Linear Regression with Regularization

CMSC 173

September 10, 2025

The Problem

- In linear regression, we minimize the cost function:

Cost Function (Ordinary Least Squares)

$$J(\theta_0, \theta_1, \ldots, \theta_p) = \frac{1}{2m} \sum_{i=1}^{m} \left(y^{(i)} - \theta_0 - \sum_{j=1}^{p} \theta_j x_j^{(i)} \right)^2$$

- The model may overfit, especially when:
  - The number of features p is large.
  - Features are highly correlated.
  - Noise dominates the data.

Problem

Overfitting $\Rightarrow$ very low training error, but poor generalization on unseen data.

Why Regularization?

- Ordinary Least Squares (OLS) tries to minimize prediction error on the training set.
- But when the model is too flexible (many parameters), it **fits noise**.
- Regularization combats this by:

Key Idea

Add a penalty term on the size of coefficients θ_j to discourage overly complex models.

- This leads to a trade-off:

Bias-Variance Tradeoff

- **High variance:** OLS with large coefficients $\Rightarrow$ overfitting.
- **High bias:** Too much penalty $\Rightarrow$ underfitting.
- Regularization balances the two.

Why Regularization?

- Ordinary Least Squares (OLS) tries to minimize prediction error on the training set.
- But when the model is too flexible (many parameters), it **fits noise**.

Key Idea

Add a penalty term on the size of coefficients θ_j to discourage overly complex models.

This leads to a trade-off:

Bias-Variance Tradeoff

- **High variance:** OLS with large coefficients $\Rightarrow$ overfitting.
- **High bias:** Too much penalty $\Rightarrow$ underfitting.
- Regularization balances the two.

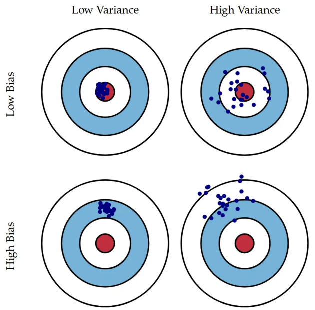


Fig. 1 Graphical illustration of bias and variance.

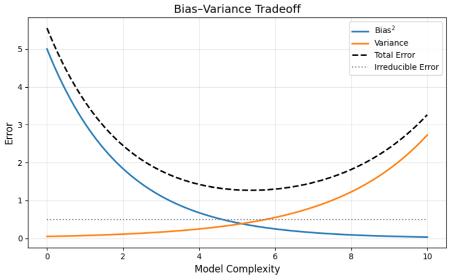
Bias–Variance Tradeoff

- Prediction error can be decomposed as:

Decomposition

$$\mathbb{E}\left[(y - \hat{f}(x))^2\right] = \underbrace{\text{Bias}[\hat{f}(x)]^2}_{\text{Systematic error}} + \underbrace{\text{Var}[\hat{f}(x)]}_{\text{Sensitivity to data}} + \underbrace{\sigma^2}_{\text{Irreducible noise}}$$

- **High variance:** Model too flexible $\Rightarrow$ fits noise.
- **High bias:** Model too simple $\Rightarrow$ misses patterns.



Interpretation

- Bias decreases with model complexity.
- Variance increases with model complexity.
- Total error (MSE) is U-shaped: **best tradeoff lies in the middle**.

Next: Regularization Techniques

Ridge and Lasso Regression

Why focus on these two?

Ridge (L2) and Lasso (L1) are the **most widely used**, forming the foundation of many modern ML models.

From general motivation $\rightarrow$ concrete methods.

Ridge Regression (ℓ_2 penalty)

Cost Function with ℓ_2 Constraint

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(y^{(i)} - \theta_0 - \sum_{j=1}^{p} \theta_j x_j^{(i)} \right)^2 + \lambda \sum_{j=1}^{p} \theta_j^2$$

- Penalizes large coefficients by shrinking them towards zero.
- Constraint set: ℓ_2 ball (circle/sphere).
- Solution: where OLS contour first touches the ℓ_2 ball.

Effect

Ridge keeps all features, but reduces their influence.

Lasso Regression (ℓ_1 penalty)

Cost Function with ℓ_1 Constraint

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(y^{(i)} - \theta_0 - \sum_{j=1}^{p} \theta_j x_j^{(i)} \right)^2 + \lambda \sum_{j=1}^{p} |\theta_j|$$

- Penalizes the absolute size of coefficients.
- Constraint set: ℓ_1 diamond (polytope).
- Solution: OLS contour often touches the corners $\Rightarrow$ many $\theta_j = 0$.

Effect

Lasso performs **feature selection** automatically.

Comparison: Ridge vs Lasso

Ridge (L2)

- Shrinks coefficients smoothly.
- Works well when many features contribute weakly.
- Never eliminates features entirely.

Lasso (L1)

- Encourages sparsity.
- Performs feature selection.
- Can be unstable if predictors are highly correlated.

Rule of Thumb

Use Ridge when *all features matter a little*. Use Lasso when you want *automatic feature selection*.

Ridge Regularization

Linear Basis Function Model:

$$y = w_0 + w_1\phi_1(x) + w_2\phi_2(x) + \cdots + w_m\phi_m(x)$$

$$y = w^T\phi(x)$$

Cost function with Ridge Regularization

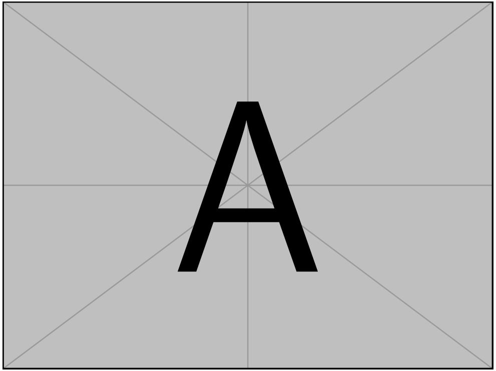
Cost function

$$\min_w f(w) = (y - \Phi w)^T(y - \Phi w) + \frac{\lambda}{2}w^T w$$

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

Example: Growth Data

Find a *10-degree polynomial* that best fits the data:



w/o regularization **with regularization**

Key Takeaways

- Regularization combats overfitting.
- Ridge shrinks coefficients $\rightarrow$ stability, no sparsity.
- Lasso shrinks and sets some coefficients to zero $\rightarrow$ sparsity, feature selection.

For Students

Experiment with Ridge vs Lasso on the Housing dataset. Which method generalizes better? Why?