

# Machine Learning in Python

## Supervised Learning - Regression and Evaluation

Cristian A. Marocico, A. Emin Tatar

Center for Information Technology  
University of Groningen

Wednesday, July 2<sup>nd</sup> 2025

# Outline

- 1 Introduction to Regression
- 2 Simple Linear Regression
- 3 Evaluation Metrics for Regression
- 4 Robust Regression
- 5 Multiple Linear Regression
- 6 Regularized Regression
- 7 Nonlinearities in Regression
- 8 Overfitting, Underfitting, and Hyperparameter Tuning

# Introduction to Regression

Regression

Definition

# Introduction to Regression

## Regression

### Definition

**Regression** is a statistical method used to model the relationship between a dependent variable and one or more independent variables.

# Simple Linear Regression

## Simple Linear Regression

Definition

# Simple Linear Regression

## Simple Linear Regression

### Definition

**Simple Linear Regression** is a method to model the relationship between two variables by fitting a linear equation to observed data.

# Simple Linear Regression

## Simple Linear Regression

### Definition

**Simple Linear Regression** is a method to model the relationship between two variables by fitting a linear equation to observed data. Mathematically:

$$y = \beta_0 + \beta_1 x + \epsilon$$

where:

- $y$  is the dependent variable (response).
- $x$  is the independent variable (predictor).
- $\beta_0$  is the y-intercept (constant term).
- $\beta_1$  is the slope of the line (coefficient).
- $\epsilon$  is the error term (residuals).

# Simple Linear Regression

A Simple Linear Regression Machine Learning model will learn the coefficients  $\beta_0$  and  $\beta_1$  from the training data to minimize the difference between the predicted values and the actual values.



# Assumptions of Simple Linear Regression

- Linearity: The relationship between the independent and dependent variable is linear.

# Assumptions of Simple Linear Regression

- Linearity: The relationship between the independent and dependent variable is linear.
- Independence: Observations are independent of each other.

# Assumptions of Simple Linear Regression

- Linearity: The relationship between the independent and dependent variable is linear.
- Independence: Observations are independent of each other.
- Homoscedasticity: Constant variance of the error terms.

# Assumptions of Simple Linear Regression

- Linearity: The relationship between the independent and dependent variable is linear.
- Independence: Observations are independent of each other.
- Homoscedasticity: Constant variance of the error terms.
- Normality: The residuals (errors) of the model are normally distributed.

# Evaluation Metrics for Regression

## Common Metrics for Regression

### Definition

# Evaluation Metrics for Regression

## Common Metrics for Regression

Definition

Common metrics to evaluate regression models include:

- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- R-squared ( $R^2$ )
- Adjusted R-squared

# Mean Absolute Error (MAE)

## Mean Absolute Error (MAE)

### Definition

# Mean Absolute Error (MAE)

## Mean Absolute Error (MAE)

Definition

**Mean Absolute Error (MAE)** is the average of the absolute differences between the predicted and the actual values:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$



# Mean Absolute Error (MAE)

## Mean Absolute Error (MAE)

Definition

**Mean Absolute Error (MAE)** is the average of the absolute differences between the predicted and the actual values:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

where:

- $n$  is the number of observations.
- $y_i$  is the actual value.
- $\hat{y}_i$  is the predicted value.

# Mean Absolute Error (MAE)

## Mean Absolute Error (MAE)

Definition

**Mean Absolute Error (MAE)** is the average of the absolute differences between the predicted and the actual values:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

where:

- $n$  is the number of observations.
- $y_i$  is the actual value.
- $\hat{y}_i$  is the predicted value.

MAE is a linear score, which can be used when all errors are equally important; it is also less sensitive to outliers compared to MSE.

# Mean Squared Error (MSE)

## Mean Squared Error (MSE)

### Definition

# Mean Squared Error (MSE)

## Mean Squared Error (MSE)

Definition

**Mean Squared Error (MSE)** is the average of the squared differences between the predicted and the actual values:

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

# Mean Squared Error (MSE)

## Mean Squared Error (MSE)

Definition

**Mean Squared Error (MSE)** is the average of the squared differences between the predicted and the actual values:

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

where:

- $n$  is the number of observations.
- $y_i$  is the actual value.
- $\hat{y}_i$  is the predicted value.

# Mean Squared Error (MSE)

## Mean Squared Error (MSE)

Definition

**Mean Squared Error (MSE)** is the average of the squared differences between the predicted and the actual values:

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

where:

- $n$  is the number of observations.
- $y_i$  is the actual value.
- $\hat{y}_i$  is the predicted value.

MSE is more sensitive to outliers than MAE because it squares the errors, which can disproportionately affect the metric if there are large errors; however, it is useful when larger errors are more significant.

# Root Mean Squared Error (RMSE)

## Root Mean Squared Error (RMSE)

Definition

# Root Mean Squared Error (RMSE)

## Root Mean Squared Error (RMSE)

Definition

**Root Mean Squared Error (RMSE)** is the square root of the average of the squared differences between the predicted and the actual values:

$$\text{RMSE} = \sqrt{\text{MSE}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$



# Root Mean Squared Error (RMSE)

## Root Mean Squared Error (RMSE)

Definition

**Root Mean Squared Error (RMSE)** is the square root of the average of the squared differences between the predicted and the actual values:

$$\text{RMSE} = \sqrt{\text{MSE}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

where:

- $n$  is the number of observations.
- $y_i$  is the actual value.
- $\hat{y}_i$  is the predicted value.

# Root Mean Squared Error (RMSE)

## Root Mean Squared Error (RMSE)

Definition

**Root Mean Squared Error (RMSE)** is the square root of the average of the squared differences between the predicted and the actual values:

$$\text{RMSE} = \sqrt{\text{MSE}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

where:

- $n$  is the number of observations.
- $y_i$  is the actual value.
- $\hat{y}_i$  is the predicted value.

RMSE is in the same units as the dependent variable, making it interpretable; it is also sensitive to outliers, similar to MSE.

# R-squared ( $R^2$ )

## R-squared ( $R^2$ )

### Definition

# R-squared ( $R^2$ )

## R-squared ( $R^2$ )

### Definition

**R-squared ( $R^2$ )** is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

where:

- $SS_{\text{res}}$  is the sum of squares of residuals (errors).
- $SS_{\text{tot}}$  is the total sum of squares (variance of the dependent variable).

# R-squared ( $R^2$ )

## R-squared ( $R^2$ )

### Definition

**R-squared ( $R^2$ )** is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

where:

- $SS_{\text{res}}$  is the sum of squares of residuals (errors).
- $SS_{\text{tot}}$  is the total sum of squares (variance of the dependent variable).

$R^2$  values range from 0 to 1, where:

- 0 indicates that the model does not explain any of the variability of the response data around its mean.
- 1 indicates that the model explains all the variability of the response data around its mean.

# Adjusted R-squared

Adjusted R-squared

Definition

# Adjusted R-squared

## Adjusted R-squared

### Definition

**Adjusted R-squared** adjusts the  $R^2$  value for the number of predictors in the model, providing a more accurate measure when comparing models with different numbers of predictors:

$$\text{Adjusted } R^2 = 1 - (1 - R^2) \frac{n - 1}{n - p - 1}$$

where:

- $n$  is the number of observations.
- $p$  is the number of predictors in the model.

# Adjusted R-squared

## Adjusted R-squared

### Definition

**Adjusted R-squared** adjusts the  $R^2$  value for the number of predictors in the model, providing a more accurate measure when comparing models with different numbers of predictors:

$$\text{Adjusted } R^2 = 1 - (1 - R^2) \frac{n - 1}{n - p - 1}$$

where:

- $n$  is the number of observations.
- $p$  is the number of predictors in the model.

Adjusted  $R^2$  can be negative, which indicates that the model is worse than a horizontal line (mean of the dependent variable); it is useful for comparing models with different numbers of predictors.



# Robust Regression

Robust Regression

Definition

# Robust Regression

## Robust Regression

### Definition

**Robust Regression** is a type of regression analysis designed to be less sensitive to outliers in the data. It provides a more reliable estimate of the relationship between variables when the data contains outliers or violations of assumptions.

# Types of Robust Regression

## Types of Robust Regression

### Definition

There are several types of robust regression techniques, including:

# Types of Robust Regression

## Types of Robust Regression

Definition

There are several types of robust regression techniques, including:

- Huber Regression

# Types of Robust Regression

## Types of Robust Regression

### Definition

There are several types of robust regression techniques, including:

- Huber Regression
- RANSAC (RANDOM Sample Consensus)

# Types of Robust Regression

## Types of Robust Regression

### Definition

There are several types of robust regression techniques, including:

- Huber Regression
- RANSAC (RANDOM Sample Consensus)
- Least Trimmed Squares (LTS)

# Types of Robust Regression

## Types of Robust Regression

### Definition

There are several types of robust regression techniques, including:

- Huber Regression
- RANSAC (RANDOM Sample Consensus)
- Least Trimmed Squares (LTS)
- Theil-Sen Estimator

# Types of Robust Regression

## Types of Robust Regression

### Definition

There are several types of robust regression techniques, including:

- Huber Regression
- RANSAC (RANDOM Sample Consensus)
- Least Trimmed Squares (LTS)
- Theil-Sen Estimator
- Quantile Regression



# Huber Regression

## Huber Regression

### Definition

# Huber Regression

## Huber Regression

### Definition

**Huber Regression** is a robust regression technique that uses a loss function that is quadratic for small residuals and linear for large residuals. This makes it less sensitive to outliers compared to traditional least squares regression.

# Huber Regression

## Huber Regression

### Definition

**Huber Regression** is a robust regression technique that uses a loss function that is quadratic for small residuals and linear for large residuals. This makes it less sensitive to outliers compared to traditional least squares regression. Mathematically, the Huber loss function is defined as:

$$L(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2 & \text{if } |y - \hat{y}| \leq \delta \\ \delta|y - \hat{y}| - \frac{1}{2}\delta^2 & \text{if } |y - \hat{y}| > \delta \end{cases}$$

where  $\delta$  is a threshold that determines the point at which the loss function transitions from quadratic to linear.

# RANSAC (RANdom SAmple Consensus)

## RANSAC (RANdom SAmple Consensus)

Definition

# RANSAC (RANDOM SAMPLE CONSENSUS)

## RANSAC (RANDOM SAMPLE CONSENSUS)

### Definition

**RANSAC (RANDOM SAMPLE CONSENSUS)** is an iterative method used to estimate parameters of a mathematical model from a dataset that contains outliers. It works by randomly selecting a subset of the data, fitting a model to this subset, and then determining how many points from the entire dataset fit this model well.

# RANSAC (RANDOM Sample Consensus)

## RANSAC (RANDOM Sample Consensus)

### Definition

**RANSAC (RANDOM Sample Consensus)** is an iterative method used to estimate parameters of a mathematical model from a dataset that contains outliers. It works by randomly selecting a subset of the data, fitting a model to this subset, and then determining how many points from the entire dataset fit this model well. The RANSAC algorithm consists of the following steps:

- 1 Randomly select a subset of the data points.
- 2 Fit a model to this subset.
- 3 Determine the inliers (points that fit the model well) and outliers (points that do not fit the model).
- 4 Repeat steps 1-3 for a specified number of iterations or until a satisfactory model is found.
- 5 Select the model with the highest number of inliers as the final model.

# Multiple Linear Regression

## Multiple Linear Regression

Definition

# Multiple Linear Regression

## Multiple Linear Regression

Definition

**Multiple Linear Regression** is an extension of simple linear regression that models the relationship between a dependent variable and multiple independent variables. It is used when there are two or more predictors.



# Multiple Linear Regression

## Multiple Linear Regression

### Definition

**Multiple Linear Regression** is an extension of simple linear regression that models the relationship between a dependent variable and multiple independent variables. It is used when there are two or more predictors. Mathematically, it is represented as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + \epsilon$$

where:

- $y$  is the dependent variable.
- $x_1, x_2, \dots, x_p$  are the independent variables (predictors).
- $\beta_0$  is the y-intercept (constant term).
- $\beta_1, \beta_2, \dots, \beta_p$  are the coefficients (slopes) for each independent variable.
- $\epsilon$  is the error term (residuals).

# Assumptions of Multiple Linear Regression

## Assumptions of Multiple Linear Regression

### Definition

# Assumptions of Multiple Linear Regression

## Assumptions of Multiple Linear Regression

Definition

The assumptions of multiple linear regression are similar to those of simple linear regression:

- **Linearity:** The relationship between the independent variables and the dependent variable is linear.
- **Independence:** Observations are independent of each other.
- **Homoscedasticity:** Constant variance of the error terms.
- **Normality:** The residuals (errors) of the model are normally distributed.
- **No multicollinearity:** The independent variables are not highly correlated with each other.

# How do Machine Learning Models Learn?

How do Machine Learning Models Learn?

Definition

# How do Machine Learning Models Learn?

## How do Machine Learning Models Learn?

Definition

Machine Learning models learn by adjusting their parameters (coefficients) to minimize the difference between the predicted values and the actual values in the training data, also known as the **cost function** or **loss function**.

# How do Machine Learning Models Learn?

## How do Machine Learning Models Learn?

Definition

Machine Learning models learn by adjusting their parameters (coefficients) to minimize the difference between the predicted values and the actual values in the training data, also known as the **cost function** or **loss function**. This process is often done using optimization algorithms that iteratively update the parameters based on the loss function, which measures the error of the predictions.

# Cost Function

| Cost Function | Definition |
|---------------|------------|
|               |            |

# Cost Function

## Cost Function

### Definition

The **cost function** (or **loss function**) is a mathematical function that quantifies the difference between the predicted values and the actual values in the training data. It is used to guide the learning process of the model.



# Cost Function

## Cost Function

### Definition

The **cost function** (or **loss function**) is a mathematical function that quantifies the difference between the predicted values and the actual values in the training data. It is used to guide the learning process of the model. Common cost functions for regression include:

- Mean Squared Error (MSE)
- Mean Absolute Error (MAE)

# Cost Function: Mean Squared Error (MSE)

## Cost Function: Mean Squared Error (MSE)

Definition

# Cost Function: Mean Squared Error (MSE)

## Cost Function: Mean Squared Error (MSE)

Definition

**Mean Squared Error (MSE)** is a common cost function used in regression tasks. It measures the average of the squares of the errors, which are the differences between the predicted values and the actual values:

$$J_{\text{MSE}}(w) = \frac{1}{2m} \sum_{i=1}^m (f(x_i, w) - y_i)^2$$

where:

- $m$  is the number of observations.
- $f(x_i, w)$  is the predicted value for the  $i$ -th observation given the input features  $x_i$  and model parameters  $w$ .
- $y_i$  is the actual value for the  $i$ -th observation.

# Cost Function: Mean Squared Error (MSE)

## Cost Function: Mean Squared Error (MSE)

Definition

**Mean Squared Error (MSE)** is a common cost function used in regression tasks. It measures the average of the squares of the errors, which are the differences between the predicted values and the actual values:

$$J_{\text{MSE}}(w) = \frac{1}{2m} \sum_{i=1}^m (f(x_i, w) - y_i)^2$$

where:

- $m$  is the number of observations.
- $f(x_i, w)$  is the predicted value for the  $i$ -th observation given the input features  $x_i$  and model parameters  $w$ .
- $y_i$  is the actual value for the  $i$ -th observation.

MSE is sensitive to outliers, and is, therefore, useful when larger errors are more significant.

# Cost Function: Mean Absolute Error (MAE)

## Cost Function: Mean Absolute Error (MAE)

Definition

# Cost Function: Mean Absolute Error (MAE)

## Cost Function: Mean Absolute Error (MAE)

Definition

**Mean Absolute Error (MAE)** is another common cost function used in regression tasks. It measures the average of the absolute differences between the predicted values and the actual values:

$$J_{\text{MAE}}(w) = \frac{1}{m} \sum_{i=1}^m |f(x_i, w) - y_i|$$

where:

- $m$  is the number of observations.
- $f(x_i, w)$  is the predicted value for the  $i$ -th observation given the input features  $x_i$  and model parameters  $w$ .
- $y_i$  is the actual value for the  $i$ -th observation.

# Cost Function: Mean Absolute Error (MAE)

## Cost Function: Mean Absolute Error (MAE)

Definition

**Mean Absolute Error (MAE)** is another common cost function used in regression tasks. It measures the average of the absolute differences between the predicted values and the actual values:

$$J_{\text{MAE}}(w) = \frac{1}{m} \sum_{i=1}^m |f(x_i, w) - y_i|$$

where:

- $m$  is the number of observations.
- $f(x_i, w)$  is the predicted value for the  $i$ -th observation given the input features  $x_i$  and model parameters  $w$ .
- $y_i$  is the actual value for the  $i$ -th observation.

MAE is less sensitive to outliers compared to MSE; useful when all errors are equally important.

# Gradient Descent

## Gradient Descent

## Definition



# Gradient Descent

## Gradient Descent

### Definition

**Gradient Descent** is an optimization algorithm used to minimize the cost function by iteratively updating the model parameters in the direction of the steepest descent of the cost function.

# Gradient Descent

## Gradient Descent

### Definition

**Gradient Descent** is an optimization algorithm used to minimize the cost function by iteratively updating the model parameters in the direction of the steepest descent of the cost function. The update rule for gradient descent is:

$$w := w - \alpha \nabla J(w)$$

where:

- $w$  is the vector of model parameters (coefficients).
- $\alpha$  is the learning rate, which controls the step size of each update.
- $\nabla J(w)$  is the gradient of the cost function with respect to the parameters  $w$ .

# Gradient Descent

## Gradient Descent

## Definition

**Gradient Descent** is an optimization algorithm used to minimize the cost function by iteratively updating the model parameters in the direction of the steepest descent of the cost function. The update rule for gradient descent is:

$$w := w - \alpha \nabla J(w)$$

where:

- $w$  is the vector of model parameters (coefficients).
- $\alpha$  is the learning rate, which controls the step size of each update.
- $\nabla J(w)$  is the gradient of the cost function with respect to the parameters  $w$ .

Gradient descent continues until convergence, which is when the change in the cost function is below a certain threshold or a maximum number of iterations is reached.

# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

The **Gradient Descent Algorithm** consists of the following steps:

- 1 Initialize the model parameters (coefficients) randomly or to zero.

# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

The **Gradient Descent Algorithm** consists of the following steps:

- 1 Initialize the model parameters (coefficients) randomly or to zero.
- 2 Compute the predicted values using the current parameters.

# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

The **Gradient Descent Algorithm** consists of the following steps:

- ➊ Initialize the model parameters (coefficients) randomly or to zero.
- ➋ Compute the predicted values using the current parameters.
- ➌ Calculate the cost function (loss) using the predicted values and actual values.

# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

The **Gradient Descent Algorithm** consists of the following steps:

- 1 Initialize the model parameters (coefficients) randomly or to zero.
- 2 Compute the predicted values using the current parameters.
- 3 Calculate the cost function (loss) using the predicted values and actual values.
- 4 Compute the gradient of the cost function with respect to the parameters.



# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

The **Gradient Descent Algorithm** consists of the following steps:

- 1 Initialize the model parameters (coefficients) randomly or to zero.
- 2 Compute the predicted values using the current parameters.
- 3 Calculate the cost function (loss) using the predicted values and actual values.
- 4 Compute the gradient of the cost function with respect to the parameters.
- 5 Update the parameters using the gradient descent update rule.

# Gradient Descent Algorithm

## Gradient Descent Algorithm

Definition

The **Gradient Descent Algorithm** consists of the following steps:

- 1 Initialize the model parameters (coefficients) randomly or to zero.
- 2 Compute the predicted values using the current parameters.
- 3 Calculate the cost function (loss) using the predicted values and actual values.
- 4 Compute the gradient of the cost function with respect to the parameters.
- 5 Update the parameters using the gradient descent update rule.
- 6 Repeat steps 2-5 until convergence (i.e., the change in the cost function is below a threshold or a maximum number of iterations is reached).

# Overfitting and Underfitting

## Overfitting and Underfitting

Definition

# Overfitting and Underfitting

## Overfitting and Underfitting

### Definition

**Overfitting** occurs when a model learns the noise in the training data instead of the underlying pattern, resulting in poor generalization to unseen data. It typically happens when the model is too complex relative to the amount of training data.

# Overfitting and Underfitting

## Overfitting and Underfitting

### Definition

**Overfitting** occurs when a model learns the noise in the training data instead of the underlying pattern, resulting in poor generalization to unseen data. It typically happens when the model is too complex relative to the amount of training data.

**Underfitting** occurs when a model is too simple to capture the underlying pattern in the data, resulting in poor performance on both training and unseen data. It typically happens when the model is not complex enough relative to the data.

# Overfitting and Underfitting

## Overfitting and Underfitting

### Definition

**Overfitting** occurs when a model learns the noise in the training data instead of the underlying pattern, resulting in poor generalization to unseen data. It typically happens when the model is too complex relative to the amount of training data.

**Underfitting** occurs when a model is too simple to capture the underlying pattern in the data, resulting in poor performance on both training and unseen data. It typically happens when the model is not complex enough relative to the data.

To mitigate overfitting, techniques such as **regularization**, **cross-validation**, and **pruning** can be used. To address underfitting, more complex models or additional features can be considered.

# Regularization

Regularization

Definition

# Regularization

## Regularization

### Definition

**Regularization** is a technique used in machine learning to prevent overfitting by adding a penalty term to the cost function. This penalty discourages overly complex models by constraining the size of the model parameters (coefficients).



# Regularization

## Regularization

### Definition

**Regularization** is a technique used in machine learning to prevent overfitting by adding a penalty term to the cost function. This penalty discourages overly complex models by constraining the size of the model parameters (coefficients).

# Types of Regularized Regression

## Types of Regularized Regression

Definition

There are two common types of regularized regression:

- Ridge Regression (L2 Regularization)
- Lasso Regression (L1 Regularization)

# Lasso Regression (L1 Regularization)

## Lasso Regression (L1 Regularization)

Definition

# Lasso Regression (L1 Regularization)

## Lasso Regression (L1 Regularization)

Definition

**Lasso Regression** adds a penalty equal to the absolute value of the magnitude of coefficients to the loss function. The objective function for lasso regression is:

$$J_{\text{Lasso}}(w) = \frac{1}{2m} \sum_{i=1}^m (f(x_i, w) - y_i)^2 + \lambda \sum_{j=1}^p |w_j|$$

where:

- $\lambda$  is the regularization parameter that controls the strength of the penalty.
- $w_j$  are the coefficients of the independent variables.

# Lasso Regression (L1 Regularization)

## Lasso Regression (L1 Regularization)

Definition

**Lasso Regression** adds a penalty equal to the absolute value of the magnitude of coefficients to the loss function. The objective function for lasso regression is:

$$J_{\text{Lasso}}(w) = \frac{1}{2m} \sum_{i=1}^m (f(x_i, w) - y_i)^2 + \lambda \sum_{j=1}^p |w_j|$$

where:

- $\lambda$  is the regularization parameter that controls the strength of the penalty.
- $w_j$  are the coefficients of the independent variables.

Lasso regression can shrink some coefficients to exactly zero, effectively performing variable selection and resulting in a simpler model.

# Ridge Regression (L2 Regularization)

## Ridge Regression (L2 Regularization)

Definition

# Ridge Regression (L2 Regularization)

## Ridge Regression (L2 Regularization)

Definition

**Ridge Regression** adds a penalty equal to the square of the magnitude of coefficients to the loss function. The objective function for ridge regression is:

$$J_{\text{Ridge}}(w) = \frac{1}{2m} \sum_{i=1}^m (f(x_i, w) - y_i)^2 + \frac{\lambda}{2m} \sum_{j=1}^p w_j^2$$

where:

- $\lambda$  is the regularization parameter that controls the strength of the penalty.
- $w_j$  are the coefficients of the independent variables.

# Ridge Regression (L2 Regularization)

## Ridge Regression (L2 Regularization)

Definition

**Ridge Regression** adds a penalty equal to the square of the magnitude of coefficients to the loss function. The objective function for ridge regression is:

$$J_{\text{Ridge}}(w) = \frac{1}{2m} \sum_{i=1}^m (f(x_i, w) - y_i)^2 + \frac{\lambda}{2m} \sum_{j=1}^p w_j^2$$

where:

- $\lambda$  is the regularization parameter that controls the strength of the penalty.
- $w_j$  are the coefficients of the independent variables.

Ridge regression helps to reduce model complexity and multicollinearity by shrinking the coefficients towards zero, but it does not set any coefficients exactly to zero.



# Nonlinearities in Regression

## Nonlinearities in Regression

Definition

# Nonlinearities in Regression

## Nonlinearities in Regression

### Definition

**Nonlinearities in Regression** refers to the presence of nonlinear relationships between the independent and dependent variables in a regression model. In such cases, a simple linear regression model may not adequately capture the underlying relationship, leading to poor predictions and insights.

# Nonlinearities in Regression

## Nonlinearities in Regression

### Definition

**Nonlinearities in Regression** refers to the presence of nonlinear relationships between the independent and dependent variables in a regression model. In such cases, a simple linear regression model may not adequately capture the underlying relationship, leading to poor predictions and insights. Nonlinear regression models can help address this limitation by allowing for more flexible relationships between the independent and dependent variables.

# Types of Nonlinear Regression Models

## Types of Nonlinear Regression Models

Definition

There are several types of nonlinear regression models, including:

- **Polynomial Regression:** Fits a polynomial equation to the data, allowing for curves and bends in the relationship between the independent and dependent variables. The degree of the polynomial can be adjusted to capture more complex relationships.

# Types of Nonlinear Regression Models

## Types of Nonlinear Regression Models

Definition

There are several types of nonlinear regression models, including:

- **Polynomial Regression:** Fits a polynomial equation to the data, allowing for curves and bends in the relationship between the independent and dependent variables. The degree of the polynomial can be adjusted to capture more complex relationships.
- **Exponential Regression:** Fits an exponential function to the data, which is useful for modeling relationships where the rate of change increases or decreases exponentially.

# Types of Nonlinear Regression Models

## Types of Nonlinear Regression Models

Definition

There are several types of nonlinear regression models, including:

- **Polynomial Regression:** Fits a polynomial equation to the data, allowing for curves and bends in the relationship between the independent and dependent variables. The degree of the polynomial can be adjusted to capture more complex relationships.
- **Exponential Regression:** Fits an exponential function to the data, which is useful for modeling relationships where the rate of change increases or decreases exponentially.
- **Logarithmic Regression:** Fits a logarithmic function to the data, which is useful for modeling relationships where the rate of change decreases as the independent variable increases.

# Overfitting, Underfitting, and Hyperparameter Tuning

Overfitting

Definition

# Overfitting, Underfitting, and Hyperparameter Tuning

## Overfitting

### Definition

**Overfitting** occurs when a model learns the noise in the training data instead of the underlying pattern, resulting in poor generalization to unseen data. It typically happens when the model is too complex relative to the amount of training data.



# Underfitting

Underfitting

Definition

# Underfitting

## Underfitting

### Definition

**Underfitting** occurs when a model is too simple to capture the underlying pattern in the data, resulting in poor performance on both training and unseen data. It typically happens when the model is not complex enough relative to the data.

# Bias-Variance Tradeoff

## Bias-Variance Tradeoff

### Definition

# Bias-Variance Tradeoff

## Bias-Variance Tradeoff

### Definition

The **Bias-Variance Tradeoff** is a fundamental concept in machine learning that describes the tradeoff between two sources of error in a model:

- **Bias:** The error introduced by approximating a real-world problem, which may be complex, with a simplified model. High bias can lead to underfitting.

# Bias-Variance Tradeoff

## Bias-Variance Tradeoff

### Definition

The **Bias-Variance Tradeoff** is a fundamental concept in machine learning that describes the tradeoff between two sources of error in a model:

- **Bias**: The error introduced by approximating a real-world problem, which may be complex, with a simplified model. High bias can lead to underfitting.
- **Variance**: The error introduced by the model's sensitivity to small fluctuations in the training data. High variance can lead to overfitting.

# Bias-Variance Tradeoff

## Bias-Variance Tradeoff

### Definition

The **Bias-Variance Tradeoff** is a fundamental concept in machine learning that describes the tradeoff between two sources of error in a model:

- **Bias**: The error introduced by approximating a real-world problem, which may be complex, with a simplified model. High bias can lead to underfitting.
- **Variance**: The error introduced by the model's sensitivity to small fluctuations in the training data. High variance can lead to overfitting.
- The goal is to find a balance between bias and variance to minimize the overall error of the model.

# Learning Curves

## Learning Curves

### Definition

# Learning Curves

## Learning Curves

### Definition

**Learning Curves** are graphical representations that show how the performance of a machine learning model changes with varying amounts of training data. They can help diagnose overfitting and underfitting by comparing the training and validation errors as the size of the training set increases.



# Hyperparameter Tuning

## Hyperparameter Tuning

Definition

# Hyperparameter Tuning

## Hyperparameter Tuning

### Definition

**Hyperparameter Tuning** is the process of selecting the best hyperparameters for a machine learning model to improve its performance. Hyperparameters are parameters that are not learned from the data but are set before training the model, such as the learning rate, regularization strength, and number of hidden layers in a neural network.

# Techniques for Hyperparameter Tuning

## Techniques for Hyperparameter Tuning

Definition

There are several techniques for hyperparameter tuning, including:

- **Grid Search:** Exhaustively searches through a specified subset of hyperparameters to find the best combination.

# Techniques for Hyperparameter Tuning

## Techniques for Hyperparameter Tuning

Definition

There are several techniques for hyperparameter tuning, including:

- **Grid Search**: Exhaustively searches through a specified subset of hyperparameters to find the best combination.
- **Random Search**: Randomly samples hyperparameters from a specified distribution to find the best combination.

# Techniques for Hyperparameter Tuning

## Techniques for Hyperparameter Tuning

Definition

There are several techniques for hyperparameter tuning, including:

- **Grid Search:** Exhaustively searches through a specified subset of hyperparameters to find the best combination.
- **Random Search:** Randomly samples hyperparameters from a specified distribution to find the best combination.
- **Bayesian Optimization:** Uses probabilistic models to find the best hyperparameters by balancing exploration and exploitation.

# Techniques for Hyperparameter Tuning

## Techniques for Hyperparameter Tuning

Definition

There are several techniques for hyperparameter tuning, including:

- **Grid Search**: Exhaustively searches through a specified subset of hyperparameters to find the best combination.
- **Random Search**: Randomly samples hyperparameters from a specified distribution to find the best combination.
- **Bayesian Optimization**: Uses probabilistic models to find the best hyperparameters by balancing exploration and exploitation.
- **Cross-Validation**: Evaluates the model's performance on different subsets of the data to find the best hyperparameters.

# Techniques for Hyperparameter Tuning

## Techniques for Hyperparameter Tuning

Definition

There are several techniques for hyperparameter tuning, including:

- **Grid Search**: Exhaustively searches through a specified subset of hyperparameters to find the best combination.
- **Random Search**: Randomly samples hyperparameters from a specified distribution to find the best combination.
- **Bayesian Optimization**: Uses probabilistic models to find the best hyperparameters by balancing exploration and exploitation.
- **Cross-Validation**: Evaluates the model's performance on different subsets of the data to find the best hyperparameters.
- **Automated Hyperparameter Tuning**: Uses algorithms to automatically search for the best hyperparameters based on the model's performance.