1 *Linear Regression:-* In Linear Regression these two variables are related through an equation, where exponent (power) of both these variables is 1. Mathematically a linear relationship represents a straight line when plotted as a graph. A non-linear relationship where the exponent of any variable is not equal to 1 creates a curve.

The general mathematical equation for a linear regression is – y = ax + b

Following is the description of the parameters used –

- y is the response variable.
- x is the predictor variable.
- a and b are constants which are called the coefficients.

lm() Function:-This function creates the relationship model between the predictor and the response variable.

Syntax: *lm(formula,data)*

Following is the description of the parameters used –

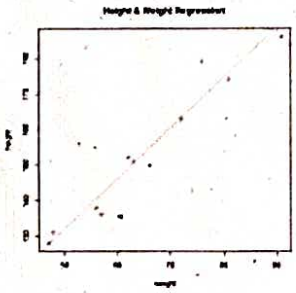
- **formula** is a symbol presenting the relation between x and y.
- **data** is the vector on which the formula will be applied.

Example:-

```
> height <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
> weight <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
> relation <- lm(weight~height)
> print(relation)

Call:
lm(formula = weight ~ height)

Coefficients:
(Intercept)       height
   -38.4551       0.6746
> plot(weight,height,col = "blue",main = "Height & Weight
Regression")
> abline(lm(height~weight),col="orange")
```



Advantages / Limitations of Linear Regression Model :

- Linear regression implements a statistical model that, when relationships between the independent variables and the dependent variable are almost linear, shows optimal results.
- Linear regression is often inappropriately used to model non-linear relationships.
- Linear regression is limited to predicting numeric output.
- A lack of explanation about what has been learned can be a problem.

Regression equation of x on y

$$x - \bar{x} = r.\frac{\sigma_x}{\sigma_y}(y - \bar{y})$$

Regression equation of y on x

$$y - \bar{y} = r.\frac{\sigma_y}{\sigma_x}(x - \bar{x})$$

where

$$r = \frac{\frac{1}{n}\sum xy - \bar{x} - \bar{y}}{\sqrt{\frac{1}{n}\sum (x-\bar{x})^2}\sqrt{\frac{1}{n}\sum (y-\bar{y})^2}}$$

2 **Logistic Regression** : The Logistic Regression is a regression model in which the response variable (dependent variable) has categorical values such as True/False or 0/1. It actually measures the probability of a binary response as the value of response variable based on the mathematical equation relating it with the predictor variables.

The general mathematical equation for logistic regression is −

$$y = 1/(1+e^{-(a+b_1x_1+b_2x_2+b_3x_3+...)})$$

- **y** is the response variable.
- **x** is the predictor variable.
- **a** and **b** are the coefficients which are numeric constants.

The function used to create the regression model is the **glm()** function.

Syntax :- *glm(formula,data,family)*

Following is the description of the parameters used −

- **formula** is the symbol presenting the relationship between the variables.
- **data** is the data set giving the values of these variables.
- **family** is R object to specify the details of the model. It's value is binomial for logistic regression.

ex:

```R
# Load the dataset
data(mtcars)

# Create a binary variable for high mileage (mpg > median(mpg))
mtcars$high_mileage <- ifelse(mtcars$mpg > median(mtcars$mpg), 1, 0)

# Fit a logistic regression model
model <- glm(high_mileage ~ cyl + hp, data = mtcars, family = binomial)

# Summary of the model
summary(model)
```

In the above code:
- We create a binary response variable `high_mileage` based on whether a car has high mileage.
- We use the `glm()` function to fit a logistic regression model, specifying the formula for the binary response and predictor variables and setting the family to "binomial."

Comparison of Linear Regression and Logistic Regression:

1. **Purpose:**
   - Linear Regression: Used for predicting continuous numerical outcomes. It models the relationship between the response variable and predictor variables to predict a numeric value.
   - Logistic Regression: Used for predicting binary categorical outcomes. It models the relationship between the probability of success (1) and predictor variables to classify data into two categories.

2. **Output:**
   - Linear Regression: The output is a continuous numeric value.
   - Logistic Regression: The output is a probability value between 0 and 1, which can be transformed into a binary prediction using a threshold (e.g., 0.5).

3. **Assumptions:**
   - Linear Regression: Assumes a linear relationship between predictor variables and the response variable.
   - Logistic Regression: Does not assume a linear relationship; it models the log-odds of the probability.

4. **Error Metric:**
   - Linear Regression: Typically uses metrics like Mean Squared Error (MSE) to evaluate the model.
   - Logistic Regression: Uses metrics like accuracy, precision, recall, and F1-score to evaluate classification performance.

3 **Poisson Regression:-** Poisson Regression involves regression models in which the response variable is in the form of counts and not fractional numbers. For example, the count of number of births or number of wins in a football match series. Also the values of the response variables follow a Poisson distribution. The general mathematical equation for Poisson regression is –

$$\log(y) = a + b_1x_1 + b_2x_2 + b_nx_n.....$$

Following is the description of the parameters used –

- y is the response variable.
- a and b are the numeric coefficients.
- x is the predictor variable.

The function used to create the Poisson regression model is the glm()function.

We have the in-built data set "warpbreaks" which describes the effect of wool type (A or B) and tension (low, medium or high) on the number of warp breaks per loom. Let's consider "breaks" as the response variable which is a count of number of breaks. The wool "type" and "tension" are taken as predictor variables.

```R
# Load the dataset (simulated data)
data <- data.frame(
  accidents = c(1, 2, 0, 3, 1, 2, 4, 2, 1, 3),
  vehicles = c(10, 12, 8, 15, 10, 13, 18, 14, 11, 16),
  speed_limit = c(30, 25, 30, 40, 35, 45, 40, 30, 35, 50)
)

# Fit a Poisson regression model
model <- glm(accidents ~ vehicles + speed_limit, data = data, family = poisson)

# Summary of the model
summary(model)
```

In the above code:
- We load a dataset containing counts of traffic accidents, the number of vehicles, and the speed limit.
- We use the `glm()` function to fit a Poisson regression model, specifying the formula for the response variable (`accidents`) and predictor variables (`vehicles` and `speed_limit`) while setting the family to "poisson."

The `summary(model)` provides information about the fitted Poisson regression model, including coefficients, their significance, and goodness-of-fit statistics.

Poisson regression is suitable for count data, and it models the relationship between the predictors and the count outcome while taking into account the Poisson distribution characteristics. It's widely used in various fields, including epidemiology, finance, and environmental studies, to analyze and predict count-based outcomes.

4 *Multiple Regression :-* **Multiple regression** is an extension of simple **linear regression**. It is used when we want to predict the value of a variable based on the value of two or more other variables. The variable we want to predict is called the dependent variable

The general mathematical equation for multiple regression is – $x_1 = a_0 + a_1x_2 + a_2x_3$

Following is the description of the parameters used –

- x_1 is the response variable.
- a_0, a_1, a_2...bn are the coefficients.
- x_1, x_2, ...xn are the predictor variables.

The Normal equations for estimating a_0,a_1 and a_2 .

$$\sum x_1 = na_0 + a_1\sum x_2 + a_2\sum x_3$$

$$\sum x_1x_2 = a_0\sum x_2 + a_1\sum x_2^{\,2} + a_2\sum x_2x_3$$

$$\sum x_1x_3 = a_0\sum x_3 + a_1\sum x_2x_3 + a_2\sum x_3^{\,2}$$

We create the regression model using the lm() function in R. The model determines the value of the coefficients using the input data. Next we can predict the value of the response variable for a given set of predictor variables using these coefficients.

lm() Function :-This function creates the relationship model between the predictor and the response variable.

Syntax :- *lm(y ~ x1+x2+x3...,data)*

Following is the description of the parameters used –

- formula is a symbol presenting the relation between the response variable and predictor variables.
- data is the vector on which the formula will be applied.

```R
# Load the dataset
data(mtcars)

# Fit a multiple regression model
model <- lm(mpg ~ hp + wt + qsec + drat, data = mtcars)

# Summary of the model
summary(model)
```

In the above code:
- We load the `mtcars` dataset, which contains information about various car attributes.
- We use the `lm()` function to fit a multiple regression model, specifying the formula for the response variable (`mpg`) and predictor variables (`hp`, `wt`, `qsec`, and `drat`) using the formula notation.

The `summary(model)` provides information about the fitted multiple regression model, including coefficients, their significance, goodness-of-fit statistics (e.g., R-squared), and more.

The example demonstrates how to implement multiple regression in R to predict a continuous dependent variable (`mpg`) based on multiple predictor variables (`hp`, `wt`, `qsec`, and `drat`). This is a basic example, and in practice, you can use as many predictor variables as needed to model the relationship with the dependent variable. Multiple regression is widely used in various fields, including economics, social sciences, and environmental sciences, to analyze complex relationships and make predictions.