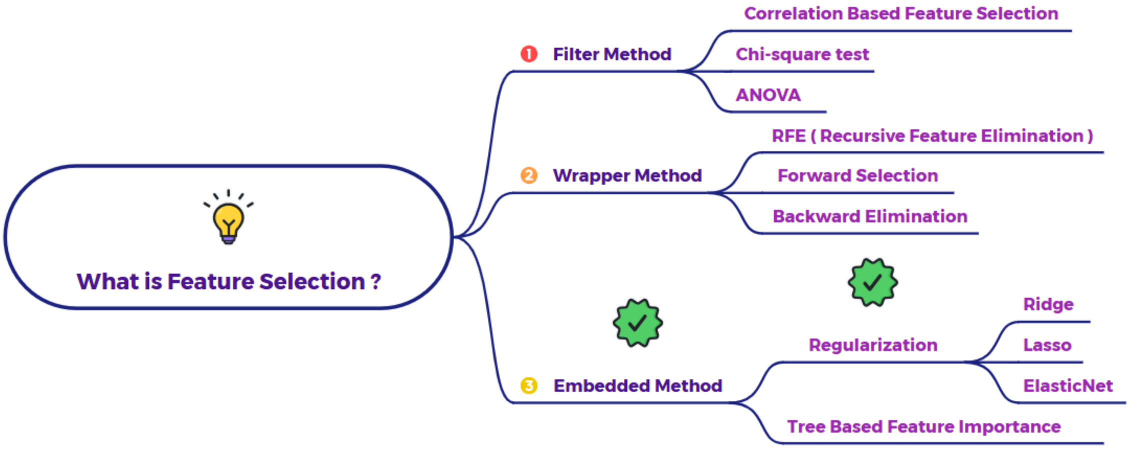
Explain Regularization based feature selection



Embedded Methods - Regularization (L1, L2, L1+L2) for feature selection

Unlike filter and wrapper methods, embedded methods perform feature selection as an integral part of the model training process itself. Regularization techniques, particularly L1 and L1+L2, have a built-in mechanism to shrink the coefficients of less important features, effectively driving some of them to zero, thus performing feature selection.

Core Idea:

Regularization adds a penalty term to the model's loss function during training. This penalty discourages overly complex models and can shrink the coefficients of features. L1 (Lasso) regularization is particularly effective for feature selection as it can drive the coefficients of irrelevant features to exactly zero, effectively removing them from the model.[1] L2 (Ridge) regularization, on the other hand, shrinks coefficients towards zero but rarely makes them exactly zero, so it's primarily used for preventing multicollinearity and overfitting rather than strict feature selection. L1+L2 (Elastic Net) combines the benefits of both.

1. L1 Regularization (Lasso):

- **Penalty Term:** The L1 penalty adds the absolute value of the coefficients to the loss function, multiplied by a regularization parameter (α or λ).

$$Loss_{Lasso} = Loss_{Original} + \alpha \sum_{i=1}^{n} |w_i|$$

- **Feature Selection Mechanism:** The nature of the absolute value penalty encourages some coefficients to become exactly zero. This means the corresponding features have no contribution to the model's prediction and are effectively selected out. The strength of the regularization (and thus the number of features driven to zero) is controlled by the α parameter. A higher α leads to more coefficients becoming zero (more aggressive feature selection).

2. **L2 Regularization (Ridge):**

- **Penalty Term:** The L2 penalty adds the squared magnitude of the coefficients to the loss function, multiplied by a regularization parameter (α or λ.

$$Loss_{Ridge} = Loss_{Original} + \alpha \sum_{i=1}^{n} w_i^2$$

- **Feature Selection Mechanism:** L2 regularization shrinks the coefficients towards zero, but it rarely makes them exactly zero. All features tend to retain some small weight. Therefore, L2 regularization is primarily used for reducing the impact of less important features and preventing multicollinearity, rather than completely eliminating features.

3. **L1+L2 Regularization (Elastic Net):**

- **Penalty Term:** Elastic Net combines both L1 and L2 penalties with a mixing parameter (ρ) that controls the balance between them.

$$Loss_{ElasticNet} = Loss_{Original} + \alpha\rho \sum_{i=1}^{n} |w_i| + \alpha(1-\rho) \sum_{i=1}^{n} w_i^2$$

- **Feature Selection Mechanism:** The L1 part of the penalty encourages sparsity (driving coefficients to zero for feature selection), while the L2 part helps with the limitations of L1 in the presence of highly correlated features (it tends to select groups of correlated features together). The α parameter controls the overall strength of regularization, and ρ controls the mix between L1 and L2.

Example: Predicting House Prices (Regression)

Let's say we want to predict the price of a house (numerical target) using a Linear Regression model with the following numerical features:

- Size of the house (in square feet)
- Number of bedrooms
- Number of bathrooms
- Age of the house (in years)
- Location score (numerical score)
- Distance to the nearest park
- Number of nearby restaurants

- Air conditioning (1 if yes, 0 if no)
- Swimming pool (1 if yes, 0 if no)
- Material of kitchen countertops (encoded numerically)

Applying L1 Regularization (Lasso):

1. **Train a Lasso Regression Model:** We train a Linear Regression model with an L1 penalty on our loss function. We also need to tune the regularization parameter α (using techniques like cross-validation) to find a value that balances model performance and sparsity.
2. **Analyze the Coefficients:** After training, we examine the learned coefficients for each feature. Features that are deemed less important by the Lasso algorithm will have their coefficients driven to exactly zero.

Let's say after training with an optimal α, the coefficients are:

  - Size of the house: 120 () ∗Numberofbedrooms: 5000()
  - Number of bathrooms: 8000 ()∗Ageofthehouse : − 200()
  - Location score: 15000 ()∗Distancetothenearestpark: − 50()
  - Number of nearby restaurants: 0 ()<−Coefficientiszero(Featureselectedout)∗Airconditioning:3000()
  - Swimming pool: 10000 ()∗Materialofkitchencountertops:0() <- Coefficient is zero (Feature selected out)

3. **Selected Features:** Based on the Lasso model, the features with non-zero coefficients are considered selected: Size of the house, Number of bedrooms, Number of bathrooms, Age of the house, Location score, Distance to the nearest park, Air conditioning, and Swimming pool. The features "Number of nearby restaurants" and "Material of kitchen countertops" have been effectively eliminated by having their coefficients set to zero.

Applying L2 Regularization (Ridge):

If we had used L2 regularization instead, the coefficients of all features would likely be non-zero, although the less important features would have coefficients closer to zero. Ridge regression helps in reducing the impact of less important features but doesn't perform explicit feature selection by setting coefficients to zero.

Applying L1+L2 Regularization (Elastic Net):

Elastic Net would provide a balance. It might drive some coefficients to zero (like L1) while also shrinking the coefficients of correlated features (like L2). The specific features selected would depend on the values of α and ρ tuned during the training process.

Advantages of Regularization for Feature Selection (Embedded Method):

- **Integrated into Model Training:** Feature selection happens directly during model training, making it computationally efficient compared to wrapper methods.
- **Considers Feature Interactions (to some extent):** The model learns the weights of features in the context of all other features, implicitly considering some level of interaction.
- **Often Leads to Good Generalization:** By penalizing complexity and shrinking coefficients, regularization helps prevent overfitting, leading to better performance on unseen data.

Disadvantages of Regularization for Feature Selection (Embedded Method):

- **Model Dependent:** The selected features are specific to the type of regularized model used (e.g., Lasso for linear models).
- **Requires Tuning of Hyperparameters:** The regularization parameter (α and ρ in Elastic Net) needs to be carefully tuned using techniques like cross-validation, which adds a computational cost.
- **Black Box Nature:** The feature selection process is embedded within the model training, which can sometimes make it less transparent compared to filter methods.

In summary, L1 (Lasso) and L1+L2 (Elastic Net) regularization are powerful embedded methods for feature selection. By adding a penalty to the loss function, they can drive the coefficients of irrelevant features to zero, effectively selecting a subset of the most important features directly during model training. L2 (Ridge) primarily helps with regularization and multicollinearity but doesn't typically perform explicit feature selection by setting coefficients to zero.