UNIT - 2

Regression

Outlines

- Regression –Intro
- Types of Regression
- Simple Linear Regression
- Least Square Method
- Multiple Regression
- Polynomial Regression
- Issues in Regression
- Overfitting
- Underfitting
- Bias and Variance
- Assumptions
- Application
- Evaluation Metrics

Supervised Learning

> Supervised Learning is a type of machine learning used to train models from labeled training data. It allows you to predict output for future or unseen data.

Supervised Learning: Case Study

Netflix uses **supervised learning** algorithms to recommend users the shows they may watch based on the viewing history and ratings by similar classes of users

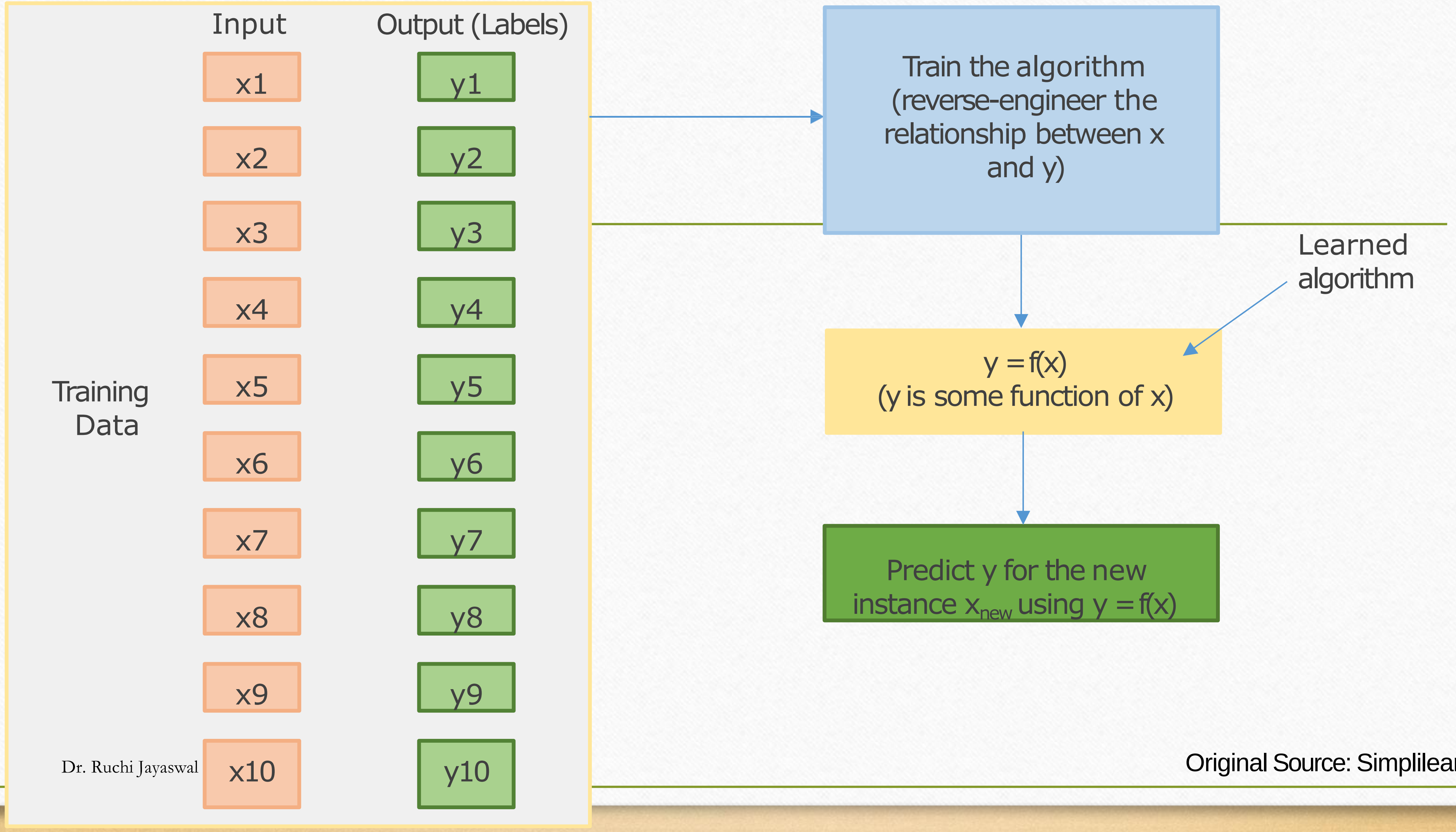
New Input

Algorithm Trained on Historical Data

Predicted Outcome

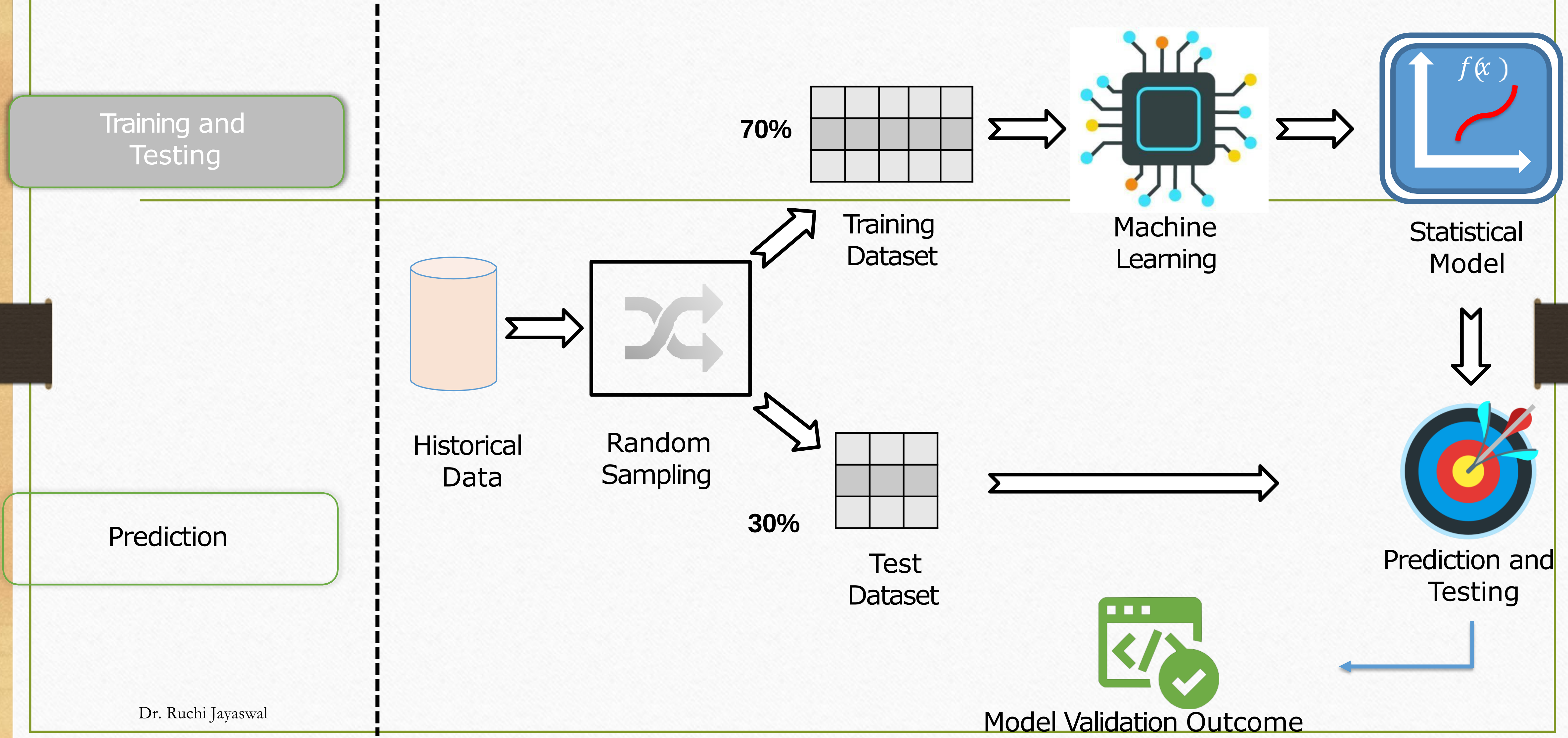
Original Source: Simplilearn.com

Understanding the Algorithm



Original Source: Simplilearn.com

Supervised Learning Flow



Dr. Ruchi Jayaswal

Pooja Kamat

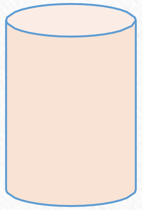
Original Source: Simplilearn.com

Supervised Learning Flow

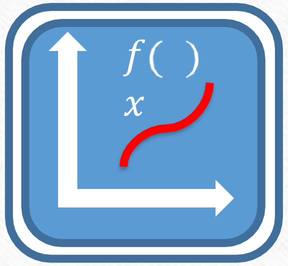
Training and Testing

Prediction

Use the learned algorithm y = f(x) to predict production data



New Data



Model

Prediction Outcome

Algorithm prediction can be improved by more training data, capacity, or algorithm redesign.

Original Source: Simplilearn.com

TYPES OF SUPERVISED LEARNING

Types of Supervised Learning

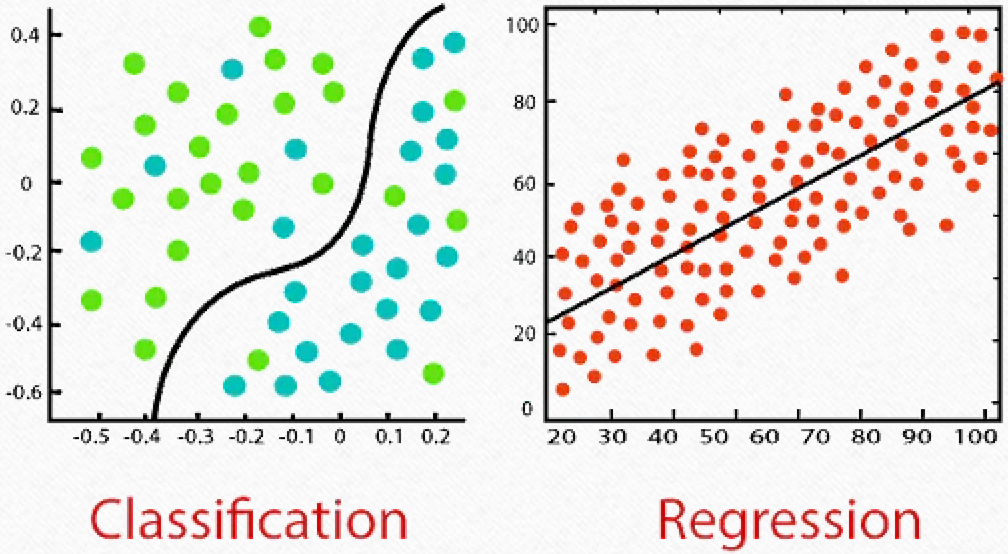
Supervised Learning

1 Classification

2 Regression

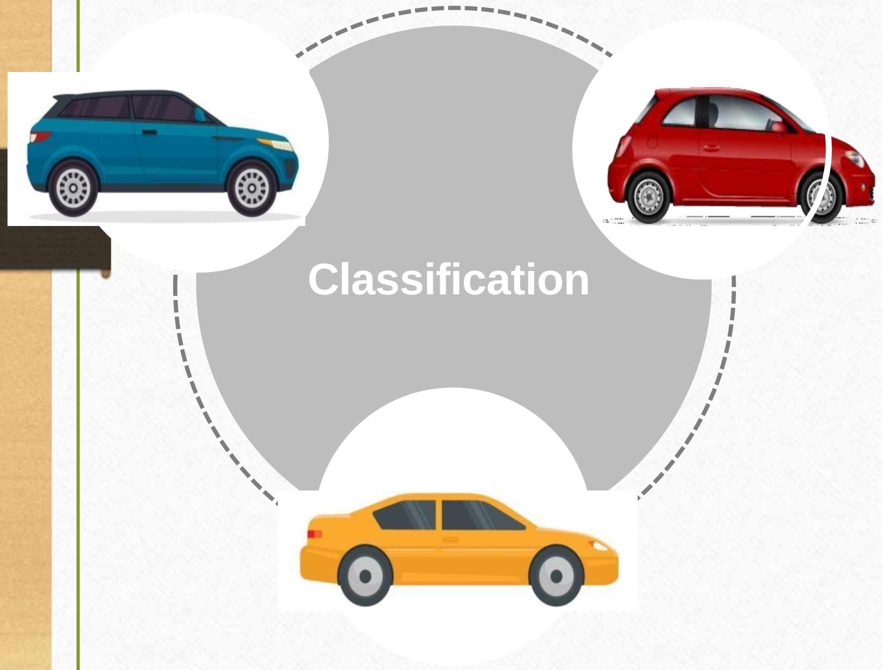
In supervised learning, algorithm is selected based on target variable.

Regression vs Classification



Types of Supervised Learning (Contd.)

If target variable is categorical (classes), then use classification algorithm.



In other words, classification is applied when the output has finite and discreet values.

Example: Predict the class of car given its features like horsepower, mileage, weight, colour, etc.

The classifier will build its attributes based on these features. Analysis has three potential outcomes - Sedan, SUV, or Hatchback

Types of Supervised Learning (Contd.)

If target variable is a continuous numeric variable (100–2000), then use a regression algorithm.

Example: Predict the price of a house given its sq. area, location, no of bedrooms, etc.

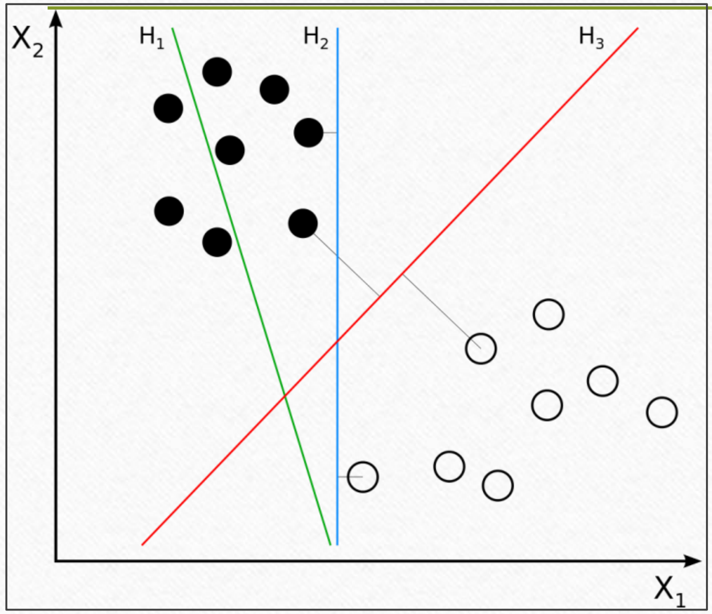
A simple regression algorithm is given below

$$y = w * x + b$$

This shows relationship between price (y) and sq. area (x)
where price is a number from a defined range.

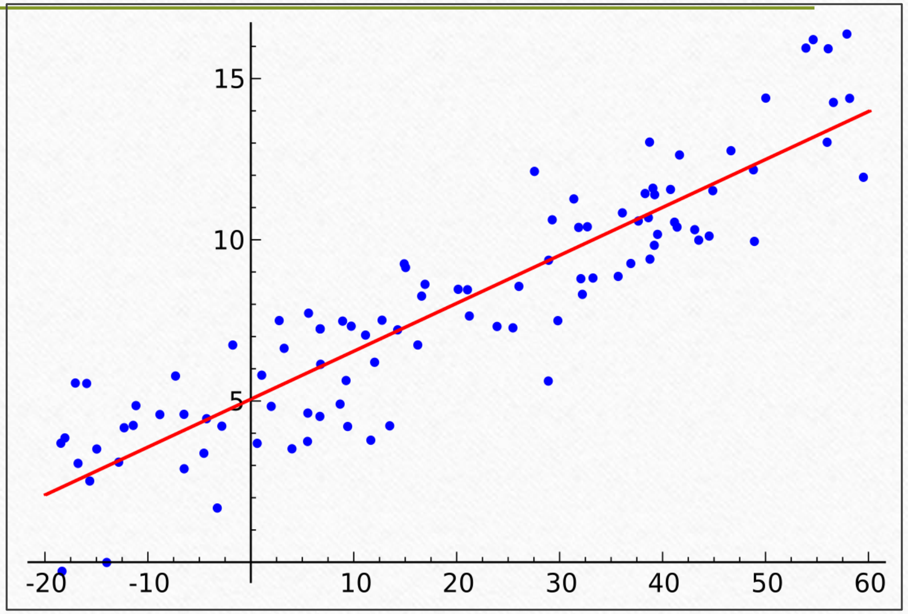
Types of Supervised Learning (Contd.)

Classification



What Class ?

Regression



How much ?

TYPES OF REGRESSION ALGORITHMS

Types of regression algorithms

Linear

Polynomial

Multiple Linear

Ridge

ElasticNet

Lasso

Types of Regression Algorithms

- Linear Regression
- Multiple Linear Regression
- Polynomial Regression

Linear Regression is a statistical model used to predict the relationship between independent and dependent variables denoted by x and y respectively

Examine 2 factors

1

How closely are x and y related ?

Linear regression gives a number between -1 and 1 indicating the strength of correlation between the two variables

1 : no correlation
2 : positively correlated
1 : negatively correlated

2

Prediction

When the relationship between x and y is known, use this to predict future values of y for a value of x

This is done by fitting a regression line and represented by a linear equation:

***y* = *m* * x + b**

Regression

Dr. Ruchi Jayaswal

Linear Regression

- Linear Regression is a supervised machine learning algorithm.
- It tries to find out the best linear relationship that describes the data you have.
- It assumes that there exists a linear relationship between a dependent variable and independent variable(s).
- The value of the dependent variable of a linear regression model is a continuous value i.e. real numbers.

Example: Suppose there is a marketing company A, who does various advertisement every year and get sales on that. The below list shows the advertisement made by the company in the ales:

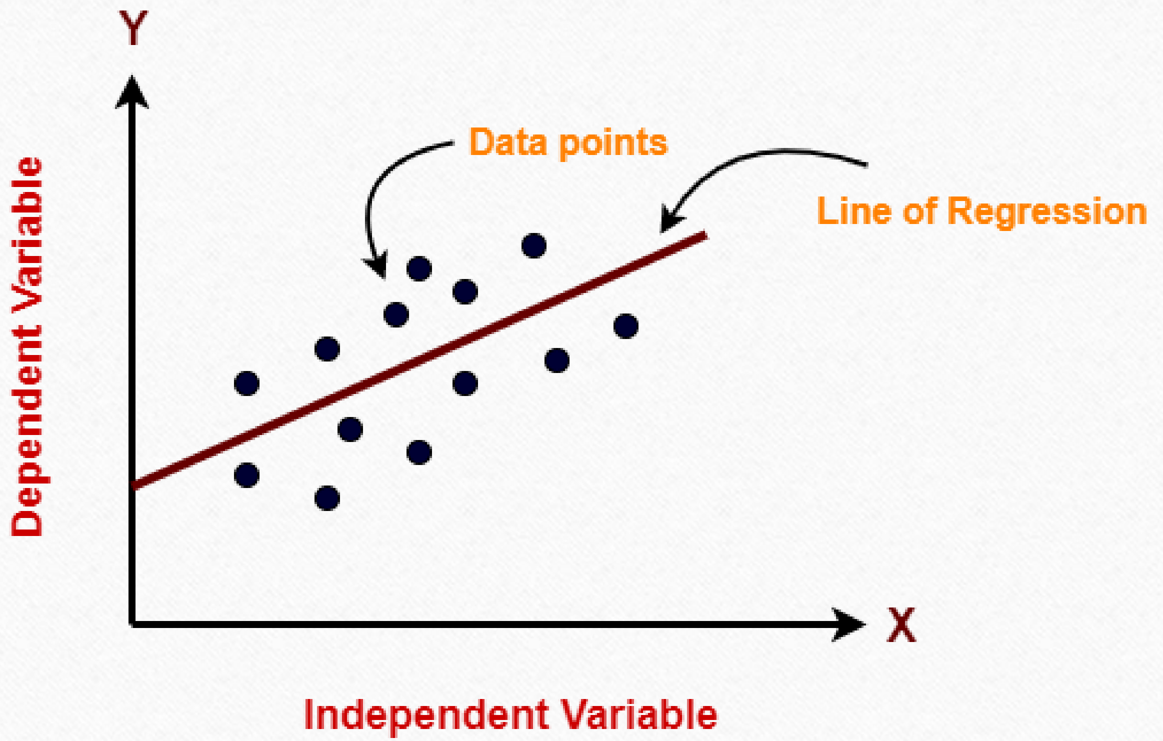
Advertisement	Sales
$90	$1000
$120	$1300
$150	$1800
$100	$1200
$130	$1380
$200	??

Now, the company wants to do the advertisement of $200 in the year 2019 **and wants to know the prediction about the sales for this year**. So to solve such type of prediction problems in machine learning, we need regression analysis.

- It is mainly used for prediction, forecasting, time series modeling, and determining the causal-effect relationship between variables.
- Regression shows a line or curve that passes through all the datapoints on target-predictor graph in such a way that the vertical distance between the datapoints and the regression line is minimum.

Representing Linear Regression Model-

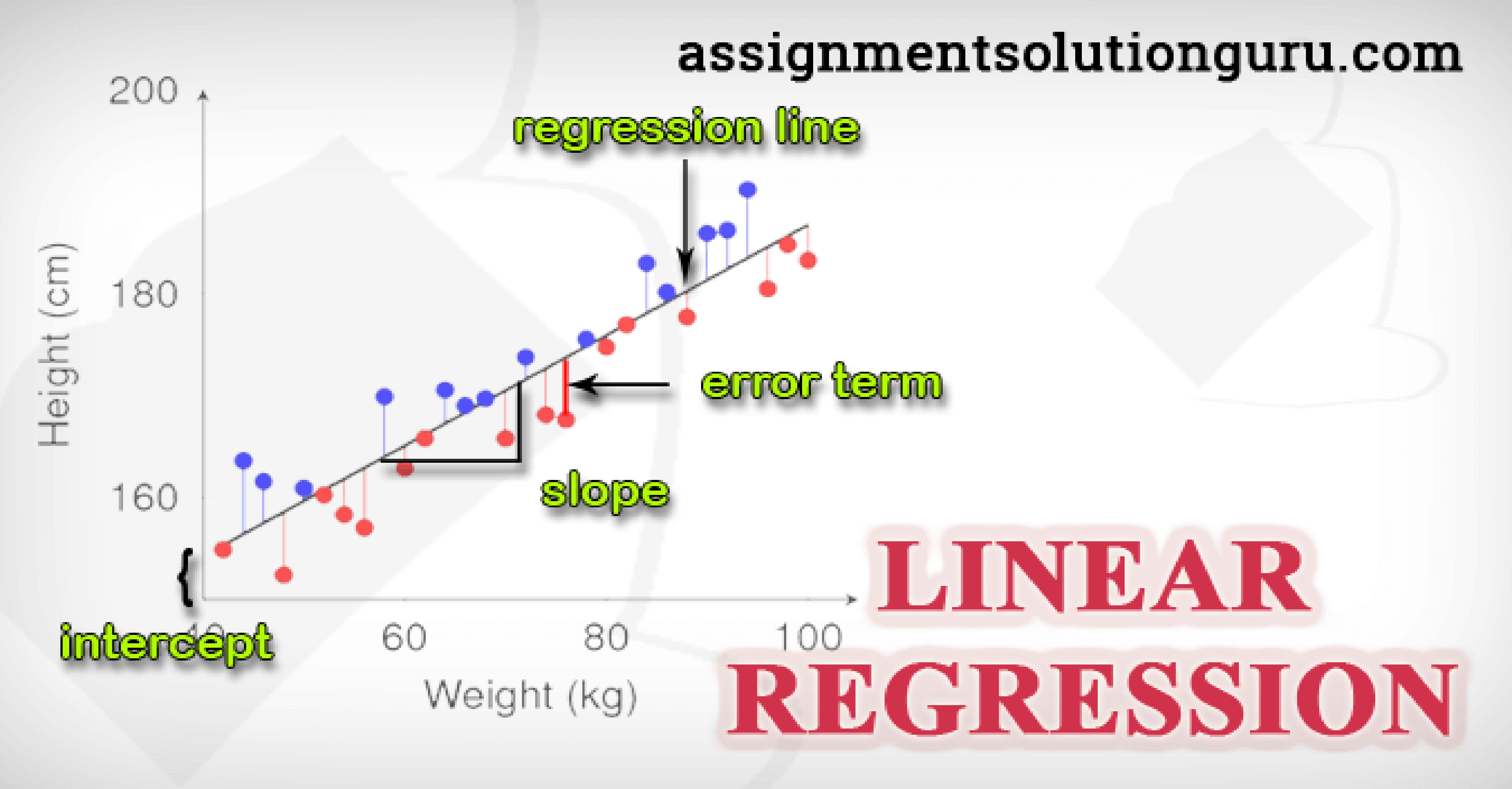
- Linear regression model represents the linear relationship between a dependent variable and independent variable(s) via a sloped straight line.



Terminologies Related to the Regression

- **Independent Variables**: The variable that are not affected by the other variables are called independent variables. For example age of a person, is an independent variable, two person's born on same date will have same age irrespective of how they lived. We presume that while independent variables are stable and cannot be manipulated by some other variable.
- **Dependent Variables**: The variables which depend on other variables or factors. We expect these variables to change when the independent variables, upon whom they depend, undergo a change. For example let us say you have a test tomorrow, then, your test score is dependent upon the amount of time you studied, so the test score is a dependent variable, and amount of time independent variable in this case.

- **Outliers:** Outlier is an observation which contains either very low value or very high value in comparison to other observed values. An outlier may hamper the result, so it should be avoided.



assignmentsolutionguru.com
200
180
160
Height (cm)
regression line
error term
slope
intercept
60
80
100
Weight (kg)
LINEAR
REGRESSION

Some examples of regression can be as:

- Prediction of rain using temperature and other factors
- Determining Market trends
- Prediction of road accidents due to rash driving.

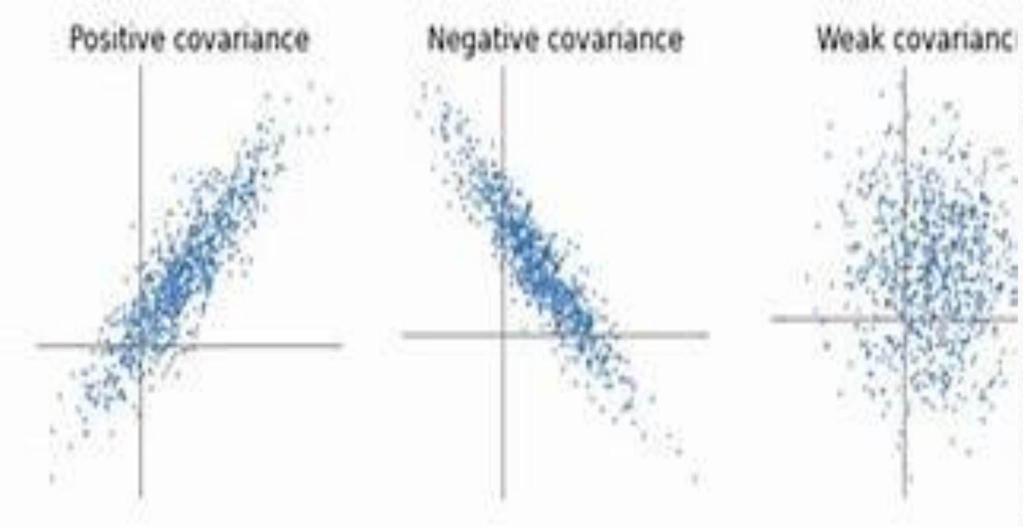
Concepts

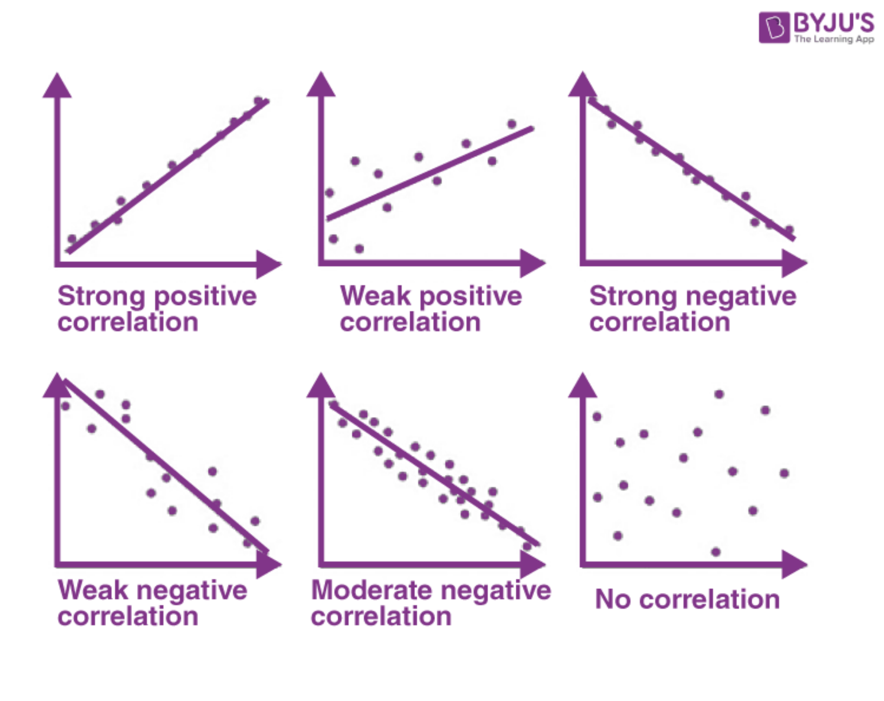
- Covariance: It is the relationship between a pair of random variables in one variable causes change in another variable. It can take any value between – infinity to +infinity.
- Correlation: It show whether and how strongly pairs of variables are related to each other.

  Range: -1 to +1

  Variables are indirectly related to each other.

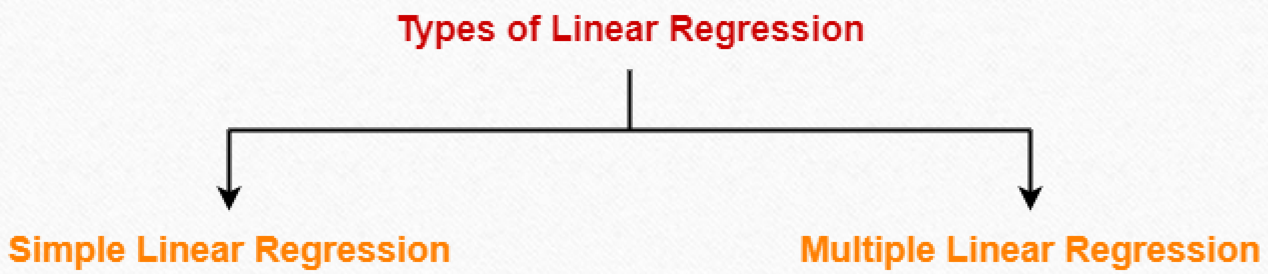
Example


Positive covariance
Negative covariance
Weak covarianc


BYJU'S
The Learning App
Strong positive correlation
Weak positive correlation
Strong negative correlation
Weak negative correlation
Moderate negative correlation
No correlation

Types of Linear Regression-

Based on the number of independent variables, there are two types of linear regression-



Simple Regression

In simple linear regression, the dependent variable depends only on a single independent variable.

For simple linear regression, the form of the model is-

$$\mathbf{Y = \beta_0 + \beta_1 X}$$

Here,

Y is a dependent variable.

X is an independent variable.

β_0 and β_1 are the regression coefficients.

β_0 is the intercept or the bias that fixes the offset to a line.

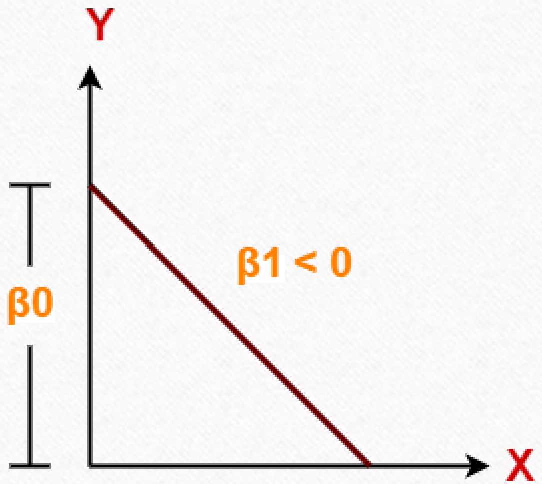
β_1 is the slope or weight that specifies the factor by which X has an impact on Y.

There are following 3 cases possible-

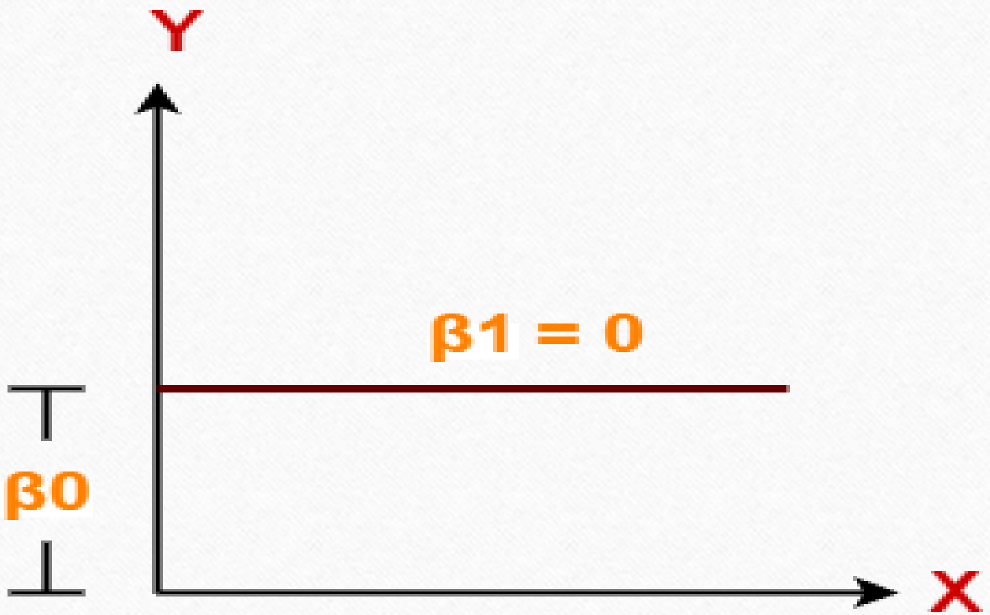
Case-01: $\beta_1 < 0$

It indicates that variable X has negative impact on Y.

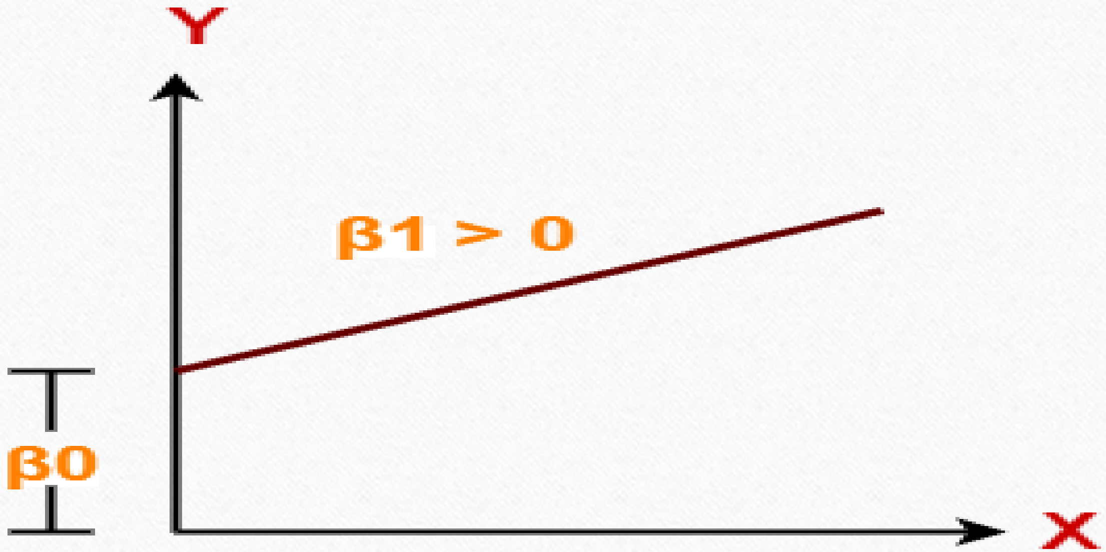
If X increases, Y will decrease and vice-versa.



- **Case-02: $\beta_1 = 0$**
- It indicates that variable X has no impact on Y.
- If X changes, there will be no change in Y.



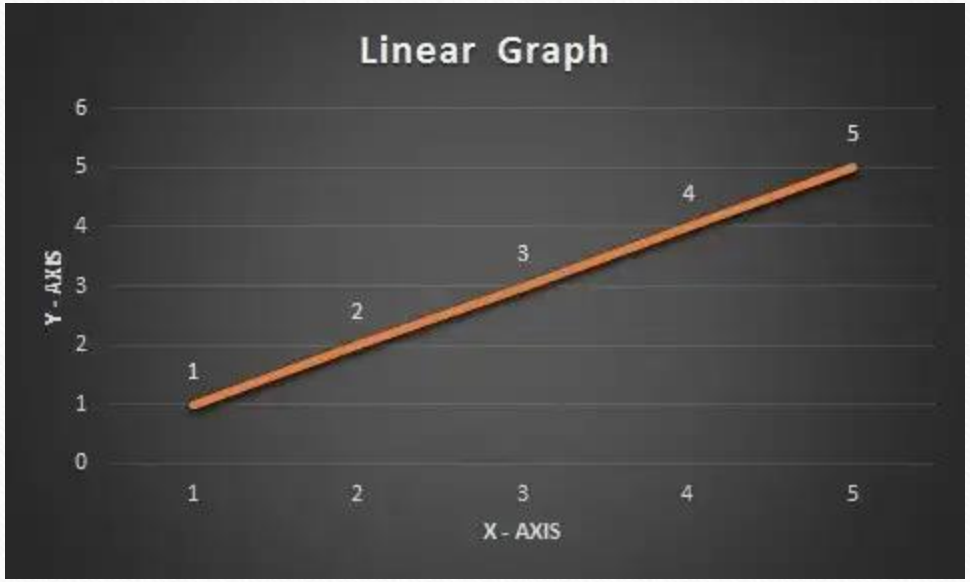
- **Case-03: $\beta_1 > 0$**
- It indicates that variable X has positive impact on Y.
- If X increases, Y will increase and vice-versa.



Simple Linear Regression

- *Linear Regression is a predictive algorithm which provides a Linear relationship between* **Prediction** *(Call it* **'Y'***) and* ***Input*** *(Call is* **'X'***).*

- (Input) X = 1,2,3,4,5
  (Prediction) Y = 1,2,3,4,5



Linear Regression with Real World Example

- Let's take a real world example of the price of agricultural products and how it varies based on the location its sold. The price will be low when bought directly from farmers and high when brought from the downtown area.
- Given this dataset, we can predict the price of the product in intermediate locations.

In this example, if we consider Input 'X — Axis' as
Sale Location and 'Y — Axis' as Price (think of any currency you're familiar with), we can plot the graph as

Problem Statement

Given this dataset, predict the price of agricultural product, if it’s sold in intermediate locations between farmers house and city downtown.

Training DataSet

The dataset provided above can be considered as Training DataSet for the problem statement stated above, If we consider these inputs as Training Data for the model, we can use that model to predict the price at locations between

- Farmers home — Village
- Village — Town
- Town — City
- City — City Downtown

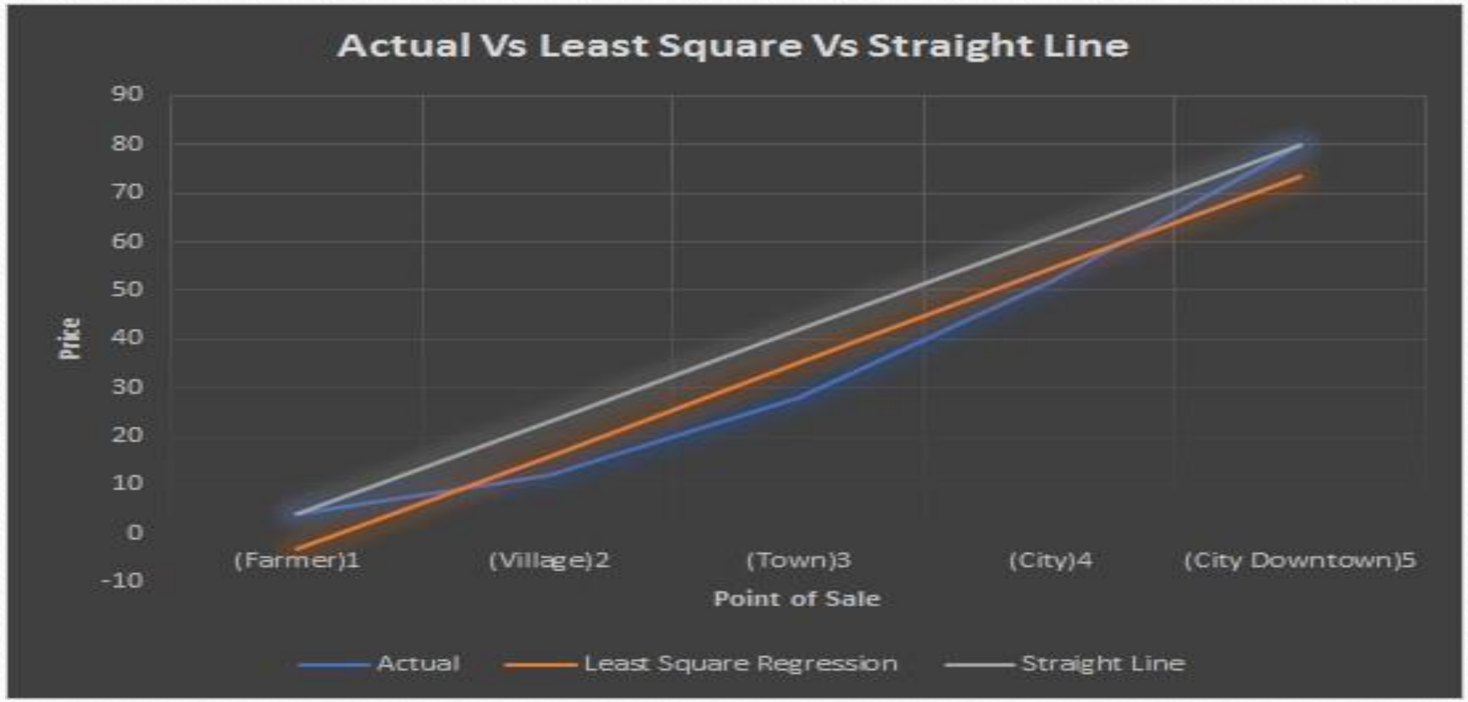
Our aim is to come with a straight line which minimizes the error between training data and our prediction model when we draw the line using the equation of straight line.

- Straight Line Equation
- Least Square Method

Least Square Method

m =   Sum of ALL (x - xmean)*(y - yMean)

Sum of (x - xmean)**2


Actual Vs Least Square Vs Straight Line
90
80
70
60
50
40
30
20
10
0
-10
Price
(Farmer)1
(Village)2
(Town)3
(City)4
(City Downtown)5
Point of Sale
Actual
Least Square Regression
Straight Line

- **Why this method is called Least Square Regression ?**
- This method is intended to reduce the sum square of all error values. The lower the error, lesser the overall deviation from the original point. We can compare the same with the errors generated out of the straight line as well as with the Least Square Regression
-

Error in 'Y' With Straight Line Equation

At X-Value	Y-Value	Actual-Value	Error	Square-Error
1	4	4	0	0
2	12	23	-11	121
3	28	42	-14	196
4	52	61	-9	81
5	80	80	0	0

Sum of Square Error = { 0 + 121 + 196 + 81 + 0 } = 398

Error in 'Y' With Least Square Equation

At X-Value	Y-Value	Actual-Value	Error	Square-Error
1	4	-3.2	7.2	51.84
2	12	16	-4	16
3	28	35.2	-7.2	51.84
4	52	54.4	-2.4	5.76
5	80	73.6	6.4	40.96

Sum of Square Error = { 51.84 + 16 + 51.84 + 5.76 + 40.96 } = 166.4

- we can see that Least Square Method provide better results than a plain straight line between two points calculation.
- Example:

x	y
2	4
3	6
2	6
3	8

Types of Regression Algorithms (Contd.)

Multiple linear regression is a statistical technique used to predict the outcome of a response variable through several explanatory variables and model the relationships between them.

Linear Regression

Multiple Linear Regression

Polynomial Regression

Equation for MLR

$$Y = m_1 * x_1 + m_2 * x_2 + m_3 * x_3 + \ldots\ldots + m_n * x_n + c$$

Dependent Variable

m1, m2, m3 …. m_n

Slopes

Coefficient

$$Y_i = w_0+w_1x_i+w_2x_{12}+\text{----------------}w_j\, x_{ij}$$

$$Y_i = w_0 + \sum w_i x_{ij}$$

Y_i= predicted label of i^{th} sample

X_{ij}= the j^{th} features of i^{th} label

W_o= intercept

W_j= j_{th} feature regression weight

In Matrix notation:

$$Y=XW$$

Types of Regression Algorithms (Contd.)

Linear Regression

Multiple Linear Regression

Polynomial Regression

Polynomial regression is applied when data is not formed in a straight line.

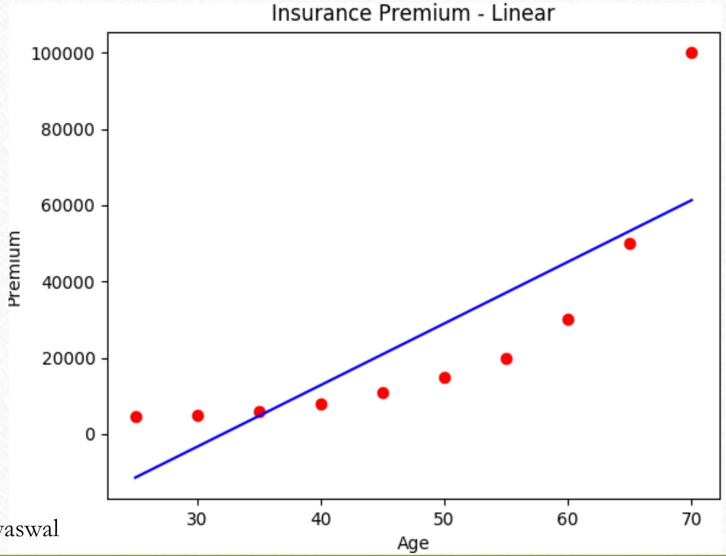
It is used to fit a linear model to non-linear data by creating new features from powers of non-linear features.

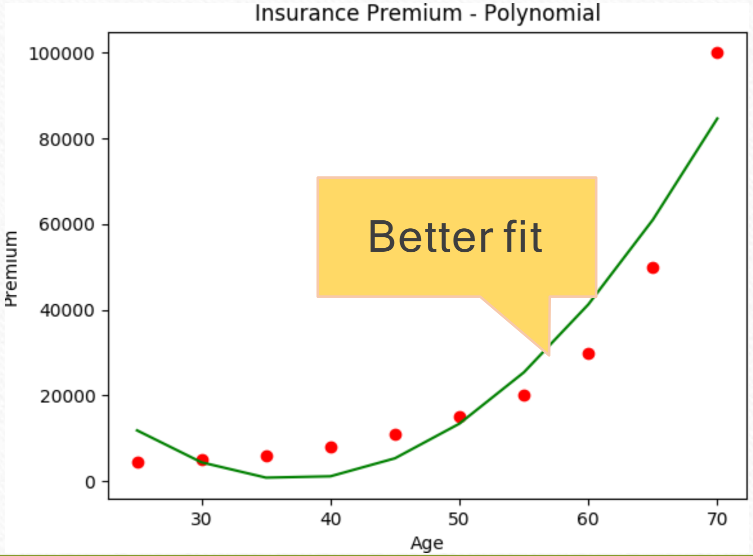
Example: Quadratic features

$x_2' = x_2^2$

$y = w_1x_1 + w_2x_2^2 + 6$

$= w_1x_1 + w_2x_2' + 6$





Polynomial Regression

$$y = b_0 + b_1 x_1$$

Simple Linear Regression

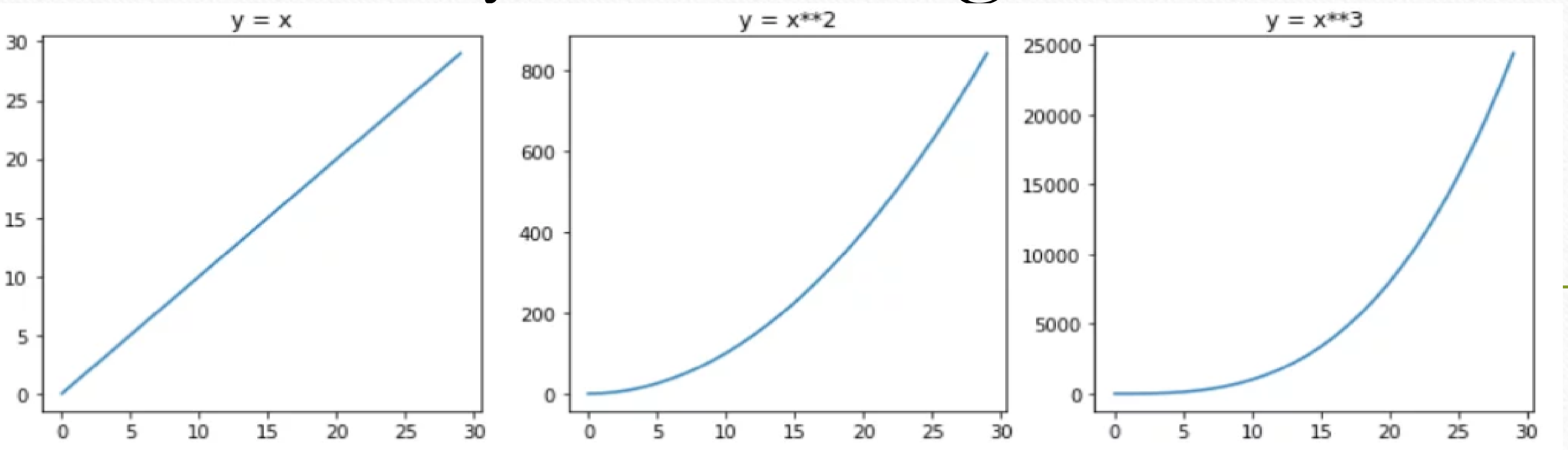
$$y = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_n x_n$$

Multiple Linear Regression

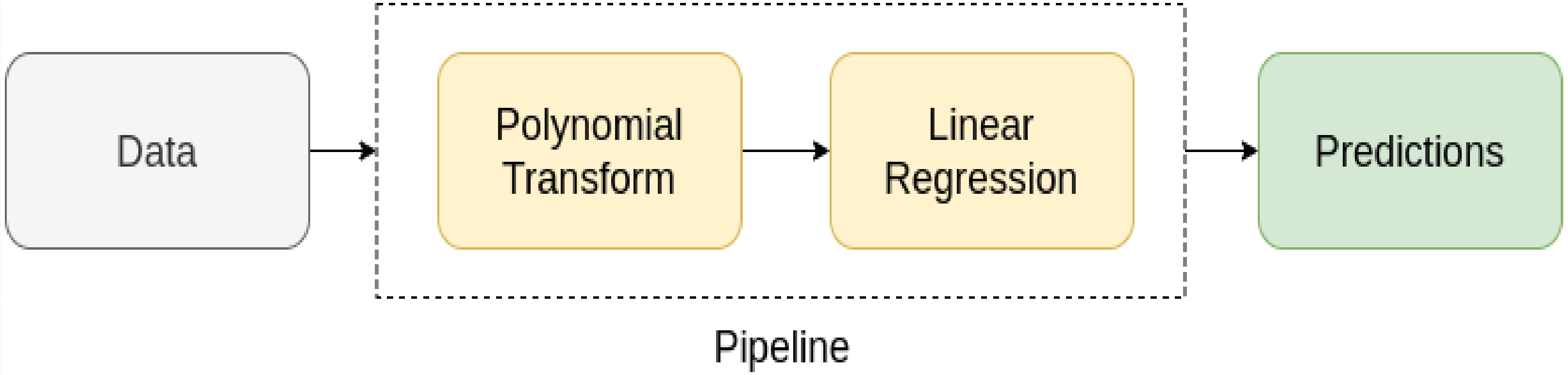
$$y = b_0 + b_1 x_1 + b_2 x_1^2 + \ldots + b_n x_1^n$$

Polynomial Regression

Polynomial Regression



poly_reg is a transformer tool that transforms the matrix of features X into a new matrix of features X_poly. It contains x1, x1^2,……, x1^n.



There are 7 assumptions taken while using Linear Regression:

- Linear Model
- No Multicolinearlity in the data
- Homoscedasticity of Residuals or Equal Variances
- No Autocorrelation in residuals
- Number of observations Greater than the number of predictors
- Each observation is unique
- Predictors are distributed Normal

(Refer: Upgrad and GeekforGeeks)

Assumptions of Linear Regression – GeeksforGeeks

Assumptions of Linear Regression: 5 Assumptions With Examples | upGrad blog

Linear model

- Relationship should be linear.
- The reason behind this relationship is that if the relationship will be non-linear which is certainly is the case in the real-world data then the predictions made by our linear regression model will not be accurate and will vary from the actual observations a lot.

The problem of Multicollinearity!

For example, let's assume that in the following linear equation:

$$Y = W0+W1*X1+W2*X2$$

- Coefficient W1 is the increase in Y for a unit increase in X1 while keeping X2 constant. But since X1 and X2 are highly correlated, changes in X1 would also cause changes in X2 and we would not be able to see their individual effect on Y.

- This makes the effects of X1 on Y difficult to distinguish from the effects of X2 on Y.

Detecting Multicollinearity using VIF (Variable Inflation Factors)

- VIF determines the strength of the correlation between the independent variables. It is predicted by taking a variable and regressing it against every other variable.

or

- VIF score of an independent variable represents how well the variable is explained by other independent variables.

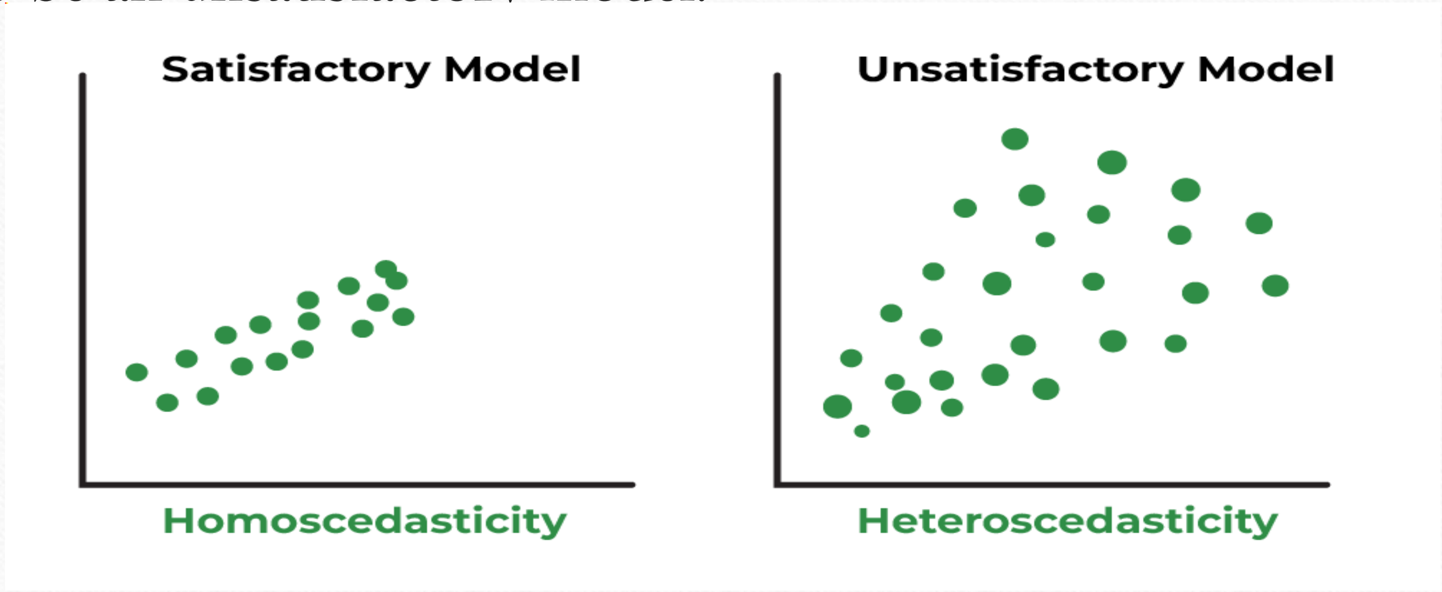
Variable Inflation Factors

$$VIF = \frac{1}{1-R^2}$$

- VIF starts at 1 and has no upper limit
- VIF = 1, no correlation between the independent variable and the other variables
- VIF exceeding 5 or 10 indicates high multi-collinearity between this independent variable and the others

Homoscedasticity of Residuals or Equal Variances

- Homoscedasticity is the term that states that the spread residuals which we are getting from the linear regression model should be homogeneous or equal spaces. If the spread of the residuals is heterogeneous then the model is called to be an unsatisfactory model.



No Autocorrelation in residuals

- There should be no autocorrelation in the residuals data.
- When the residuals are dependent on each other, there is autocorrelation. This factor is visible in the case of stock prices when the price of a stock is not independent of its previous one.
  Plotting the variables on a graph like a scatterplot or a line plot allows you to check for autocorrelations if any.
- In other words, there is no correlation between the consecutive error terms of the time series data. The presence of correlation in the error terms drastically reduces the accuracy of the model. If the error terms are correlated, the estimated standard error tries to deflate the true standard error.

- **How to determine if the assumption is met?**
- Conduct a Durbin-Watson (DW) statistic test. The values should fall between 0-4. If DW=2, no auto-correlation; if DW lies between 0 and 2, it means that there exists a positive correlation. If DW lies between 2 and 4, it means there is a negative correlation. Another method is to plot a graph against residuals vs time and see patterns in residual values.

Number of observations Greater than the number of predictors

- For a better-performing model, the number of training data or observations should be always greater than the number of test or prediction data. However greater the number of observations better the model performance.

Each observation is unique

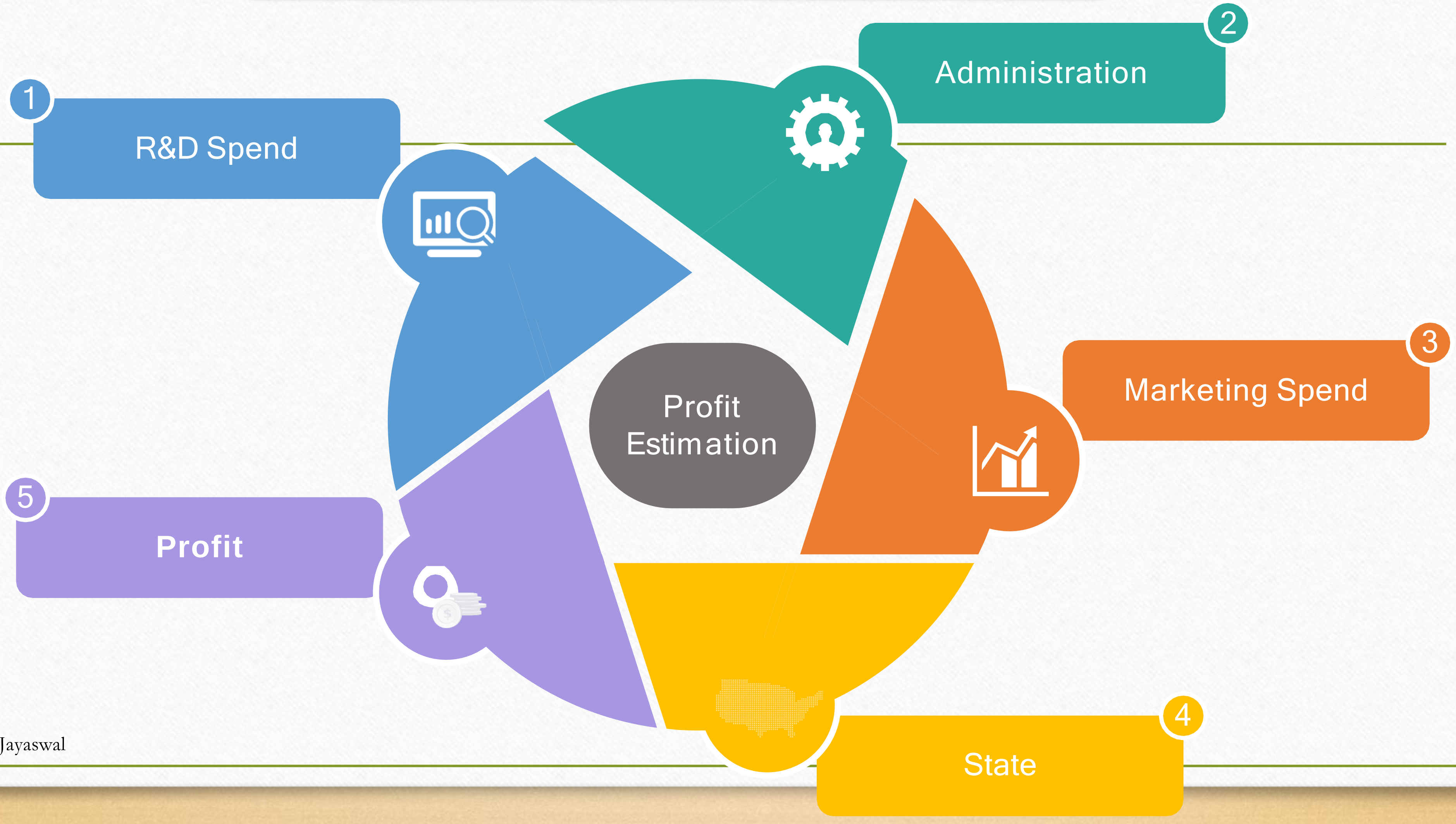
- It is also important to ensure that each observation is independent of the other observation.
- Meaning each observation in the data set should be measured separately on a unique occurrence of the event that caused the observation.

Predictors are distributed Normally

- This assumption ensures that you have equally distributed observations for the range of each predictor. So at the end of the model training, the predicted values for each test data should be a normal distribution.
- One can get an idea of the distribution of the predicted values by plotting density, KDE, or QQ plots for the predictions.

Regression Use Case

Predicting profit based on expenditures of the company



Gradient Descent

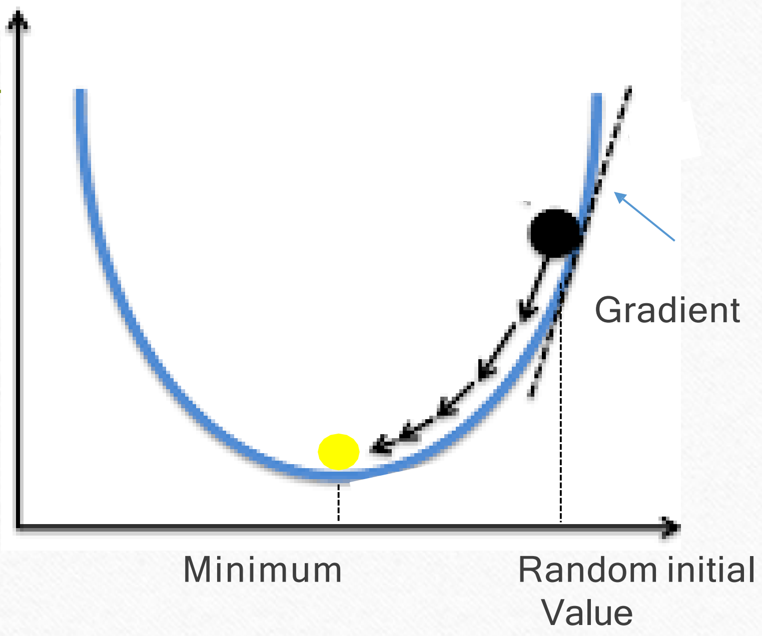
Gradient descent is another algorithm used to reduce the loss function.

It is an optimization algorithm that tweaks it's parameters (coefficients) iteratively to minimize a given cost function to its minimum.

Model stops learning when the gradient (slope) is zero

Algorithm : 1) Initialize parameter by some value

2) For each iteration calculate the derivative of the cost function and simultaneously update the parameters until a global minimum

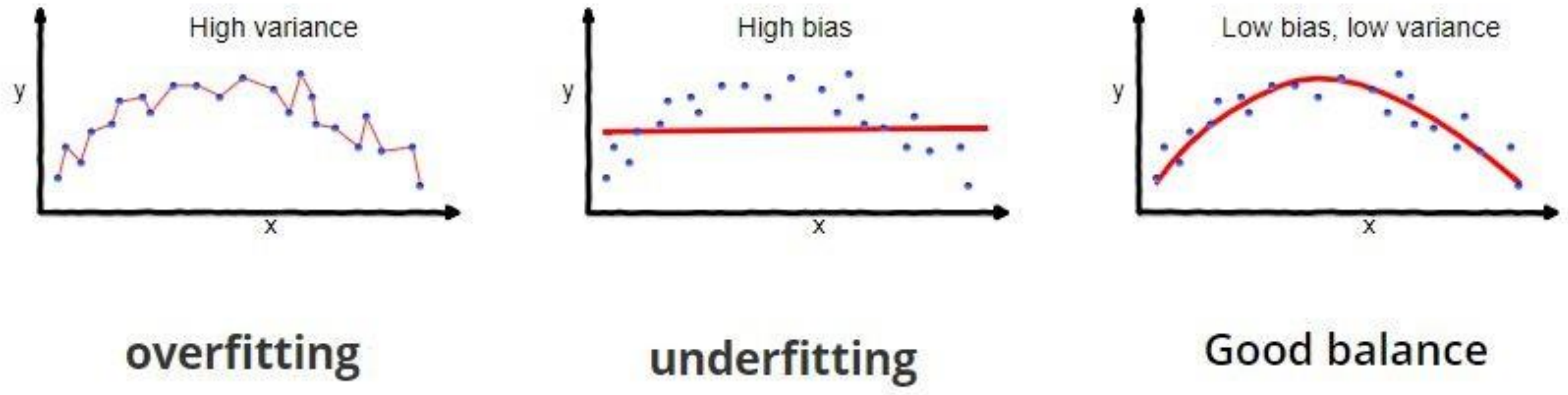


$$\theta := \theta - \alpha \frac{\delta}{\delta\theta} J(\theta)$$

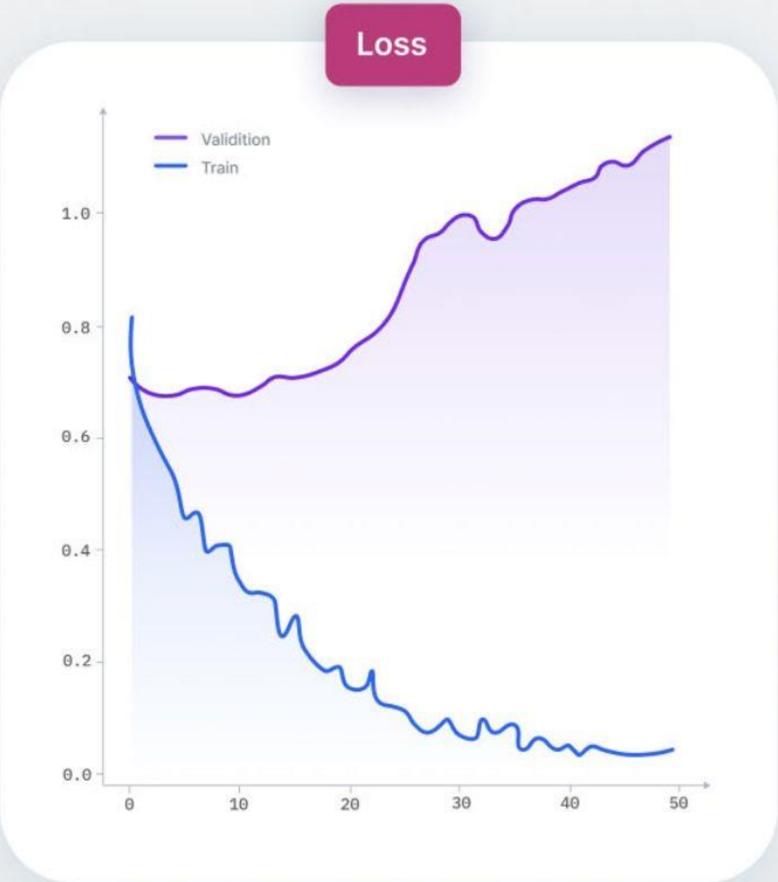
where ἀis the learning rate

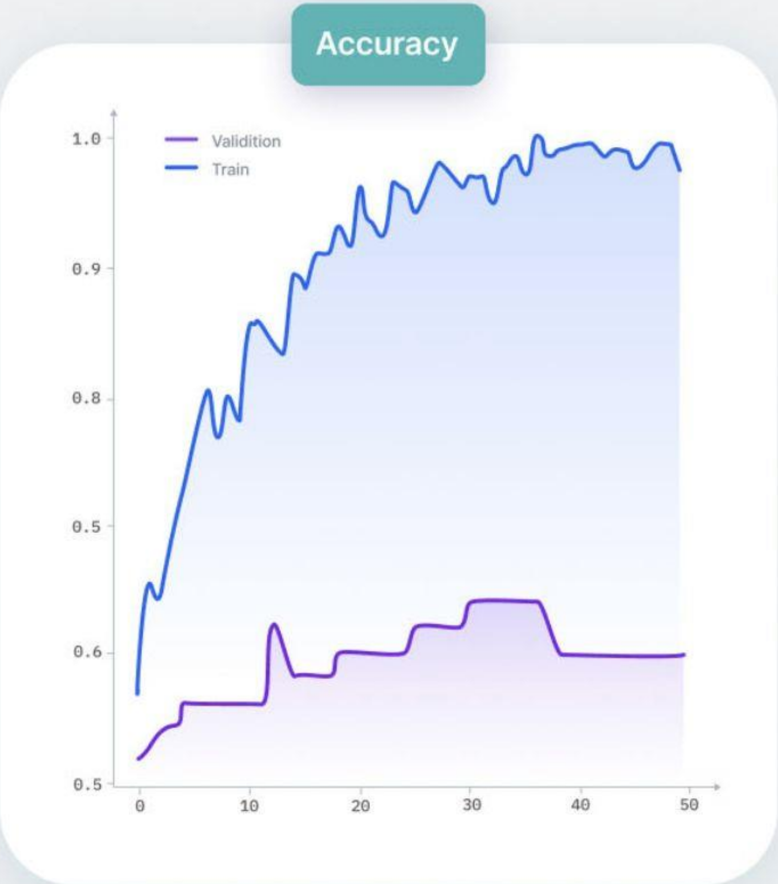
- For more – Refer Director sir’s slides and Class Notes

Issues -Regression



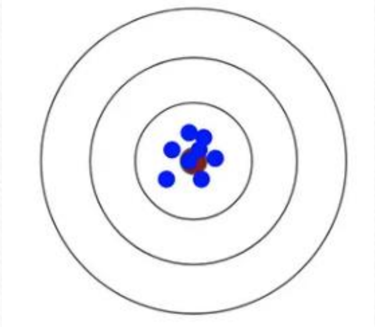
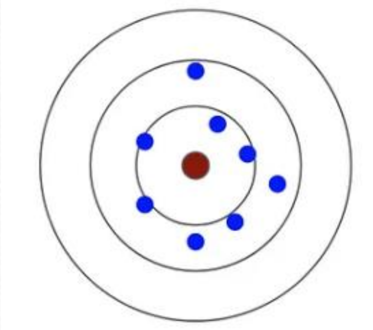
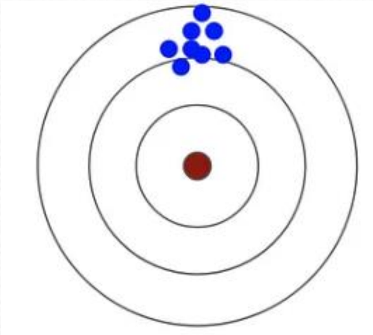
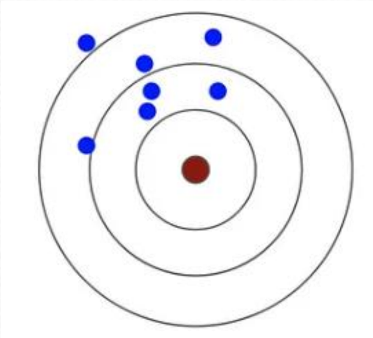
Dr. Ruchi Jayaswal





Dr. Ruchi Jayaswal

Bias-Variance Example!

	Low Variance	High Variance
Low Bias		
High Bias		

- **Bias:** Bias measures the difference between the model's prediction and the target value. If the model is oversimplified, then the predicted value would be far from the ground truth resulting in more bias.

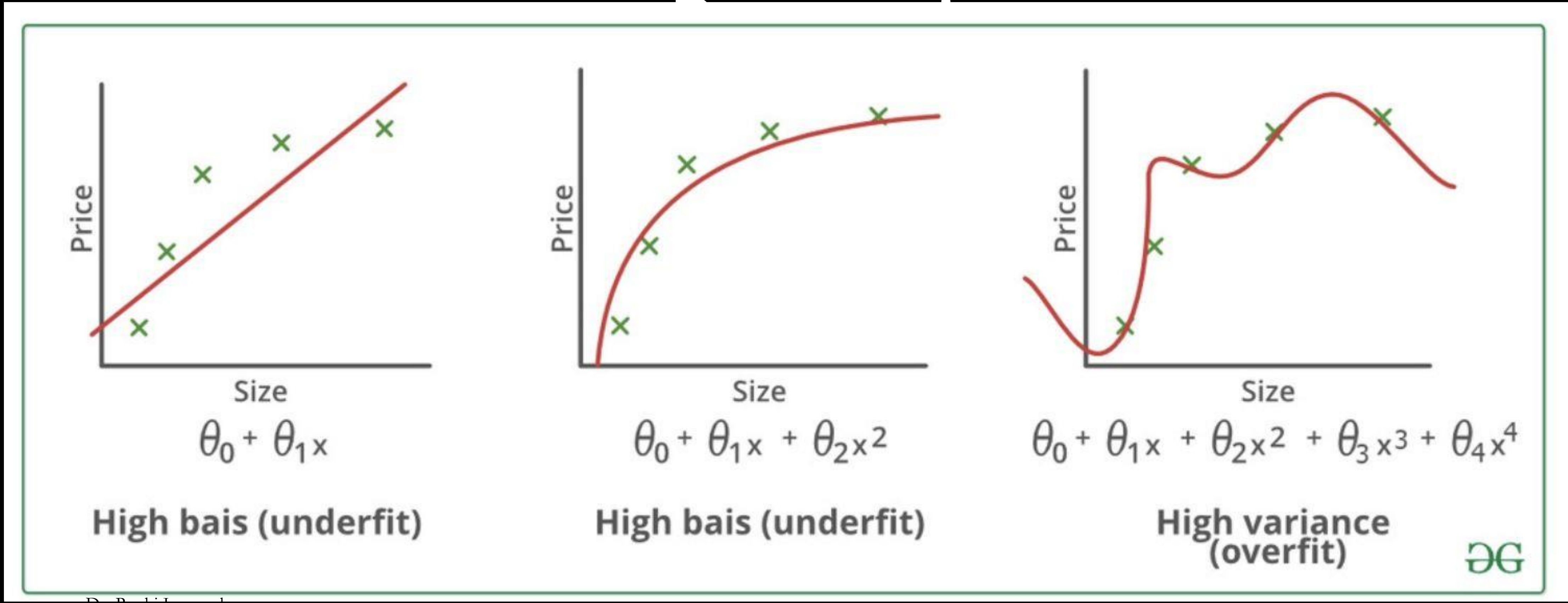
- **Variance:** Variance is the measure of the inconsistency of different predictions over varied datasets. If the model's performance is tested on different datasets, the closer the prediction, the lesser the variance. Higher variance is an indication of overfitting in which the model loses the ability to generalize.

- **Bias-variance tradeoff:** A simple linear model is expected to have a high bias and low variance due to less complexity of the model and fewer trainable parameters. On the other hand, complex non-linear models tend to observe an opposite behavior. In an ideal scenario, the model would have an optimal balance of bias and variance.

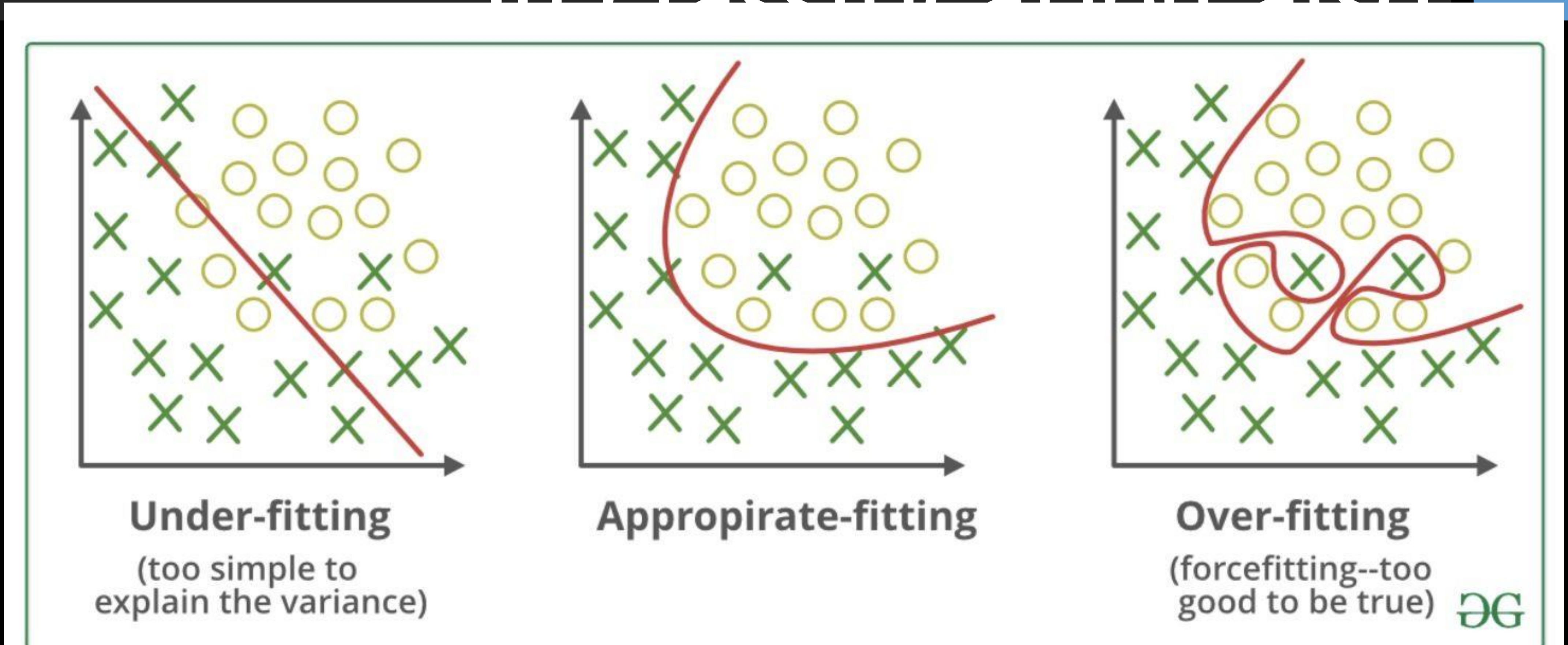
- **Model generalization:** Model generalization means how well the model is trained to extract useful data patterns and classify unseen data samples.

- **Feature selection:** It involves selecting a subset of features from all the extracted features that contribute most towards the model performance. Including all the features unnecessarily increases the model complexity and redundant features can significantly increase the training time.

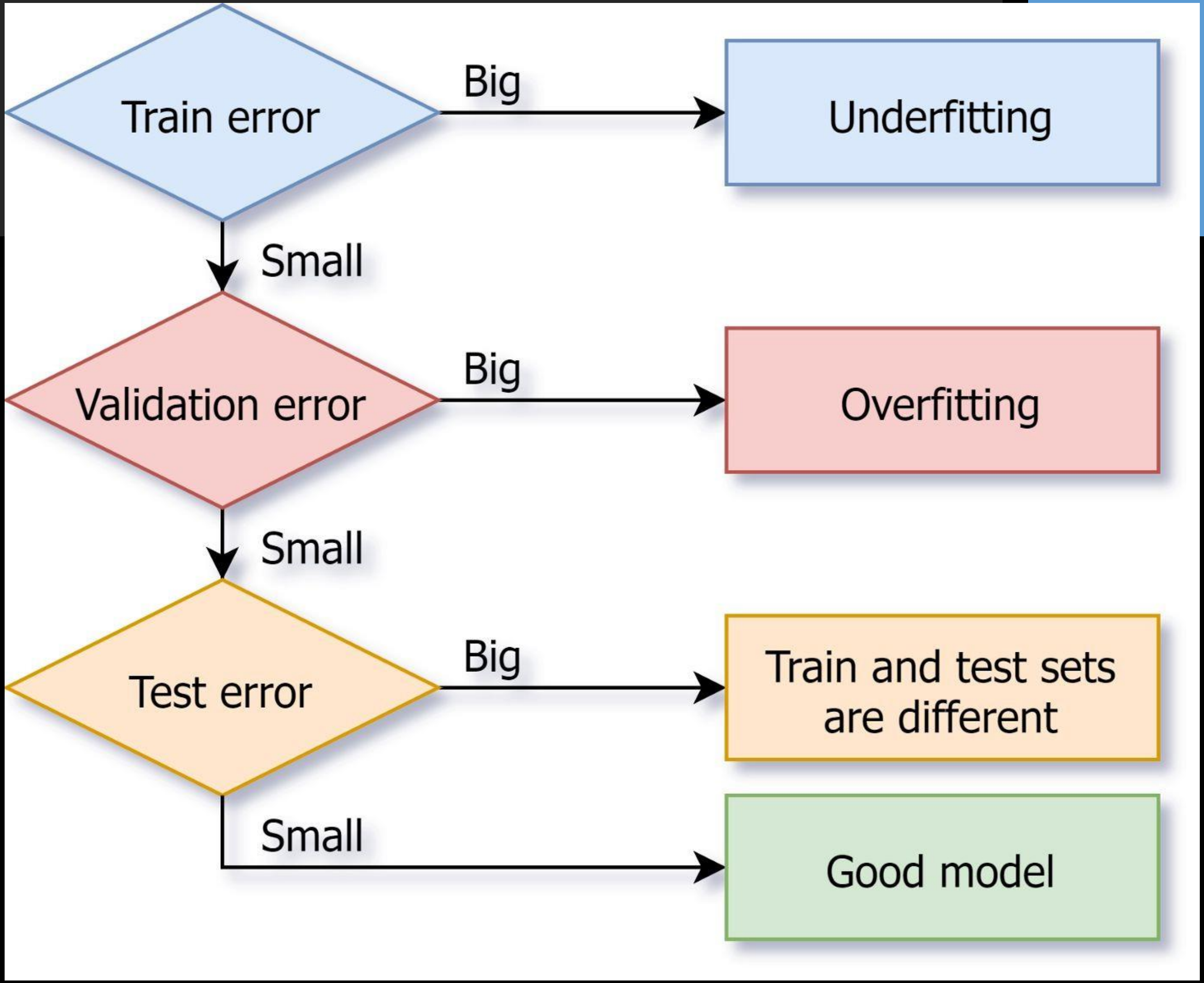
Underfitting Vs Overfitting in

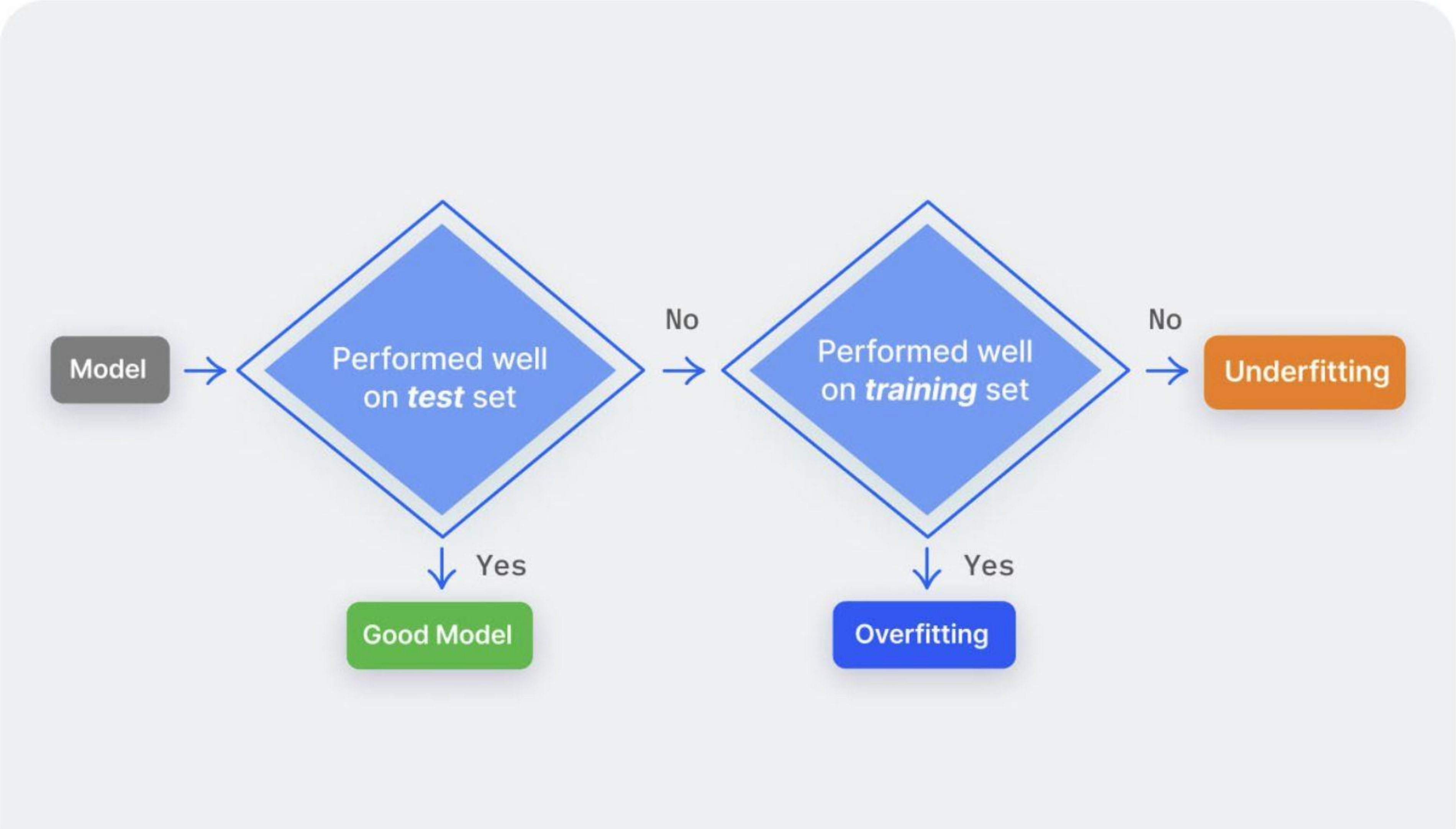


In classification also



Underfitting Vs Overfitting




Model
Performed well on *test* set
No
Performed well on *training* set
No
Underfitting
Yes
Good Model
Yes
Overfitting

Overfitting happens when:

1. The data used for training is not cleaned and contains garbage values. The model captures the noise in the training data and fails to generalize the model's learning.
2. The model has a high variance.
3. The training data size is not enough, and the model trains on the limited training data for several epochs.
4. The architecture of the model has several neural layers stacked together. Deep neural networks are complex and require a significant amount of time to train, and often lead to overfitting the training set.

Underfitting happens when:

1. Unclean training data containing noise or outliers can be a reason for the model not being able to derive patterns from the dataset.
2. The model has a high bias due to the inability to capture the relationship between the input examples and the target values.
3. The model is assumed to be too simple. For example, training a linear model in complex scenarios.

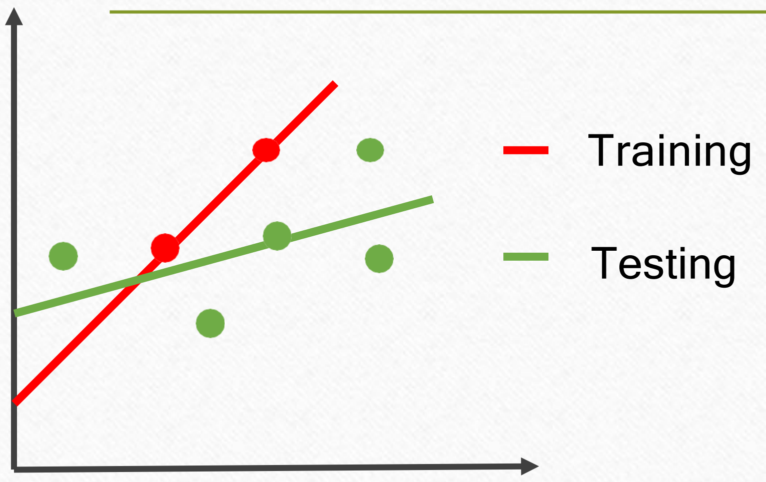
The goal is to find a good fit such that the model picks up the patterns from the training data and does not end up memorizing the finer details.

This, in turn, would ensure that the model generalizes and accurately predicts other data samples.

Regularization

Regularization solves overfitting to the training data.

Used to restrict the parameters values that are estimated in the model



$$L = \sum(\hat{Y}_i - Y_i)^2 + \lambda \sum \beta^2$$

This loss function includes 2 elements.

1) the sum of square distances between predicted and actual value

2) the second element is the regularization term

Types of Regression (Contd.)

Ridge Regression

Lasso Regression

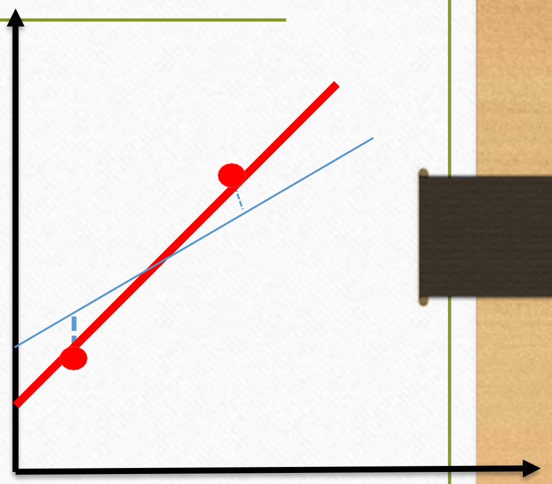
ElasticNet Regression

Ridge Regression (L2) is used when there is a problem of multicollinearity.

By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

The main idea is to find a new line that has some bias with respect to the training data

In return for that small amount of bias, a significant drop in variance is achieved



Minimization objective = LS Obj + λ * (sum of the square of coefficients)

LS Obj refers to least squares objective

λ controls the strength of the penalty term

Types of Regression (Contd.)

Ridge Regression

Lasso Regression

ElasticNet Regression

Lasso Regression (L1) is similar to ridge, but it also performs feature selection.

It will set the coefficient value for features that do not help in decision making very low, potentially zero.

Minimization objective = LS Obj + λ * (sum of absolute coefficient values)

Lasso regression tends to exclude variables that are not required from the equation, whereas ridge tends to do better when all variables are present.

Types of Regression (Contd.)

- Ridge Regression
- Lasso Regression
- **ElasticNet Regression**

ElasticNet regression combines
the strength of **lasso and ridge regression**

$\downarrow$

the sum of the squared residuals

+

$$\boxed{\underbrace{\lambda_1 \times |\text{variable}_1| + \ldots + |\text{variable}_x|}_{\text{Lasso penalty}} + \underbrace{\lambda_2 \times \text{variable}_1^2 + \ldots + \text{variable}_x^2}_{\text{Ridge penalty}}}$$

If you are not sure whether to use lasso or ridge, use ElasticNet

Evaluation Metrics for Regression

- Various regression evaluation metrics is used to measure how well our model fits the data.
- There is no "one function to rule them all".
- Residual
- Loss function
- Cost function

Evaluation Metrics for Regression (to be cont'd)

- Loss function : A function that calculates loss for 1 data point is called Loss function.

  Squared Error: $(yi - yi\Lambda)^2$

- Cost function: A function that calculates loss for the entire data being is called the cost function.

  MSE= 1/n $\sum_{i=1}^{n}(yi - yi\Lambda)^2$

- Residual: When produces the smallest difference between actual and predicted values, where these differences are unbiased as well. The difference or error is also known as residuals. Unbiased means there is no systematic pattern of distributed of predicted values.
- Residual = Actual Value – Predicted Values
- E= $(yi - yi\Lambda)^2$

MAE (Mean Absolute Error)

- Mean absolute error, also known as L1 loss is one of the simplest loss functions.
- It is calculated by taking the absolute difference between the predicted values and the actual values and averaging it across the dataset,

- MAE measures only the magnitude of the errors and doesn't concern itself with their direction. The lower the MAE, the higher the accuracy of a model.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} | y_i - \hat{y}_i |$$

R Square/Adjusted R Square

- R Square measures how much variability in dependent variable can be explained by the model. It is the square of the Correlation Coefficient(R) and that is

$$R^2 = 1 - \frac{SS_{Regression}}{SS_{Total}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

MSE (Mean Squared Error)

- The most common metric for regression tasks is MSE. It has a convex shape. It is the average of the squared difference between the predicted and actual value.
- It is also known as L2 loss
- MSE penalizes large errors.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

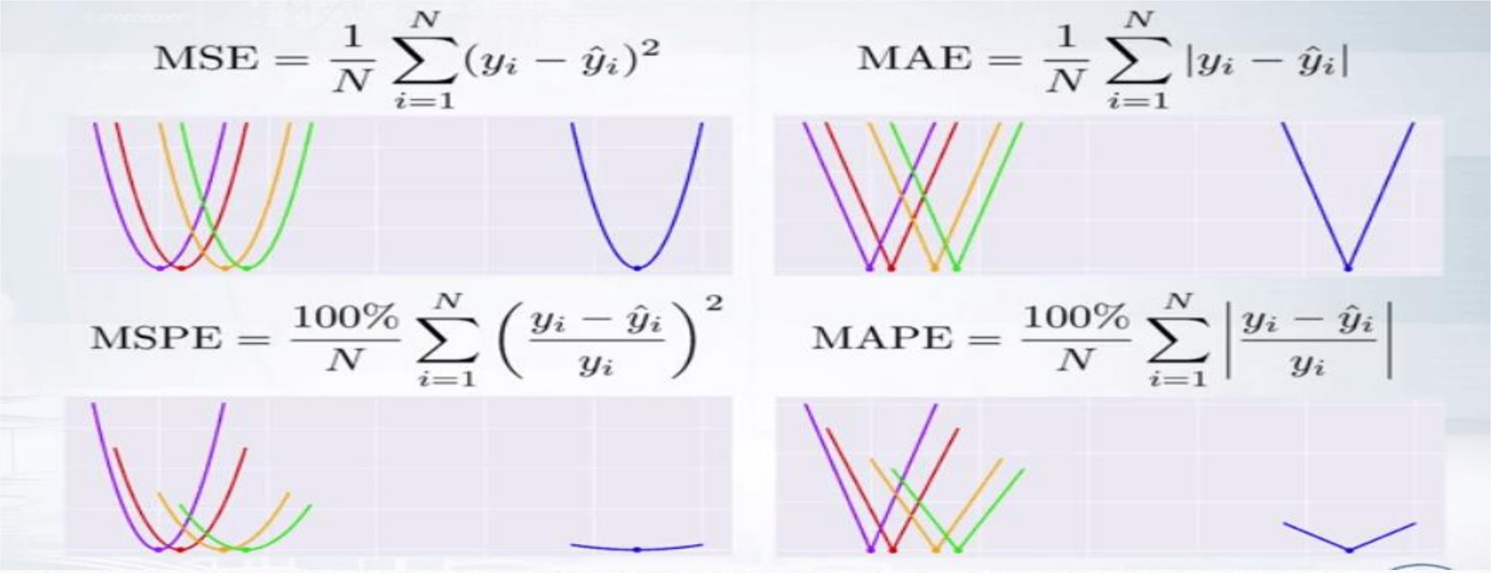
Root Mean Square Error(RMSE)

- The Root Mean Square Error is measured by taking the square root of the average of the squared difference between the prediction and the actual value. It represents the sample standard deviation of the differences between predicted values and observed values(also called residuals).

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

MSPE (Mean Square Percentage Error) & MAPE (Mean Absolute Percentage Error)

- These measures relative error. MSPE and MAPE can be thought of as weighted versions of MSE and MAE respectively.



For comparison purpose

- Evaluation Metric for Regression Models - Analytics Vidhya

- The Scikit-learn library in Python offers us a plethora of metrics to choose from as seen below:

explained_variance’	metrics.explained_variance_score
‘max_error’	metrics.max_error
‘neg_mean_absolute_error’	metrics.mean_absolute_error
‘neg_mean_squared_error’	metrics.mean_squared_error
‘neg_root_mean_squared_error’	metrics.mean_squared_error
‘neg_mean_squared_log_error’	metrics.mean_squared_log_error
‘neg_median_absolute_error’	metrics.median_absolute_error
‘r2’	metrics.r2_score
‘neg_mean_poisson_deviance’	metrics.mean_poisson_deviance
‘neg_mean_gamma_deviance’	metrics.mean_gamma_deviance
‘neg_mean_absolute_percentage_error’	metrics.mean_absolute_percentage_error
‘d2_absolute_error_score’	metrics.d2_absolute_error_score
‘d2_pinball_score’	metrics.d2_pinball_score
‘d2_tweedie_score’	metrics.d2_tweedie_score