📘 Ridge and Lasso Regression – In-depth Intuition

🔹 Linear Regression Recap

- Equation:

$y = mx + c$

- Goal: Minimize the **Sum of Residuals** (errors):

$\text{Cost Function (MSE)} = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$

🔍 Overfitting vs Underfitting (Intuition)

- **Training dataset**: low error
- **Testing dataset**: high error → leads to **overfitting**

📉 If the model fits too closely to the training data:

- Low bias
- High variance

📈 If the model is too simple:

- High bias
- Low variance

📊 Visual Intuition

A simple linear regression line fits a straight line through the data points.
If there's a **huge slope**, even a small change in input x causes large change in output y → leads to **overfitting**.

🧠 Regularization: Ridge and Lasso

To reduce overfitting and improve generalization, **regularization** adds a **penalty term** to the loss function.

🧮 Ridge Regression (L2 Regularization)

- Adds square of coefficients to the cost function:

$\text{Cost} = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{n} \beta_j^2$

- Where:

    - λ\lambdaλ: regularization parameter
    - β\betaβ: model coefficients
- Helps **shrink** coefficients but doesn't set them to zero
- Reduces overfitting

🗒️ **Note**: Small changes in x won't cause large changes in y, making the model more stable.

🧱 **Lasso Regression (L1 Regularization)**

- Adds absolute value of coefficients:

Cost=1m∑i=1m(yi−y^i)2+λ∑j=1n|βj|\text{Cost} = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{n} |\beta_j|Cost=m1i=1∑m(yi−y^i)2+λj=1∑n|βj|

- Can set some coefficients to **zero** → useful for **feature selection**

🧩 **Summary Points**

- **Overfitting**: High variance, low bias
- **Regularization** helps reduce overfitting
- **Ridge**: Shrinks coefficients
- **Lasso**: Shrinks and selects (can make coefficients zero)
- λ>0\lambda > 0λ>0: Any positive value used to apply regularization