
- Problem equivalent to

\[
\hat{\beta}_{\text{ridge}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X\beta \|_2^2 \text{ subject to } \sum_{i=1}^{p} \beta_i^2 \leq t.
\]

- 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

\[
\frac{\partial}{\partial \beta} \left( \| y - X \beta \|_2^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^T X \beta - X^T y) + 2\lambda \beta
\]

\[
= 2 \left( (X^T X + \lambda I) \beta - X^T y \right).
\]
We have
\[
\frac{\partial}{\partial \beta} \left( \|y - X\beta\|_2^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^TX\beta - X^Ty) + 2\lambda \beta \\
= 2 \left( (X^TX + \lambda I)\beta - X^Ty \right). 
\]

Therefore, the critical points satisfy
\[
(X^TX + \lambda I)\beta = X^Ty. 
\]
Ridge regression: closed form solution

We have

$$\frac{\partial}{\partial \beta} \left( \|y - X\beta\|^2_2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^T X \beta - X^T y) + 2\lambda \beta$$

$$= 2 \left( (X^T X + \lambda I) \beta - X^T y \right).$$

Therefore, the critical points satisfy

$$(X^T X + \lambda I) \beta = X^T y.$$  

**Note:** $(X^T X + \lambda I)$ is positive definite, and therefore invertible.
Ridge regression: closed form solution

We have
\[ \frac{\partial}{\partial \beta} \left( \|y - X\beta\|^2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^TX\beta - X^Ty) + 2\lambda\beta \]
\[ = 2 \left( (X^TX + \lambda I)\beta - X^Ty \right). \]

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,
\[ \beta_{\text{ridge}} = (X^TX + \lambda I)^{-1}X^Ty. \]
We have
\[
\frac{\partial}{\partial \beta} \left( \| y - X \beta \|^2_2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^T X \beta - X^T y) + 2\lambda \beta \\
= 2 \left( (X^T X + \lambda I) \beta - X^T y \right).
\]

Therefore, the critical points satisfy
\[
(X^T X + \lambda I) \beta = X^T y.
\]

Note: \((X^T X + \lambda I)\) 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,
\[
\beta^{\text{ridge}} = (X^T X + \lambda I)^{-1} X^T y.
\]

Remarks:
- When \(\lambda > 0\), the estimator is defined even when \(n < p\).
Ridge regression: closed form solution

We have
\[
\frac{\partial}{\partial \beta} \left( \|y - X\beta\|^2_2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^TX\beta - X^Ty) + 2\lambda\beta \\
= 2 \left( (X^TX + \lambda I)\beta - X^Ty \right).
\]

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.
Ridge regression: closed form solution

We have

\[
\frac{\partial}{\partial \beta} \left( \|y - X\beta\|^2_2 + \lambda \sum_{i=1}^{p} \beta_i^2 \right) = 2(X^TX\beta - X^Ty) + 2\lambda \beta
\]

\[
= 2 \left( (X^TX + \lambda I)\beta - X^Ty \right).
\]

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 (Least Absolute Shrinkage and Selection Operator):

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X \beta \|_2^2 + \lambda \sum_{i=1}^{p} |\beta_i| \right). \]
The Lasso (Least Absolute Shrinkage and Selection Operator):

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

- Introduced in 1996 by Robert Tibshirani.
- Equivalent to

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2 \text{ subject to } \| \beta \|_1 = \sum_{i=1}^{p} |\beta_i| \leq t.$$
The Lasso (Least Absolute Shrinkage and Selection Operator):

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X\beta \|_2^2 + \lambda \sum_{i=1}^{p} |\beta_i| \right). \]

- Introduced in 1996 by Robert Tibshirani.
- Equivalent to

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X\beta \|_2^2 \text{ subject to } \| \beta \|_1 = \sum_{i=1}^{p} |\beta_i| \leq t. \]

- Sets coefficients to zero (model selection) and shrinks them.
- More “global” approach to selecting variables compared to previously discussed greedy approaches.
The Lasso (Least Absolute Shrinkage and Selection Operator):

\[
\hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \| y - X \beta \|_2^2 + \lambda \sum_{i=1}^{p} |\beta_i| \right).
\]

- Introduced in 1996 by Robert Tibshirani.
- Equivalent to

\[
\hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2 \text{ subject to } \| \beta \|_1 = \sum_{i=1}^{p} |\beta_i| \leq t.
\]

- Sets coefficients to zero (model selection) and shrinks them.
- 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.
The Lasso (Least Absolute Shrinkage and Selection Operator):

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \left( \|y - X\beta\|_2^2 + \lambda \sum_{i=1}^{p} |\beta_i| \right). \]

- Introduced in 1996 by Robert Tibshirani.
- Equivalent to

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 \text{ subject to } \|\beta\|_1 = \sum_{i=1}^{p} |\beta_i| \leq t. \]

- Sets coefficients to zero (model selection) and shrinks them.
- 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

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2 \]
subject to \( \| \beta \|_1 = \sum_{i=1}^{p} |\beta_i| \leq t \)
Important model selection property

$$\hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X\beta \|_2^2$$
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.*
Important model selection property

\[ \hat{\beta}_{\text{lasso}} = \arg\min_{\beta \in \mathbb{R}^p} \| y - X \beta \|_2^2 \]
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.
- Likely to “hit” corners. Thus, the solution usually has many zeros.
Note: We usually do not penalize the intercept (variable “0” on the figure).
Elastic net (Zou and Hastie, 2005)

$$\hat{\beta}_{e-net} = \arg\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|^2_2 + \lambda_2 \|\beta\|^2_2 + \lambda_1 \|\beta\|_1.$$
Elastic net (Zou and Hastie, 2005)

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

- Benefits from both $\ell_1$ (model selection) and $\ell_2$ regularization.
Elastic net

Elastic net (Zou and Hastie, 2005)

\[ \hat{\beta}_{\text{e-net}} = \text{argmin}_{\beta \in \mathbb{R}^p} \| y - X \beta \|^2 + \lambda_2 \| \beta \|^2 + \lambda_1 \| \beta \|_1. \]

- Benefits from both \( \ell_1 \) (model selection) and \( \ell_2 \) regularization.
- Downside: Two parameters to choose instead of one (can increase the computational burden quite a lot in large experiments).
Ridge, lasso, elastic net have regularization parameters.

Cross-validation is a popular approach for rigorously choosing parameters.

K-fold cross-validation:
- Split data into $K$ equal (or almost equal) parts/folds at random.
- For each parameter $\lambda_i$ do:
  - Fit model on data with fold $j$ removed.
  - Test model on remaining fold $\rightarrow j$-th test error.
- Compute average test errors for parameter $\lambda_i$.
- Pick parameter with smallest average error.
Choosing parameters: cross-validation

- Ridge, lasso, elastic net have regularization parameters.
- We obtain a family of estimators as we vary the parameter(s).
Choosing parameters: cross-validation

- Ridge, lasso, elastic net have regularization parameters.
- We obtain a family of estimators as we vary the parameter(s).
- An *optimal* parameter needs to be chosen in a principled way.

Cross-validation is a popular approach for rigorously choosing parameters.

**K-fold cross-validation:**

1. Split data into $K$ equal (or almost equal) parts/folds at random.
2. For each parameter $\lambda_i$ do
   - For $j = 1, \ldots, K$
     - Fit model on data with fold $j$ removed.
     - Test model on remaining fold $\rightarrow j$-th test error.
3. Compute average test errors for parameter $\lambda_i$.
4. Pick parameter with smallest average error.
Choosing parameters: cross-validation

- Ridge, lasso, elastic net have regularization parameters.
- We obtain a family of estimators as we vary the parameter(s).
- An *optimal* parameter needs to be chosen in a principled way.
- **Cross-validation** is a popular approach for rigorously choosing parameters.
Ridge, lasso, elastic net have regularization parameters. 
We obtain a family of estimators as we vary the parameter(s). 
An *optimal* parameter needs to be chosen in a principled way. 
**Cross-validation** is a popular approach for rigorously choosing parameters.

*K-fold cross-validation:*

Split data into *K* equal (or almost equal) parts/folds at random.

```plaintext
for each parameter \( \lambda_i \) do
  for \( j = 1, \ldots, K \) do
    Fit model on data with fold \( j \) removed.
    Test model on remaining fold \( \rightarrow j \)-th test error.
  end for
  Compute average test errors for parameter \( \lambda_i \).
end for
```

Pick parameter with smallest average error.
More precisely,

- Split data into $K$ folds $F_1, \ldots, F_K$.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train</td>
<td>Train</td>
<td>Validation</td>
<td>Train</td>
<td>Train</td>
</tr>
</tbody>
</table>

Let $L(y, \hat{y})$ be a loss function. For example, $L(y, \hat{y}) = \|y - \hat{y}\|^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$.

Let $f_{-k}(x)$ be the model trained on all but the $k$-th fold.

Let $CV(\lambda) := \frac{1}{n} \sum_{k=1}^{K} \sum_{i \in F_k} L(y_i, f_{-i}(x_i))$.

Pick $\lambda$ among a relevant set of parameters $\hat{\lambda} = \arg\min_{\lambda \in \{\lambda_1, \ldots, \lambda_m\}} CV(\lambda)$. 

10/13
More precisely,

- Split data into $K$ folds $F_1, \ldots, F_K$.

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>Train</td>
<td>Validation</td>
<td>Train</td>
<td>Train</td>
</tr>
</tbody>
</table>

- Let $L(y, \hat{y})$ be a \textit{loss function}. For example, $L(y, \hat{y}) = \|y - \hat{y}\|_2^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$. 

More precisely,

- Split data into $K$ folds $F_1, \ldots, F_K$.

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>Train</td>
<td>Validation</td>
<td>Train</td>
<td>Train</td>
</tr>
</tbody>
</table>

- Let $L(y, \hat{y})$ be a loss function. For example,
  \[
  L(y, \hat{y}) = \| y - \hat{y} \|_2^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2.
  \]

- Let $f_\lambda^{-k}(x)$ be the model fitted on all, but the $k$-th fold.
K-fold CV

More precisely,

- Split data into $K$ folds $F_1, \ldots, F_K$.

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>Train</td>
<td>Validation</td>
<td>Train</td>
<td>Train</td>
</tr>
</tbody>
</table>

- Let $L(y, \hat{y})$ be a loss function. For example,
  \[ L(y, \hat{y}) = \|y - \hat{y}\|_2^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \]
- Let $f_{\lambda}^{-k}(x)$ be the model fitted on all, but the $k$-th fold.
- Let $CV(\lambda) := \frac{1}{n} \sum_{k=1}^{n} \sum_{i \in F_k} L(y_i, f_{\lambda}^{-i}(x_i))$
More precisely,

- Split data into $K$ folds $F_1, \ldots, F_K$.

- Let $L(y, \hat{y})$ be a loss function. For example,
  \[ L(y, \hat{y}) = \| y - \hat{y} \|_2^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \]

- Let $f_{\lambda}^{-k}(x)$ be the model fitted on all, but the $k$-th fold.

- Let
  \[ CV(\lambda) := \frac{1}{n} \sum_{k=1}^{n} \sum_{i \in F_k} L(y_i, f_{\lambda}^{-i}(x_i)) \]

- Pick $\lambda$ among a relevant set of parameters
  \[ \hat{\lambda} = \arg\min_{\lambda \in \{\lambda_1, \ldots, \lambda_m\}} CV(\lambda) \]
Model selection vs Model assessment

Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

Generally speaking, the CV error provides a good estimate of the prediction error.

When enough data is available, it is better to separate the data into three parts: train/validate, and test. Typically: 50% train, 25% validate, 25% test.

Test data is kept in a vault, i.e., not used for fitting or choosing the model.

Other methods (e.g. AIC, BIC, etc.) can be used when working with very little data.
Model selection vs Model assessment

Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

- **Model assessment**: having chosen a final model, estimating its prediction error (generalization error) on new data.
Model selection vs Model assessment

Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

- **Model assessment**: having chosen a final model, estimating its prediction error (generalization error) on new data.

Model assessment: is the estimator really good? compare different models with their own sets of parameters.
Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

- **Model assessment**: having chosen a final model, estimating its prediction error (generalization error) on new data.

Model assessment: is the estimator really good? compare different models with their own sets of parameters.

Generally speaking, the CV error provides a good estimate of the prediction error.
Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

- **Model assessment**: having chosen a final model, estimating its prediction error (generalization error) on new data.

Model assessment: is the estimator really good? compare different models with their own sets of parameters.

Generally speaking, the CV error provides a good estimate of the prediction error.

- When *enough* data is available, it is better to separate the data into three parts: train/validate, and test.
Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

- **Model assessment**: having chosen a final model, estimating its prediction error (generalization error) on new data.

Model assessment: is the estimator really good? compare different models with their own sets of parameters.

Generally speaking, the CV error provides a good estimate of the prediction error.

- When *enough* data is available, it is better to separate the data into three parts: train/validate, and test.

- Typically: 50% train, 25% validate, 25% test.
Two related, but different goals:

- **Model selection**: estimating the performance of different models in order to choose the “best” one.

- **Model assessment**: having chosen a final model, estimating its prediction error (generalization error) on new data.

Model assessment: is the estimator really good? compare different models with their own sets of parameters.

Generally speaking, the CV error provides a good estimate of the prediction error.

- When *enough* data is available, it is better to separate the data into three parts: train/validate, and test.

- Typically: 50% train, 25% validate, 25% test.

- Test data is “kept in a vault”, i.e., not used for fitting or choosing the model.

- Other methods (e.g. AIC, BIC, etc.) can be used when working with very little data.
Summary of the regression methods seen so far

1. Ordinary least squares (OLS)
   - Minimizes sum of squares.
   - Solution not unique when $n < p$.
   - Estimate unstable when the predictors are collinear.
   - Generally does not lead to best prediction error.
Summary of the regression methods seen so far

1. Ordinary least squares (OLS)
   - Minimizes sum of squares.
   - Solution not unique when $n < p$.
   - Estimate unstable when the predictors are collinear.
   - Generally does not lead to best prediction error.

2. Ridge regression ($\ell_2$ penalty)
   - Regularized solution.
   - Estimator exists and is stable, even when $n < p$.
   - Easy to compute (add multiple of identity to $X^T X$).
   - Coefficients not set to zero (no model selection).
3 Subset selection methods (best subset, stepwise and stagewise approaches)

- Generally leads to a favorable bias-variance trade-off.
- Model selection. Leads to models that are easier to interpret and work with.
- Can be computationally intensive (e.g. best subset can only be computed for small $p$)
- Some of the approaches are greedy/less-rigorous.
Summary of the regression methods seen so far (cont.)

3. Subset selection methods (best subset, stepwise and stagewise approaches)
   - Generally leads to a favorable bias-variance trade-off.
   - Model selection. Leads to models that are easier to interpret and work with.
   - Can be computationally intensive (e.g. best subset can only be computed for small $p$)
   - Some of the approaches are greedy/less-rigorous.

4. Lasso ($\ell_1$ penalty)
   - Shrinks and sets to zero the coefficients (shrinkage + model selection).
   - Generally leads to a favorable bias-variance trade-off.
   - Model selection. Leads to models that are easier to interpret and work with.
   - Can be efficiently computed.
   - Supporting theory. Active area of research.