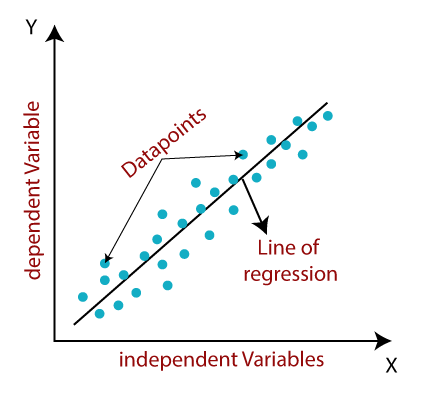
**Linear Regression**

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables. Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

The linear regression model provides a sloped straight line representing the relationship between the variables. Consider the below image:



**Here,**

* Y= Dependent Variable (Target Variable)
* X= Independent Variable (predictor Variable)
* a0= intercept of the line (Gives an additional degree of freedom)
* a1 = Linear regression coefficient (scale factor to each input value).
* ε = random error

The values for x and y variables are training datasets for Linear Regression model representation

Linear regression can be further divided into two types of the algorithm:

* **Simple Linear Regression:**  
  If a single independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Simple Linear Regression.
* **Multiple Linear regression:**  
  If more than one independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression.

When working with linear regression, our main goal is to find the best fit line that means the error between predicted values and actual values should be minimized. The best fit line will have the least error. The different values for weights or the coefficient of lines (a0, a1) gives a different line of regression, so we need to calculate the best values for a0 and a1 to find the best fit line, so to calculate this we use cost function.

For Linear Regression, we use the **Mean Squared Error (MSE)** cost function, which is the average of squared error occurred between the predicted values and actual values. It can be written as:

A close-up of a number

Description automatically generated

**Residuals:** The distance between the actual value and predicted values is called residual. If the observed points are far from the regression line, then the residual will be high, and so cost function will high. If the scatter points are close to the regression line, then the residual will be small and hence the cost function.

**Gradient Descent:**

* Gradient descent is used to minimize the MSE by calculating the gradient of the cost function.
* A regression model uses gradient descent to update the coefficients of the line by reducing the cost function.
* It is done by a random selection of values of coefficient and then iteratively update the values to reach the minimum cost function.

**Overview of Linear Regression Model:**

A math equations on a white background

Description automatically generated

The Goodness of fit determines how the line of regression fits the set of observations. The process of finding the best model out of various models is called **optimization**. It can be achieved by below method:

**R-squared method:**

* R-squared is a statistical method that determines the goodness of fit.
* It measures the strength of the relationship between the dependent and independent variables on a scale of 0-100%.
* The high value of R-square determines the less difference between the predicted values and actual values and hence represents a good model.
* It is also called a **coefficient of determination,** or **coefficient of multiple determination** for multiple regression.
* It can be calculated from the below formula:

A black text on a white background

Description automatically generated

Where:

* SSResidual ​: Sum of squared residuals (unexplained variance).
* SSTotal ​: Total variance in the dependent variable.

Range: 0≤R2≤1

* R2=1: Perfect fit (model explains 100% of the variance).
* R2=0: Model explains none of the variance.
* R2<0: Indicates a poor fit, potentially worse than a horizontal line (mean prediction).

General Guidelines:

* High R²: For problems where precise predictions are crucial (e.g., engineering or physical sciences), R² values of 0.9 or above may be desirable.
* Moderate R²: In fields like social sciences, economics, or psychology, where data is often noisy, an R² between 0.3 and 0.7 is often considered acceptable.
* Low R²: In certain real-world problems (e.g., stock market predictions, human behavior modeling), an R² of 0.1 to 0.3 can still be meaningful if it provides actionable insights.

**Assumptions of Linear Regression**:

* **Linear relationship between the features and target:** Linear regression assumes the linear relationship between the dependent and independent variables.
* **Small or no multicollinearity between the features:**  
  Multicollinearity means high correlation between the independent variables. Due to multicollinearity, it may be difficult to find the true relationship between the predictors and target variables. So, the model assumes either little or no multicollinearity between the features or independent variables.

**Logistic Regression**

**Logistic regression** is a **supervised machine learning algorithm**used for **classification tasks** where the goal is to predict the probability that an instance belongs to a given class or not.

**Assumptions of Logistic Regression:**

We will explore the assumptions of logistic regression as understanding these assumptions is important to ensure that we are using appropriate application of the model. The assumption include:

* Independent observations: Each observation is independent of the other. meaning there is no correlation between any input variables.
* Binary dependent variables: It takes the assumption that the dependent variable must be binary or dichotomous, meaning it can take only two values. For more than two categories SoftMax functions are used.
* Linearity relationship between independent variables and log odds: The relationship between the independent variables and the log odds of the dependent variable should be linear.
* No outliers: There should be no outliers in the dataset.
* Large sample size: The sample size is sufficiently large

**Overview**

Similar to linear regression, logistic regression starts by computing a **linear combination** of input features:



Or in vectorized form:



where:

*  is the feature vector.
* are the model weights.
* **b** is the bias term.

To convert the linear combination zzz into a probability value between 0 and 1, we apply the **sigmoid function**:

A math equation with numbers and symbols

Description automatically generated

Thus, the probability of the output being class **1** (instead of 0) is:

A black and white math equation

Description automatically generated

Since probabilities range between 0 and 1, we can classify the output using a threshold (P(y=1)>0, predict class **1**, otherwise predict **0**).

To train the model, we need a **loss function** that measures how well the predicted probabilities match the actual class labels. We use the **log-likelihood** function (also known as binary cross-entropy):

A black text with a white background

Description automatically generated

where:

* ​ is the actual class label (0 or 1).
* ​ is the predicted probability σ(z).



Since we want to **maximize** likelihood, we typically **minimize** the negative log-likelihood:

A close up of a word

Description automatically generated

This is called the **Binary Cross-Entropy Loss**/**Log Likelihood Loss Function.**

To find the optimal **w** and **b**, we use **Gradient Descent**, an iterative optimization algorithm. The partial derivatives of the loss function with respect to the weights and bias b are:

A mathematical equations with numbers and symbols

Description automatically generated with medium confidence

where:

*  is the j-th feature of the i-th training sample.
* is the error (difference between prediction and actual label).

Using **Gradient Descent**, we update the parameters as:

A math equations with numbers

Description automatically generated with medium confidence

where α is the **learning rate**.

The decision boundary is the function that separates the two classes. Since logistic regression outputs probabilities, we define the decision rule as:

A number symbols with numbers and arrows

Description automatically generated with medium confidence