Linear regression:

What is linear regression?

🡪 Regression is one of the easiest and most popular machine learning algorithms

🡪 it is based on supervised leaning.

🡪 it performs a regression task.

🡪 it is a statistical method that is used for predictive analysis.

Linear regression algorithm shows a relationship between dependent(y) target variables) and one or more independent(x) predictor variables. hence it is called as linear regression.

The linear regression model provides a sloped straight line representing the relationship between the variables.



Mathematically, we can represent a linear regression as:

y= a0+a1x+ ε

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.

Types of Linear Regression:

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

**🡪Simple Linear Regression**

**🡪Multiple Linear regression**

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:

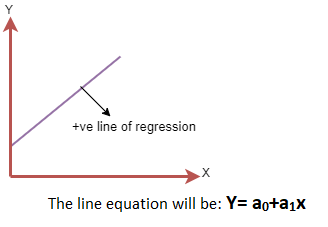
If more than one independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression.

## Linear Regression Line:

A linear line showing the relationship between the dependent and independent variables is called a **regression line**. A regression line can show two types of relationship:

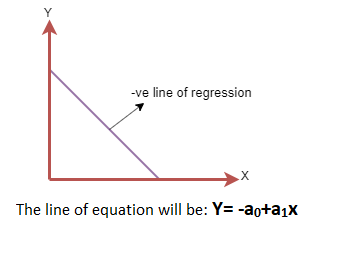
**Positive Linear Relationship:**

* If the dependent variable increases on the Y-axis and independent variable increases on X-axis, then such a relationship is termed as a Positive linear relationship.



**Negative Linear Relationship:**

* If the dependent variable decreases on the Y-axis and independent variable increases on the X-axis, then such a relationship is called a negative linear relationship.



## Finding the best fit line:

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.

## Assumptions of linear regression:

## *1.linearity:* The relationship between x and the mean of y is linear.

## *2.homoscedasticity:* The variance of residual is the same for any value of x.

## *3.independence:* observations are independent of each other.

## *4.normality:* for any fixed value of x, y is normally distributed.

## Advantages of linear regression:

## 1.linear regression perform exceptionally well for linearly separable data.

## 2.easy to implement and train the model.

## 3.it can handle overfitting using dimensionality reduction techniques and cross validity regularization.

## Disadvantages of linear regression:

## 1.sometimes lot of feature engineering is required

## 2.if the independent features are corelated it may affect performance

## 3.it is often quite prone to noise and overfitting.

## Evaluation metrics for a linear regression model:

## Evaluation metrics are a measure of how good a model performs and how well it approximates the relationship.

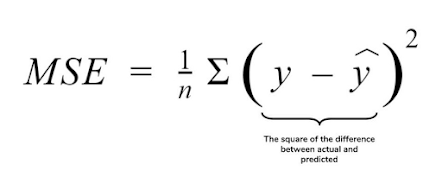
## Let us look at MSE, MAE, R-squared, Adjusted R-squared, and RMSE.

To evaluate the performance of the model is known as metrics.

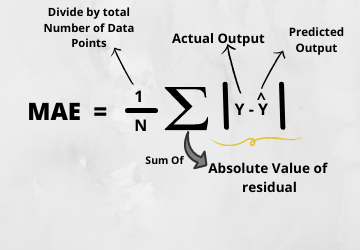
* [1) Mean Absolute Error (MAE)](https://www.analyticsvidhya.com/blog/2021/05/know-the-best-evaluation-metrics-for-your-regression-model/#h2_5)
* [2) Mean Squared Error (MSE)](https://www.analyticsvidhya.com/blog/2021/05/know-the-best-evaluation-metrics-for-your-regression-model/#h2_6)
* [3) Root Mean Squared Error (RMSE)](https://www.analyticsvidhya.com/blog/2021/05/know-the-best-evaluation-metrics-for-your-regression-model/#h2_7)
* [4) Root Mean Squared Log Error (RMSLE)](https://www.analyticsvidhya.com/blog/2021/05/know-the-best-evaluation-metrics-for-your-regression-model/#h2_8)
* [**5) R Squared (R2)**](https://www.analyticsvidhya.com/blog/2021/05/know-the-best-evaluation-metrics-for-your-regression-model/#h2_9)

## **Residual = actual value — predicted value**

## Mean Squared Error (MSE)



## Mean Absolute Error (MAE)

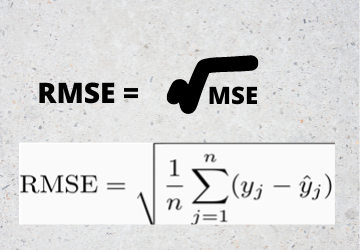


## N=the count of data points,

## y=the actual value in the data set,

## y^= the model of predicted value.

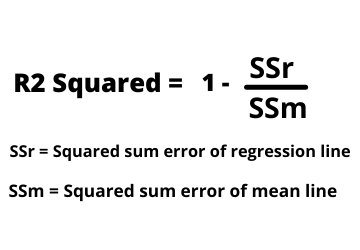
## Root Mean Squared Error(RMSE)



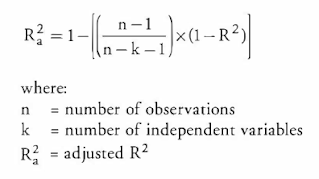
## Root Mean Squared Log Error(RMSLE)

print("RMSE",np.log(np.sqrt(mean\_squared\_error(y\_test,y\_pred))))

## R Squared (R2)



### Adjusted R Squared



 MAE,MSE,RMSE,🡪We can expect closer to 0

Rsq,adjR,🡪here we can expect closer to 1.(i.e. it shows a good model)

## What is Lasso Regression?

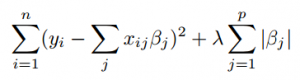
**Lasso regression** is a type of [**linear regression**](https://www.statisticshowto.com/probability-and-statistics/regression-analysis/find-a-linear-regression-equation/)that uses [shrinkage](https://www.statisticshowto.com/shrinkage-estimator/). Shrinkage is where data values are shrunk towards a central point, like the [mean](https://www.statisticshowto.com/mean/). The lasso procedure encourages simple, sparse models (i.e. models with fewer parameters). This particular type of regression is well-suited for models showing high levels of [muticollinearity](https://www.statisticshowto.com/multicollinearity/) or when you want to automate certain parts of model selection, like variable selection/parameter elimination.

The acronym “LASSO” stands for **L**east **A**bsolute **S**hrinkage and **S**election **O**perator.

## L1 Regularization

Lasso regression performs L1 [regularization](https://www.statisticshowto.com/regularization/), which adds a penalty equal to the[absolute value](https://www.statisticshowto.com/integer/#abs)of the magnitude of coefficients. This type of regularization can result in sparse models with few coefficients; Some coefficients can become zero and eliminated from the model. Larger penalties result in coefficient values closer to zero, which is the ideal for producing simpler models. On the other hand, L2 regularization (e.g. [Ridge regression](https://www.statisticshowto.com/ridge-regression/)) doesn’t result in elimination of coefficients or sparse models. This makes the Lasso far easier to interpret than the Ridge.

## Performing the Regression

Lasso solutions are quadratic programming problems, which are best solved with software (like [Matlab](https://www.mathworks.com/help/stats/lasso.html)). The goal of the algorithm is to minimize:  
[](https://www.statisticshowto.com/wp-content/uploads/2015/09/lasso-regression.png)  
  
  
Which is the same as minimizing the [sum of squares](https://www.statisticshowto.com/residual-sum-squares/)with constraint Σ |Bj≤ s (Σ = [summation notation](https://www.calculushowto.com/calculus-definitions/summation-notation-sigma-function/)). Some of the βs are shrunk to exactly zero, resulting in a regression model that’s easier to interpret.

A [**tuning parameter**](https://www.statisticshowto.com/tuning-parameter/), λ controls the strength of the L1 penalty. λ is basically the amount of shrinkage:

* When λ = 0, no parameters are eliminated. The estimate is equal to the one found with linear regression.
* As λ increases, more and more coefficients are set to zero and eliminated (theoretically, when λ = ∞, all coefficients are eliminated).
* As λ increases, [bias](https://www.statisticshowto.com/what-is-bias/)increases.
* As λ decreases, [variance](https://www.statisticshowto.com/probability-and-statistics/variance/)increases.

If an [intercept](https://www.calculushowto.com/problem-solving/find-intercepts-x-y/) is included in the model, it is usually left unchanged.

## Lasso Regression

LASSO stands for Least Absolute Shrinkage and Selection Operator. I know it doesn’t give much of an idea but there are 2 key words here – ‘absolute‘ and ‘selection‘.

Lets consider the former first and worry about the latter later.

Lasso regression performs **L1 regularization**, i.e. it adds a factor of sum of absolute value of coefficients in the optimization objective. Thus, lasso regression optimizes the following:

#### **Objective = RSS + α \* (sum of absolute value of coefficients)**

Here, α (alpha) works similar to that of ridge and provides a trade-off between balancing RSS and magnitude of coefficients. Like that of ridge, α can take various values. Lets iterate it here briefly:

1. α = 0: Same coefficients as simple linear regression
2. α = ∞: All coefficients zero (same logic as before)
3. 0 < α < ∞: coefficients between 0 and that of simple linear regression

## ****What is Ridge Regression?****

Ridge [regression](https://www.mygreatlearning.com/blog/what-is-regression/) is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values.

Contributed by: [*Prashanth Ashok*](https://www.linkedin.com/in/prashanth-a-bb122425/)

The cost function for ridge regression:

**Min(||Y – X(theta)||^2 + λ||theta||^2)**

Lambda is the penalty term. λ given here is denoted by an alpha parameter in the ridge function. So, by changing the values of alpha, we are controlling the penalty term. The higher the values of alpha, the bigger is the penalty and therefore the magnitude of coefficients is reduced.

* It shrinks the parameters. Therefore, it is used to prevent multicollinearity
* It reduces the model complexity by coefficient shrinkage

## ****Ridge Regression Models****

For any type of regression machine learning model, the usual regression equation forms the base which is written as:

**Y = XB + e**

Where Y is the dependent variable, X represents the independent variables, B is the regression coefficients to be estimated, and e represents the errors are residuals.

Once we add the lambda function to this equation, the variance that is not evaluated by the general model is considered. After the data is ready and identified to be part of L2 regularization, there are steps that one can undertake.

## ****Standardization****

In ridge regression, the first step is to standardize the variables (both dependent and independent) by subtracting their means and dividing by their standard deviations. This causes a challenge in notation since we must somehow indicate whether the variables in a particular formula are standardized or not. As far as standardization is concerned, all ridge regression calculations are based on standardized variables. When the final regression coefficients are displayed, they are adjusted back into their original scale. However, the ridge trace is on a standardized scale.

## ****Bias and variance trade-off****

Bias and variance trade-off is generally complicated when it comes to building ridge regression models on an actual dataset. However, following the general trend which one needs to remember is:

1. The bias increases as λ increases.
2. The variance decreases as λ increases.

## ****Assumptions of Ridge Regressions****

The assumptions of ridge regression are the same as that of linear regression: linearity, constant variance, and independence. However, as ridge regression does not provide confidence limits, the distribution of errors to be normal need not be assumed.

Now, let’s take an example of a linear regression problem and see how ridge regression if implemented, helps us to reduce the error.

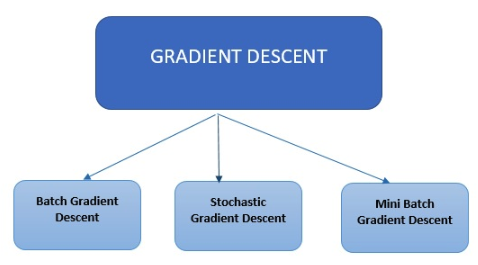
We shall consider a data set on Food restaurants trying to find the best combination of food items to improve their sales in a particular region.

**Gradient descent:**

Gradient descent is the most commonly used optimization method deployed in machine learning and deep learning algorithms. It’s used to train a machine learning model.

**There are three types of gradient descent learning algorithms: batch gradient descent, stochastic gradient descent and mini-batch gradient descent.**

* Batch gradient descent. ...
* Stochastic gradient descent. ...
* Mini-batch gradient descent.



### **Batch Gradient Descent**

Batch Gradient Descent is the most straightforward type. It calculates the error for each example in the training dataset, however, it only updates the model after all training examples have been evaluated.

### **Stochastic Gradient Descent**

## Stochastic Gradient Descent calculates the error and updates the model for each example in the training dataset.

### **Mini Batch Gradient Descent** SGD can be used when the dataset is large. Batch Gradient Descent converges directly to minima. SGD converges faster for larger datasets. But, since in SGD we use only one example at a time, we cannot implement the vectorized implementation on it.

**Mini Batch Gradient Descent Batch : A Compromise**

* Easily fits in the memory.
* It is computationally efficient.
* Benefit from vectorization.
* If stuck in local minimums, some noisy steps can lead the way out of them.
* Average of the training samples produces stable error gradients and convergence

## Mathematical & Geometrical Intuition of Linear Regression:

## Step 1: Plot of the Independent & Dependent Variables. Draw the best fit line (Approx.).

## Best Fit Line is determined based on the sum of errors to be minimum.s

## Step 2: Calculate the individual errors.

## Error is defined based on the actual & the predicted value. Below is the formula to calculate the error for individual points.

## In our Sample Example, we have 3 points which are not in line with our best fit line. So we need to calculate the error function.

## Error=y-y^

## The calculation of error can conclude the status of selecting the best fit line.

## Step 3: Calculating the minimum sum of squares of errors or Ordinary Least-squares.

## As we have calculated the individual Error for all the points, we are now going to sum & consider the min value to evaluate the best fit line by using the below formula.

## 

## Where,

## 

## 

## Once we have our Best Fit Line on the Linear Model, we are having the freedom to predict the dependent values based on the given independent variables.

## 