Linear Regression

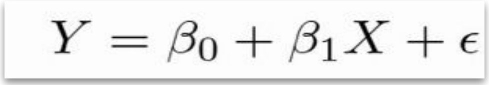
We will talk about…

- Introduction
- Hypothesis of linear regression
- Assumption of regression
- Properties of regression line
- Advantages of linear regression
- Limitations of linear regression
- Data preparation for linear regression
- Regression Model Evaluation Metrics

Introduction

Linear regression is one of the most basic types of regression in supervised machine learning. The linear regression model consists of a predictor variable and a dependent variable related ***linearly*** to each other. We try to find the ***relationship*** between ***independent variable*** (input) and a corresponding ***dependent variable*** (output).

This can be expressed in the form of a straight line

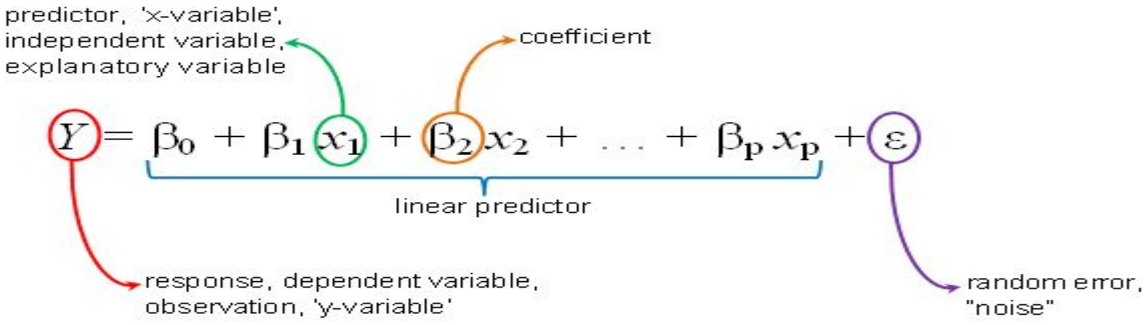


$$Y = \beta_0 + \beta_1 X + \epsilon$$

Linear regression, also known as ***ordinary least squares (OLS)*** and ***linear least squares***, is the real workhorse of the regression world.

Hypothesis of Linear Regression

The linear regression model can be represented by the following equation:



predictor, 'x-variable', independent variable, explanatory variable

coefficient

$$Y = \underbrace{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p}_{\text{linear predictor}} + \varepsilon$$

response, dependent variable, observation, 'y-variable'

random error, "noise"

Y is the actual value

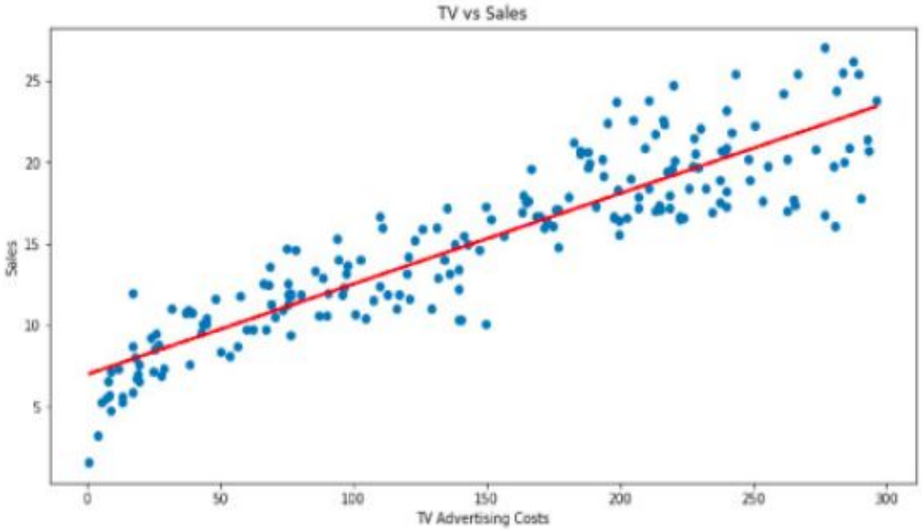
β_0 is the bias term.

$\beta_1,\ldots,\beta_p$ are the model parameters

x_1, $x_2,\ldots,x_p$ are the feature values.

ROI of TV ads?

Looking at the past investment on TV ads vs Sales done by company. Company wants to estimate return on investment (ROI) which can be calculated by estimating sales through past investment on TV ads.



The goal of linear regression is to create a ***trend line*** *or* ***best fit line*** based on the past data of investment on TV ads vs Sales.

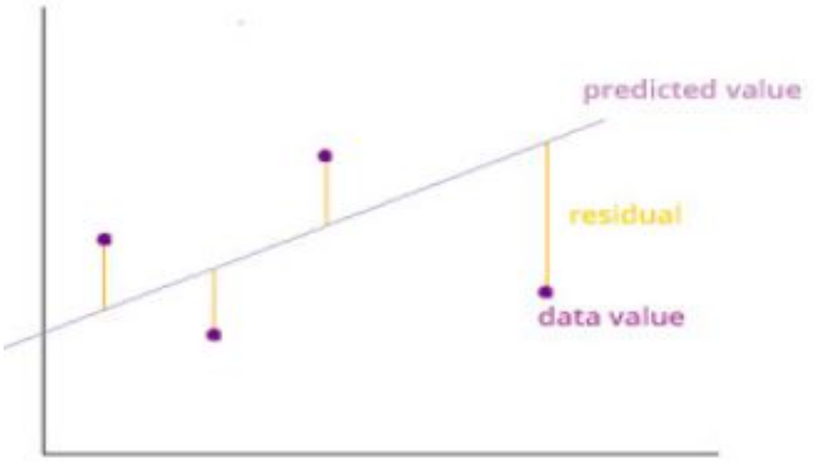
Trend line which can then be used to confirm or deny the relationship whether TV ads create impact on Sales and also predict Sales based on TV ads.

How do we determine the best fit line?

The best fit line is considered to be the line for which the error between the predicted values and the observed values is minimum.

It is also called the regression line and the errors are also known as residuals.

It can be visualized by the vertical lines from the observed data value to the regression line.



$$\min(SSE) = \sum_{i=1}^{n} (\textit{actual output} - \textit{predicted output})^2$$

SSE : Sum square error

Predicted Output?

Linear regression equation without error term gives predicted output.

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots + \beta_p X_p$$

By varying different parameter values ($\boldsymbol{\beta}_p$) , predicted value can be determined and value which gives minimum SSE will produce the best fit line.

SSE : $$\sum_{i=1}^{n} (y - \hat{y})^2$$

Note: Predicted value can have positive or negative error. If we don’t square the error, then the positive and negative points will cancel each other out. Absolute function(|Actual- Pred|) can be one option but it is not differentiable at Actual = Pred

Least Squares Estimators of Parameters

Linear regression equation : $Y = \beta_0 + \beta_1 X + \epsilon$

$$SSE = \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i \right)^2$$

The method of least squares chooses estimators of β that minimize SSE. To determine these estimators, we differentiate SSE first with respect to β_0 and then to β_1 as follows :

$$\frac{\partial SSE}{\partial \beta_0} = -2 \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i \right)$$

$$\frac{\partial SSE}{\partial \beta_1} = -2 \sum_{i=1}^{n} x_i \cdot \left(y_i - \beta_0 - \beta_1 x_i \right)$$

Derivative = 0

Setting these partial derivatives equal to zero yields the following equations for the minimizing values β_0 and β_1

$$\sum_{i=1}^{n} y_i = n\beta_o + \beta_1 \sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} x_i y_i = \beta_o \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^{\,2}$$

These equations are known as the **normal equations.**

contd..

If we let

$$\bar{y} = \sum_i \frac{y_i}{n}$$

$$\bar{x} = \sum_i \frac{x_i}{n}$$

then we can write the first normal equation as $\beta_0 = \bar{y} - \beta_1 \bar{x}$

Substituting this value of β_0 into the second normal equation yields

$$\sum_i x_i y_i = \left(\bar{y} - \beta_1 \bar{x}\right) \cdot n\bar{x} + \beta_1 \sum_i x_i^2$$

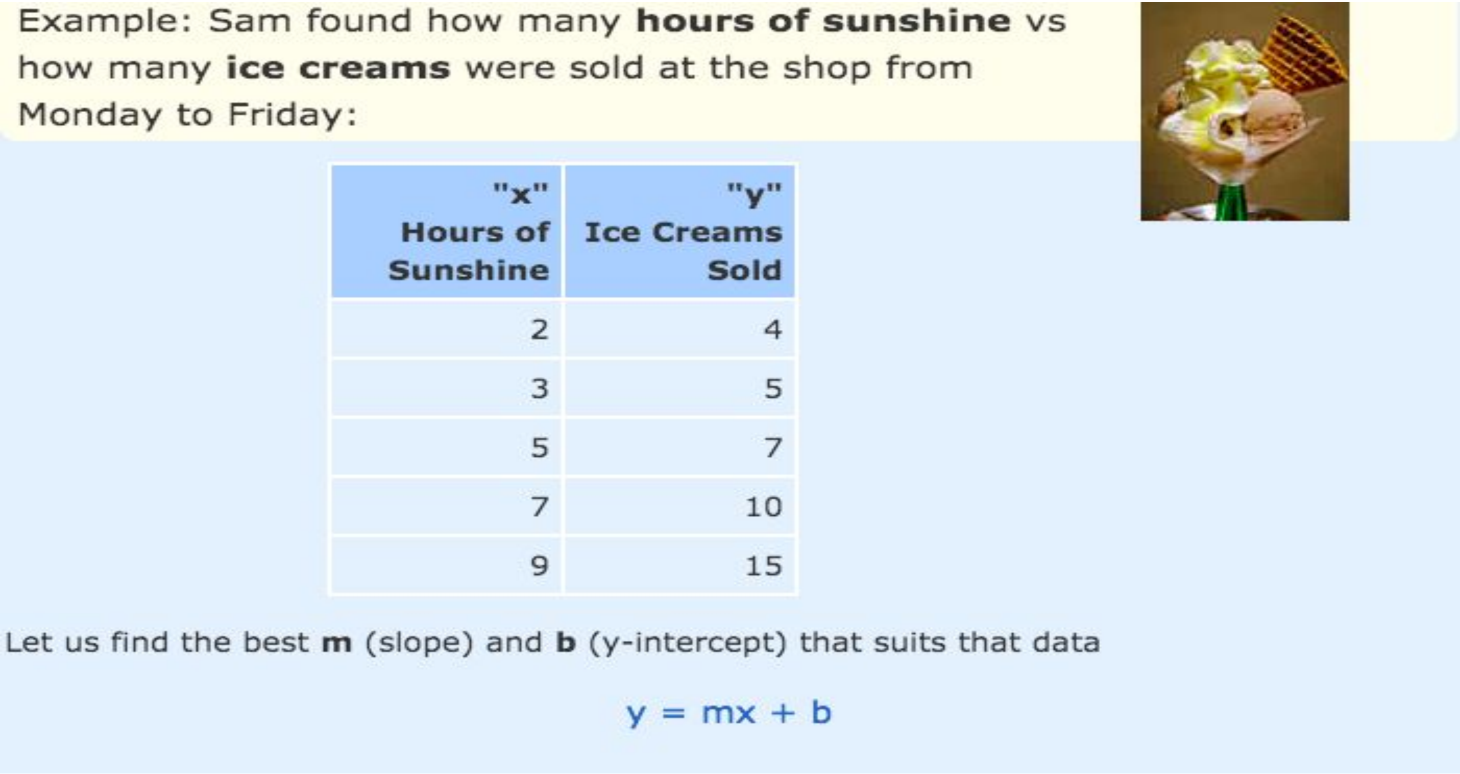
Or

$$\beta_1 \left(\sum_i x_i^2 - n\bar{x}^2 \right) = \sum_i x_i y_i - n \cdot \bar{x}\,\bar{y}$$

$$\beta_1 = \frac{\left(\sum_i x_i y_i - n \cdot \bar{x} \cdot \bar{y} \right)}{\left(\sum_i x_i^2 - n \cdot (\bar{x})^2 \right)}$$

$$\beta_1 = \frac{n \sum_i x_i y_i - \sum_i x \sum_i y}{n \sum_i x_i^2 - \left(\sum_i x_i \right)^2}$$

Example

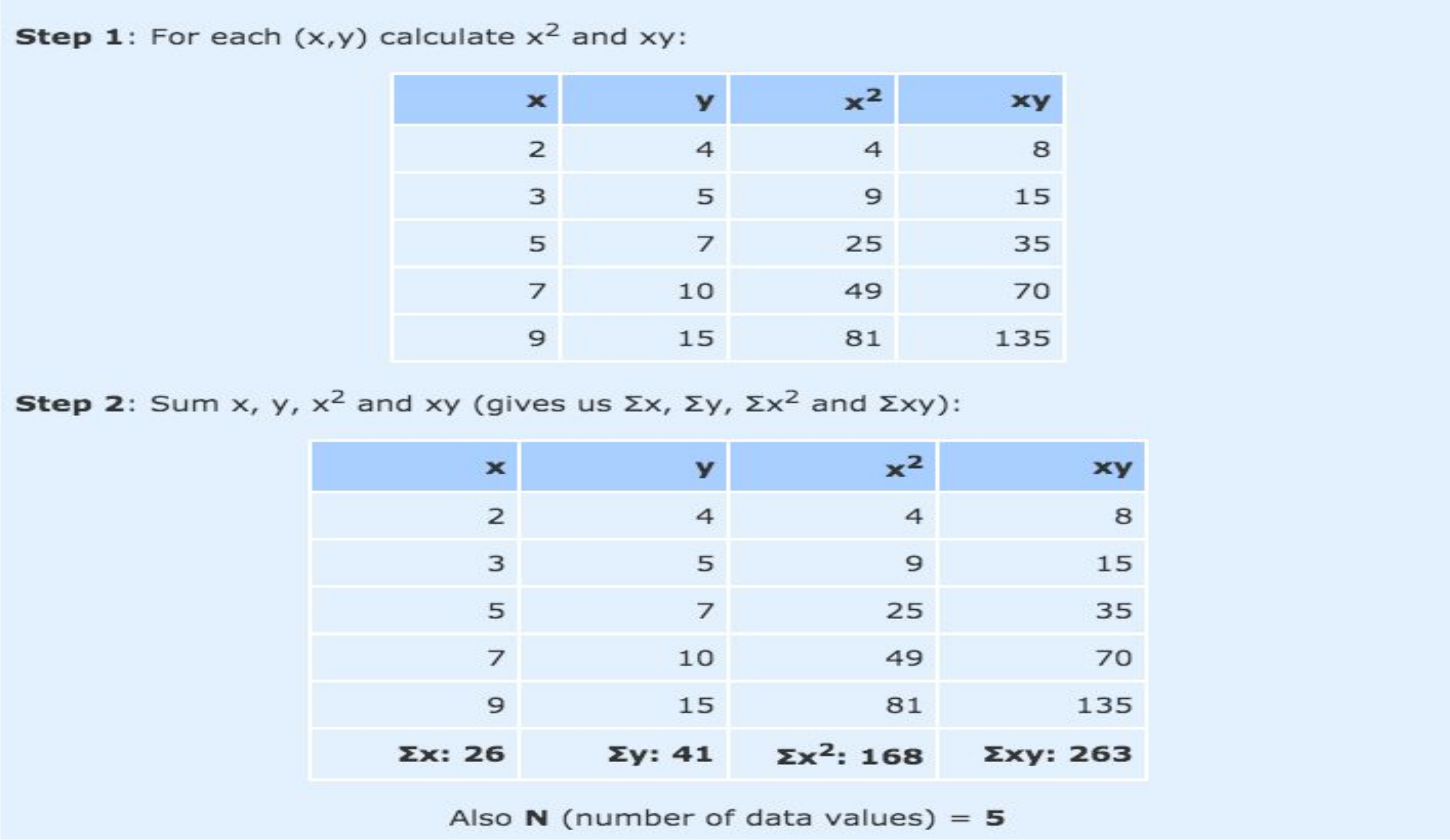


Example: Sam found how many **hours of sunshine** vs how many **ice creams** were sold at the shop from Monday to Friday:

"x" Hours of Sunshine	"y" Ice Creams Sold
2	4
3	5
5	7
7	10
9	15

Let us find the best **m** (slope) and **b** (y-intercept) that suits that data

$$y = mx + b$$



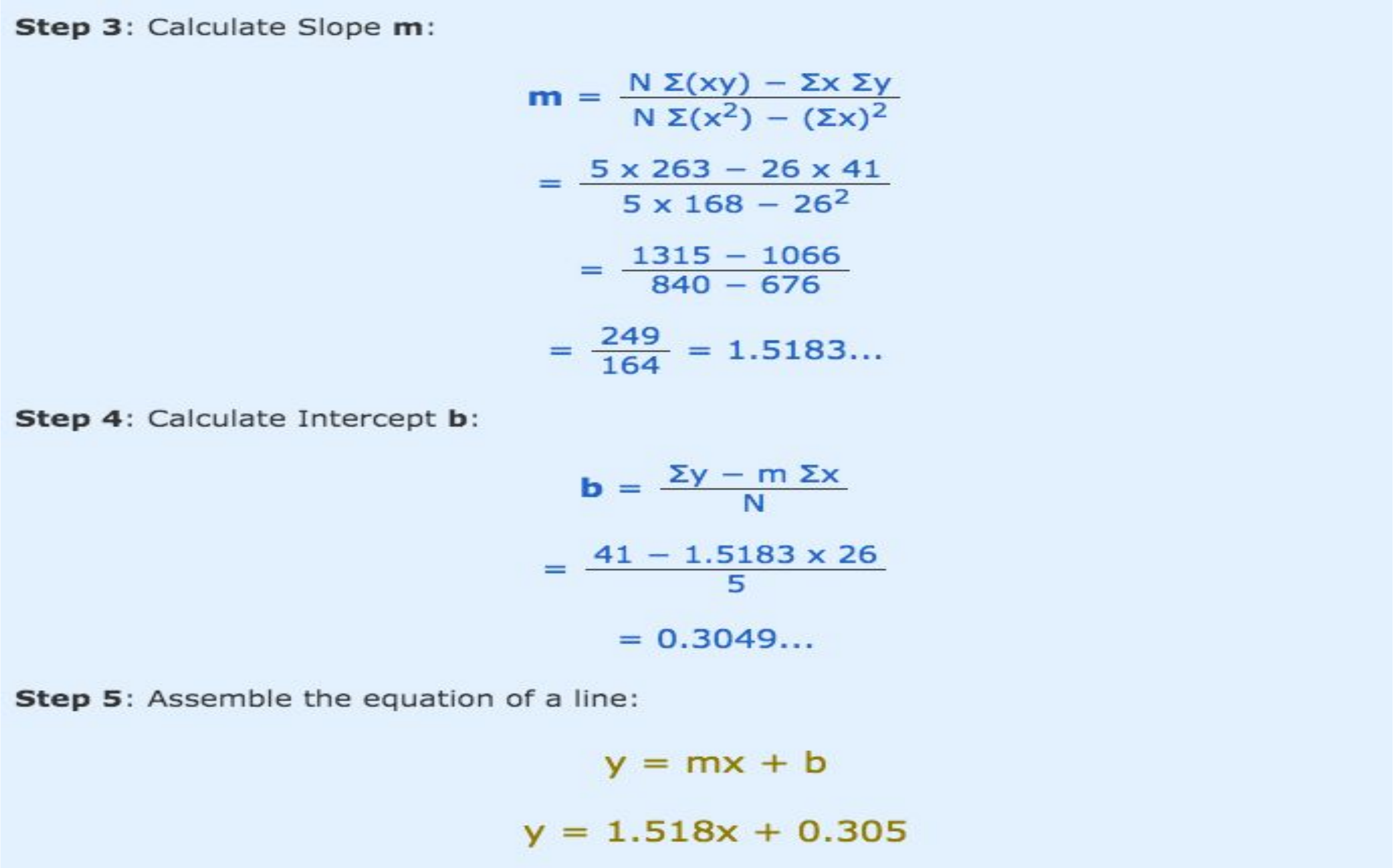
Step 1: For each (x,y) calculate x^2 and xy:

x	y	x^2	xy
2	4	4	8
3	5	9	15
5	7	25	35
7	10	49	70
9	15	81	135

Step 2: Sum x, y, x^2 and xy (gives us Σx, Σy, $Σx^2$ and Σxy):

x	y	x^2	xy
2	4	4	8
3	5	9	15
5	7	25	35
7	10	49	70
9	15	81	135
Σx: 26	**Σy: 41**	**$Σx^2$: 168**	**Σxy: 263**

Also **N** (number of data values) = **5**



Step 3: Calculate Slope **m**:

$$\mathbf{m} = \frac{N\ \Sigma(xy) - \Sigma x\ \Sigma y}{N\ \Sigma(x^2) - (\Sigma x)^2}$$

$$= \frac{5 \times 263 - 26 \times 41}{5 \times 168 - 26^2}$$

$$= \frac{1315 - 1066}{840 - 676}$$

$$= \frac{249}{164} = 1.5183...$$

Step 4: Calculate Intercept **b**:

$$\mathbf{b} = \frac{\Sigma y - m\ \Sigma x}{N}$$

$$= \frac{41 - 1.5183 \times 26}{5}$$

$$= 0.3049...$$

Step 5: Assemble the equation of a line:

$$y = mx + b$$

$$y = 1.518x + 0.305$$

Best possible β_p

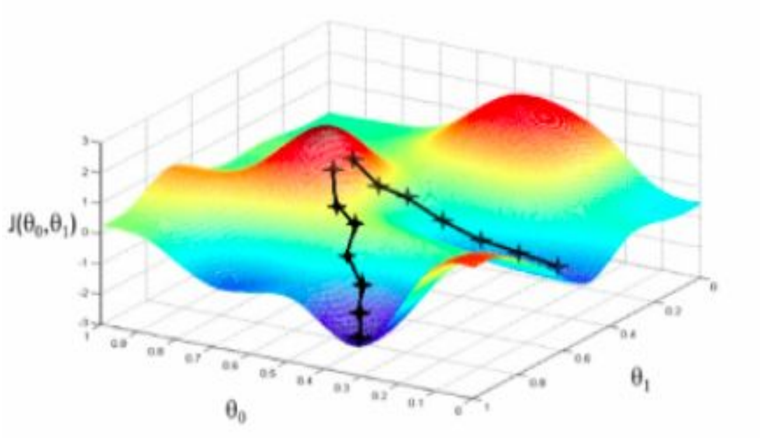
With infinite possibilities of values which parameter can take, how do we arrive at the best possible β_p with minimum iteration.

Gradient Descent is an optimization algorithm that helps machine learning models to find out paths to a minimum value using repeated steps.

Gradient descent is used to minimize a function so that it gives the lowest output of that function. This function is called the Loss Function.

Gradient Descent

Let us relate gradient descent with a real-life analogy for better understanding. Think of a valley you would like to descend when you are blind-folded. Any sane human will take a step and look for the slope of the valley, whether it goes up or down. Once you are sure of the downward slope you will follow that and repeat the step again and again until you have descended completely (or reached the minima).

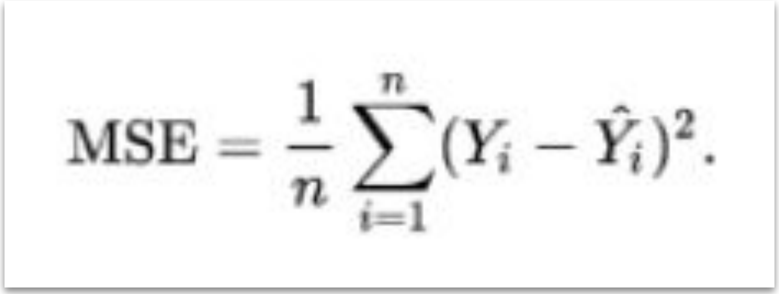


This is exactly what happens in gradient descent. The inclined and/or irregular is the cost function when it is plotted and the role of gradient descent is to provide direction and the velocity (learning rate) of the movement in order to attain the minima of the function i.e where the cost is minimum.

How does Gradient Descent work?

It is always the primary goal of any Gradient descent to minimize/maximize the Cost Function. Minimizing/maximize cost functions will also result in a lower error between the predicted values and the actual values which also denotes that the algorithm has performed well in learning.

The function which is used to minimize for
linear regression model
is the mean squared error.



$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 .$$

Note: MSE is used instead of SSE to save space and avoid memory explosion with large # of observations(n)

Initialize

Regression Equation : Sales(y) = $\boldsymbol{\beta}_0$ + $\boldsymbol{\beta}_1$ * TV ads(X)

Loss function : J($\boldsymbol{\beta}_0$,$\boldsymbol{\beta}_1$) = $\frac{1}{2n}\sum_{i=1}^{n}\left(y - \left(\beta_0 + \beta_1 X\right)\right)^2$

Let us now start by initializing β_0 and β_1. to any value, say 0 for both, and start from there. The algorithm is as follows:

$$\beta_j := \beta_j - \alpha \frac{\partial J\left(\beta_0, \beta_1\right)}{\partial \beta_j} \quad (\,For\ j\ = 0\ and\ j = 1)$$

where α, alpha, is the **learning rate**, or how rapidly do we want to move towards the minimum. We can always overshoot if the value of α is too large.

Delta

Delta value change in parameter is the derivative which refers to the slope of the function. It also gives us to know the direction (sign) in which the coefficient values should move so that they attain a lower cost on the following iteration.

$$\beta_0 : \frac{\partial J(\beta_0, \beta_1)}{\partial \beta_0} = \frac{-1}{n} \sum_{i=1}^{n} \left(y - \left(\beta_0 + \beta_1 X \right) \right)$$

$$\beta_1 : \frac{\partial J(\beta_0, \beta_1)}{\partial \beta_1} = \frac{-1}{n} \sum_{i=1}^{n} \left(y - \left(\beta_0 + \beta_1 X \right) \right) . X$$

Repeat until Convergence

Once we know the direction from the derivative, we can update the coefficient values. Now you need to specify a learning rate parameter which will control how much the coefficients can change on each update.

coefficient = coefficient – (alpha * derivative value)

$$\beta_0 := \beta_0 - \alpha \frac{-1}{n} \sum_{i=1}^{n} \left(y - \left(\beta_0 + \beta_1 X \right) \right)$$

$$\beta_1 := \beta_1 - \alpha \frac{-1}{n} \sum_{i=1}^{n} \left(y - \left(\beta_0 + \beta_1 X \right) \right) . X$$

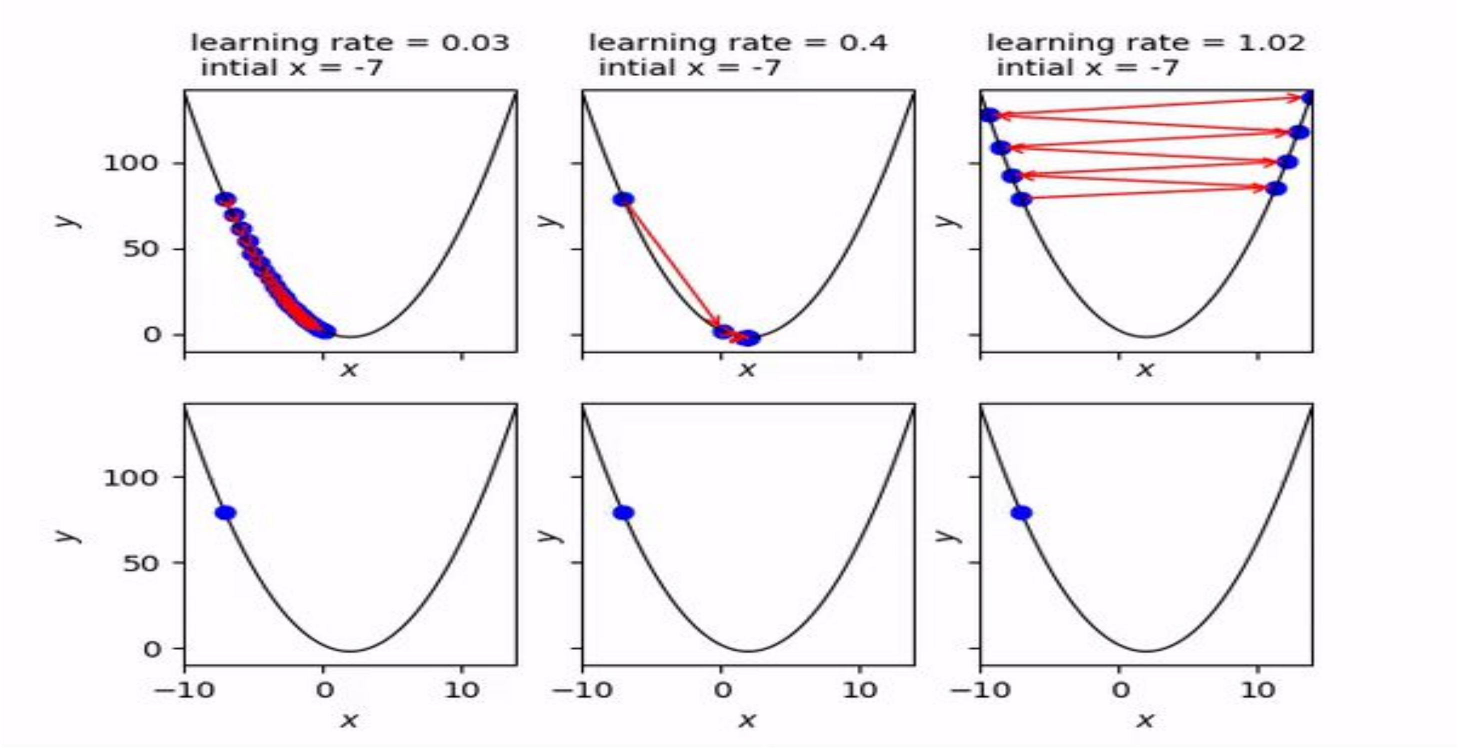
This particular process is repeated as long as the change in coefficients is 0.0 or close enough to zero.

Learning Rate(α)

The learning rate (the size of the adjustments made) is an important hyper-parameter to set carefully.

In practise, it’s better to keep the learning rate small. However, too small and it would take a long to descend toward the global minima. Too large, however, and it may overshoot the global minimum altogether, or even diverge away from it (thus increasing the error).

Learning Rate(α)..contd



Exploring β

β_p :

1. If $\beta_p > 0$, then x (predictor) and y(target) have a positive relationship. That is an increase in x will increase y.
2. If $\beta_p < 0$ then x (predictor) and y(target) have a negative relationship. That is an increase in x will decrease y.

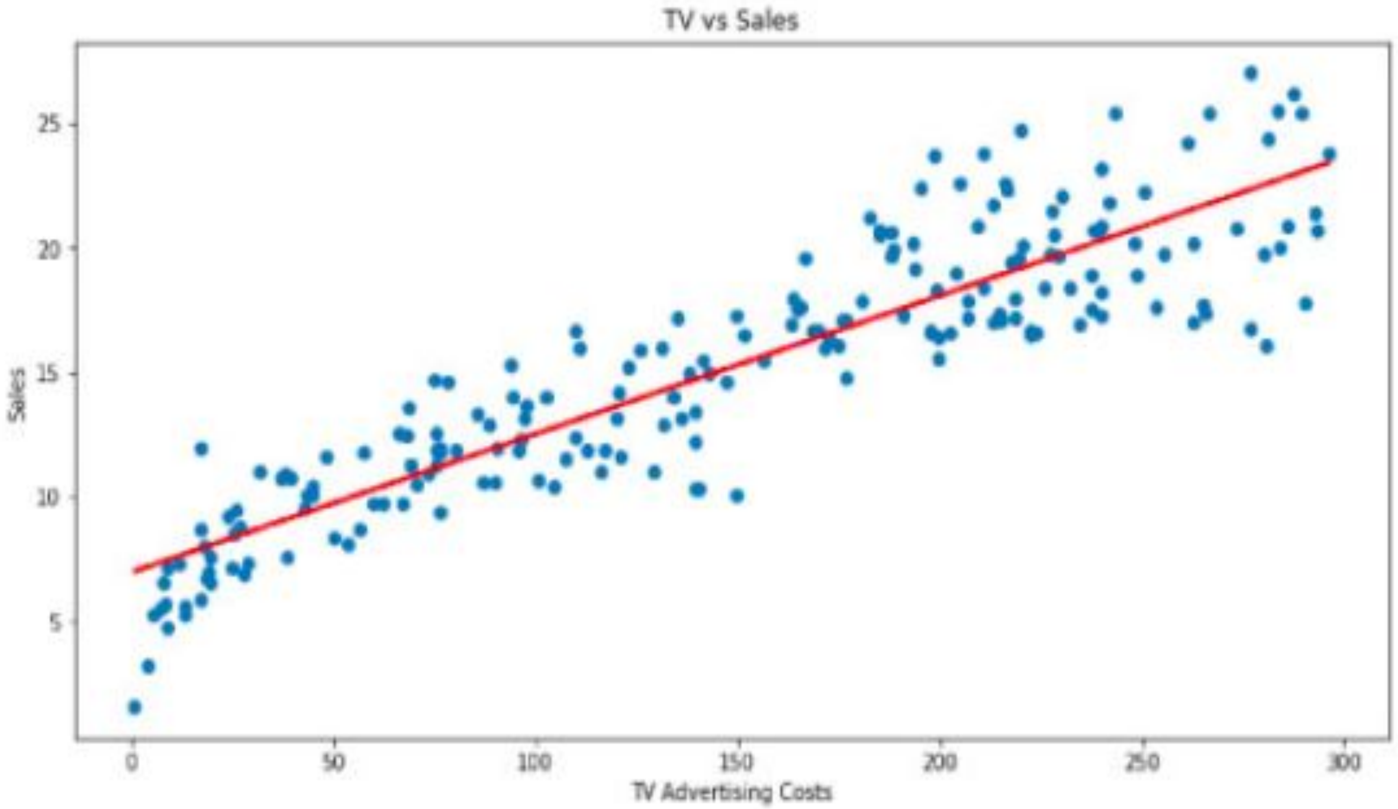
β_0 :

The value of β_0 guarantees that the residual will have mean zero. If there is no $\beta_0$term, then the regression will be forced to pass over the origin. Both the regression coefficient and prediction will be biased.

Assumption of regression line

1. The relation between the dependent and independent variables should be almost linear.
2. Mean of residuals should be zero or close to 0 as much as possible. It is done to check whether our line is actually the line of "best fit".
3. There should be homoscedasticity or equal variance in a regression model. This assumption means that the variance around the regression line is the same for all values of the predictor variable (X).
4. There should not be multicollinearity in regression model. Multicollinearity generally occurs when there are high correlations between two or more independent variables.

Linear Relationship

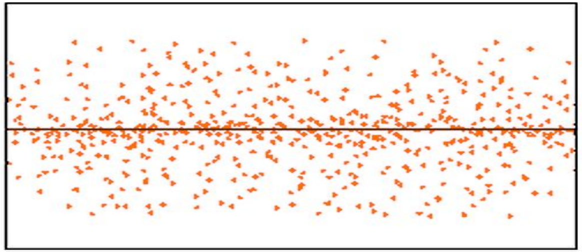


Homoscedasticity

Homoscedasticity (meaning “same variance”) describes a situation in which the error term is the same across all values of the independent variables.

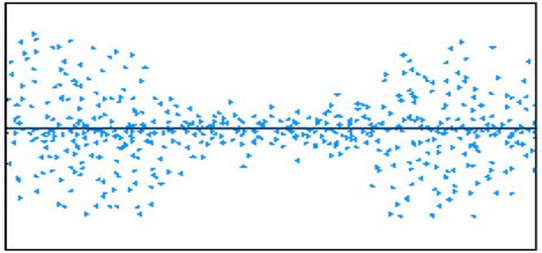
Heteroscedasticity (the violation of homoscedasticity) is present when the size of the error term differs across values of an independent variable. (Scatter plot : Residual vs Fitted value)

Homoscedasticity



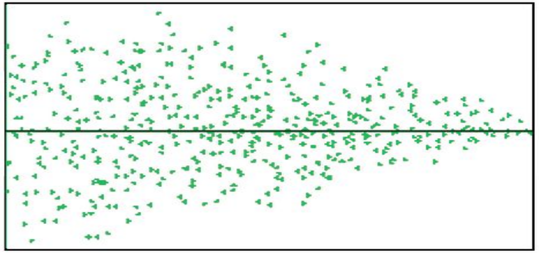
Random Cloud (No Discernible Pattern)

Heteroscedasticity



Bow Tie Shape (Pattern)

Heteroscedasticity



Fan Shape (Pattern)

What if *Heteroscedasticity*?

Heteroscedasticity does not cause bias in the coefficient estimates, it does make them less precise.

Dealing with Heteroscedasticity:

1. Log-transformation of features
2. Outlier treatment
3. Try polynomial fit

Multicollinearity

Multicollinearity occurs when independent variables in a regression model are correlated.

This correlation is a problem because independent variables should be independent. If the degree of correlation between variables is high enough, it can cause problems when you fit the model and interpret the results.

Variance inflation factor(VIF) detects multicollinearity.

A rule of thumb for interpreting the variance inflation factor:

1 = not correlated.

Between 1 and 5 = moderately correlated.

Greater than 5 = highly correlated.

$$\mathrm{VIF} = \frac{1}{1 - R_i^2}$$

What if *Multicollinearity*?

1. Remove some of the highly correlated independent variables.
2. Linearly combine the independent variables, such as adding them together.
3. Perform an analysis designed for highly correlated variables, such as principal components analysis

Properties of Regression Line

1. Regression line passes through the mean of independent variable (x) as well as mean of the dependent variable (y).
2. β_p explains the change in Y with a change in x by one unit. In other words, if we increase the value of ‘x’ it will result in a change in value of Y.
3. The regression constant (β_0) is equal to the y intercept of the regression line.
4. The line minimizes the sum of squared differences between observed values (the y values) and predicted values (the ŷ values computed from the regression equation).

Note: The least squares regression line is the only straight line that has all of these properties.

Advantages of Linear Regression

1. Linear regression is simple to implement and easier to interpret the output coefficients
2. When you know the relationship between the independent and dependent variable is linear, this algorithm is the best to use because it’s less complex as compared to other algorithms
3. It works well irrespective of data size

Limitations of Linear Regression

1. Outliers can have huge effect on the regression line.
2. Linear regression assumes a linear relationship between dependent and independent variables, which is not the case in most of the real world problems.
3. Prone to underfitting - Linear regression sometimes fails to capture the underneath pattern in data properly due to simplicity of the algorithm.

Data Preparation for Linear Regression

1. **Linear Assumption:** Linear regression assumes that the relationship between your independent and dependent is linear. It does not support anything else. This may be obvious, but it is good to remember when you have a lot of attributes. You may need to transform data to make the relationship linear (e.g. log transform for an exponential relationship).
2. **Remove Outlier:** Linear regression assumes that your independent and dependent variables are not noisy. Consider using data cleaning operations that let you better expose and clarify the signal in your data. This is most important for the output variable and you want to remove outliers in the output variable (y) if possible.
3. **Remove Collinearity:** Linear regression will over-fit your data when you have highly correlated input variables. Consider calculating pairwise correlations for your input data and removing the most correlated.
4. **Gaussian Distributions:** Linear regression will make more reliable predictions if your independent and dependent variables have a Gaussian distribution. You may get some benefit using transforms (e.g. log or BoxCox) on you variables to make their distribution more Gaussian looking.
5. **Rescale Inputs:** Linear regression will often make more reliable predictions if you rescale input variables using standardization or normalization.

Regression Model Evaluation Metrics

After the model is built, if we see that the difference in the values of the predicted and actual data is not much, it is considered to be a good model and can be used to make future predictions.

Few metric tools we can use to calculate error in the model

1. MSE (Mean Squared Error)
2. RMSE (Root Mean Squared Error)
3. MAE (Mean Absolute Error)
4. MAPE (Mean Absolute Percentage Error)
5. R^2 (R – Squared)
6. Adjusted R^2

1. Mean Squared Error (MSE)

MSE or Mean Squared Error is one of the most preferred metrics for regression tasks. It is simply the average of the squared difference between the target value and the value predicted by the regression model.

$$MSE = \frac{1}{n} \Sigma \underbrace{\left(y - \widehat{y} \right)^{2}}_{\text{The square of the difference between actual and predicted}}$$

2. Root Mean Squared Error (RMSE)

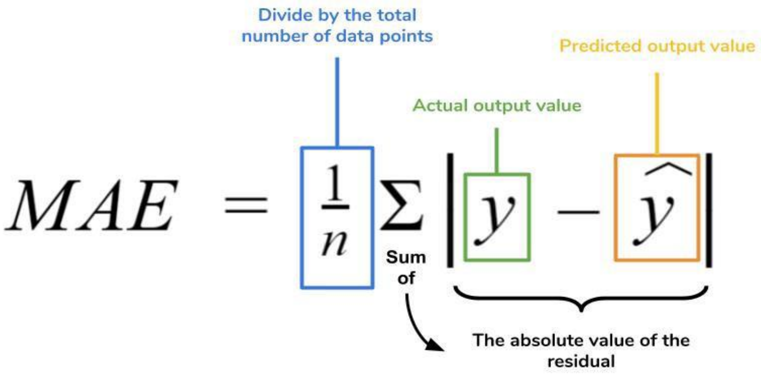
RMSE is the most widely used metric for regression tasks and is the square root of the averaged squared difference between the target value and the value predicted by the model. It is preferred more in some cases because the errors are first squared before averaging which poses a high penalty on large errors.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

3. MAE (Mean Absolute Error)

MAE is the absolute difference between the target value and the value predicted by the model. The MAE is more robust to outliers and does not penalize the errors as extremely as MSE. MAE is a linear score which means all the individual differences are weighted equally.

It is not suitable for applications where you want to pay more attention to the outliers.



$$MAE = \boxed{\frac{1}{n}} \underbrace{\sum}_{\text{Sum of}} \underbrace{\left| \boxed{y} - \boxed{\hat{y}} \right|}_{\text{The absolute value of the residual}}$$

Divide by the total number of data points

Actual output value

Predicted output value

Sum of

The absolute value of the residual

4. MAPE (Mean Absolute Percentage Error)

The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), is a measure of prediction accuracy of a forecasting method in statistics, for example in trend estimation, also used as a loss function for regression problems in machine learning.

$$\mathrm{MAPE} = \frac{100\%}{N} \sum_{i=1}^{N} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

4. R^2 (R – Squared)

Coefficient of Determination or R^2 is another metric used for evaluating the performance of a regression model.

The metric helps us to compare our current model with a constant baseline and tells us how much our model is better.

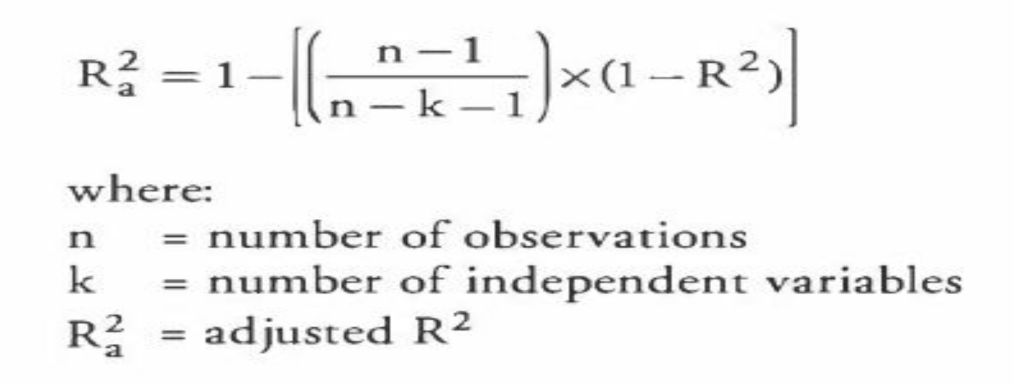
The constant baseline is chosen by taking the mean of the data and drawing a line at the mean.

R^2 is a scale-free score that implies it doesn't matter whether the values are too large or too small, the R^2 will always be less than or equal to 1.

$$R^2 = 1 - \frac{\mathrm{MSE(model)}}{\mathrm{MSE(baseline)}}$$

5. Adjusted R^2

Adjusted R² depicts the same meaning as R² but is an improvement of it. R² suffers from the problem that the scores improve on increasing terms even though the model is not improving which may misguide the researcher.



$$R_a^2 = 1 - \left[\left(\frac{n-1}{n-k-1}\right) \times (1 - R^2)\right]$$

where:

n = number of observations

k = number of independent variables

R_a^2 = adjusted R^2

Adjusted R² is always lower than R² as it adjusts for the increasing predictors and only shows improvement if there is a real improvement.

Properties of R^2

1. R^2 ranges between 0* to 1.
2. R^2 of 0 means that there is no correlation between the dependent and the independent variable.
3. R^2 of 1 means the dependent variable can be predicted from the independent variable without any error.
4. An R^2 of 0.20 means that 20% variance in Y is predictable from X; an R^2 of 0.40 means that 40% variance is predictable.

*Note : R^2 score may range from $-\infty$ to 1 if OLS is not used to get the predictions.