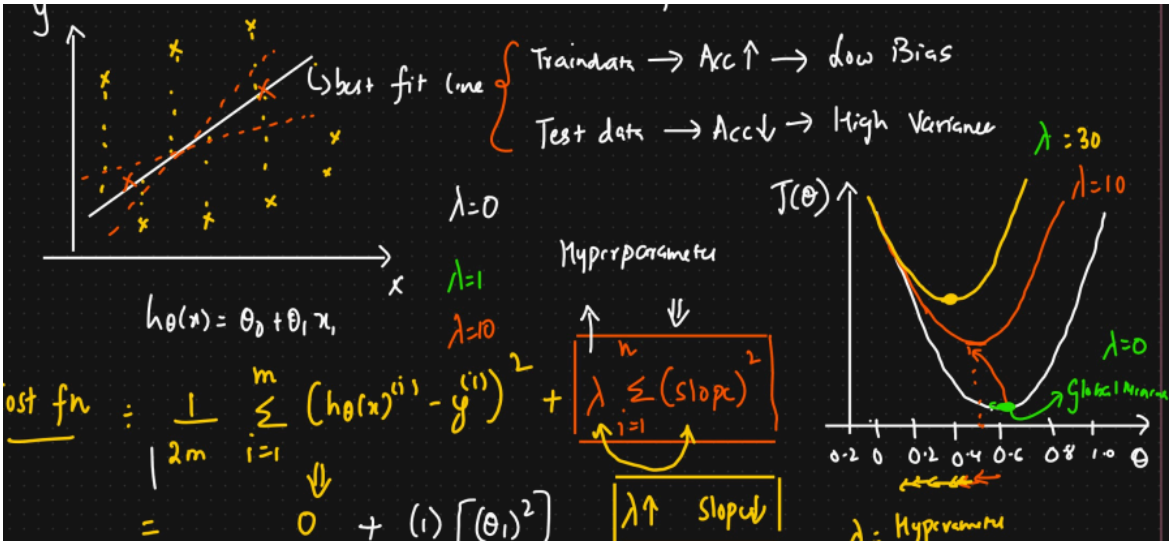
Ridge Regression

- **Type:** Linear Regression Model + Regularization Technique.
- **Purpose:** To address problems encountered in Ordinary Least Squares (OLS) regression, specifically:
    - **Overfitting:** When the model learns the training data too well, including noise, and performs poorly on unseen data.
    - **Multicollinearity:** When predictor variables are highly correlated with each other, leading to unstable and high-variance coefficient estimates in OLS.
- **Mechanism:** It adds a **penalty term** to the OLS cost function. This penalty discourages the model coefficients(b1, b2, …) from becoming too large.



2. The Problem with OLS (Why Ridge is Needed)

- **OLS Cost Function:** Aims to minimize the Sum of Squared Errors (SSE) or Residual Sum of Squares (RSS): $\text{Cost(OLS)} = \Sigma(y_i - \hat{y}_i)^2 = \Sigma(y_i - (\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p))^2$ where y_i is the actual value, $\hat{y}_i$ is the predicted value, and β_p are the model coefficients.
- **Issues:**
    - If the number of predictors (p) is large, or close to the number of observations (n), OLS can overfit.
    - If predictors are highly correlated (multicollinearity), the coefficient estimates (β_p) can become very large and sensitive to small changes in the data, making the model unreliable.

$$\text{Cost Function (Ridge)} = \underbrace{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}_{\text{MSE}} + \underbrace{\lambda \sum_{j=1}^{p} \beta_j^2}_{\text{Penalty Term}}$$

Where:

- y_i: actual output
- $\hat{y}_i$: predicted output
- β_j: model coefficients
- $\lambda \geq 0$: regularization strength

3. How Ridge Regression Works

- **Ridge Cost Function:** Ridge adds a penalty term proportional to the **sum of the squares of the coefficients** (excluding the intercept β_0). This is known as **L2 Regularization**. Cost(Ridge) = $\Sigma(y_i - \hat{y}_i)^2 + \alpha * \Sigma(\beta_j)^2$ (summation for j is from 1 to p)
- **Components:**
  - $\Sigma(y_i - \hat{y}_i)^2$: The standard OLS term (measures goodness of fit).
  - α: The **tuning parameter** (also often denoted as lambda, λ). It's a non-negative value that controls the strength of the penalty.
    - $\alpha = 0$: The penalty term vanishes, and Ridge Regression becomes identical to OLS.
    - $\alpha > 0$: The penalty is active. As α increases, the coefficients are "shrunk" more strongly towards zero.
    - $\alpha \to \infty$: The coefficients are forced very close to (not exactly) zero.
  - $\Sigma(\beta_j)^2$: The **L2 penalty** (squared magnitude of the coefficient vector). It penalizes large coefficients.
- **Objective:** The algorithm now tries to minimize this combined cost function. This involves finding a balance between:
  - Fitting the training data well (minimizing SSE).
  - Keeping the coefficient magnitudes small (minimizing the penalty term).

4. Key Characteristics of Ridge Regression

- **Coefficient Shrinkage:** Ridge shrinks the coefficients towards zero, reducing their variance and the model's complexity.
- **Does Not Perform Feature Selection:** Unlike Lasso Regression (L1 penalty), Ridge generally does *not* force coefficients to be exactly zero. It keeps all predictors in the model but reduces their influence.

- **Handles Multicollinearity:** By penalizing large coefficients, Ridge produces more stable estimates even when predictors are highly correlated. It tends to distribute the effect among correlated predictors.
- **Bias-Variance Trade-off:** Ridge introduces a small amount of bias into the coefficient estimates (they are no longer unbiased like OLS) but significantly reduces the variance, often leading to a better overall model performance (lower Mean Squared Error) on unseen data.
- **Requires Feature Scaling:** The L2 penalty is sensitive to the scale of the predictor variables. Variables with larger scales will have disproportionately larger penalties. Therefore, it's crucial to **standardize** (e.g., using StandardScaler: mean=0, std=1) or **normalize** features before applying Ridge Regression.

5. Tuning the Hyperparameter (α / λ)

- The optimal value of α is problem-dependent and usually found using **cross-validation**.
- Techniques like Grid Search or Randomized Search are used to test different α values and select the one that yields the best performance (e.g., lowest Mean Squared Error or highest R^2) on validation data.

6. Advantages of Ridge Regression

- Effectively reduces overfitting compared to OLS.
- Handles multicollinearity well, leading to more stable models.
- Improves model generalization on unseen data.
- Computationally efficient (has a closed-form solution, similar to OLS).

7. Disadvantages of Ridge Regression

- Includes all predictors in the final model (no automatic feature selection). If interpretability or a sparse model is needed, Lasso might be preferred.
- Introduces bias into the coefficient estimates.
- Requires tuning the hyperparameter α.
- Performance heavily depends on proper feature scaling.

8. When to Use Ridge Regression

- When you suspect multicollinearity among predictor variables.
- When you want to prevent overfitting in a linear model, especially if you have many predictors.
- When you believe most predictors are relevant to the outcome and don't necessarily want to eliminate any.
- As a baseline regularized regression model.

Relationship Between λ and Coefficients

1. As λ→0:

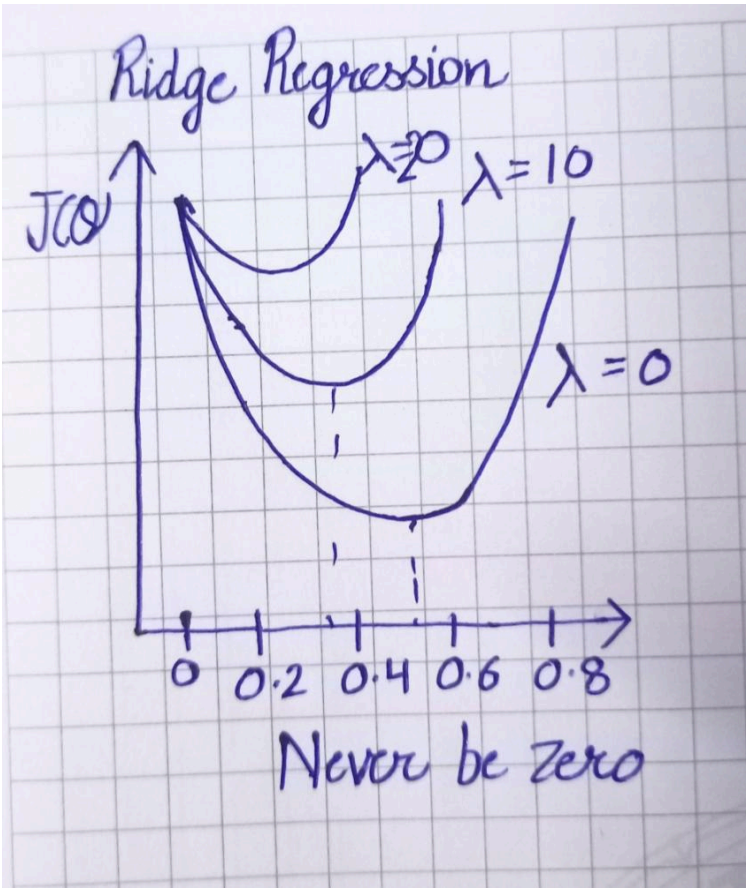
- Ridge regression becomes **ordinary least squares (OLS)**.
- Coefficients are calculated only based on minimizing the **MSE** (no penalty).
- So, slopes can be **large** (especially if features are correlated or data is noisy).

2. As λ increases:

- A **penalty term λ∑βj2** is added to the loss.
- Coefficients are **shrunk toward zero**.
- This leads to **smaller slopes**, reducing model variance and overfitting.

3. As λ→∞:

- The penalty dominates the loss function.
- Coefficients are **forced close to zero**, but not exactly zero (unlike Lasso).
- The model may **underfit** the data.



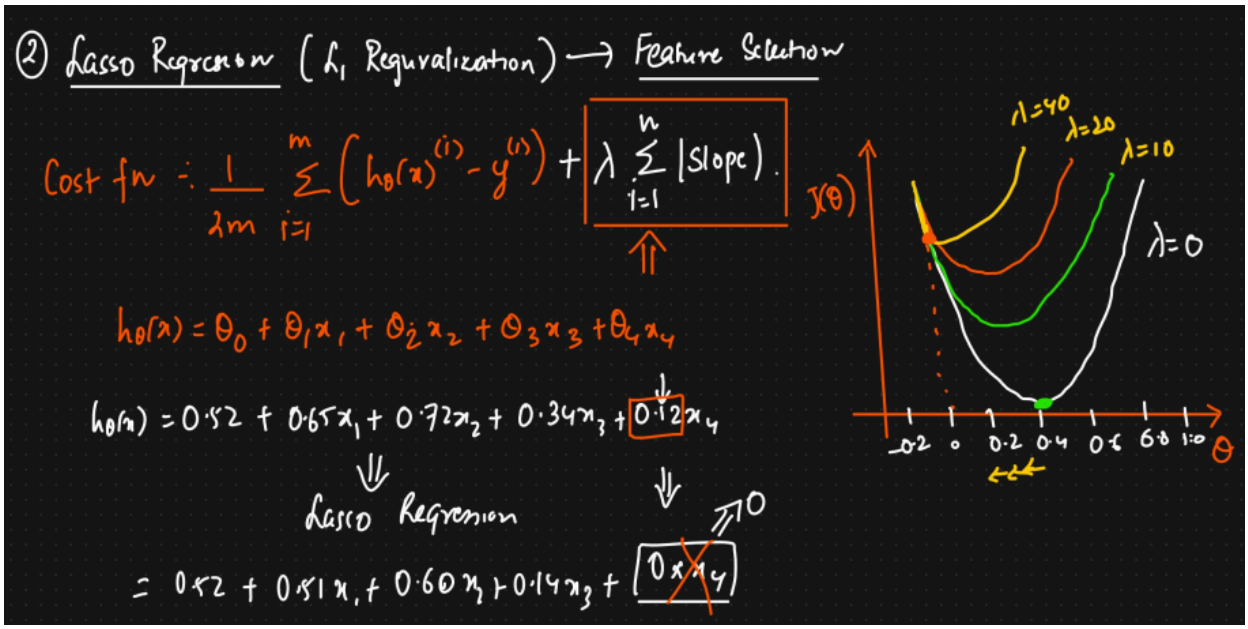
Lasso Regression (L1 Regularization)

- **Type:** Linear Regression Model + Regularization Technique. Lasso stands for **L**east **A**bsolute **S**hrinkage and **S**election **O**perator.
- **Purpose:** Similar to Ridge, it addresses overfitting and can handle multicollinearity. However, its primary distinguishing feature is its ability to perform **automatic feature selection**.
- **Mechanism:** It adds a penalty term to the OLS cost function based on the **sum of the absolute values of the coefficients**.

2. How Lasso Regression Works

- **Lasso Cost Function:** Lasso adds an **L1 Regularization** penalty term to the OLS cost function: $\text{Cost(Lasso)} = \Sigma(y_i - \hat{y}_i)^2 + \alpha * \Sigma|\beta_j|$ (summation for j is from 1 to p)
- **Components:**
    - $\Sigma(y_i - \hat{y}_i)^2$: The standard OLS term (measures goodness of fit).
    - α: The non-negative **tuning parameter** (lambda, λ) controlling the penalty strength.

        - $\alpha = 0$: Lasso becomes OLS.
        - $\alpha > 0$: The penalty is active. As α increases, coefficients are shrunk towards zero, and some are forced to be *exactly* zero.
    - $\Sigma|\beta_j|$: The **L1 penalty** (sum of absolute values of the coefficients). It penalizes the sum of coefficients' magnitudes.
- **Objective:** Minimize the combined cost function. The L1 penalty's nature (using absolute values) leads to solutions where some coefficients become precisely zero.



3. Key Characteristics of Lasso Regression

- **Automatic Feature Selection:** By forcing some coefficients to exactly zero, Lasso effectively removes irrelevant features from the model, creating a **sparse model**.
- **Coefficient Shrinkage:** Like Ridge, Lasso shrinks coefficients towards zero, but the L1 penalty allows some to reach zero.
- **Handling Multicollinearity:** When faced with a group of highly correlated predictors, Lasso tends to arbitrarily select one (or a few) predictor(s) from the group and shrink the coefficients of the others to zero. This behavior can be unstable.
- **Bias-Variance Trade-off:** Introduces bias (like Ridge) but can significantly reduce variance, partly by reducing model complexity via feature selection.
- **Requires Feature Scaling:** Like Ridge, the L1 penalty is sensitive to feature scales. **Standardization** or **normalization** is crucial before applying Lasso.

4. Tuning the Hyperparameter (α / λ)

- The optimal α is found using **cross-validation** (e.g., Grid Search, Randomized Search) to minimize prediction error on validation data.

5. Advantages of Lasso Regression

- Performs automatic feature selection, leading to simpler and more interpretable models.
- Effective in high-dimensional settings (where predictors p > observations n), as it can produce a sparse solution.
- Reduces overfitting.

6. Disadvantages of Lasso Regression

- **Instability with Correlated Predictors:** The selection among highly correlated predictors can be arbitrary and unstable.
- **Limited Predictor Selection:** If there are k highly correlated predictors, Lasso might arbitrarily pick only one and zero out others. In some cases, it tends not to select more predictors than the number of samples (n).
- Can exhibit high variance in the presence of strong multicollinearity (though often less than OLS).
- Introduces bias into the estimates.
- Requires tuning α.

7. When to Use Lasso Regression

- When you suspect many features are irrelevant and want a simpler, sparser model.
- When dealing with very high-dimensional data (e.g., genomics, text analysis).
- When interpretability (knowing which features are most important) is a key goal.

Elastic Net Regression

1. What is Elastic Net Regression?

- **Type:** Linear Regression Model + Regularization Technique.
- **Purpose:** To combine the strengths of both Ridge (L2 penalty) and Lasso (L1 penalty) while mitigating their weaknesses. Specifically, it aims to handle highly correlated predictors more effectively than Lasso while still performing feature selection.
- **Mechanism:** Adds a penalty term that is a *mix* of the L1 and L2 penalties to the OLS cost function.

2. How Elastic Net Regression Works

- **Elastic Net Cost Function:** $\text{Cost(ElasticNet)} = \Sigma(y_i - \hat{y}_i)^2 + \alpha * [\ \rho * \Sigma|\beta_j| + (1 - \rho)/2 * \Sigma(\beta_j)^2\ ]$ *(Note: Some formulations use different parameterizations like λ_1 and λ_2 directly for L1 and L2 penalties, but the α and ρ (or $l1\_ratio$) version is common in libraries like scikit-learn).*
- **Components:**

    - $\Sigma(y_i - \hat{y}_i)^2$: The standard OLS term.
    - α: The **overall penalty strength** parameter (non-negative). Controls the total amount of regularization.
    - ρ (rho): The **mixing parameter** (often called l1_ratio in libraries), where $0 \le \rho \le 1$. It controls the balance between L1 and L2 penalties.
        - $\rho = 0$: Elastic Net becomes Ridge Regression.
        - $\rho = 1$: Elastic Net becomes Lasso Regression.
        - $0 < \rho < 1$: A combination of L1 and L2 penalties.
    - $\Sigma|\beta_j|$: The L1 penalty component (encourages sparsity).
    - $\Sigma(\beta_j)^2$: The L2 penalty component (encourages smaller coefficients and handles correlations).
- **Objective:** Minimize this combined cost function. The combination allows Elastic Net to shrink coefficients and perform feature selection while grouping and handling correlated predictors more stably than Lasso.

$$\text{Cost Function (Elastic Net)} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \left[\alpha \sum |\beta_j| + (1-\alpha)\sum \beta_j^2\right]$$

- λ: overall regularization strength
- $\alpha \in [0, 1]$: mixing parameter
    - $\alpha = 1$: Lasso
    - $\alpha = 0$: Ridge
    - $0 < \alpha < 1$: Elastic Net

3. Key Characteristics of Elastic Net Regression

- **Combines L1 and L2 Regularization:** Gets benefits from both penalties.
- **Feature Selection:** Like Lasso, it can produce sparse models by setting some coefficients to zero.
- **Handles Correlated Predictors:** Behaves like Ridge with groups of correlated predictors – it tends to select or drop them together rather than arbitrarily picking one like Lasso. This is known as the "grouping effect".
- **Overcomes Lasso's Limitations:** Can select more than n predictors when $p > n$. More stable than Lasso when predictors are highly correlated.
- **Requires Feature Scaling:** Absolutely necessary due to the scale sensitivity of both L1 and L2 penalties. Standardize or normalize features first.

4. Tuning Hyperparameters (α and ρ / l1_ratio)

- Requires tuning **two** hyperparameters: α (overall strength) and ρ (the mix ratio).

- This is typically done using **cross-validation**, often with a grid search over combinations of α and ρ values.

5. Advantages of Elastic Net Regression

- Often performs better than Lasso when predictors are highly correlated.
- Combines feature selection with the stability of Ridge regression.
- Robust choice when dealing with high-dimensional data and potential multicollinearity.
- Can select groups of correlated features.

6. Disadvantages of Elastic Net Regression

- Has two hyperparameters to tune, making the tuning process more complex and computationally expensive than Ridge or Lasso.
- Model interpretation can be slightly less straightforward than pure Lasso (due to the L2 component).

7. When to Use Elastic Net Regression

- When dealing with datasets with a large number of predictors ($p > n$).
- When predictors are known or suspected to be highly correlated.
- When you want feature selection but Lasso performs poorly or unstably due to correlations.
- As a general robust alternative if you are unsure whether Ridge or Lasso is optimal (tuning ρ allows the model to find the best mix).