Linear Model Selection and Regularization

In earlier chapters, we explored the linear regression model and how it can be used to fit data using least squares. While least squares provide an unbiased and simple way to estimate the relationship between predictors and a response, it often struggles when:

The number of predictors p is:

- When $n \gg p$, the model has low variance and performs well on the test set.
- When $n \approx p$, it results in high variability, leading to overfitting and poor test performance.
- When $n < p$, there are infinitely many solutions with zero training error, which typically perform poorly on the test set.

We are concerned about model interpretability, prediction accuracy, or overfitting. Often, some variables may not be associated with the response but still increase model complexity unnecessarily.

To address these challenges, Chapter 6 introduces three powerful extensions to linear regression that enhance both performance and generalization.

In This Chapter, We Explore Three Major Approaches:

- **Subset Selection:** Selecting a subset of the most relevant predictors to include in the model.
- **Shrinkage (Regularization):** Including all predictors but applying a penalty to shrink the magnitude of less important coefficients — depending on the type of shrinkage, some coefficients may even be reduced to exactly zero.
- **Dimension Reduction:** Reducing the predictor space by projecting the original variables into a smaller number of informative components.

These approaches help balance the bias-variance trade-off, improve model interpretability, and reduce the risk of overfitting, especially in high-dimensional settings.

Subset Selection

Best Subset Selection

We take all p possible variables and create linear models for each of the 2^p subsets, and select the best among them. It is not trivial, so this is usually broken up into two stages in the algorithm.

Algorithm:

1. Let M_0 denote the null model (which contains no predictors). This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, 3, \ldots, p$:
    - Fit all $\binom{p}{k}$ models that contain exactly k predictors.
    - Pick the best among these models, and call it M_k. Here, the best model has the smallest RSS or the largest R^2.
3. Select a single best model from among $M_0, M_1, M_2, \ldots, M_p$ using the prediction error on a validation set, C_p, AIC, BIC, or adjusted R^2, or use the cross-validation method.

    Although we select the best model having less RSS and higher R^2 on the basis of C_p (Mallows' C_p), AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion), and cross-validation.

Pros:

- Gives the best model among all possible models.
- Very thorough and accurate (if p is small).

Cons:

- If the number of predictors is large, then total subsets = 2^p, which is very large to search for the best one.

Forward Stepwise Selection

- Computationally efficient alternative to best-subset selection.

- Models are $\frac{p(p+1)}{2} + 1$ — it guides the search over the model space.
- Though forward stepwise tends to do well in practice, it is not guaranteed to find the best out of 2^p models.

Algorithm:

1. Let M_0 be the null model (no predictors), which simply predicts the mean of the response variable: $M_0 = \beta_0 = \bar{y}$.
2. For $k = 1$ to p:
   - Add one new predictor to M_k.
   - Add only predictors not already in M_k.
   - Among the $p - k$ remaining predictors, choose the one that decreases the RSS the most or increases R^2 the most.
   - Call this new model M_{k+1}.

   Repeat until all predictors have been added.
3. Select a single best model from among $M_0, M_1, M_2, \ldots, M_p$ using prediction error on the validation set, C_p, AIC, BIC, adjusted R^2, or cross-validation.

Although we select the best model by comparing RSS and R^2, we can use criteria like:

- C_p (Mallows' C_p)
- AIC (Akaike Information Criterion)
- BIC (Bayesian Information Criterion)
- Cross-validation

Pros:

- Faster than best subset selection — doesn't try all combinations.
- Still gives pretty good results.

Cons:

- Once a predictor is added, it can't be removed.
- Might miss the globally best model.

Backward Stepwise Selection

Algorithm:

1. Start with the full model (M_p).
2. For $k = p$ to 1:
   - Remove one predictor at a time.
   - For the current model, remove each predictor one at a time, estimate the model, and compute RSS and R^2.
   - Choose the best among these p models (with lowest RSS or highest R^2), and call it M_{k-1}.

   Repeat until only one predictor remains.
3. Select a single best model from among $M_0, M_1, M_2, \ldots, M_p$ using prediction error on a validation set, C_p, AIC, BIC, adjusted R^2, or cross-validation.

Although we select the best model by comparing RSS and R^2, we can use criteria like:

- C_p (Mallows' C_p)
- AIC (Akaike Information Criterion)
- BIC (Bayesian Information Criterion)
- Cross-validation

Pros:

- Often better than best subset selection computationally.

Cons:

- Once a predictor is removed, it cannot be re-added.
- The estimated model might not be globally optimal.

Hybrid Approach (Mix forward & backward)

- Adding predictor from a side (forward).
- Based on their updated importance, can remove predictor (like backward).

Algorithm:

1. Let M_0 denote the null model, which contains no predictor. This model simply predicts the sample mean for each observation.

2. Repeat until stopping criteria met:

    - **Forward step:** Add predictor (one) that improves model most.
    - **Backward step:** After adding, check if any of the previously added predictors have become insignificant & remove them if needed. Alternate between adding and removing predictor.

3. Select a single best model from among $M_0, M_1, M_2, \ldots, M_p$ using the prediction error on validation set, C_p, AIC, BIC or adjusted R^2 or use the cross-validation method.

    Although we select best model having less RSS and higher R^2 on basis of C_p (Mallow's C_p), AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion) and cross-validation.

Comparison Summary:

Method	**Add Predictor**	**Remove Predictor**	**Tries All Combination**	**Fast**	**Accurac**
Best Subset	Yes	Yes	Yes	No	Highest
Forward	Yes	No	No	Medium	Medium
Backward	No	Yes	No	Yes	Medium
Hybrid	Yes	Yes	No	Fast	Better

How to choose best optimal model:

1. Not always smallest RSS and highest R^2 give best model because:

    - RSS always decreases when new predictor added (overfitting).
    - R^2 always increases when new predictor added.

    And training error improves but does not mean that error decreases on test data.

2. Best model based on test error where test error calculated by two approaches:

    - Indirect estimate test error (C_p, AIC, BIC, Adjusted R^2).
    - Direct estimate test error (Cross-validation).

Indirect estimation of test error:

1. C_p: Mallow's C_p:

- C_p is used to select among many linear regression models.
- It balances the *fit of the model* (how well it matches the data) and the *complexity* (number of predictors used).
- Derived from:

$$\mathbb{E}[\text{Test MSE}] \approx \text{Training MSE} + \frac{2d\sigma^2}{n} \quad \text{(Optimism)}$$

- C_p tries to minimize test MSE by choosing a sweet spot between bias and variance.
- **Mathematical Form (for a model with d predictors):**

$$C_p = \frac{1}{n}(\text{RSS} + 2d\sigma^2)$$

  Where:

  - RSS = Residual Sum of Squares for the model
  - d = number of predictors used in the model
  - σ^2 = estimate of error variance from the full model (with all predictors):

$$\sigma^2 = \frac{\text{RSS}_{\text{full}}}{n - d_{\text{full}} - 1}$$

- As you increase d, training MSE decreases but test MSE may increase (overfitting).
- Lower $C_p \Rightarrow$ better model

2. AIC: Akaike Information Criterion:

- AIC balances model fit (maximize log-likelihood) and complexity (parameters).

$$\text{AIC} = -2\log(L) + 2d$$

  Where:

  - L = Maximum value of likelihood of the model
  - d = number of parameters in the model

- More parameters → better log-likelihood, but might overfit.
- AIC adds penalty $2d$ to discourage adding useless terms.
- **Akaike's Philosophy:** AIC approximates Kullback-Leibler divergence between true model $g(x)$ and estimated model $f(x)$.
- Lower AIC $\Rightarrow$ better model

3. BIC: Bayesian Information Criterion:

- Similar to AIC, but stronger penalty for complexity:

$$\text{BIC} = -2\log(L) + d\log(n)$$

- Where:
  - L = likelihood
  - d = number of parameters
  - n = number of observations
- Penalty term $d\log(n)$ increases faster than AIC's $2d$, especially for large n.
- Derived from Bayesian marginal likelihood.
- Lower BIC $\Rightarrow$ better model

4. Adjusted R^2:

$$\text{Adjusted } R^2 = 1 - \left(\frac{\text{RSS}(n-1)}{\text{TSS}(n-d-1)}\right)$$

- R^2 always increases with more predictors.
- Adjusted R^2 increases only if new predictor is helpful.
- Maximum Adjusted R^2 $\Rightarrow$ optimal model size.

Comparison Table:

Criteria	**Full Form**	**Formula**	**Penalty?**	**Based On**	**Bette**
C_p	Mallow's C_p	$\frac{1}{n}(\text{RSS} + 2d\sigma^2)$	Yes (2dσ^2)	Training Error	Lower
AIC	Akaike Info Criterion	$-2\log L + 2d$	Yes (2d)	Likelihood	Lower
BIC	Bayesian Info Criterion	$-2\log L + d\log n$	Yes (stronger)	Likelihood	Lower
Adjusted R^2	Adj. R-squared	$1 - \frac{(1-R^2)(n-1)}{(n-p-1)}$	Yes (implicit)	Variance explained	Highe

Direct estimation of test error:

1. Validation set:

- Split data into two parts:

  - Training set: used to train the model
  - Validation set: used to estimate test error

2. Cross-validation:

- More robust, splits data into multiple parts

  Types of Cross-validation:

- **K-fold cross-validation:**
  - Split into k equal parts (folds)
  - Each fold used as validation once, others as training
  - Average error across k runs
- **LOOCV (Leave-One-Out Cross-Validation):**
  - Leave one observation out as validation
  - Train on remaining $n - 1$
  - Repeat for all n, then average test errors

1 Shrinkage Methods

We fit a model using all p predictors with techniques that constrain or regularize the coefficient estimates, shrinking coefficients toward zero.

Why Shrink?

- Using all predictors can cause:
  1. Overfitting
  2. High variance (sensitivity)
- Shrinkage:
  - Reduces variance
  - Improves model's generalization on new data
  - Improves bias-variance trade-off (introduces bias while reducing variance)

1.1 Ridge Regression

$$\text{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .$$

- Similar to least squares, but we minimize:

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^{p} \beta_j^2,$$

  where λ is a hyperparameter controlling shrinkage.

- Solution:

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

- Properties:
  - Minimizes RSS while penalizing coefficients toward zero
  - Decreases variance and overfitting
  - β_0 has no penalty
  - Uses L_2 norm: $\|\beta\|_2 = \sqrt{\sum_{j=1}^{p} {\beta_j}^2}$

Why Standardization is Mandatory in Ridge Regression

- OLS is scale-invariant (multiplying predictor by 1000 has no effect)
- Ridge Regression is scale-variant (penalty term changes with scaling)
- Standardization formula:

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \overline{x}_j)^2}}$$

2 The Lasso

- Overcomes Ridge's disadvantage of keeping all predictors
- Minimizes:

$$\text{RSS} + \lambda \sum_{j=1}^{p} |\beta_j|$$

Table 1: Comparison: OLS vs Ridge Regression

Features	OLS	Ridge Regression
Bias	Low	Comparatively High
Variance	High	Low
Overfitting	High	Low
Prediction stability	Low	High
Works when $p > n$	No	Yes
Works when $p = n$	No	Yes
Scaling needed	No	Yes
Test error	High	Lower (at best λ)

- Combination of Ridge and Best Subset selection
- Uses L_1 penalty: $\|\beta\|_1 = \sum |\beta_j|$
- Can force coefficients to exactly zero (variable selection)

Comparison of Methods

- **Ridge**:
$$\min \text{RSS} \quad \text{subject to} \quad \sum \beta_j^2 \leq s$$
- **Lasso**:
$$\min \text{RSS} \quad \text{subject to} \quad \sum |\beta_j| \leq s$$
- **Best Subset**:
$$\min \text{RSS} \quad \text{subject to} \quad \sum I(\beta_j \neq 0) \leq s$$

Geometric Interpretation

- **Lasso**:
  - Diamond constraint region (L_1 norm)
  - Often intersects axes, setting coefficients to zero
- **Ridge**:
  - Circular constraint region (L_2 norm)
  - Rarely intersects axes, keeping all coefficients non-zero

Ridge	Lasso
Good when response depends on many predictors with roughly equal coefficients	Good when few predictors have substantial coefficient and others are small/zero
Smooth shrinkage	Non-smooth (some coefficients exactly zero)
No variable selection	Performs variable selection
	Automatically selects important predictors

Prediction Accuracy Comparison

3 Bayesian Interpretation

- Treats coefficients β as random variables with prior distributions
- **Ridge**: Gaussian prior (smooth shrinkage, no zeros)
- **Lasso**: Laplace (double-exponential) prior
- Posterior mode = most likely β given data and prior
  - Ridge posterior mode = Ridge solution
  - Lasso posterior mode = Lasso solution (sparse)

4 Selecting the Tuning Parameter

- λ controls amount of regularization
- High λ = high penalty = smaller coefficients
- Cross-validation used to choose optimal λ
- Lasso with CV performs feature selection
- Ridge shrinks all variables but keeps all
- Least squares doesn't use λ and can overfit when $p \approx n$ or $p > n$

Dimension Reduction Method

- Dimension Reduction: transform original features into new ones (Z1, Z2, Z3...Zm)
- Each Zm are linear combination of original x_i

- Controls variance ⇒ improves stability and generalization
- Two step:
  1. Create Z's
  2. Fit model on them
- Works best when:
  - $p \gg n$
  - High correlation among X's

Principal Component Regression (PCR)

- PCA finds new axes (Z1, Z2, Z3...Zm) called principal components.
- PCA works by computing eigenvectors of the covariance matrix of X.
- Z_1: Direction with max variance; Z_2: Left max variance $\perp Z_1$, and so on.
- Orthogonal components ⇒ uncorrelated variables
- PCR does not use the response y → unsupervised learning
- First few components capture most of the variance; rest can be ignored
- Standardize data before applying PCA:

$$X_{\text{standardized}} = \frac{X - \text{mean}(X)}{\text{std}(X)}$$

Why do this?

Problem	Solution
Multicollinearity	PCA gives orthogonal Zs
Overfitting (too many predictors)	Only top m components used (m ¡ p)
Interpretability	Focus on most informative directions
Noise	Lower principal components capture less signal – drop them

PCR Steps:

1. Compute covariance matrix: $S = \frac{1}{n-1} X^T X$
2. Compute eigenvectors: $V = \text{eig}(S)$
3. Sort eigenvectors by eigenvalue

4. Select top m eigenvectors: V_m
5. Compute principal components: $Z = XV_m$
6. Fit model: $y_i = \beta_0 + \beta_1 Z_1 + \cdots + \beta_m Z_m$

Partial Least Squares (PLS)

- PCR problem: finds directions that explain X, not necessarily related to y
- PLS combines PCA + regression — finds components (Z1, Z2...Zm) that explain X and are predictive of y
- First direction $Z_1 = \sum \phi_j X_j$, where $\phi_j \propto \text{corr}(X_j, y)$
- Iteratively compute new X and y residuals and update directions
- Components selected using cross-validation
- Fit model: $y_i = \beta_0 + \beta_1 Z_1 + \cdots + \beta_m Z_m$

Ridge vs Lasso Regression – Detailed Comparison

Feature / Aspect	**Ridge Regression**	**Lasso Regression**
Type of Penalty	L2 Norm: $\sum \beta_j^2$	L1 Norm: $\sum \lvert\beta_j\rvert$
Objective Function	$\text{RSS} + \lambda \sum \beta_j^2$	$\text{RSS} + \lambda \sum \lvert\beta_j\rvert$
Shrinkage Effect	Shrinks coefficients toward zero but none become exactly zero	Shrinks coefficients, some become exactly zero
Feature Selection	No (includes all features)	Yes (performs automatic feature selection)
Bias	Moderate increase	Can be high (especially when many coefficients forced to zero)
Variance	Substantial reduction	Strong reduction
Overfitting Risk	Lower than OLS	Lower than OLS; can be lower than Ridge in sparse models
Model Interpretability	Harder (uses all predictors)	Easier (produces sparse model)
Performance in High Dimensions (p ¿ n)	Works well	Works well
Correlation Between Predictors	Spreads coefficient values among correlated variables	May randomly pick one predictor among correlated ones
When Most Features Are Relevant	Performs better (all contribute small effects)	Underperforms (eliminates some useful predictors)
When Few Features Are Relevant	Underperforms	Performs better (focuses on relevant predictors)
Solution Geometry	Constrained to a circle (L2 ball)	Constrained to a diamond (L1 ball)
Zero Coefficients Allowed?	No	Yes
Computational Efficiency	Fast (closed-form solution via matrix operations)	Slower (requires convex optimization algorithms like coordinate descent)
Bayesian Interpretation	Gaussian (Normal) prior on coefficients	Laplace (Double Exponential) prior on coefficients
Posterior Mode Interpretation	Ridge estimate	Lasso estimate
Posterior Mean Interpretation	Matches posterior mode	Does not match mode (not sparse)
Stability of Predictions	High	High (but may vary more due to variable selection)
Scaling Requirement	Mandatory	Mandatory
Use Cases	Many features, none dominant	Sparse models, few strong predictors
Lambda Effect	High $\lambda \to$ All coefficients small, but none zero	High $\lambda \to$ Many coefficients exactly zero
Selection of Lambda	Via cross-validation	Via cross-validation