JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

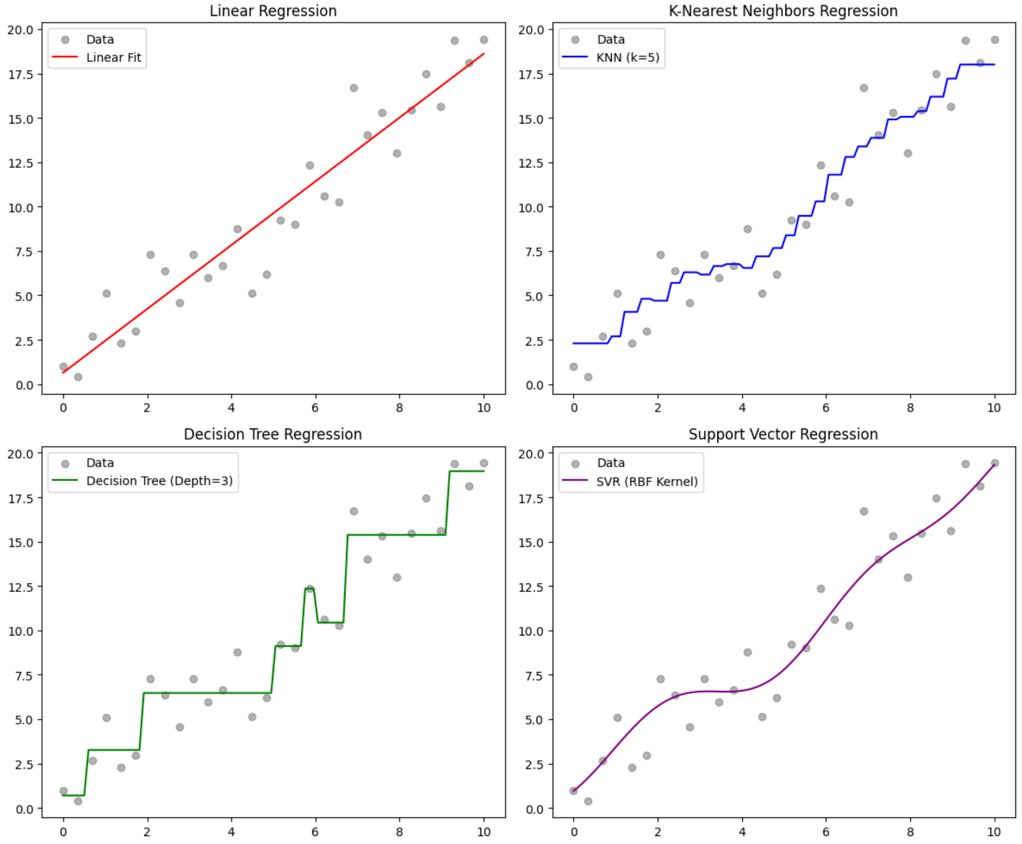
685.621 Algorithms for Data Science

Supervised Learning: Regression Algorithms

How Regression Predicts Values

Regression Types

- **Linear Regression**
  - Mathematical Equation
- **K-Nearest Neighbors**
  - Influenced by nearby points
- **Decision Tree**
  - Rule Based
- **Support Vector**
  - Hyperplane within a margin of error



How Regression Models Learn

Algorithm	Key Characteristics
Linear Regression	Finds best-fit line by minimizing squared errors.
K-Nearest Neighbors (KNN) Regression	Stores data and predicts based on nearest neighbors
Decision Tree Regression	Recursively splits data into segments to minimize variance
Support Vector Regression	Finds a function that fits most data within a margin
Ridge/Lasso Regression	Modifies Linear Regression by penalizing large coefficients

Choosing the Right Model

Algorithm	Type	Key Characteristics
Linear Regression	Linear	Simple, interpretable, assumes linearity
KNN Regression	Instance-based	No training phase, works well for small datasets
Decision Trees Regression	Rule-based	Captures nonlinear relationships, ensemble methods for robustness
Support Vector Regression	Hyperplane-based	Handles outliers, defines optimal margin for predictions
Ridge & Lasso Regression	Linear	Regularization techniques to prevent overfitting

Linear Regression

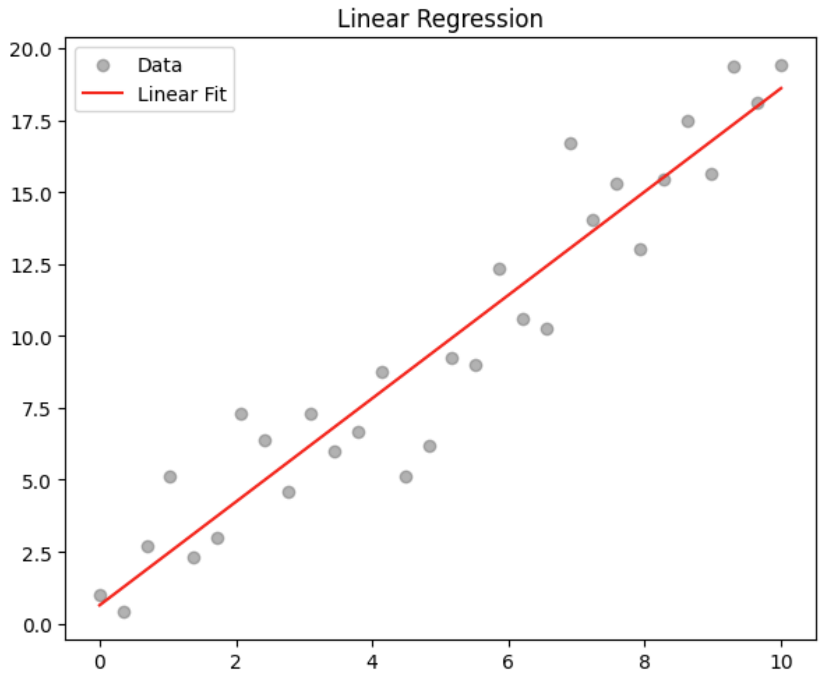
$$\hat{y} = w^T x + b$$

Advantages

- **Interpretable**
- Works well when data has **linear relationship**
- **Efficient** & scalable

Limitations

- Assumes **linearity** and **independence**
- **Sensitive** to outliers
- Can **underfit** complex relationships
- **Cannot model interactions** unless explicitly added

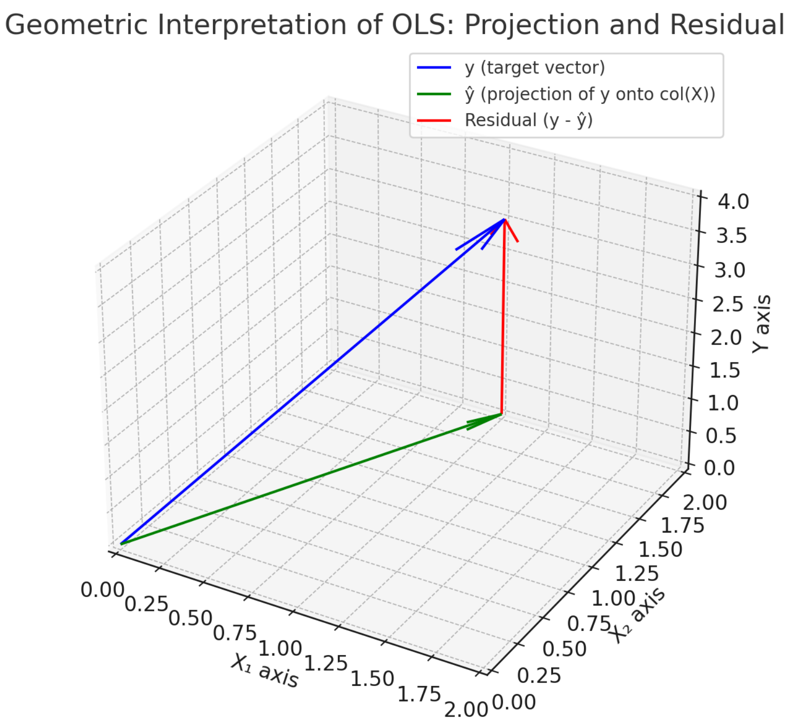


Ordinary Least Squares (OLS)

Ordinary Least Squares (OLS) minimizes the sum of squared residuals to find the best linear model

$$y = \boldsymbol{X}\beta + \varepsilon \quad \beta = (X^T X)^{-1} X^T y$$

It assumes a linear relationship, projects the target onto the featu space, and provides the most unbiased linear estimator under Gaussian noise.

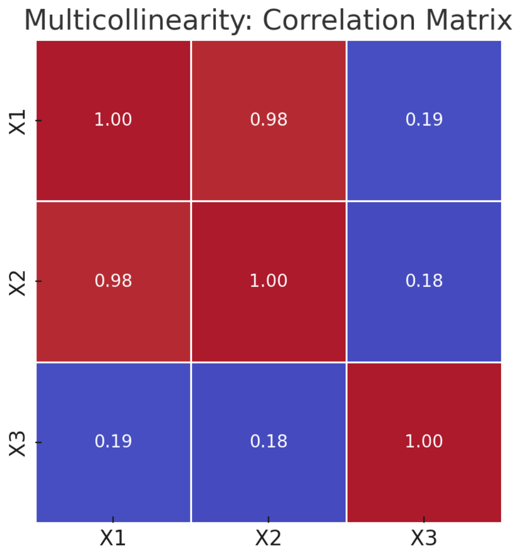


Multicollinearity

Occurs when two or more independent variables in a regression model are highly correlated.

- **Why is it a Problem?**
  - Makes it difficult to determine the individual effect of each variable.
  - Leads to unstable coefficients
  - Reduces interpretability of the model
- **How to Detect it:**
  - Variance Inflation Factor (VIF)
  - Correlation Matrix

$$VIF(X_i) = \frac{1}{1 - R_i^2}$$



Dealing with Multicollinearity

- **VIF-Based Feature Selection**
  - If two features have a high VIF (>10) remove one.
- **Principal Component Analysis (PCA)**
  - Transform correlated features into independent principal components
- **Ridge Regression (L2 Regularization)**
  - Reduces the impact of multicollinearity by shrinking coefficients
- **Lasso Regression (L1 Regularization)**
  - Can set some coefficients to zero, performing automatic feature selection

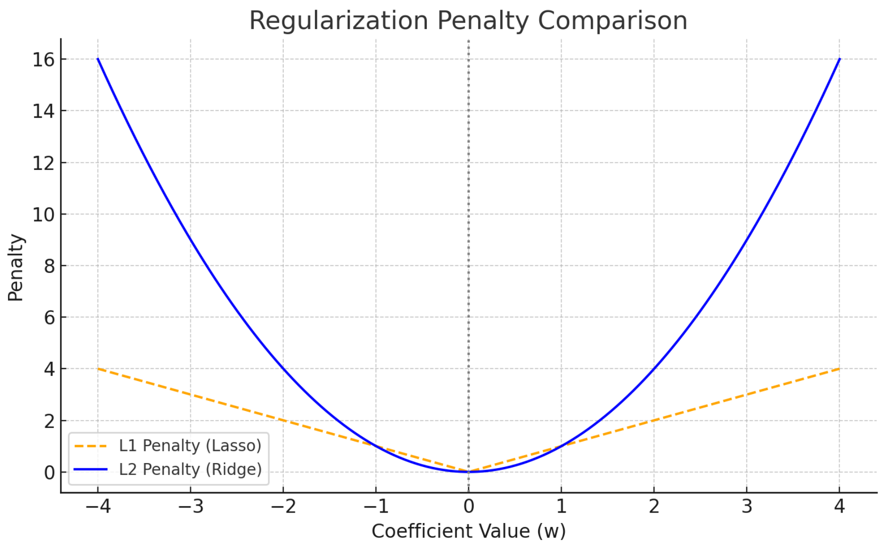
Regularization for Linear Regression

Penalty on absolute values of coefficients

$$\widehat{\beta}_{\text{lasso}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \beta \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

Penalty on squared coefficients

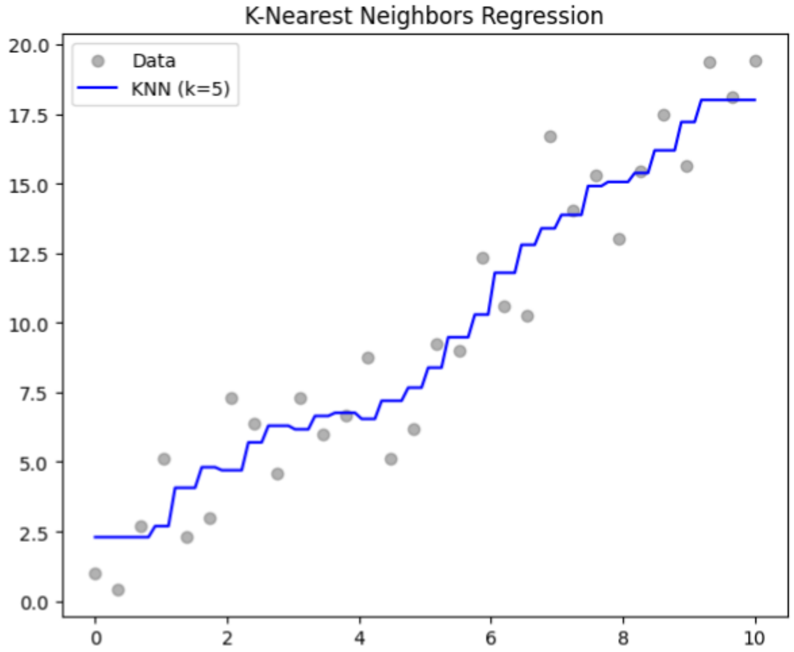
$$\widehat{\beta}_{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \beta \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$



K-Nearest Neighbors (KNN) Regression

$$d(x, x') = \sqrt{\sum_{i=1}^{n} (x_i - x'_i)^2}$$

$$\hat{y} = \frac{1}{k} \sum_{i \in neighbors} y_i$$



Advantages

- **Easy** to implement and **understand**
- Captures local patterns and **non-linearities**
- Naturally handles **multi-modal distributions**

Limitations

- **Computationally expensive** at prediction time
- **Poor performance** in higher dimensions
- Choice of **k matters**
- **Sparse data** is a problem

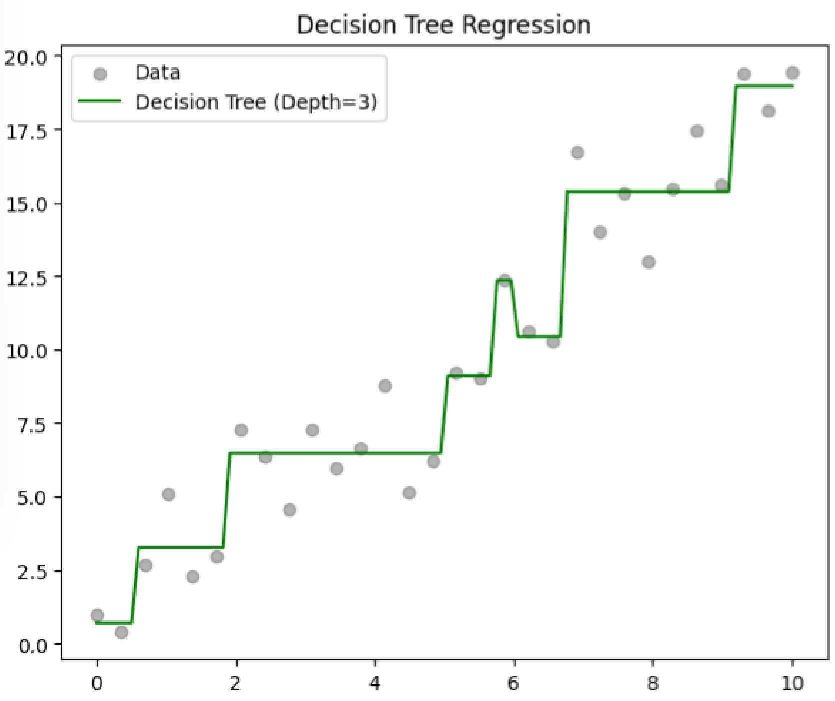
Rule-Based Regression

Advantages

- Captures **nonlinear and interaction** effects
- **Easy** to visualize and **interpret**
- **No scaling** and handles missing values
- **Handles** both **categorical** and **numerical**

Limitations

- **Prone to overfitting –** deep trees memorize
- **Unstable–** Small changes in data can produce very different trees
- May create biased splits with **imbalanced** target distributions



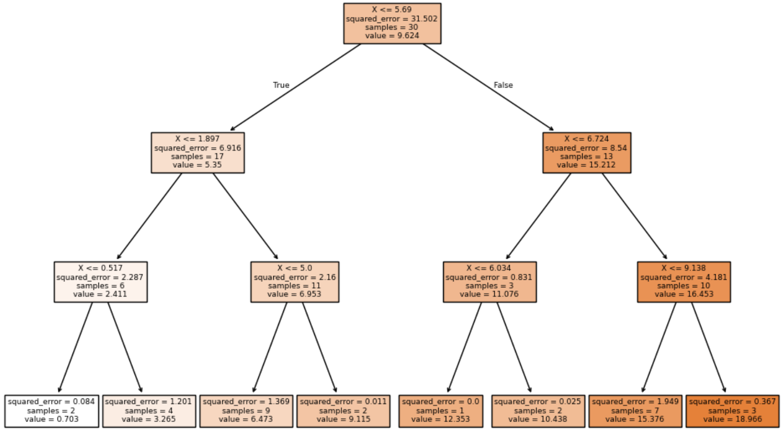
The Power of Ensembles

Advantages

- **Reduces overfitting –** More stable than a single Decision Tree.
- **Handles high-dimensional data well –** Works even when many features exist.
- **Works with missing data –** Can still make predictions even if some values are missing.

Limitations

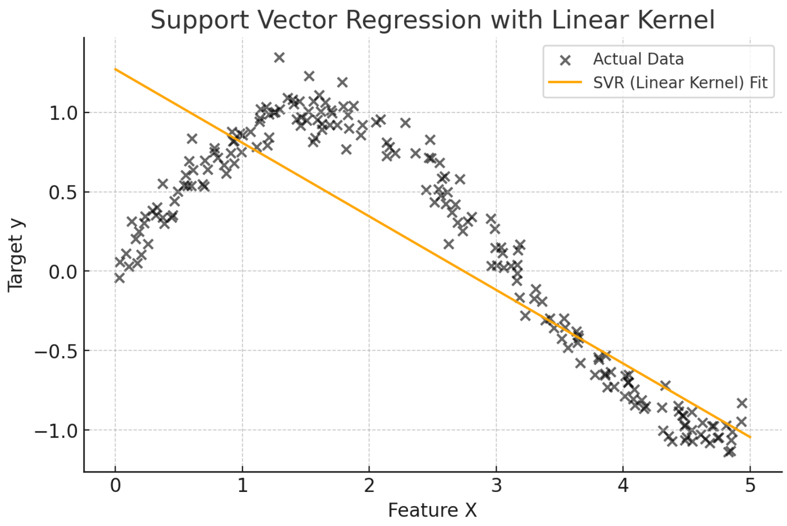
- **Less interpretable –** Unlike a single Decision Tree, it’s hard to visualize.
- **Computationally expensive –** Training multiple trees takes more time than a single model.
- **May not work well for small datasets –** Too many trees can lead to unnecessary complexity.

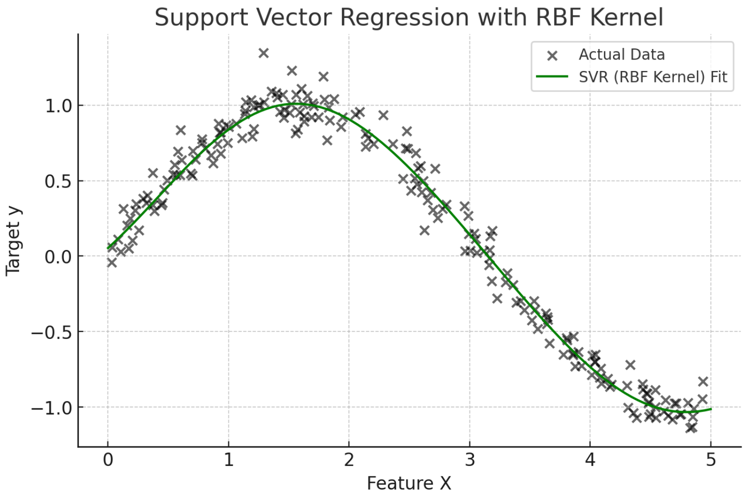


Support Vector Regression (SVR)

Soft Margin $$\min_{\mathbf{w}, b, \xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \xi_i,$$

Function $$\hat{y} = \sum_{i=1}^{N} \left(\alpha_i - \alpha_i^*\right) K\left(\mathbf{x}_i, \mathbf{x}\right) + b.$$





JOHNS HOPKINS

WHITING SCHOOL *of* ENGINEERING