Ridge Regression with Gradient Descent

1. Introduction

- Previously, Ridge Regression solution was derived analytically:

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

- But for **large datasets with many features**, computing the **matrix inverse** is computationally expensive.
- **Gradient Descent (GD)** offers an **iterative optimization method** to minimize the Ridge cost function without directly computing the inverse.

2. Ridge Regression Cost Function

Ridge Regression minimizes:

$$J(\beta) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

Where:

- $\hat{y}_i = x_i^T \beta$
- λ = regularization parameter
- n = number of samples

3. Gradient of the Cost Function

We take the derivative of $J(\beta)$ with respect to β.

First term (MSE):

$$\frac{\partial}{\partial\beta}\frac{1}{n}(y - X\beta)^T(y - X\beta) = -\frac{2}{n}X^T(y - X\beta)$$

Second term (regularization):

$$\frac{\partial}{\partial\beta}\lambda\beta^T\beta = 2\lambda\beta$$

Combining both:

$$\nabla J(\beta) = -\frac{2}{n}X^T(y - X\beta) + 2\lambda\beta$$

---

4. Gradient Descent Update Rule

We update coefficients iteratively:

$$\beta^{(t+1)} = \beta^{(t)} - \alpha\nabla J(\beta^{(t)})$$

Substituting the gradient:

$$\beta^{(t+1)} = \beta^{(t)} - \alpha\Big(-\frac{2}{n}X^T(y - X\beta^{(t)}) + 2\lambda\beta^{(t)}\Big)$$

Simplify:

$$\beta^{(t+1)} = \beta^{(t)} + \frac{2\alpha}{n}X^T(y - X\beta^{(t)}) - 2\alpha\lambda\beta^{(t)}$$

Where:

- α = learning rate.

---

5. Interpretation

- The update has **two parts:**
  1. **Gradient from errors** ($X^T(y - X\beta)$) → same as normal linear regression.
  2. **Shrinkage term** ($-2\alpha\lambda\beta$) → pushes coefficients closer to zero.

This **penalization term prevents coefficients from growing too large**, controlling overfitting.

---

6. When to Use Gradient Descent for Ridge Regression

- **Large datasets** (matrix inversion costly).
- **Streaming data** or **online learning** scenarios.
- Useful when:
  1. **Number of features (p)** is very large (e.g., >10,000).
  2. Sparse data where matrix inversion is inefficient.

---

7. Effect of Hyperparameters

- **Learning rate (α\alpha):**
  1. Too high → algorithm may diverge.
  2. Too low → very slow convergence.
- **Regularization parameter (λ\lambda):**
  1. Large → more shrinkage (higher bias, lower variance).
  2. Small → closer to linear regression.

🧠 Key Takeaways

- **Gradient Descent version of Ridge Regression** avoids direct matrix inversion.
- Cost function:

$$J(\beta) = \frac{1}{n}\|y - X\beta\|^2 + \lambda\|\beta\|^2$$

- Update rule:

$$\boxed{\beta^{(t+1)} = \beta^{(t)} + \frac{2\alpha}{n}X^T(y - X\beta^{(t)}) - 2\alpha\lambda\beta^{(t)}}$$

- Balances **fit to data** and **regularization shrinkage**.
