Regression

Regression is defined as a statistical method that helps us to analyze and understand the relationship between two or more variables of interest.

Regression searches for relationships among **variables**. For example, you can observe several employees of some company and try to understand how their salaries depend on their **features**, such as experience, education level, role, city of employment, and so on.This is a regression problem where data related to each employee represents one **observation**. The presumption is that the experience, education, role, and city are the independent features, while the salary depends on them.

In other words, you need to find a **function that maps some features or variables to others** sufficiently well.

The dependent features are called the **dependent variables**, **outputs**, or **responses**. The independent features are called the **independent variables**, **inputs**, **regressors**, or **predictors**.

- Regression estimates the relationship between the target and the independent variable.
- It is used to find the trends in data.
- It helps to predict real/continuous values.
- By performing the regression, we can confidently determine the **most important factor, the least important factor, and how each factor is affecting the other factors**.

Supervised and Unsupervised Learning algorithms:

Supervised learning algorithms are trained using labeled data. Unsupervised learning algorithms are trained using unlabeled data. Supervised learning model takes direct feedback to check if it is predicting correct output or not. Unsupervised learning model does not take any feedback. In supervised learning, the algorithm “learns” from the training dataset by iteratively making predictions on the data and adjusting for the correct answer.

Types of Regression

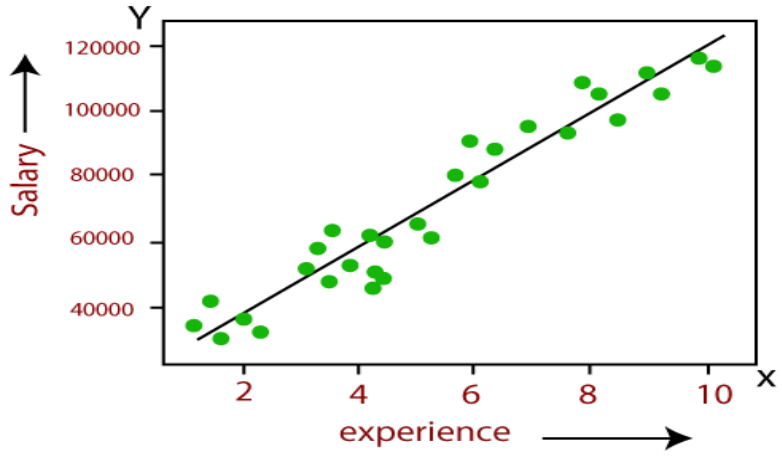
There are various types of regressions which are used in data science and machine learning. Each type has its own importance on different scenarios, but at the core, all the regression methods analyze the effect of the independent variable on dependent variables. Here we are discussing some important types of regression which are given below:

- **Linear Regression**
- **Logistic Regression**

- **Polynomial Regression**
- **Decision Tree Regression**
- **Ridge Regression**

Linear Regression:

- Linear regression is a statistical regression method which is used for predictive analysis.
- It is one of the very simple and easy algorithms which works on regression and shows the relationship between the continuous variables.
- It is used for solving the regression problem in machine learning.
- Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence called linear regression.
- If there is only one input variable (x), then such linear regression is called **simple linear regression**. And if there is more than one input variable, then such linear regression is called **multiple linear regression**.
- The relationship between variables in the linear regression model can be explained using the below image. Here we are predicting the salary of an employee on the basis of **the year of experience**.



Logistic Regression:

- Logistic regression is another supervised learning algorithm (where machine is fed with Labeled Data) which is used to solve the classification problems. In **classification problems**, are used to

forecast or classify the distinct values such as Real or False, Male or Female, Spam or Not Spam, etc.

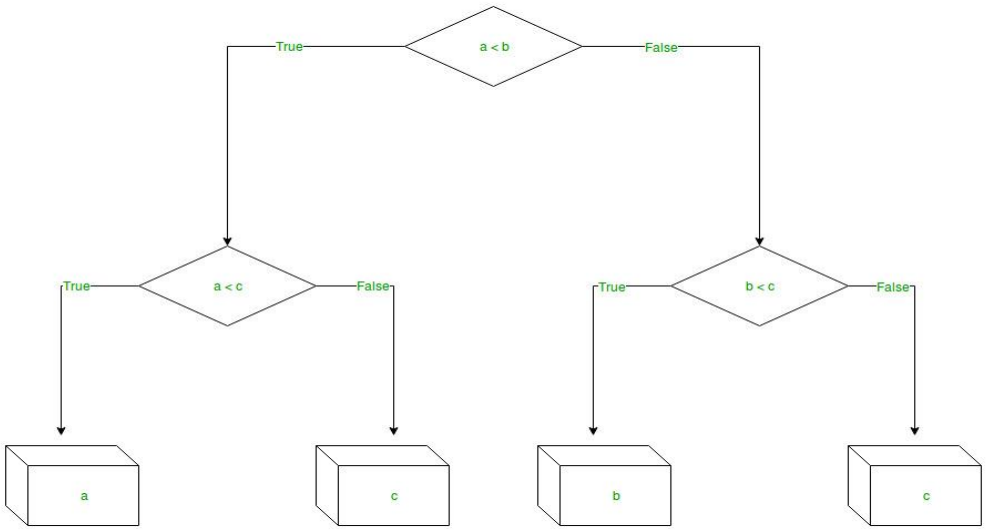
- Logistic regression algorithm works with the categorical variable such as 0 or 1, Yes or No, True or False, Spam or not spam, etc.
- It is a predictive analysis algorithm which works on the concept of probability.

Polynomial Regression:

- Polynomial Regression is a type of regression which models the **non-linear dataset** using a linear model.
- It is similar to multiple linear regression, but it fits a non-linear curve between the value of x and corresponding conditional values of y.
- Suppose there is a dataset which consists of datapoints which are present in a non-linear fashion, so for such case, linear regression will not best fit to those datapoints. To cover such datapoints, we need Polynomial regression.

Decision Tree Regression:

- Decision Tree is a supervised learning algorithm which can be used for solving both classification and regression problems.
- It can solve problems for both categorical and numerical data
- Decision Tree regression builds a tree-like structure in which each internal node represents the "test" for an attribute, each branch represent the result of the test, and each leaf node represents the final decision or result.
- A decision tree is constructed starting from the root node/parent node (dataset), which splits into left and right child nodes (subsets of dataset). These child nodes are further divided into their children node, and themselves become the parent node of those nodes.



Ridge Regression:

- Ridge regression is one of the most robust versions of linear regression in which a small amount of bias is introduced so that we can get better long term predictions.
- The amount of bias added to the model is known as **Ridge Regression penalty**.
- This is basically used where we have large number of predictor variables but less no of observations.

Exploratory Data Analysis (EDA)

This is an approach to analyze the data using visual techniques. It is used to discover trends, patterns, or to check assumptions with the help of statistical summary and graphical representations.
To understand the concept we will use a single dataset i.e. employee data for this. It contains 8 columns namely – First Name, Gender, Start Date, Last Login, Salary, Bonus%, Senior Management, and Team.

Dataset Used: Employees.csv
i) To read the dataset using the Pandas module and print the 1st five rows, will use the **head()** function.

```
import pandas as pd
import numpy as np

df = pd.read_csv('employees.csv')
df.head(5)
```

ii) To print the last five rows we will use the **tail()** function.
df.tail(5)

iii) We can get the total number of rows and columns from the data set using ".shape"
df.shape()

Output:
```
(1000, 8)
```

This means that this dataset has 1000 rows and 8 columns.

iv) describe() method. The describe() function applies basic statistical computations on the dataset like extreme values, count of data points standard deviation, etc. Any missing value or NaN value is automatically skipped. describe() function gives a good picture of the distribution of data.

df.describe()

v) To know about the columns and their data types,we will use the info() method.

```
df.info()
```

Handling Missing Values

It can occur when no information is provided for one or more items or for a whole unit. For Example, Suppose different users being surveyed may choose not to share their income, some users may choose not to share the address in this way many datasets went missing. Missing Data is a very big problem in real-life scenarios. Missing Data can also refer to as NA(Not Available) values in pandas. There are several useful functions for detecting, removing, and replacing null values in Pandas DataFrame :

- isnull()
- notnull()
- dropna()
- fillna()
- replace()
- interpolate()

Data visualization

Data Visualization is the process of analyzing data in the form of graphs or maps, making it a lot easier to understand the trends or patterns in the data. There are various types of visualizations –

- **Univariate analysis:** This type of data consists of only one variable. The main purpose of the analysis is to describe the data and find patterns that exist within it.

- **Bi-Variate analysis:** This type of data involves two different variables. The analysis is done to find out the relationship among the two variables.
- **Multi-Variate analysis:** When the data involves three or more variables, it is categorized under multivariate.

Using Matplotlib and Seaborn library for the data visualization.

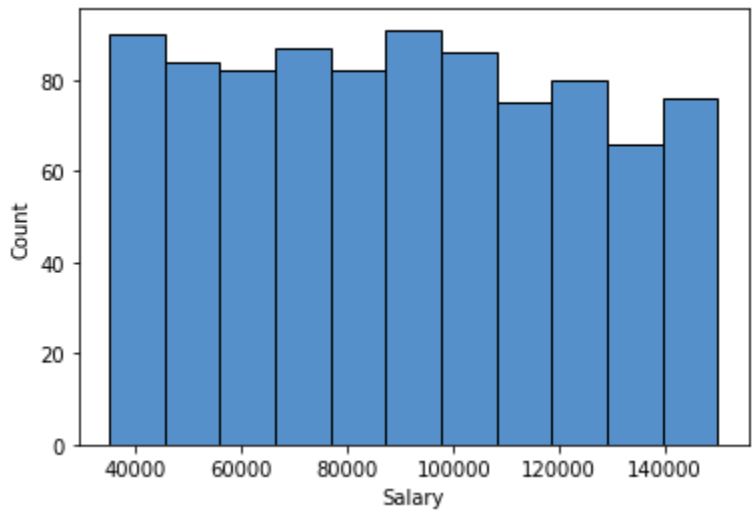
Histogram

It can be used for both uni and bivariate analysis.

Example:

```
# importing packages

import seaborn as sns

import matplotlib.pyplot as plt

sns.histplot(x='Salary', data=df, )

plt.show()
```

Output:

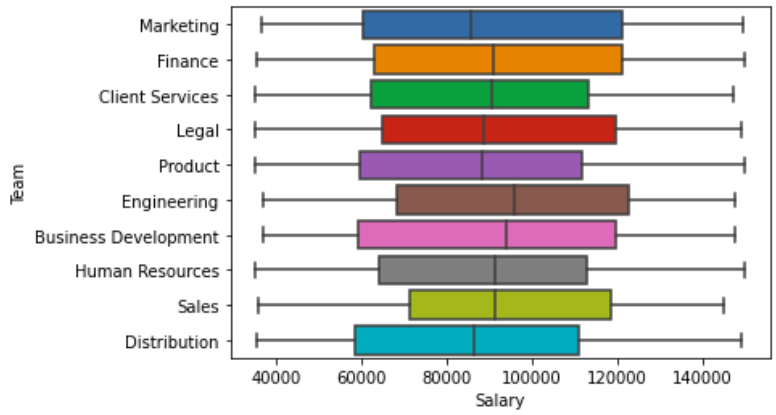


Boxplot

It can also be used for univariate and bivariate analyses.

```
# importing packages

import seaborn as sns

import matplotlib.pyplot as plt

sns.boxplot( x="Salary", y='Team', data=df, )

plt.show()
```

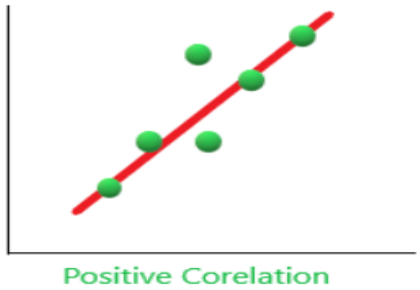
Output:



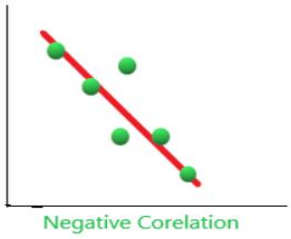
Correlation Matrix:

Correlation means an association, It is a measure of the extent to which two variables are related.

1. Positive Correlation: When two variables increase together and decrease together. They are positively correlated. '1' is a perfect positive correlation. For example – demand and profit are positively correlated the more the demand for the product, the more profit hence positive correlation.



2. Negative Correlation: When one variable increases and the other variable decreases together and vice-versa. They are negatively correlated. For example, If the distance between magnet increases their attraction decreases, and vice-versa. Hence, a negative correlation. '-1' is no correlation



3. Zero Correlation (No Correlation): When two variables don't seem to be linked at all. '0' is a perfect negative correlation. For Example, the amount of tea you take and level of intelligence.

- **The libraries needed:**
    1. Sklearn
    2. Numpy
    3. Matplotlib
    4. Pandas

Given data about two variables, we can find the correlation between the two variables using Pandas:

```
import pandas as p
var1 = p. Series ([1, 3, 4, 6, 7, 9])
var2 = p. Series ([2, 4, 7, 8, 9, 11])
correlation = var2. corr (var1)
```

```
print (correlation)
correlation = var1. corr (var2)
print (correlation)
```

Output:

```
0.9793792286287205

0.9793792286287205
```

- If two variables are in correlation, the first variable is dependent on the second variable just as much as the second variable is dependent on the first. Hence, the two values are the same.