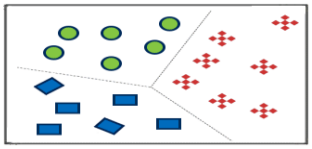
Supervised Learning

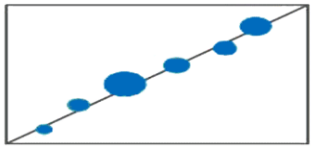
SUPERVISED LEARNING

EXAMPLES



Classification

Is this data input red, blue or green?



Regression

What is the impact of product price on number of sales?
What is the impact of years of experience on salary?

Outline

- Supervised Learning
- Regression
- Linear and Logistic Regression
- Classifier
- KNN
- Decision Trees
- Random Forest
- Support Vector Machine
- Confusion Matrix

Supervised Learning

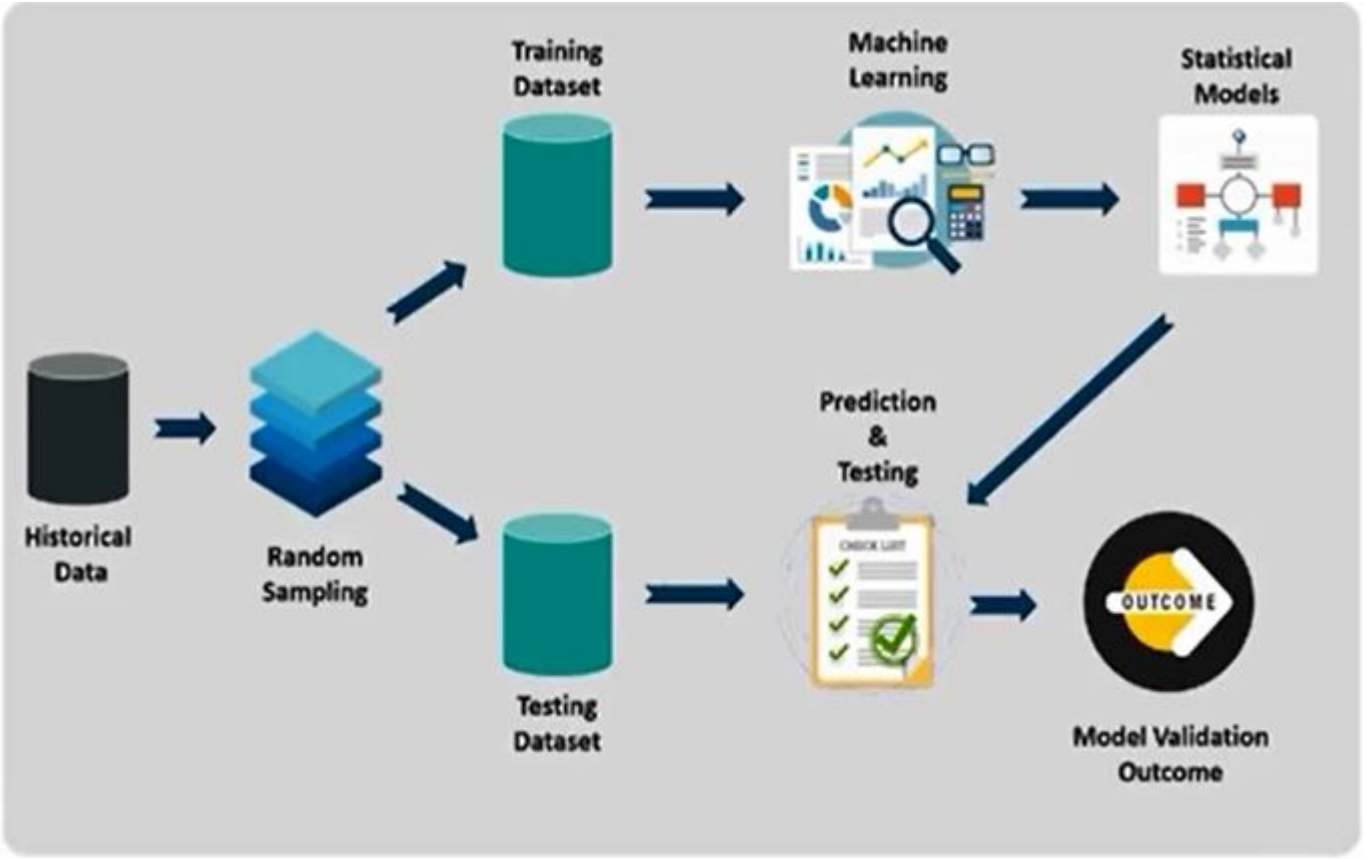
Supervised Learning is where you have labelled input variables(x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to output.

It is called **Supervised Learning** because the process of an algorithm learning from the training dataset can be thought as a teacher supervising the learning process.

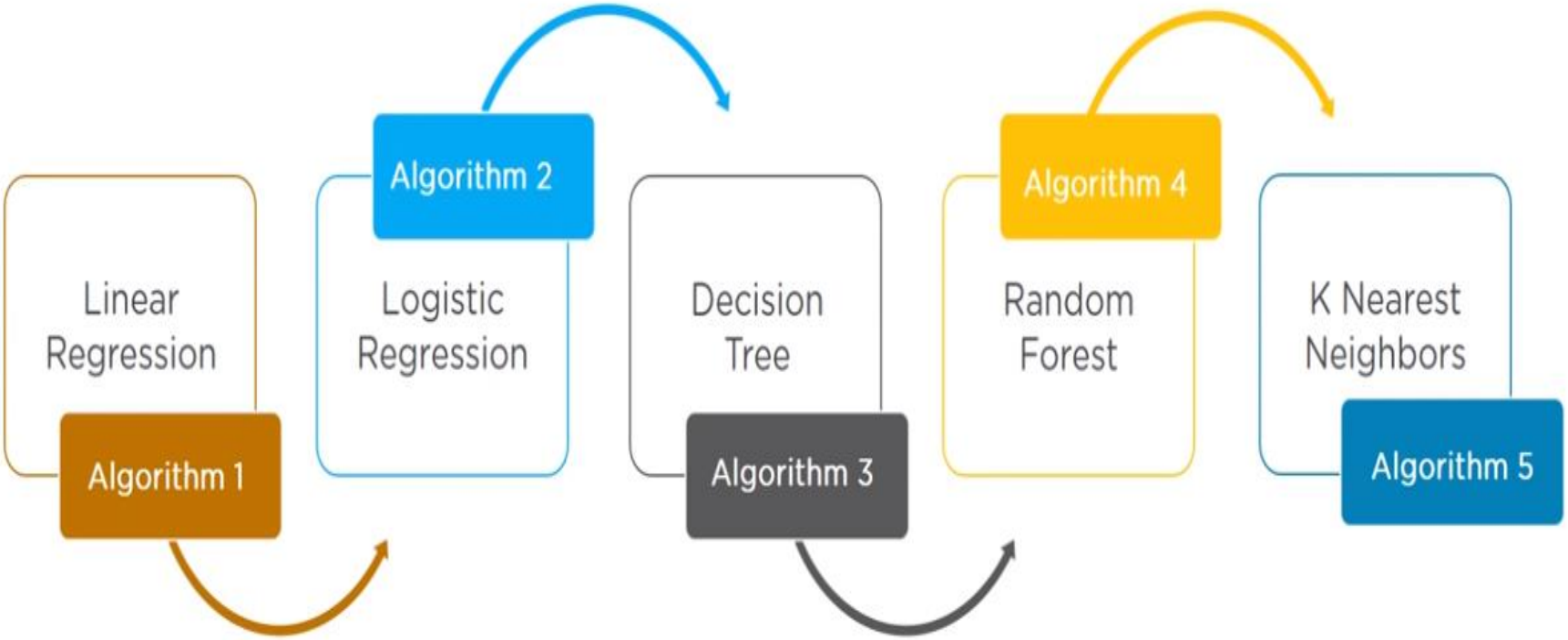
Supervised Learning

Training & Testing

Prediction



Popular Algorithms in Machine Learning

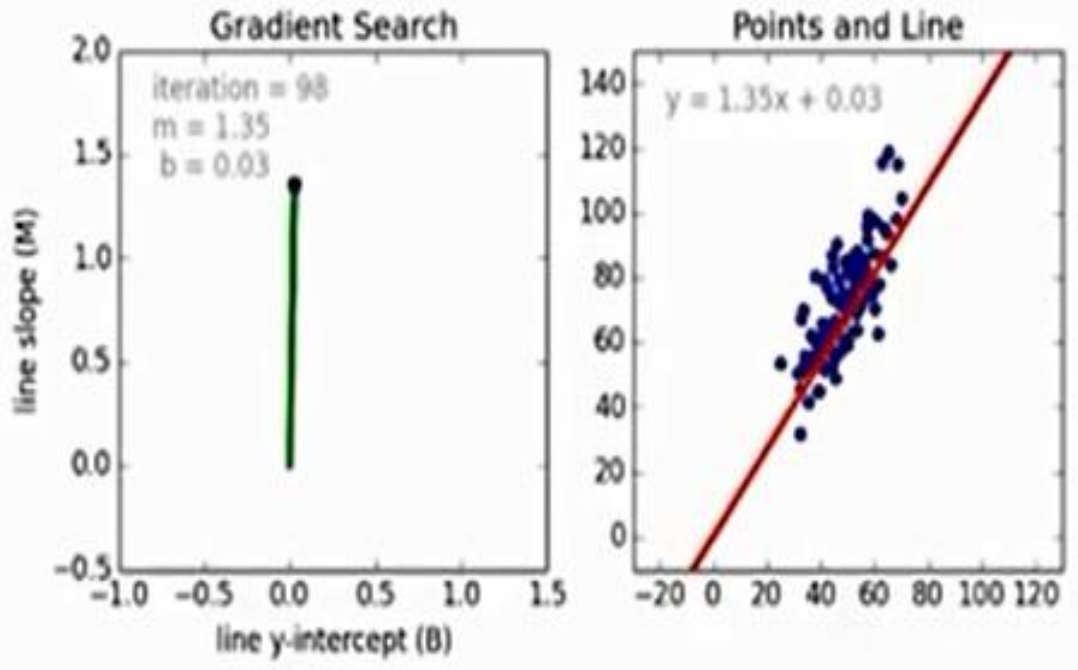


Supervised Learning Algorithm

- **Regression**
  - Linear Regression
  - Logistic Regression
- **Classification**
  - K-Nearest Neighbour
  - Naïve Bayes
  - Support Vector Machines
  - Decision Trees
  - Random Forest
- **Classifier Performance**

Regression

- What is Regression?
- Regression Use case
- Types of Regression-Linear Vs Logistics Regression
- What is Linear regression
- Finding best fit regression line using Least Square Method
- Checking goodness of fit using R squared method



What is Regression

A simple supervised learning technique used to find the **Best Trendline** to describe a dataset.

Regression analysis is a form of predictive modeling technique which investigate the relationship between a dependent and independent variables.

The dependent variable whose variation is being studied, by altering inputs, also known as **regressors** in a statistical context.

What is Regression

- Predict a value of a given continuous valued variables based on the values of the other variables, assuming a linear or non linear model of dependency.
- Greatly studied in statistics and Neural Network fields.

Examples

- Predicting sales amount of new product based on advertising expenditure.
- Predicting wind velocities as a function of temperature, humidity, air pressure etc.
- Time Series prediction of stock market indices.

Uses of Regression

Three major uses for Regression Analysis for

1. Determining the strength of Predictors.
2. Forecasting an effect, and
3. Trend Forecasting

Types of Regression

1. Linear Regression
2. Non-Linear Regression
3. Logistic Regression
4. Multiple Linear Regression etc.

Types of Regression

1. Linear Regression

Linear regression always uses a linear equation, **Y = mx + c**, where x is the explanatory variable and Y is the dependent variable.

2. Non Linear Regression

If the model equation does not follow the **Y = mx + c** form then the relationship between the dependent and independent variables will not be linear. There are many different forms of non-linear models. A random forest regression is considered a non-linear model.

3. Logistic Regression

It is used to classify elements of a set into two groups (binary classification) by calculating the probability of each element of the set.

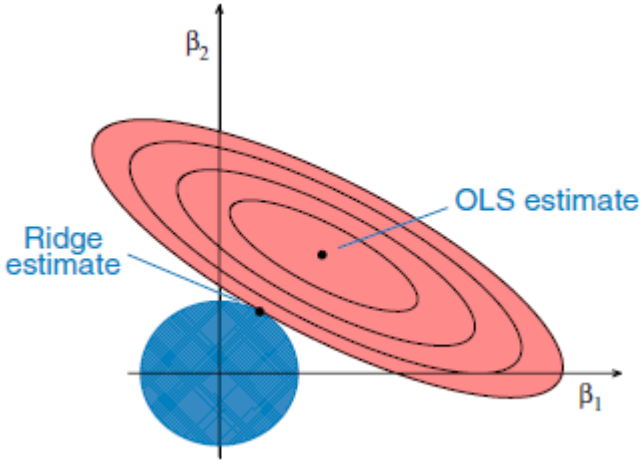
4. Multiple Linear Regression

Multiple linear regression refers to a statistical technique that is used to predict the outcome of a variable (Dependent Variable) based on the value of two or more Independent variables.

Types of Regression

5. Ridge Regression

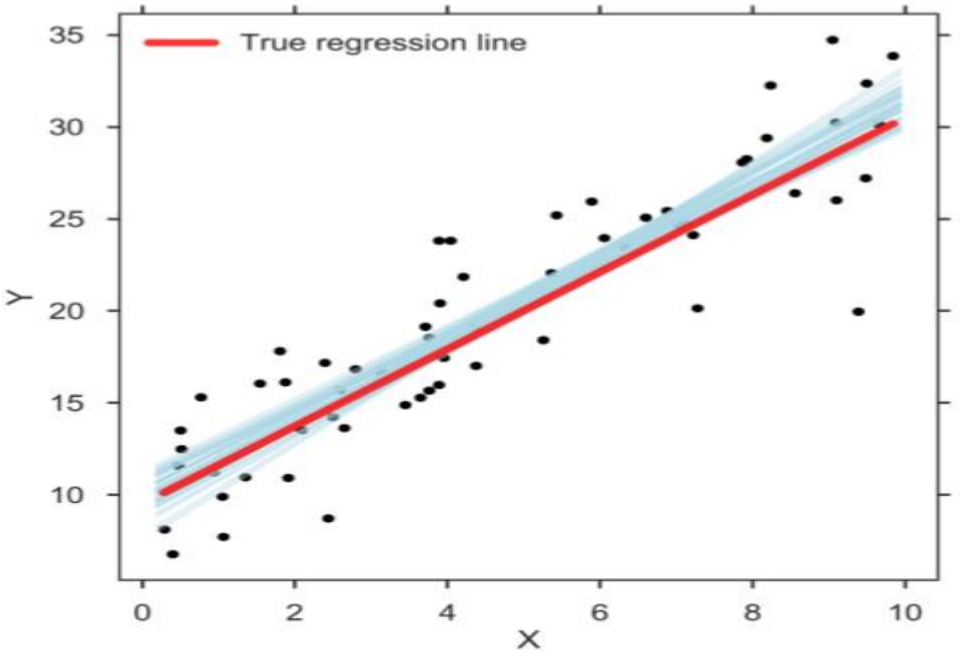
.



- This is another one of the **types of regression in machine learning**.
- It is usually used when there is a high correlation between the independent variables. This is because, in the case of multi collinear data, the least square estimates give unbiased values. But, in case the collinearity is very high, there can be some bias value.
- Therefore, a bias matrix is introduced in the equation of Ridge Regression.
- This is a powerful regression method where the model is less susceptible to overfitting.

Types of Regression

6. Bayesian Linear Regression



- It uses **Bayes theorem** to find out the value of regression coefficients.
- In this method of regression, the posterior distribution of the features is determined instead of finding the least-squares.
- **Bayesian Linear Regression** is like **both Linear Regression and Ridge Regression** but is more stable than the simple Linear Regression.

Linear Regression Selection Criteria

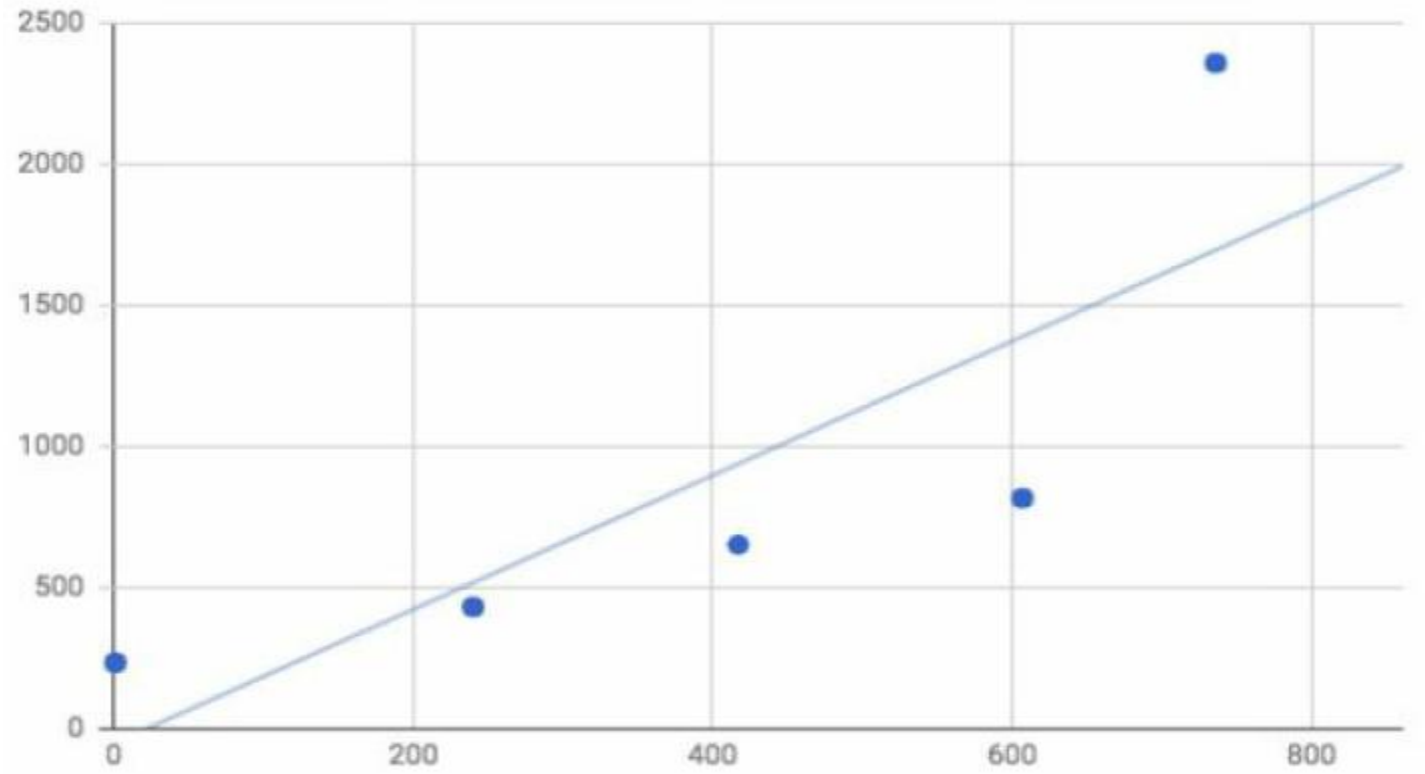
- Classification and Regression Capabilities
- Data Quality
- Computational Complexity
- Comprehensible and Transparent

Where is Linear Regression used

- Evaluating Trends and Sales Estimates
- Analyzing the Impact of Price Changes
- Assessment of Risk in Financial services and Insurance domain.

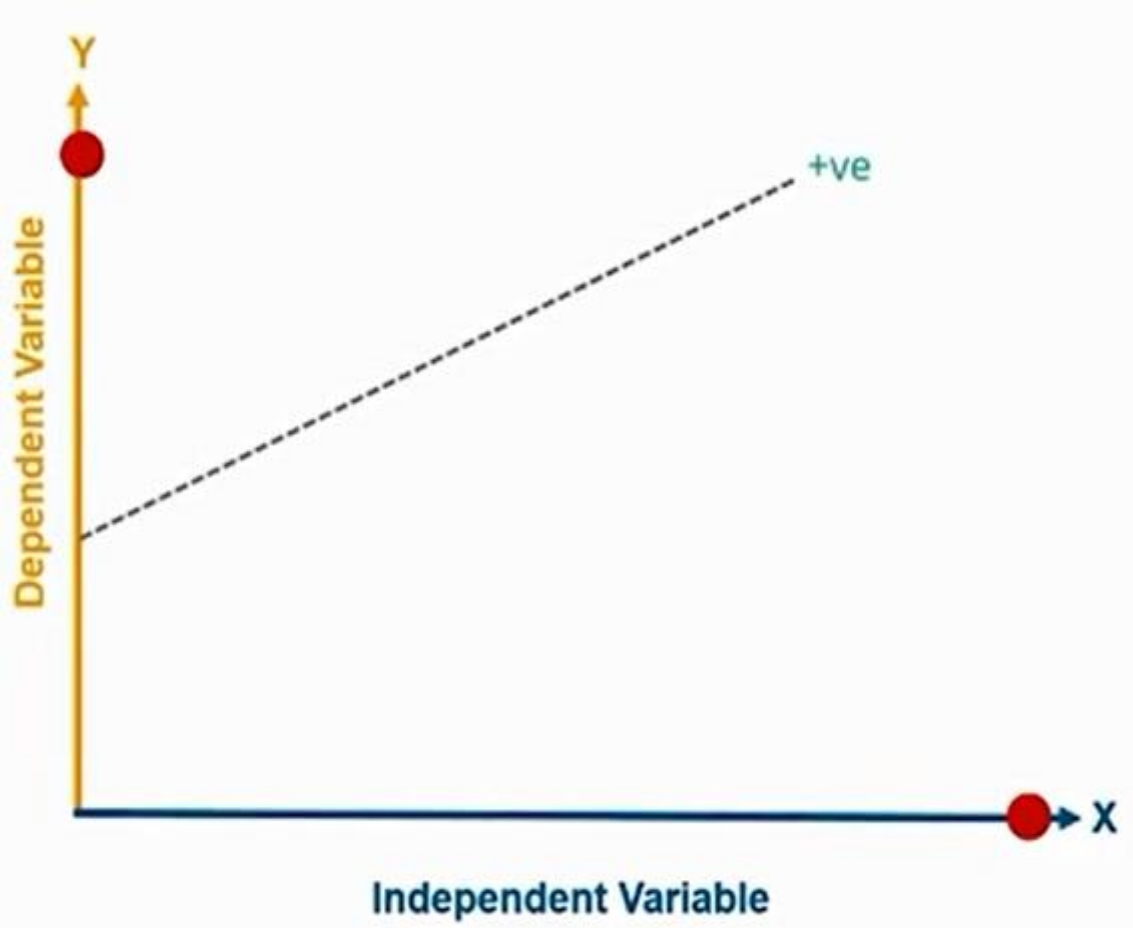
Linear Regression

- The goal of linear regression is to split your data in a way that minimizes the distance between the regression line and all data points on the scatterplot.
- If draw a vertical line from the regression line to each data point on the graph, the aggregate distance of each point would equate to the smallest possible distance to the regression line. **Hyper plane, Slope, Deviation**

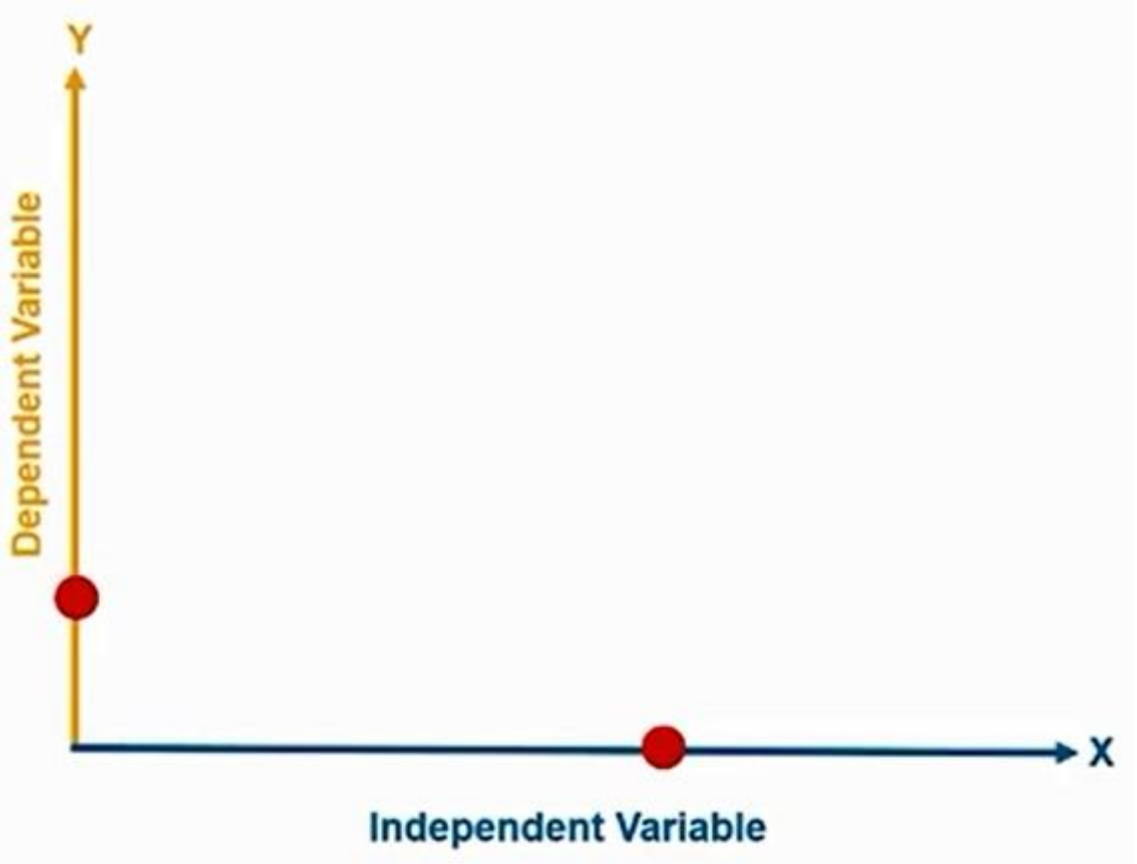


Understanding of Linear Regression Algorithm

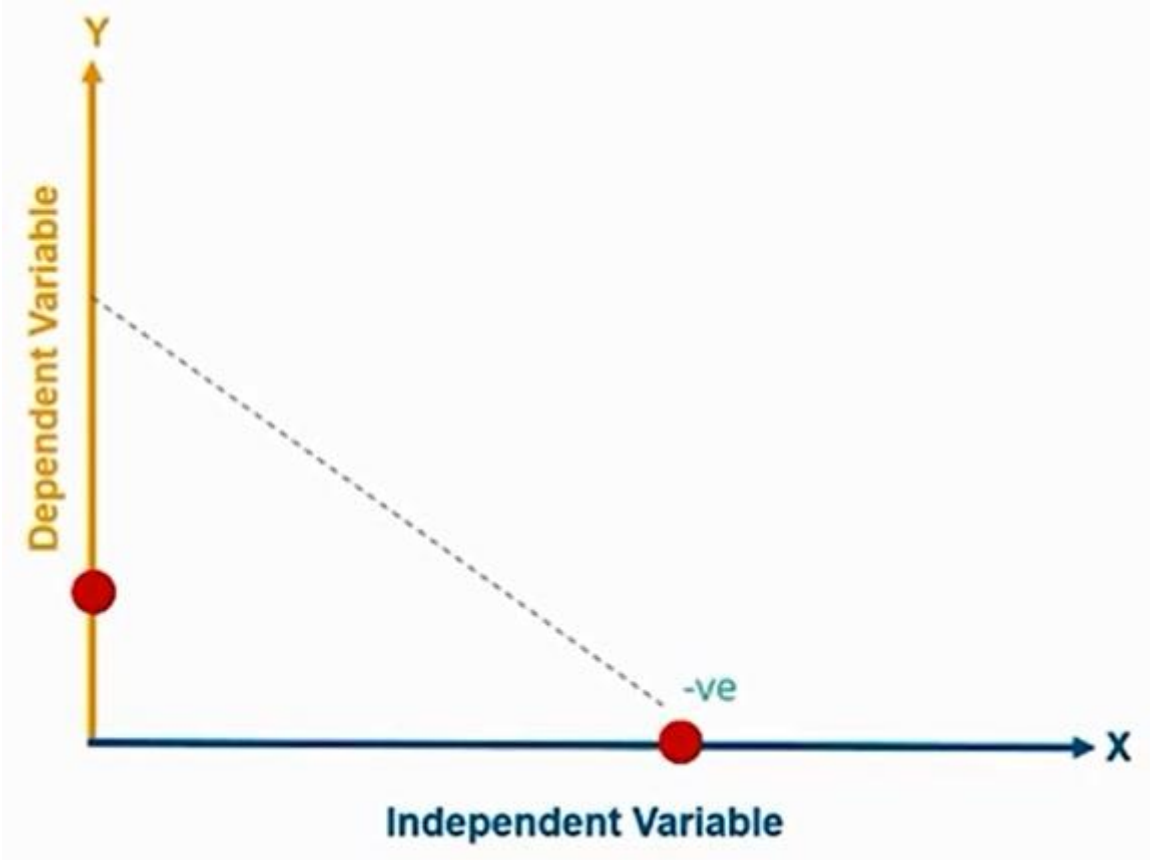
Understanding of Linear Regression Algorithm



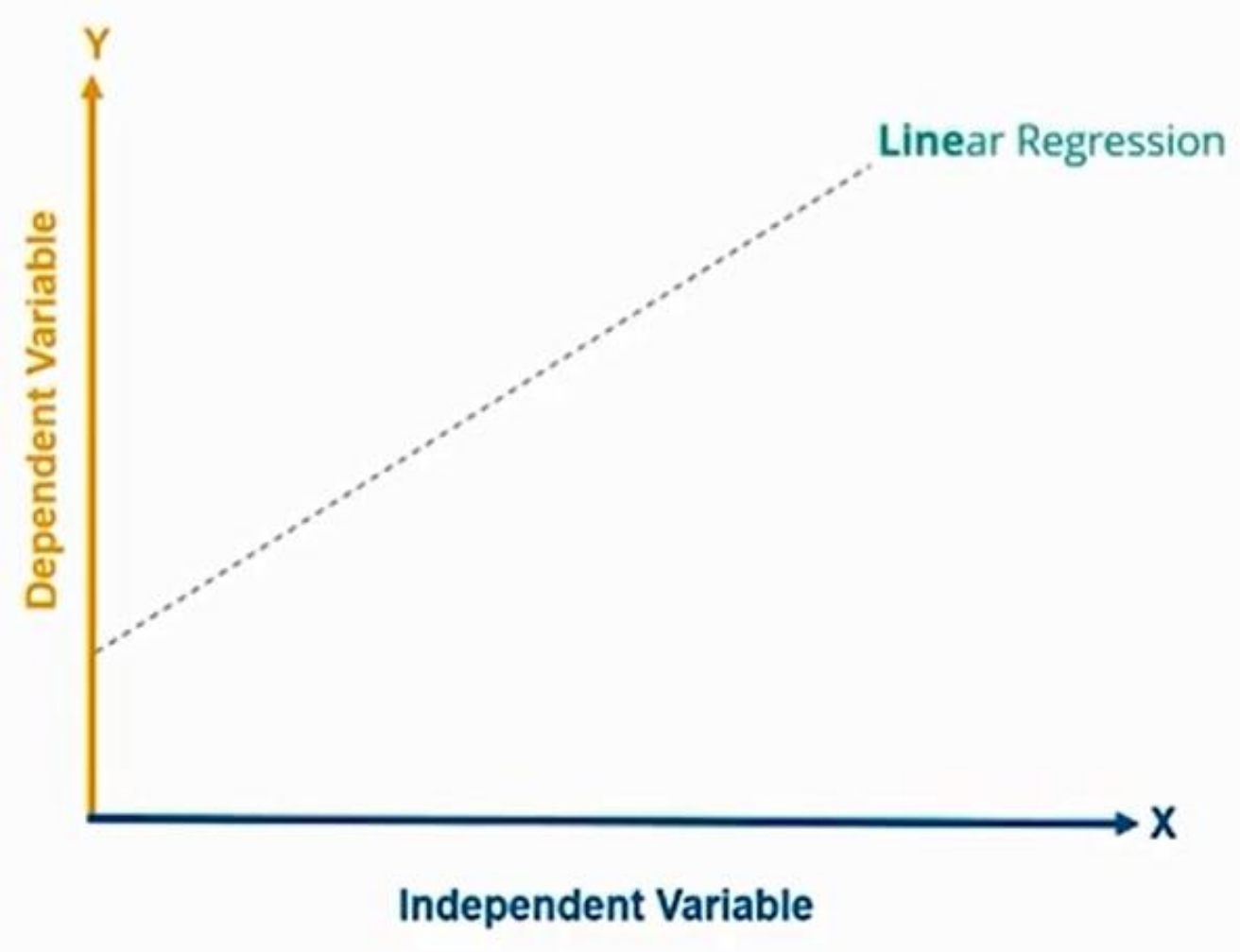
Understanding of Linear Regression Algorithm



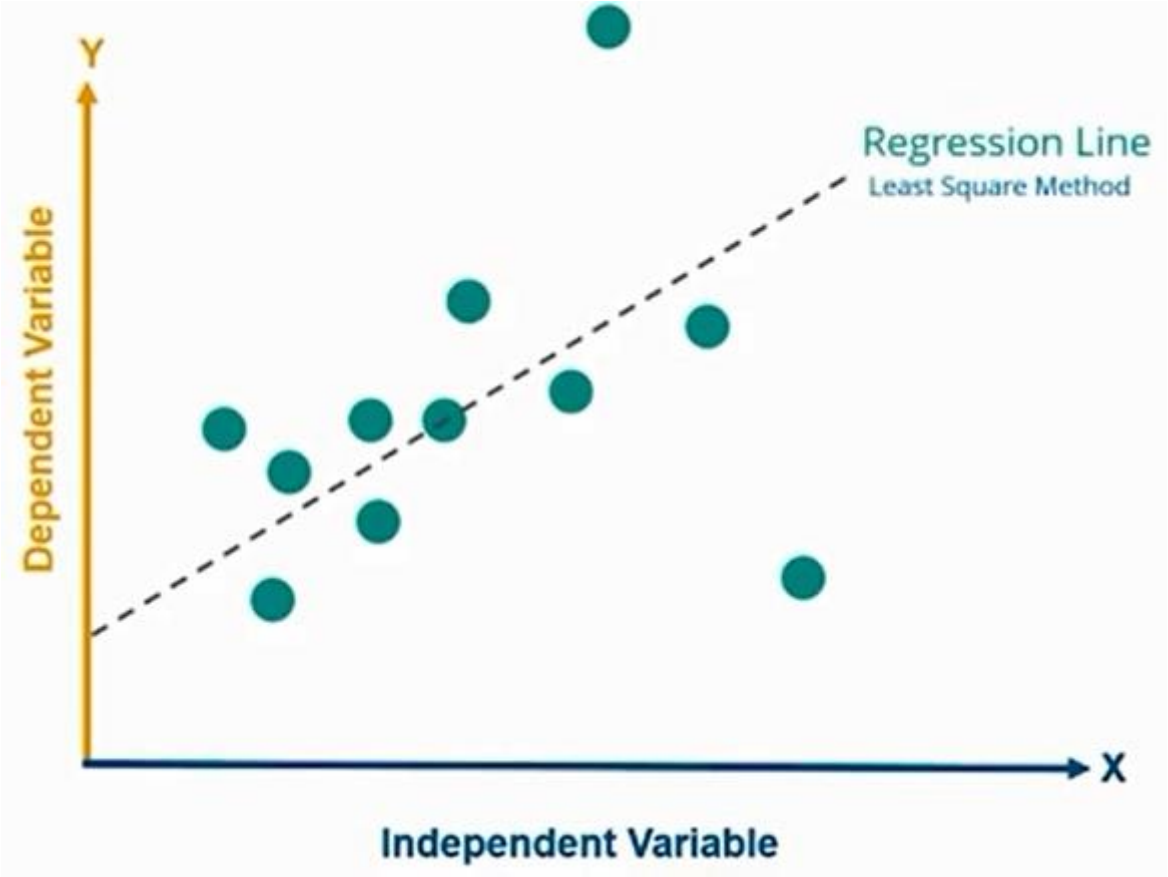
Understanding of Linear Regression Algorithm



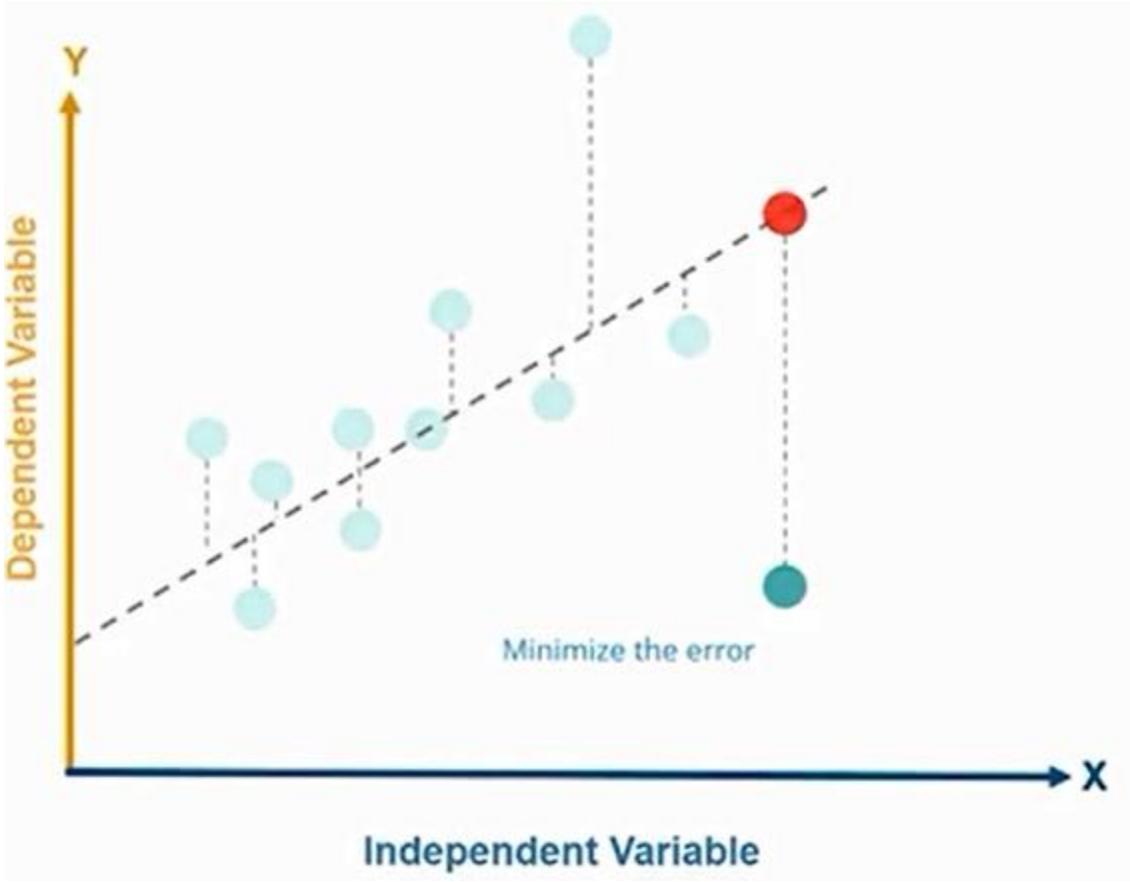
Understanding of Linear Regression Algorithm



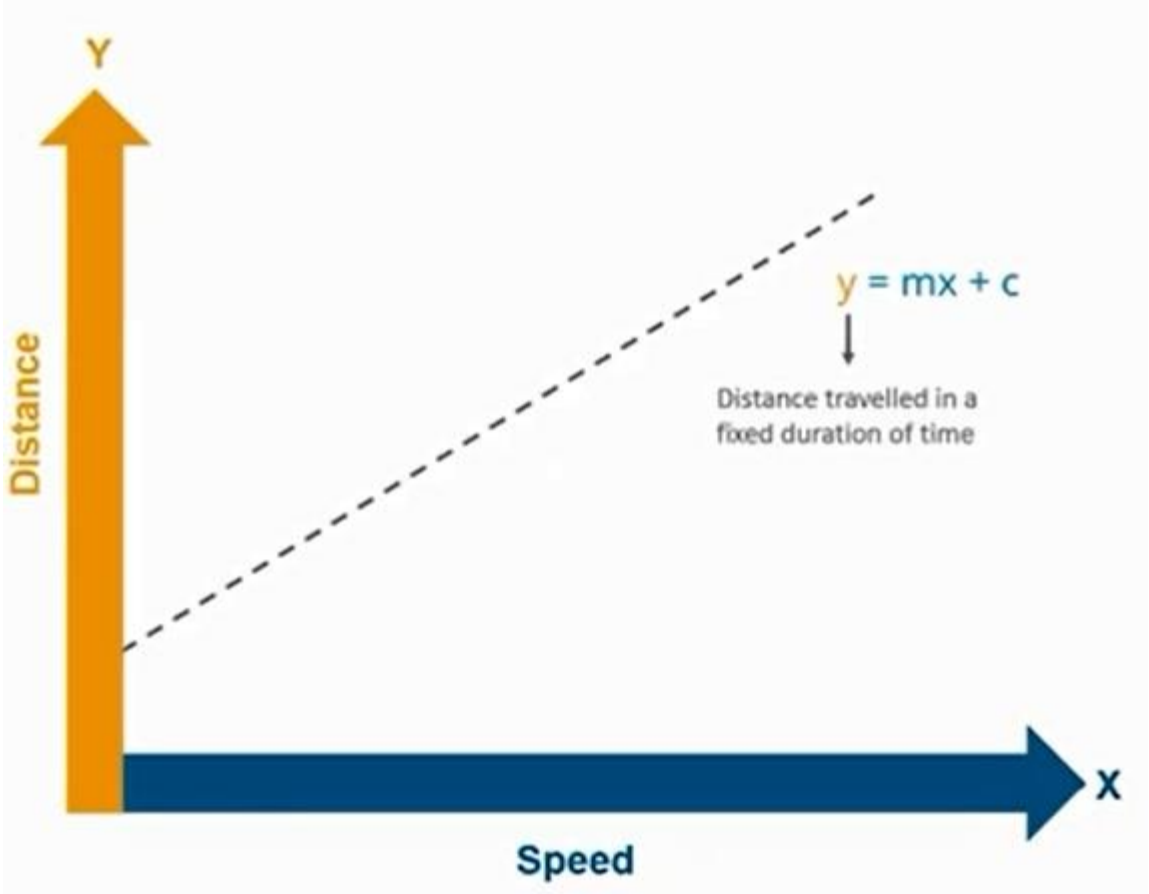
Understanding of Linear Regression Algorithm



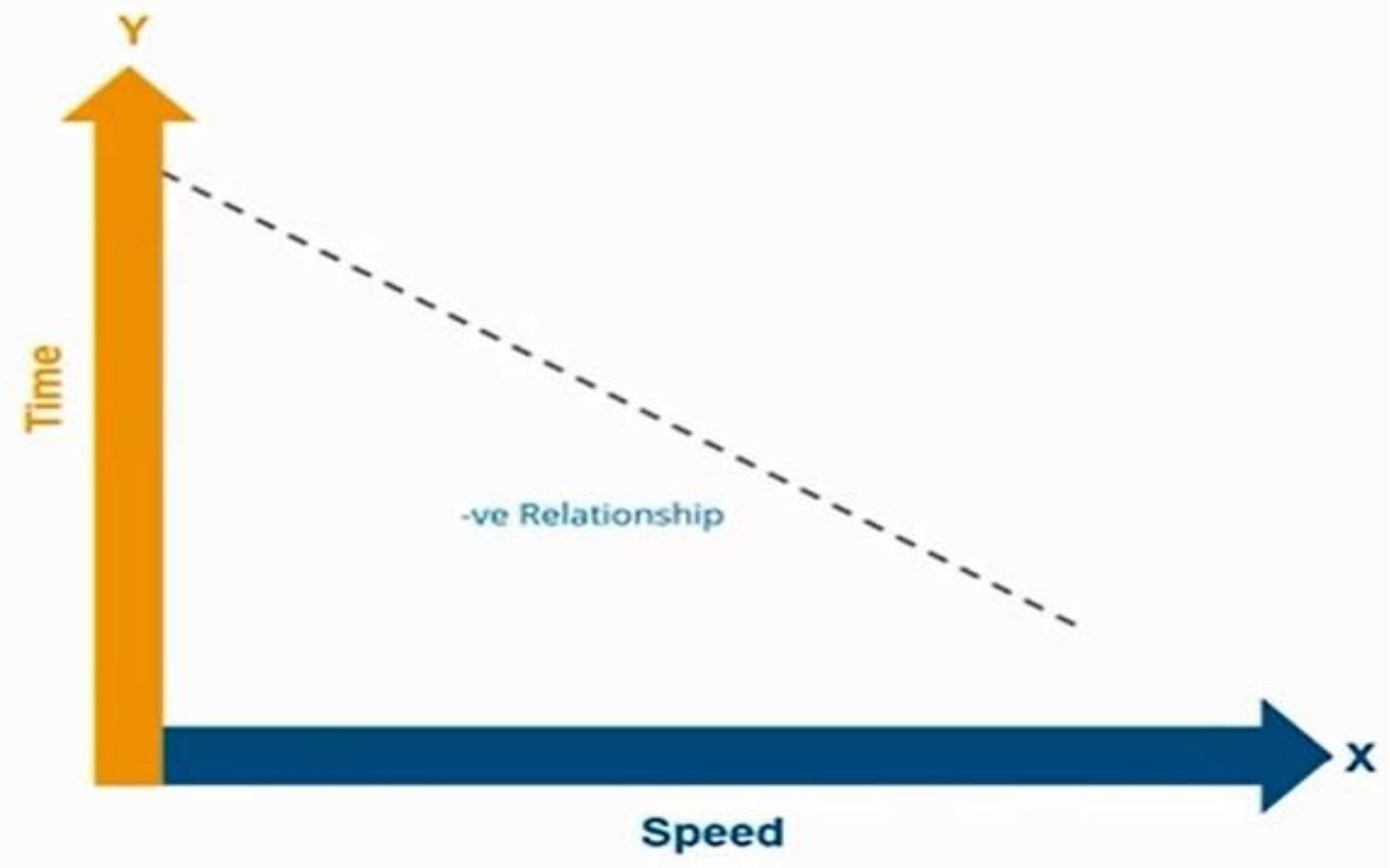
Understanding of Linear Regression Algorithm



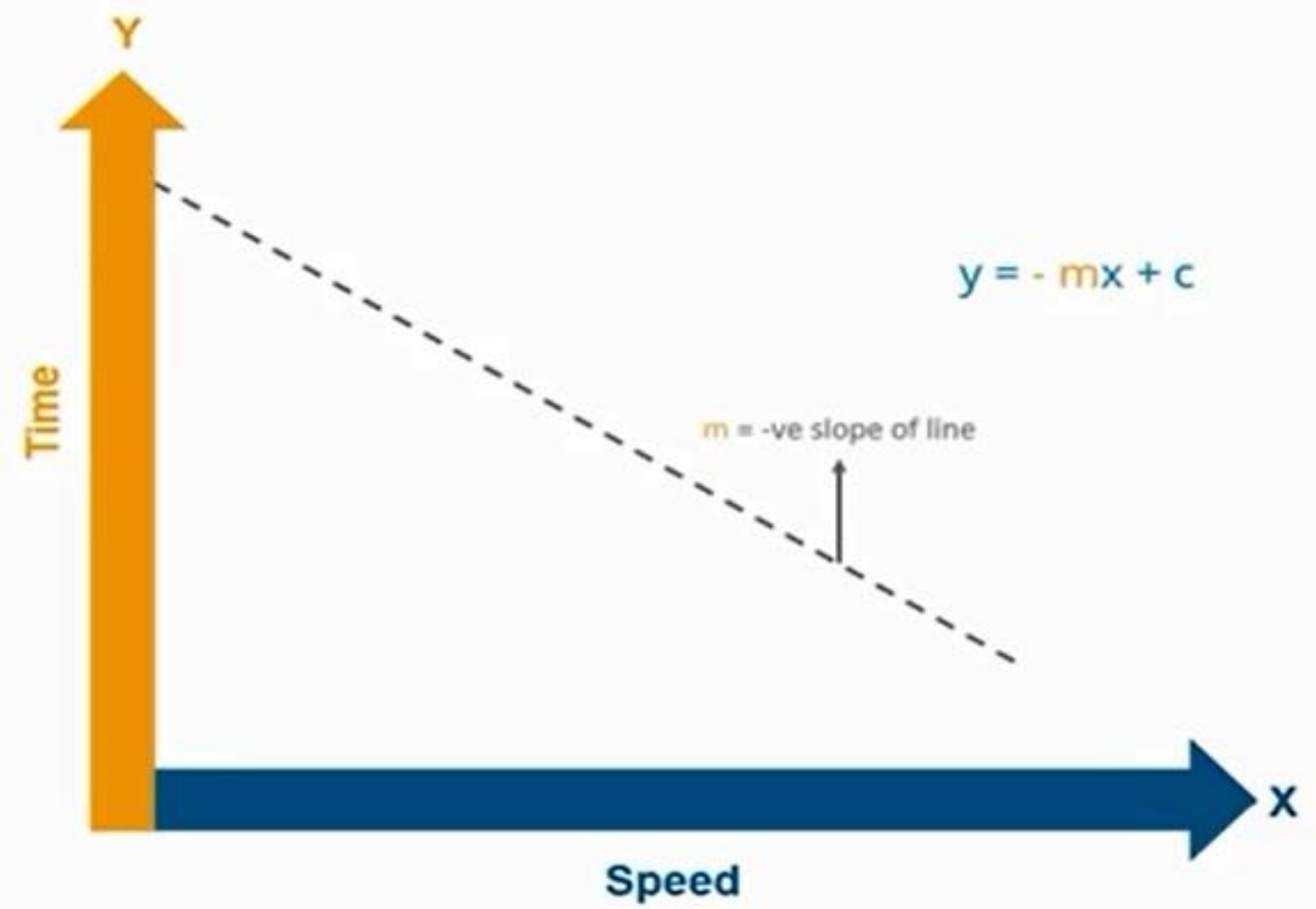
Understanding of Linear Regression Algorithm



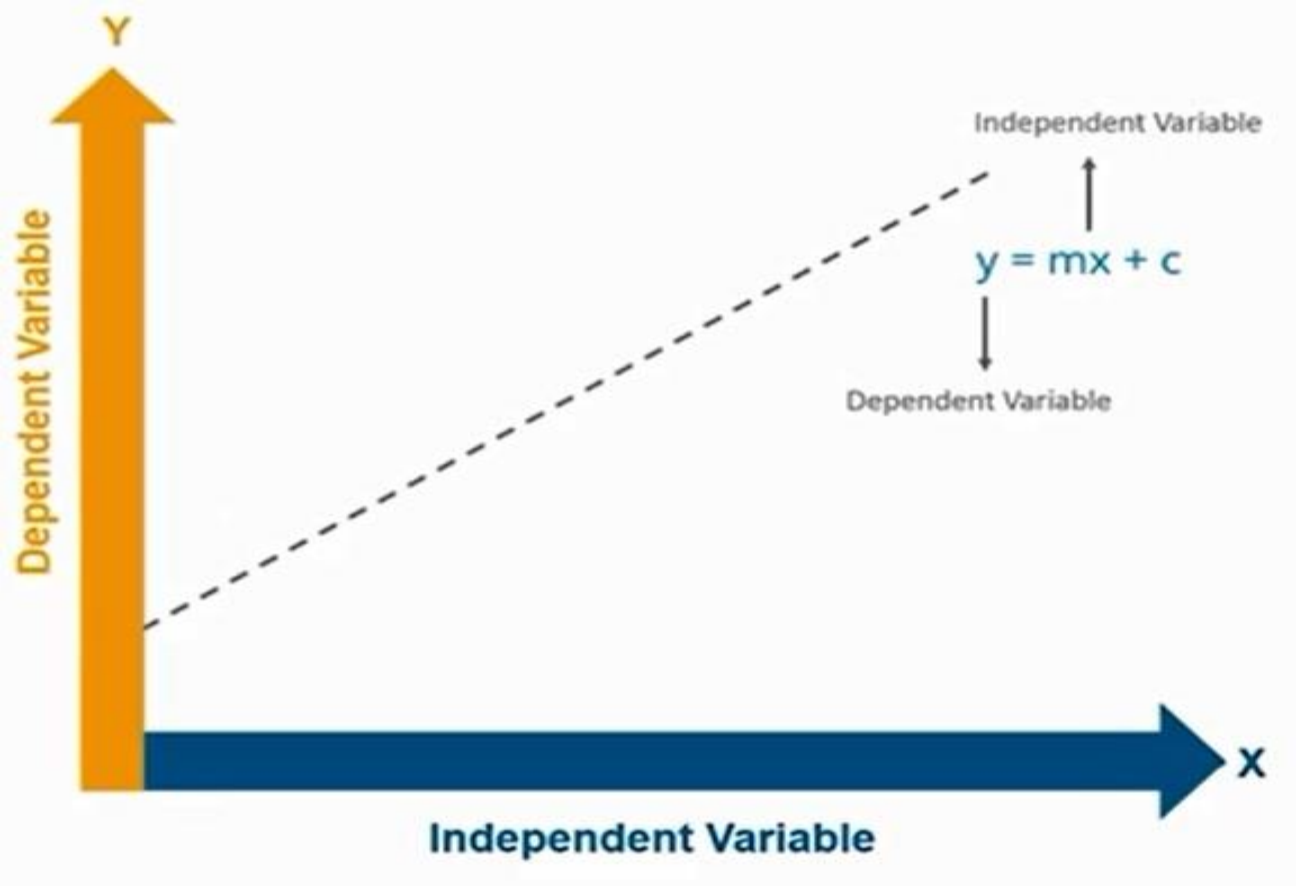
Understanding of Linear Regression Algorithm



Understanding of Linear Regression Algorithm



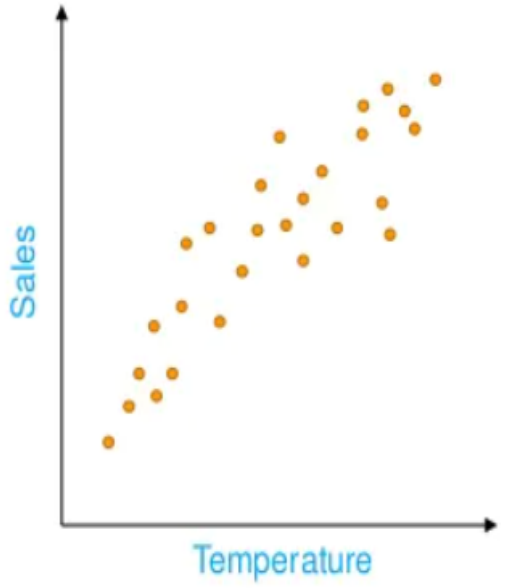
Understanding of Linear Regression Algorithm



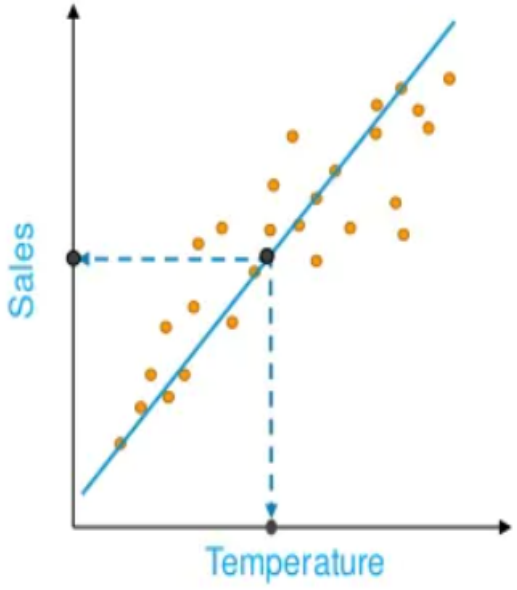
Linear Regression Method

Linear Regression is a linear modeling approach to find relationship between one or more independent variables (predictors) denoted as X and dependent variable (target) denoted as Y.

Predict sales of Ice Cream based on temperature:



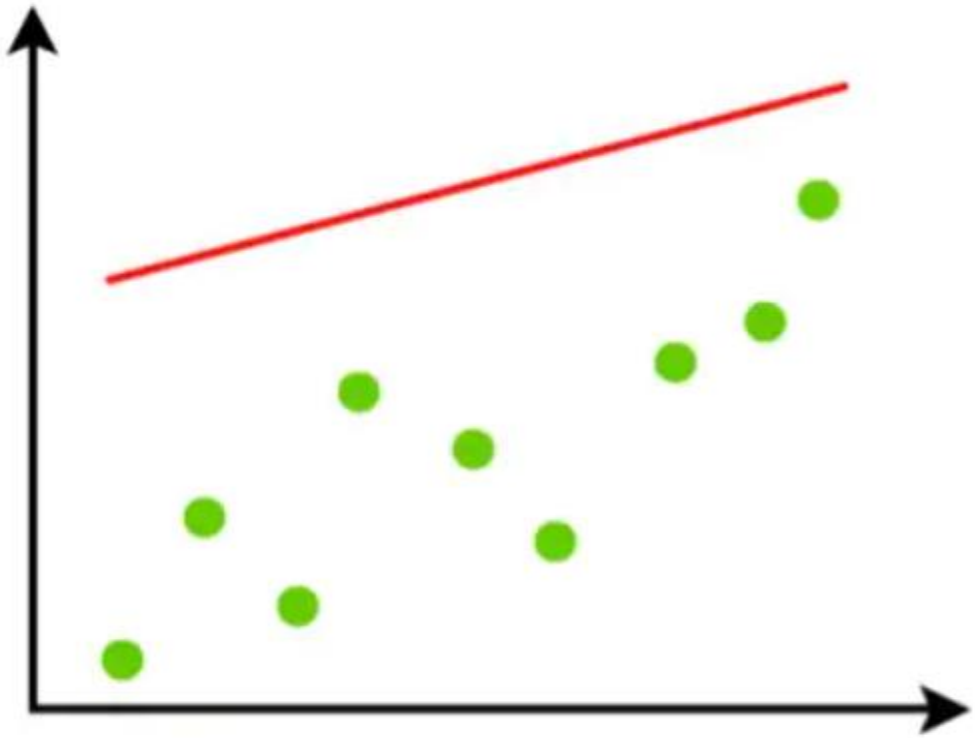
Plotting the sales of ice cream based on temperature



Regression line to predict the sales

Linear Regression Method

Finding the Best Fit Line-The best fit line can be found out by minimizing the distance between all the data points and the distance to the Regression Line. Ways to minimize this distance are sum of Squared error, sum of Absolute error.

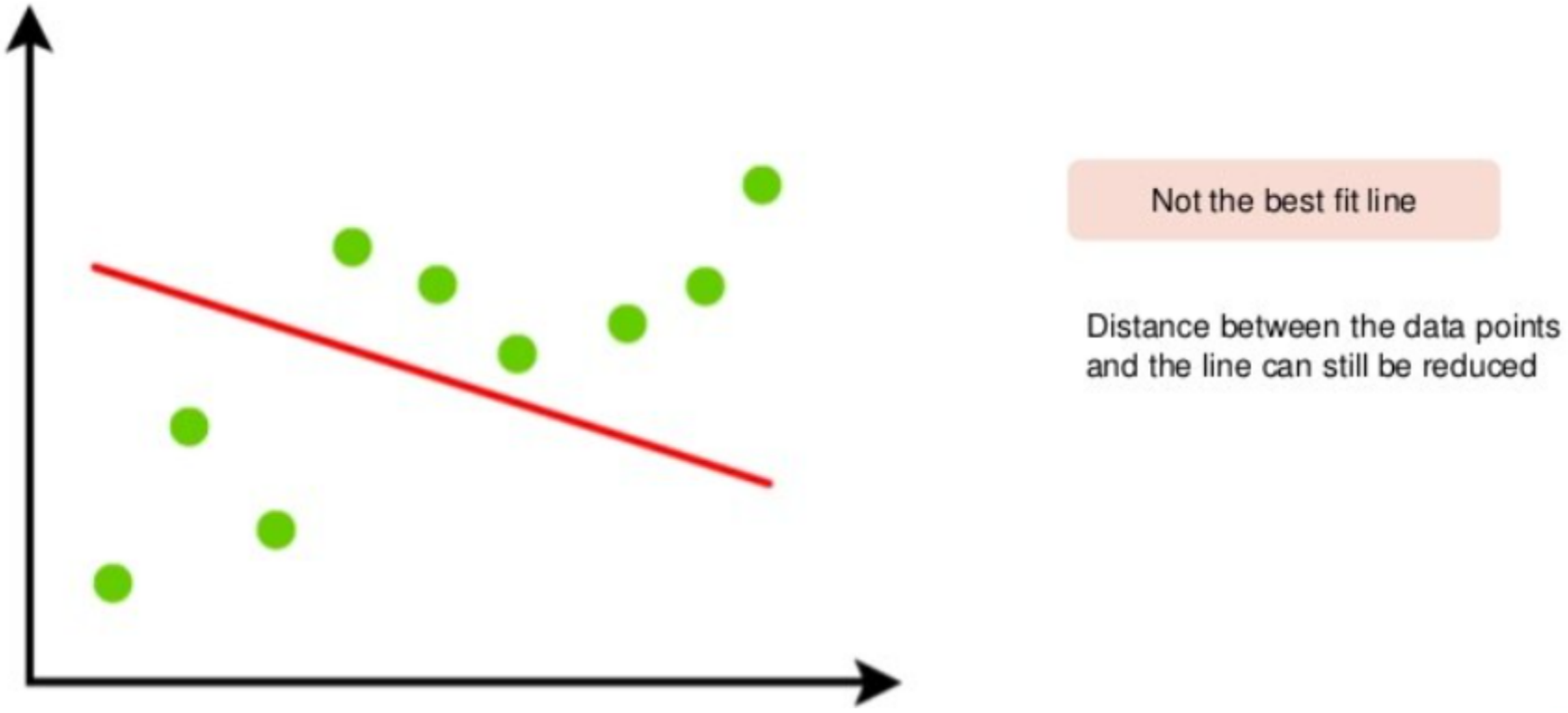


Not the best fit line

Distance between the data points and the line is maximum

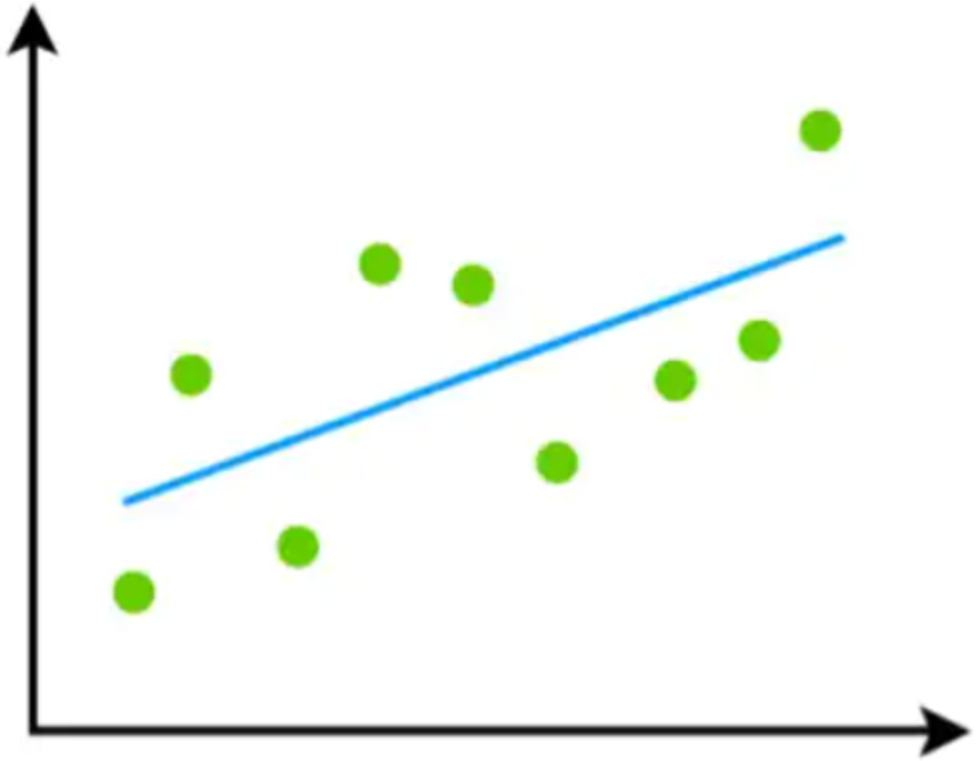
Linear Regression Method

Finding the Best Fit Line-The best fit line can be found out by minimizing the distance between all the data points and the distance to the Regression Line. Ways to minimize this distance are sum of Squared error, sum of Absolute error.



Linear Regression Method

Finding the Best Fit Line-The best fit line can be found out by minimizing the distance between all the data points and the distance to the Regression Line. Ways to minimize this distance are sum of Squared error, sum of Absolute error.

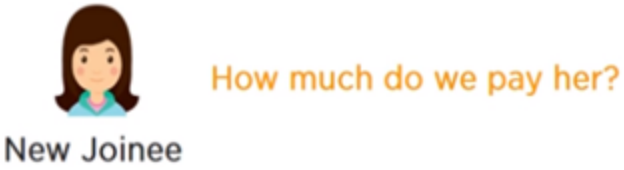


Best fit regression line

Distance between the data points and the line is the least

Implementation of Linear Regression

Linear Regression : Predict Employee Salary based on Years of Experience.



HR team trying to figure out how much to pay to a new joinee?

Employee data of Salary based on Years of experience

Predict Salary

Linear Regression Method

- If the output of a prediction is numerical, Regression should be used.
- Regression is one of the methods used for numerical prediction.
- Regression analysis models the relationship between one or more independent variables(results) and a dependent variable(input attributes).
- The simplest regression is fitting a line to a set of points. It can be described as

$$Y = W_0 + W_1 x \quad \textbf{or } y = mx + c$$

Where, W_0 & W_1 are the weights of the regression coefficients.

The coefficients can be calculated by the method of least squares to fit a line that minimizes the error between the actual data and the estimate. If D is the training set,

$$Wi = \frac{\sum_{i=1}^{|D|}(xi - \bar{x})(yi - \bar{y})}{\sum_{i=1}^{|D|}(xi - \bar{x})^2}$$

$W_0 = \bar{Y} - W_1 \bar{X}$, where $\bar{x}$ and $\bar{y}$ are the mean values of data x and y, respectively.

Regression Method-Numerical

1. Consider the sample dataset in table. What is the regression analysis?

Soln- A linear regression model can be obtained as follows. A line can be fitted to the given data as $y=W_0 + W_1x$

Size (x)	Number of pixels (y)
1	1
2	4
3	9
4	16
5	25

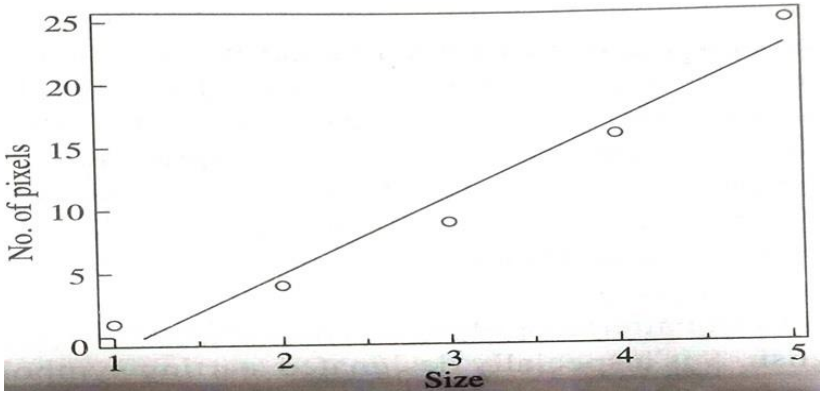
$x=15/5=3$, $y=55/5=11$

$$Wi = \frac{\sum_{i=1}^{n}(xi-\bar{x})(yi-\bar{y})}{\sum_{i=1}^{n}(xi-\bar{x})^2} = \frac{(1-3)(1-11)+(2-3)(4-11)+(3-3)(9-11)+(4-3)(16-11)+(5-3)(25-11)}{[(1-3)^2+(2-3)^2+(3-3)^2+(4-3)^2+(5-3)^2]}$$

$=60/10=6$, $W_0 = \bar{y}-W_1\bar{x} = 11-(6x3)=-7$

Regression Method-Numerical

The scatter plot and the regression line are shown in figure



By substituting these parameters, we get the regression equation for the dataset

As $\mathbf{y=W_0 + W_1x}$ **or** $\mathbf{y = mx+c}$

That is, $\mathbf{y= -7 + 6x}$

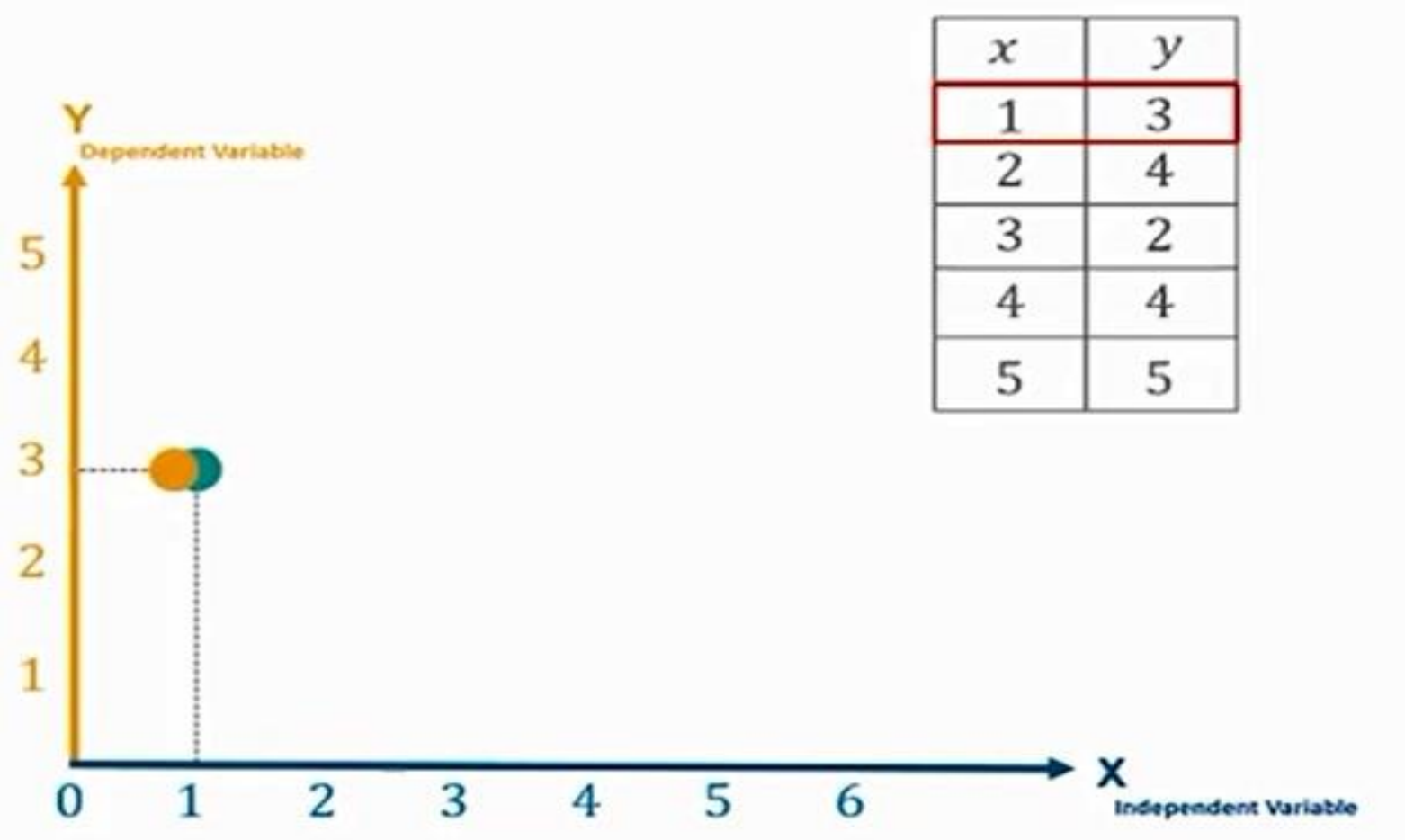
For example, when **x=3, this yields y=-7+6(3)=11.**

However, **the actual value is 9**. **The difference of 2 is called the prediction error.** The model is accurate when more data is present along the direction of the regression line.

If three attributes are involved (say A_1, A_2, and A_3), the linear regression model would be

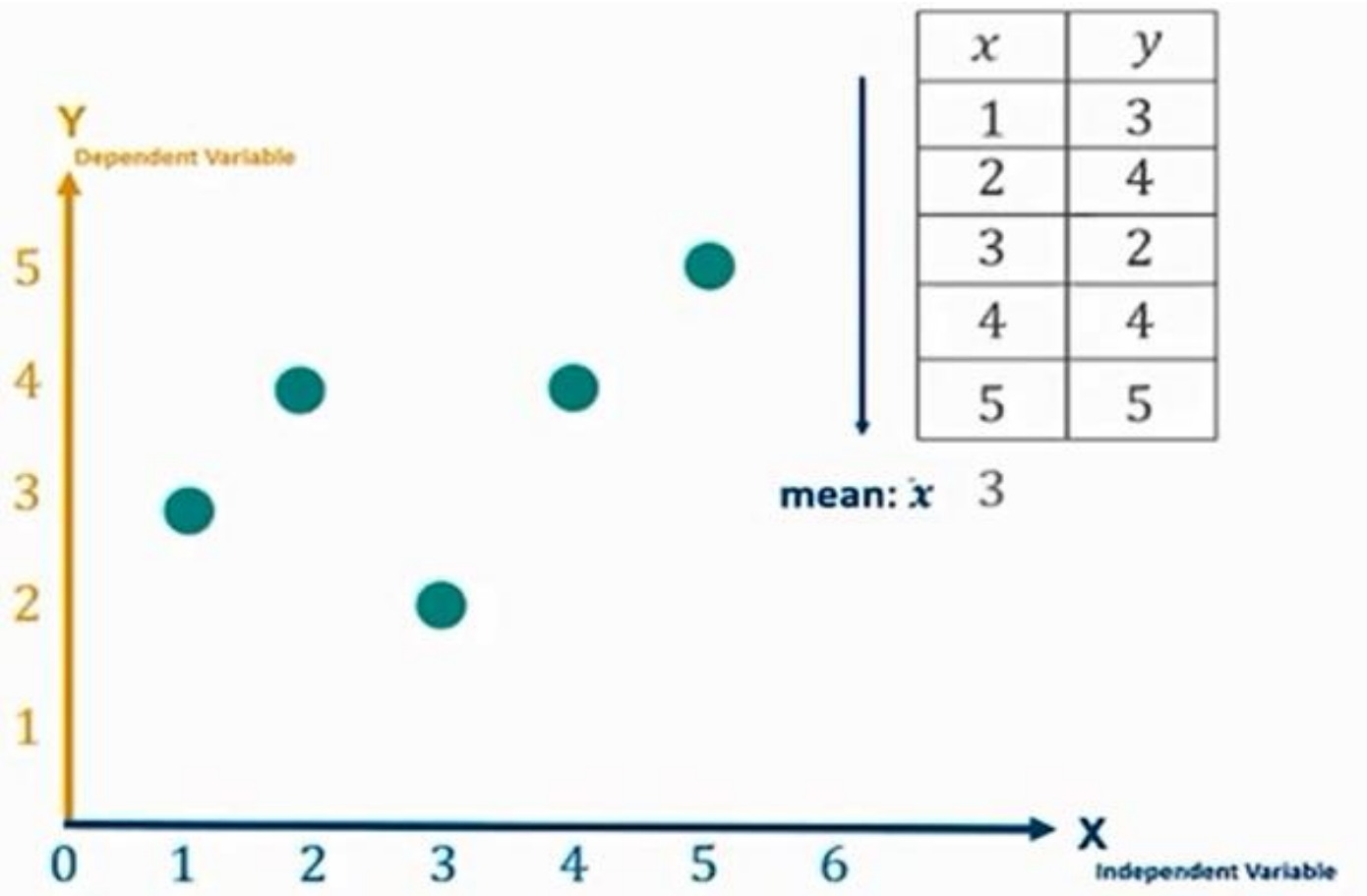
$$y=w_0+w_1x_1+w_2x_2+w_3x_3$$

Understanding of Linear Regression Algorithm



x	y
1	3
2	4
3	2
4	4
5	5

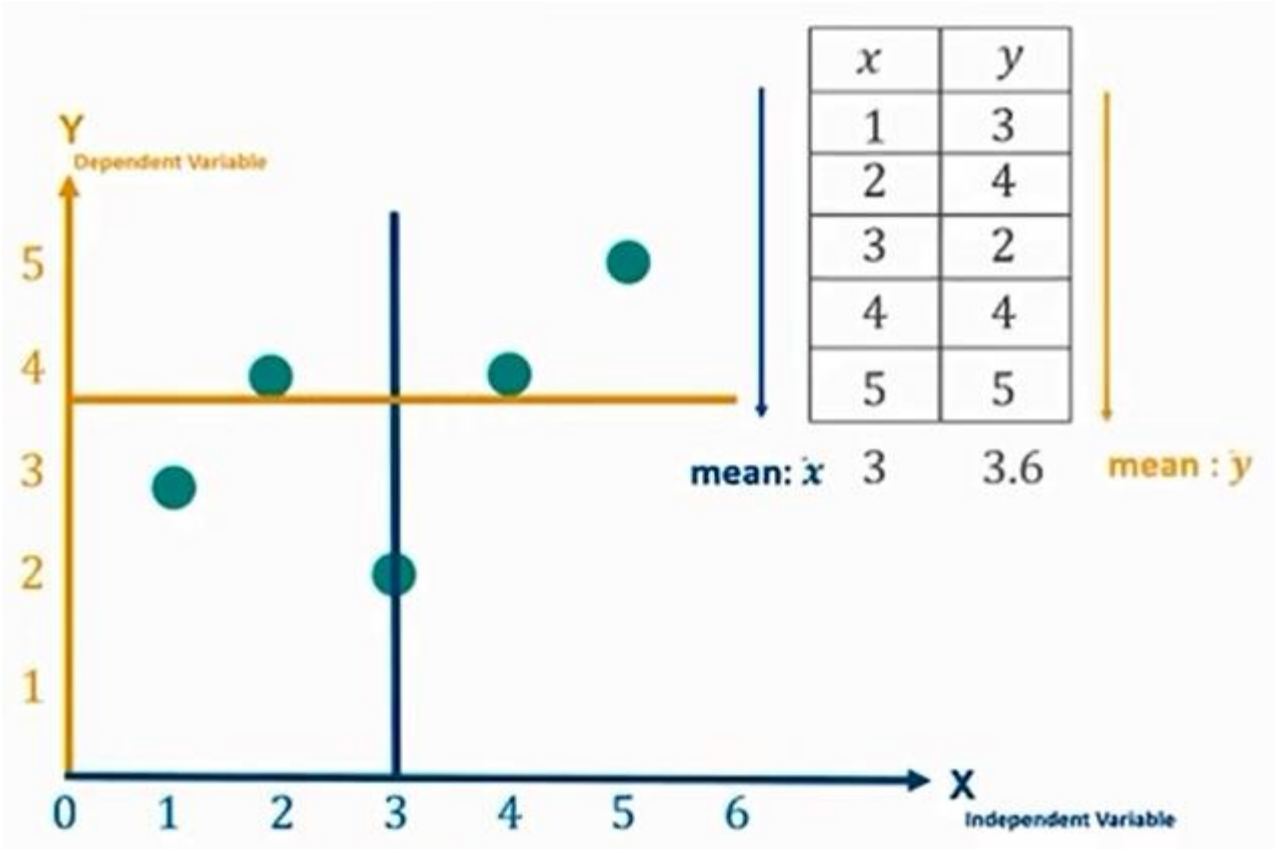
Understanding of Linear Regression Algorithm



x	y
1	3
2	4
3	2
4	4
5	5

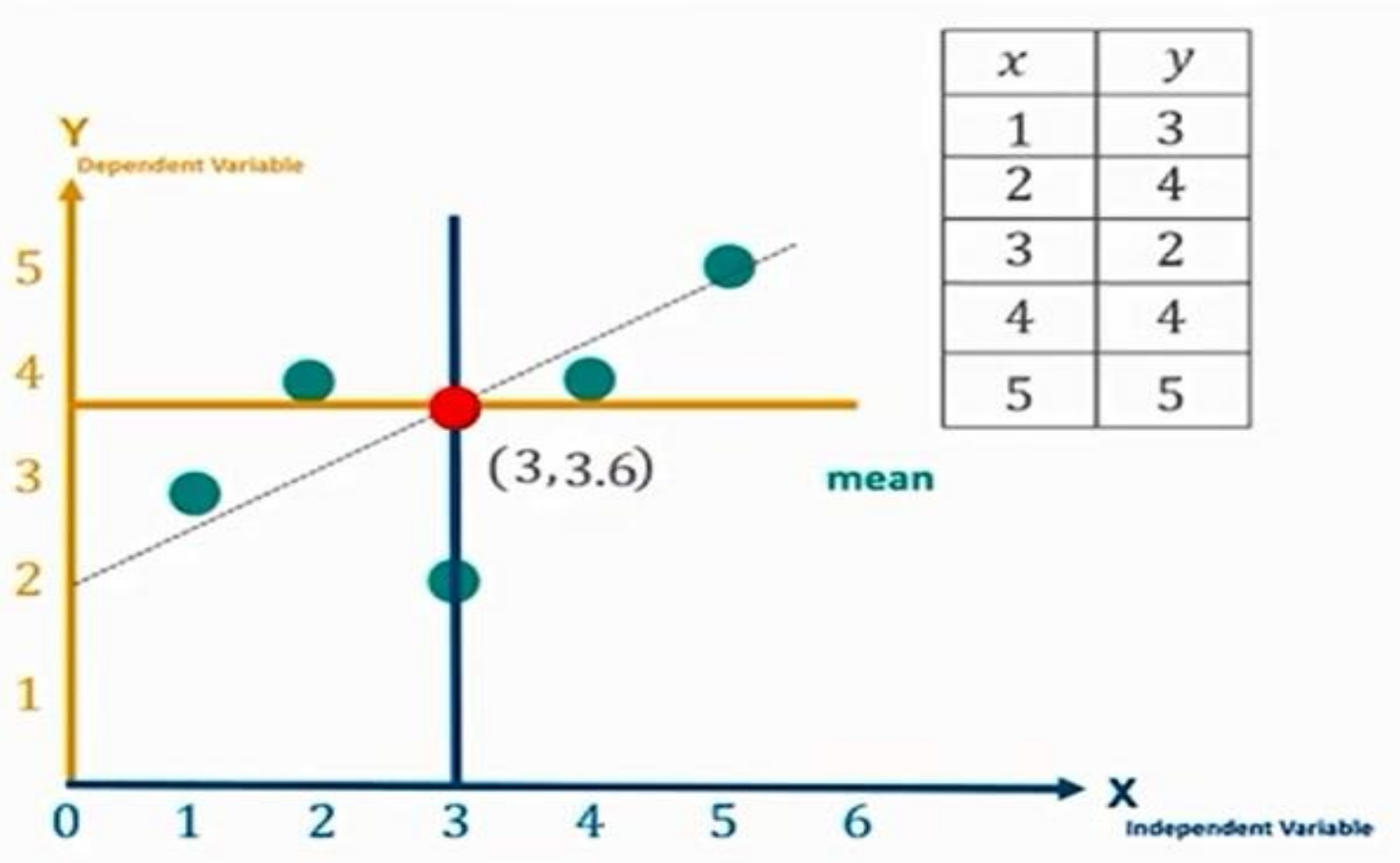
mean: $\bar{x}$ 3

Understanding of Linear Regression Algorithm



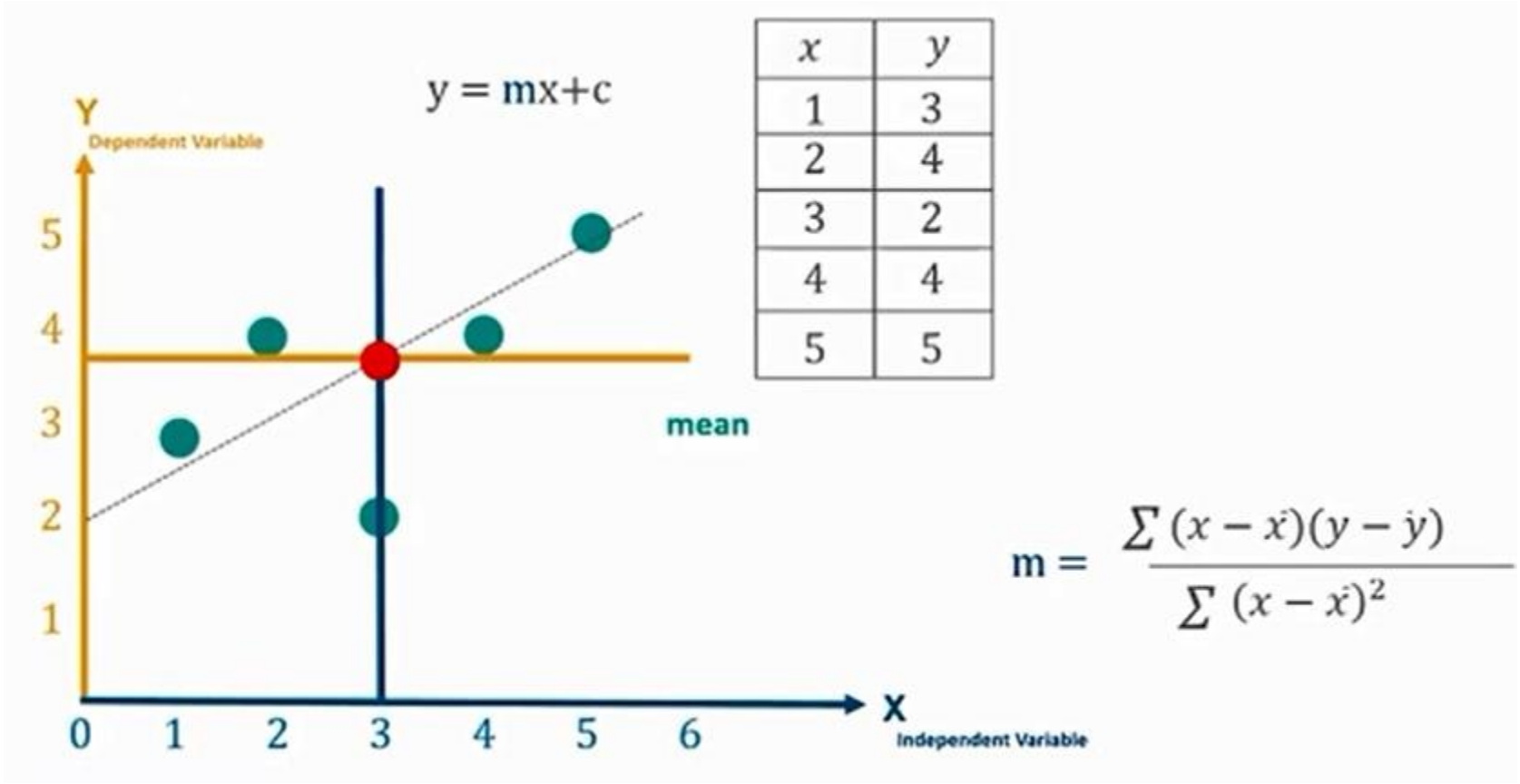
x	y
1	3
2	4
3	2
4	4
5	5

Understanding of Linear Regression Algorithm



x	y
1	3
2	4
3	2
4	4
5	5

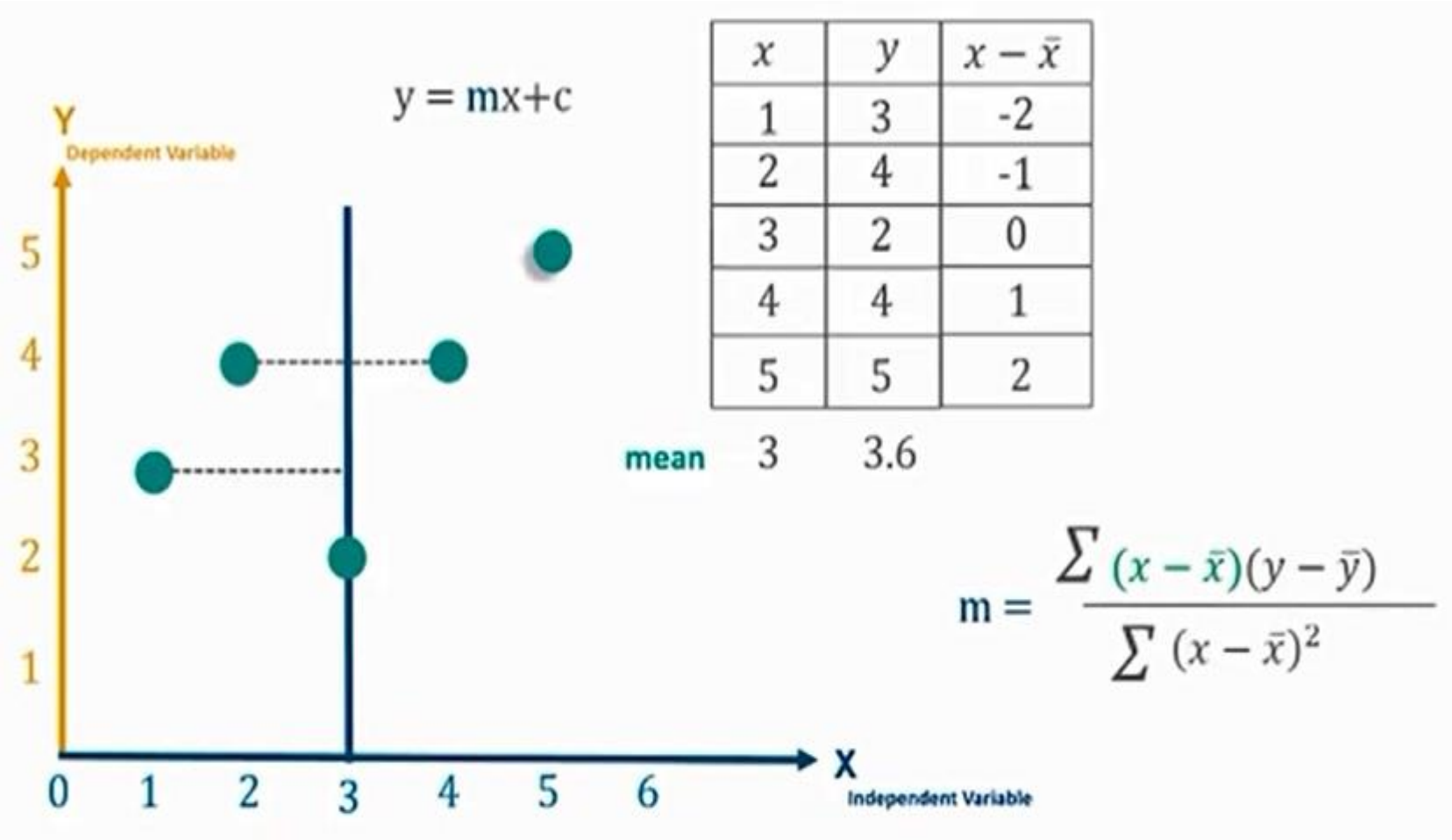
Understanding of Linear Regression Algorithm



x	y
1	3
2	4
3	2
4	4
5	5

$$m = \frac{\sum (x - \dot{x})(y - \dot{y})}{\sum (x - \dot{x})^2}$$

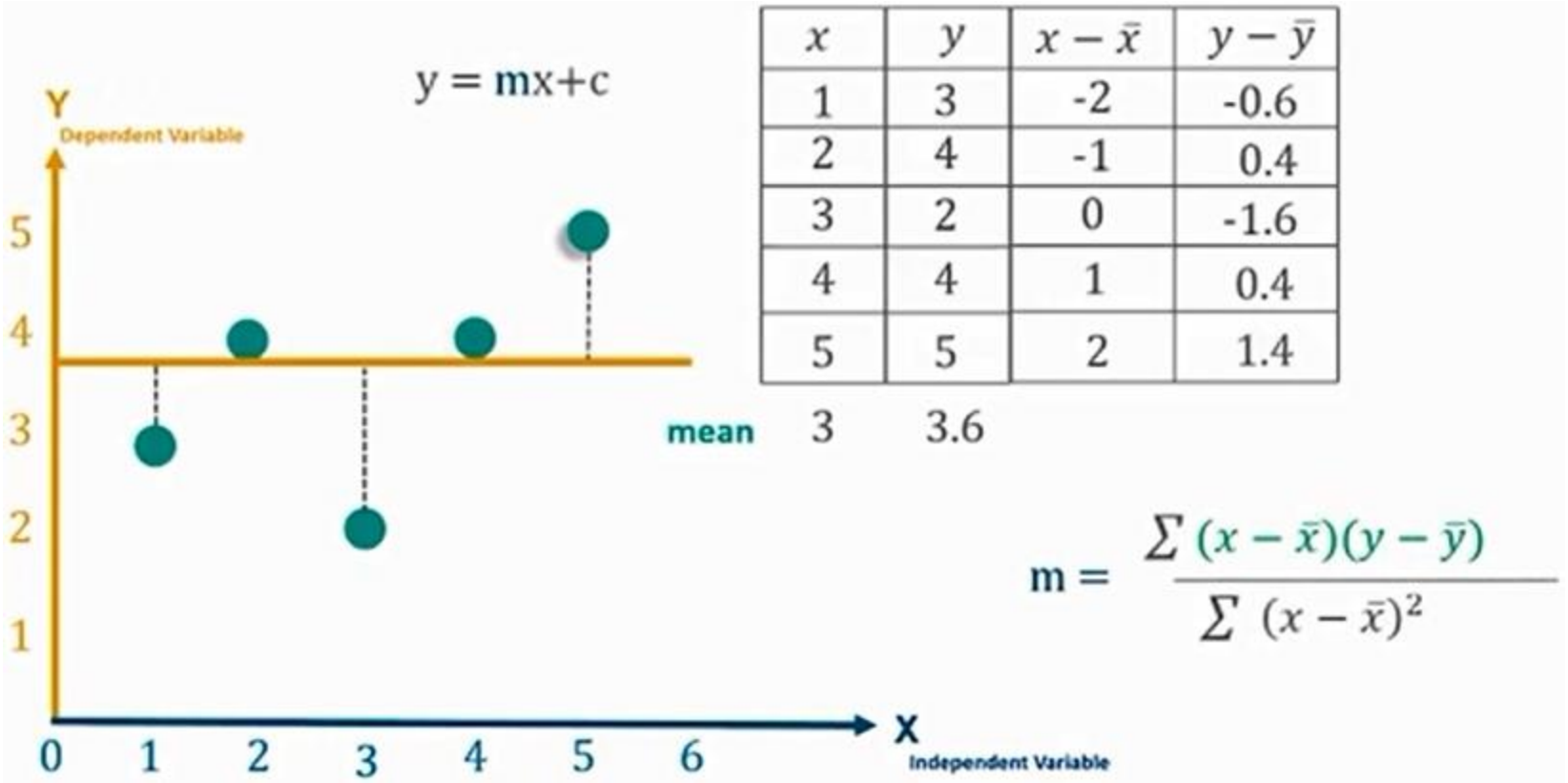
Understanding of Linear Regression Algorithm



x	y	$x - \bar{x}$
1	3	-2
2	4	-1
3	2	0
4	4	1
5	5	2
mean	3	3.6

$$m = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2}$$

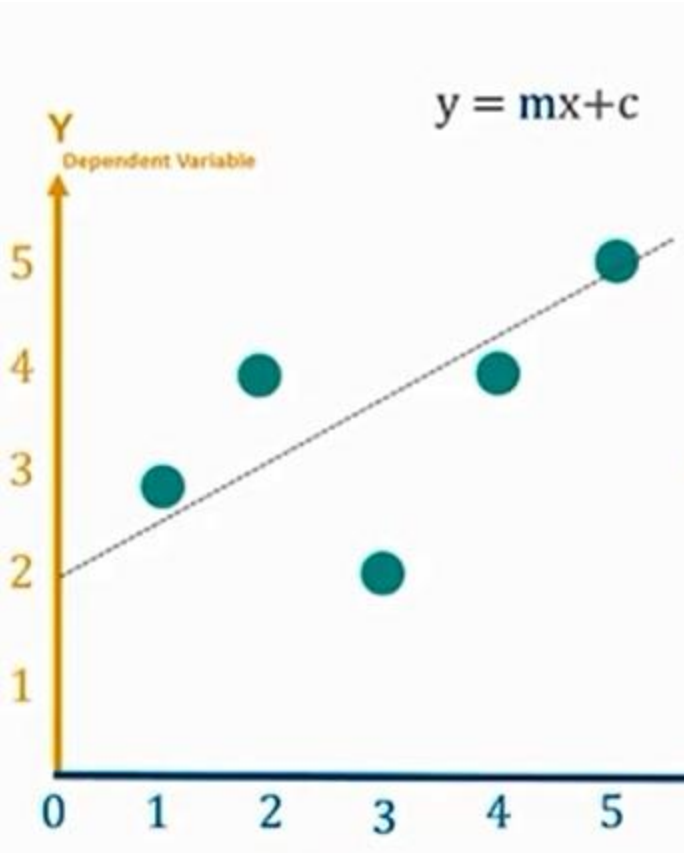
Understanding of Linear Regression Algorithm



x	y	$x-\bar{x}$	$y-\bar{y}$
1	3	-2	-0.6
2	4	-1	0.4
3	2	0	-1.6
4	4	1	0.4
5	5	2	1.4
mean	3	3.6	

$$m = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sum (x-\bar{x})^2}$$

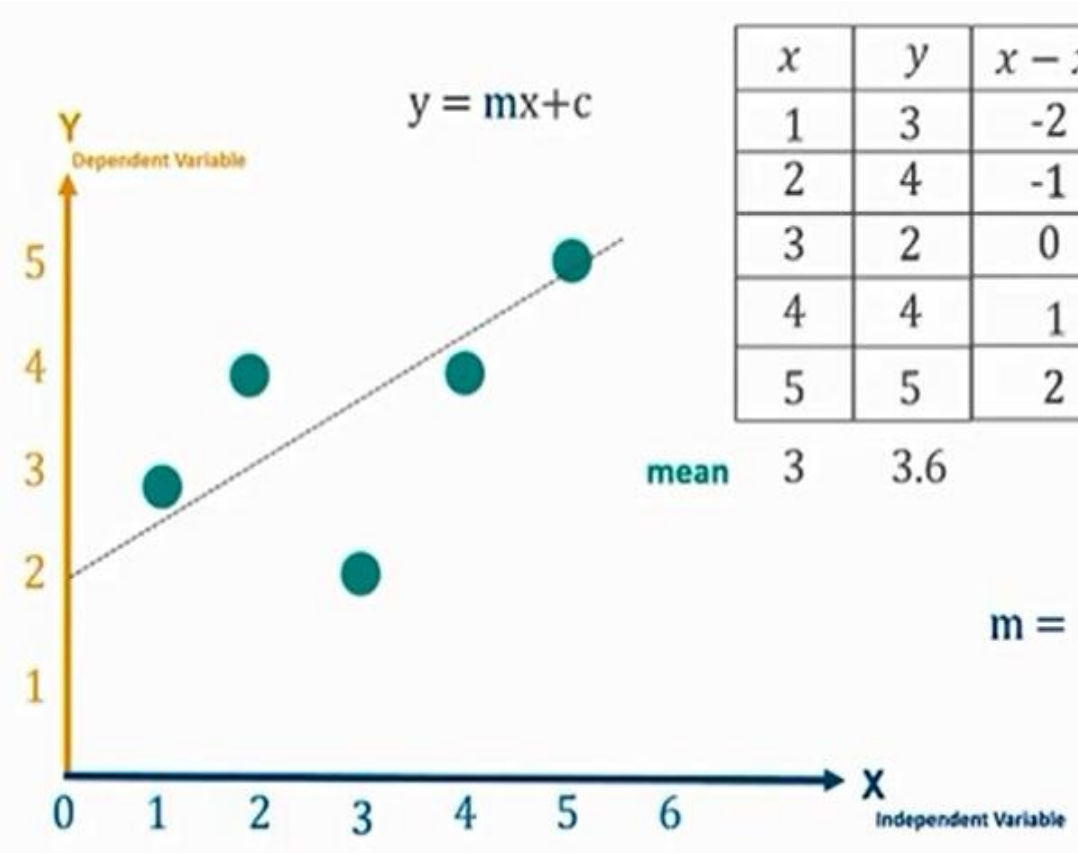
Understanding of Linear Regression Algorithm



x	y	$x - \dot{x}$	$y - \dot{y}$	$(x - \dot{x})^2$
1	3	-2	-0.6	4
2	4	-1	0.4	1
3	2	0	-1.6	0
4	4	1	0.4	1
5	5	2	1.4	4
mean	3		3.6	

$$m = \frac{\sum (x - \dot{x})(y - \dot{y})}{\sum (x - \dot{x})^2}$$

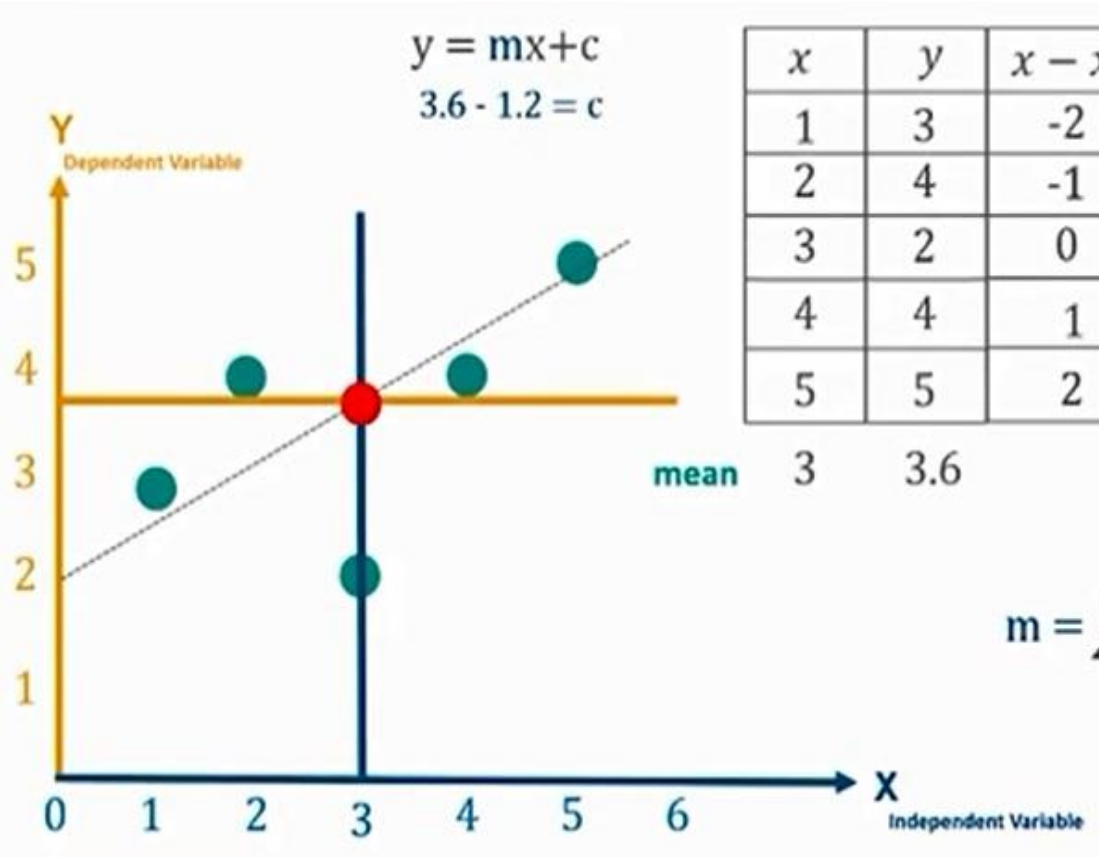
Understanding of Linear Regression Algorithm



x	y	$x-\dot{x}$	$y-\dot{y}$	$(x-\dot{x})^2$	$(x-\dot{x})(y-\dot{y})$
1	3	-2	-0.6	4	1.2
2	4	-1	0.4	1	-0.4
3	2	0	-1.6	0	0
4	4	1	0.4	1	0.4
5	5	2	1.4	4	2.8
mean	3	3.6		$\Sigma = 10$	$\Sigma = 4$

$$m = \frac{\sum (x-\dot{x})(y-\dot{y})}{\sum (x-\dot{x})^2} = \frac{4}{10}$$

Understanding of Linear Regression Algorithm



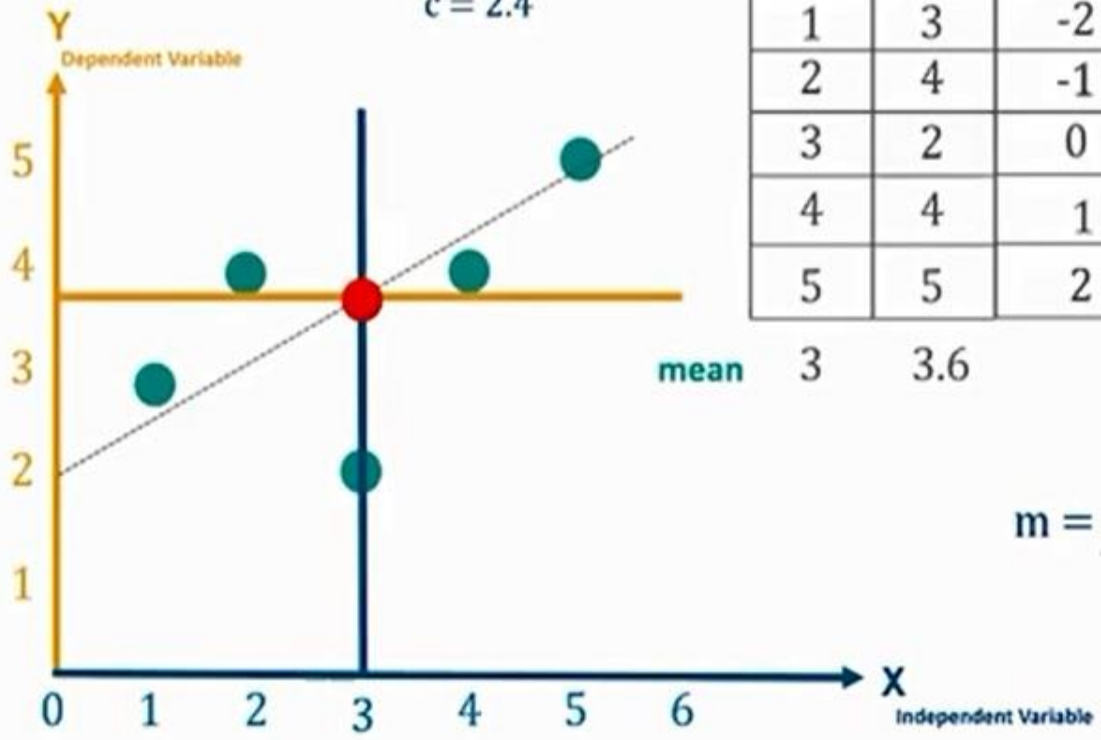
x	y	$x - \dot{x}$	$y - \dot{y}$	$(x - \dot{x})^2$	$(x - \dot{x})(y - \dot{y})$
1	3	-2	-0.6	4	1.2
2	4	-1	0.4	1	-0.4
3	2	0	-1.6	0	0
4	4	1	0.4	1	0.4
5	5	2	1.4	4	2.8
mean	3	3.6		$\Sigma = 10$	$\Sigma = 4$

$$m = \sum \frac{(x - \dot{x})(y - \dot{y})}{(x - \dot{x})^2} = \frac{4}{10}$$

Understanding of Linear Regression Algorithm

$y = mx+c$

$c = 2.4$



x	y	$x-\dot{x}$	$y-\dot{y}$	$(x-\dot{x})^2$	$(x-\dot{x})(y-\dot{y})$
1	3	-2	-0.6	4	1.2
2	4	-1	0.4	1	-0.4
3	2	0	-1.6	0	0
4	4	1	0.4	1	0.4
5	5	2	1.4	4	2.8
mean	3	3.6		$\Sigma = 10$	$\Sigma = 4$

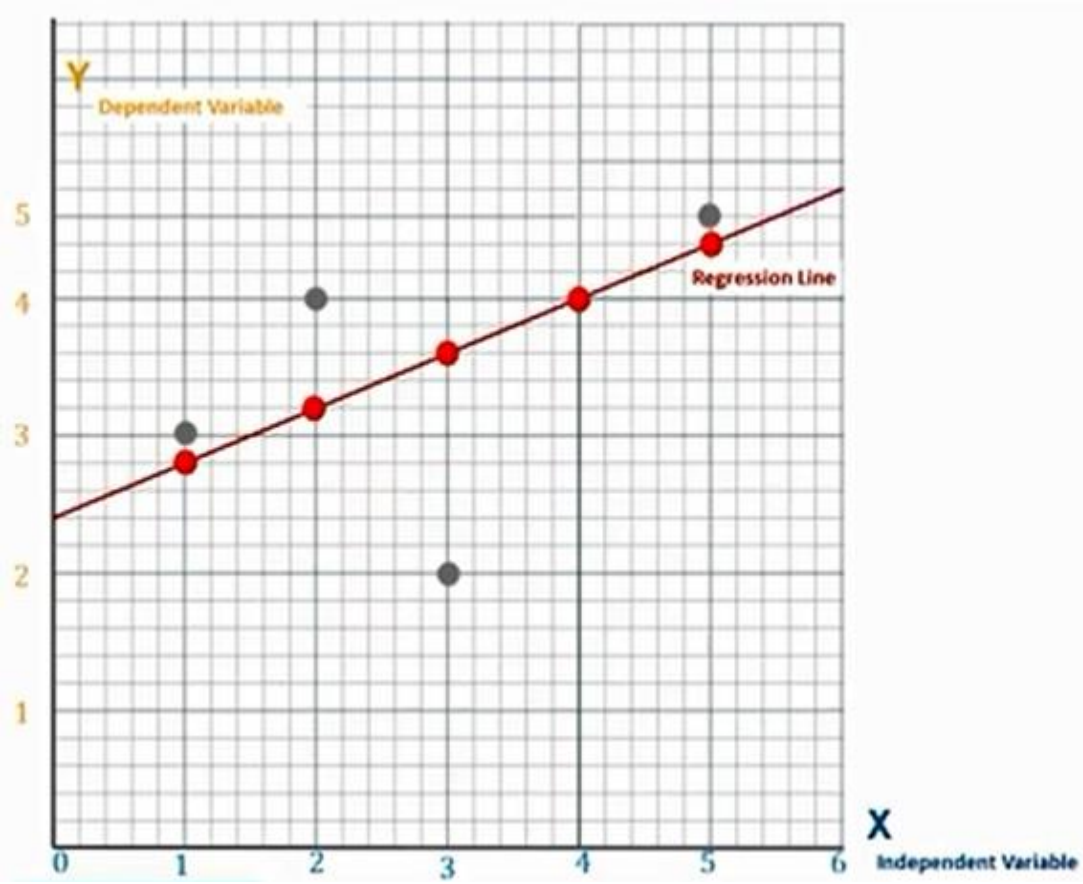
$$m = \sum \frac{(x-\dot{x})(y-\dot{y})}{(x-\dot{x})^2} = \frac{4}{10}$$

$m = 0.4$

$c = 2.4$

$y = 0.4x + 2.4$

Understanding of Linear Regression Algorithm



$m = 0.4$

$c = 2.4$

$y = 0.4x + 2.4$

For given $m = 0.4$ & $c = 2.4$, lets predict values for y for $x = \{1,2,3,4,5\}$

$y = 0.4 \times 1 + 2.4 = 2.8$

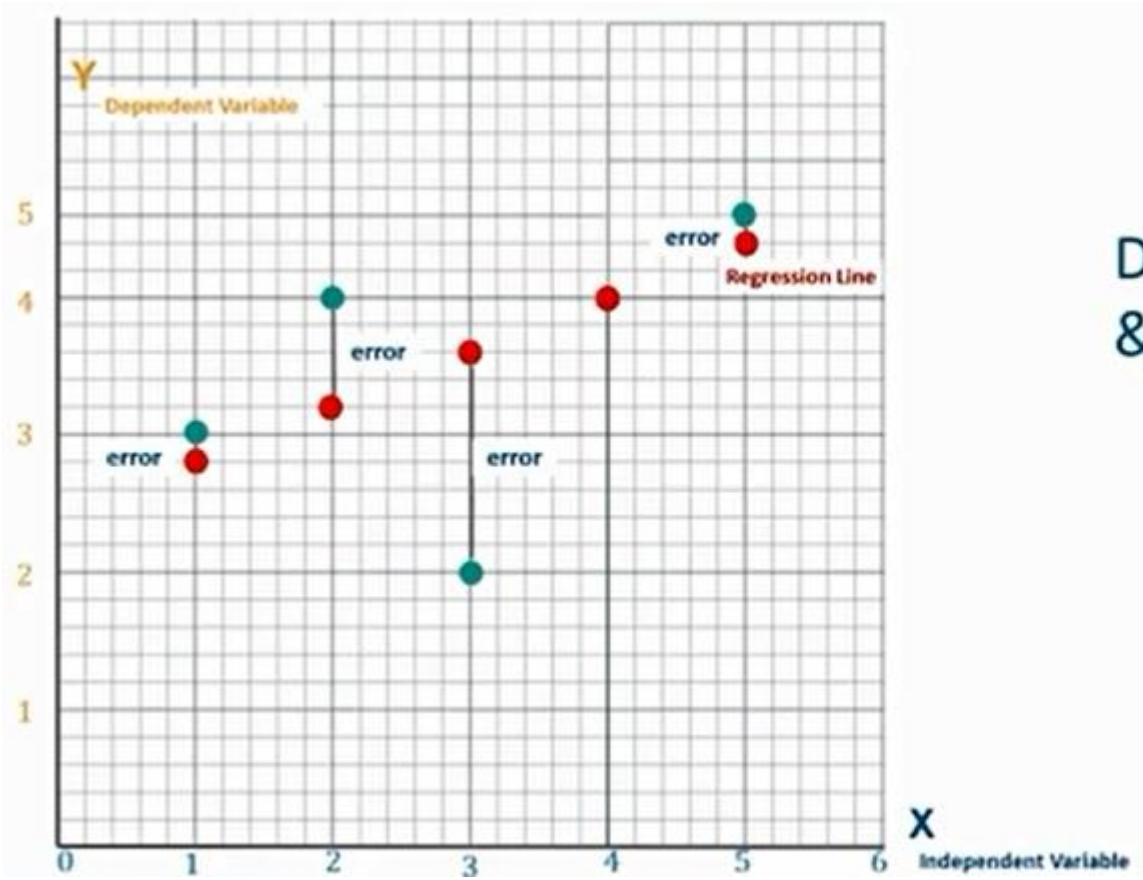
$y = 0.4 \times 2 + 2.4 = 3.2$

$y = 0.4 \times 3 + 2.4 = 3.6$

$y = 0.4 \times 4 + 2.4 = 4.0$

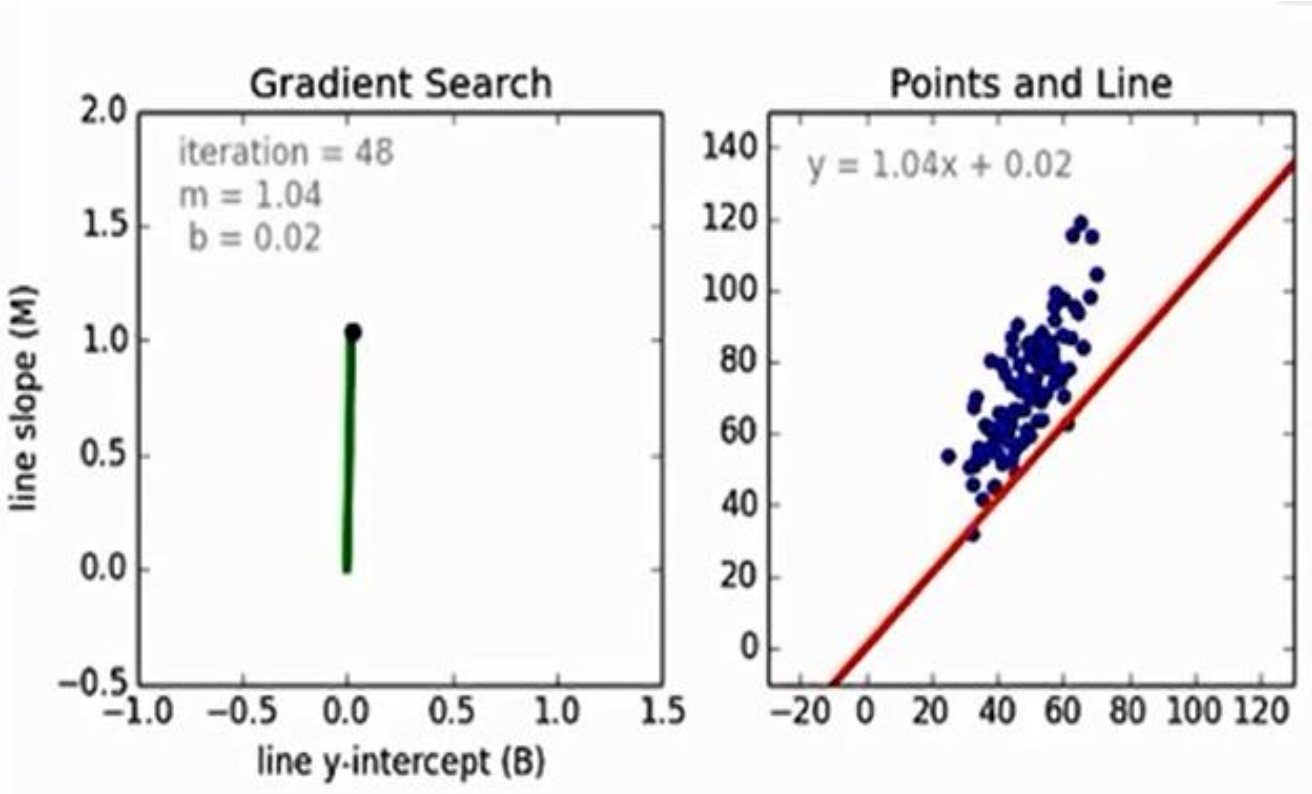
$y = 0.4 \times 5 + 2.4 = 4.4$

Understanding of Linear Regression Algorithm



Distance between actual & predicted value

Finding the best fit line

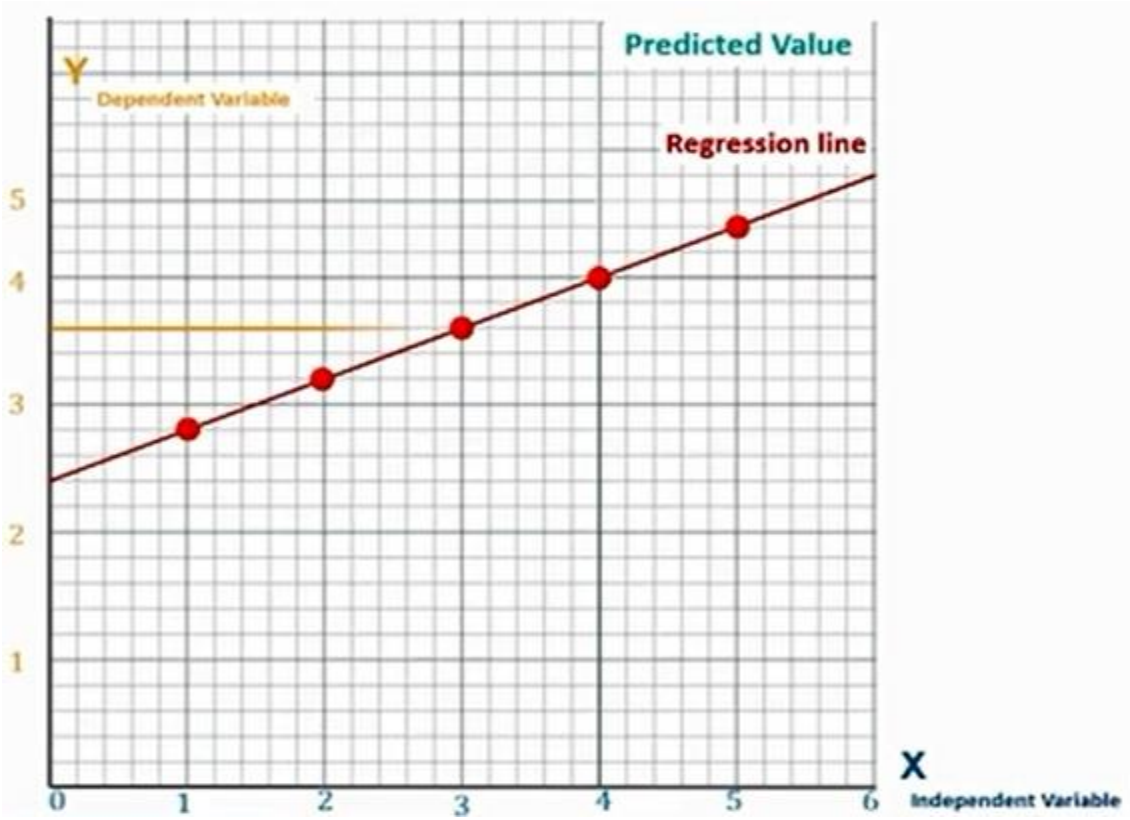


Lets check the goodness of fit

What is R-Square method

R-squared value is a statistical measure of how close the data are to be fitted **Regression Line**.

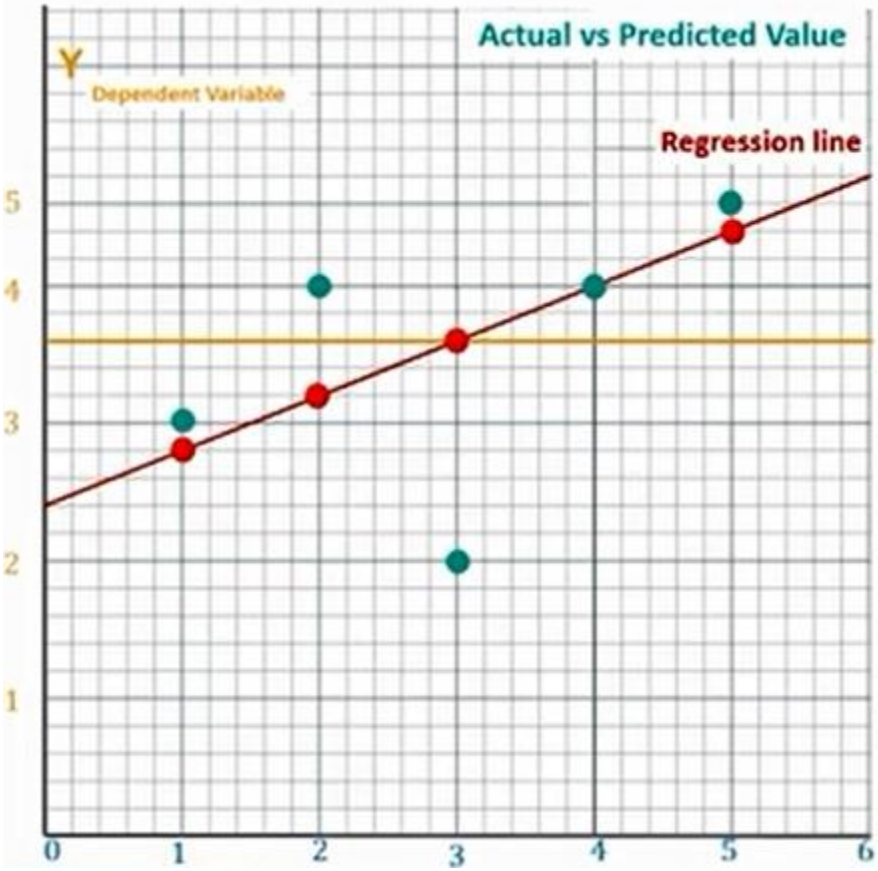
It is also known as coefficient of determination, or the coefficient of multiple determination.



x	y_p
1	2.8
2	3.2
3	3.6
4	4.0
5	4.4

Lets check the goodness of fit

R-Squared method



Distance actual - mean

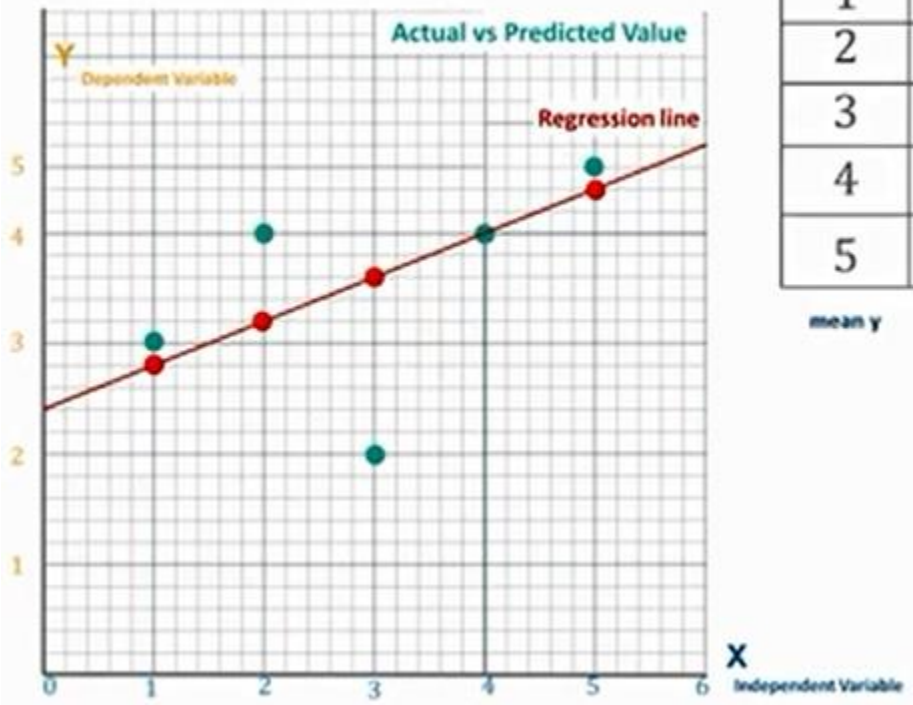
vs

Distance predicted - mean

This is nothing but $R^2 = \frac{\sum (y_p - \bar{y})^2}{\sum (y - \bar{y})^2}$

Lets check the goodness of fit

R-Squared method

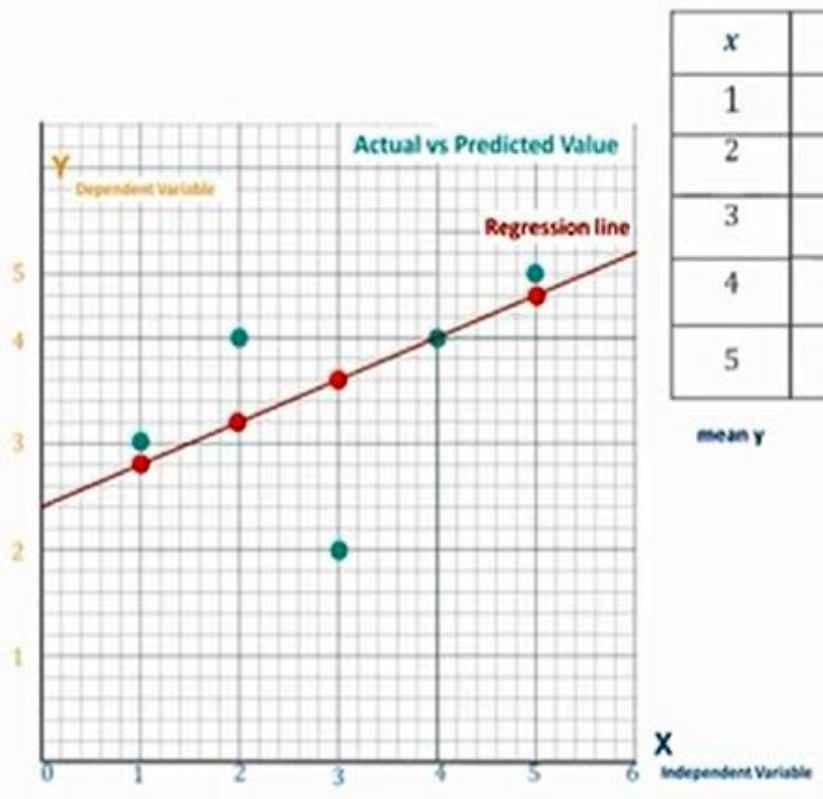


x	y	$y-\bar{y}$	$(y-\bar{y})^2$	y_p	$(y_p-\bar{y})$
1	3	-0.6	0.36	2.8	-0.8
2	4	0.4	0.16	3.2	-0.4
3	2	-1.6	2.56	3.6	0
4	4	0.4	0.16	4.0	0.4
5	5	1.4	1.96	4.4	0.8
mean y	3.6				

$$R^2 = \frac{\sum (y_p - \bar{y})^2}{\sum (y - \bar{y})^2}$$

Lets check the goodness of fit

R-Squared method

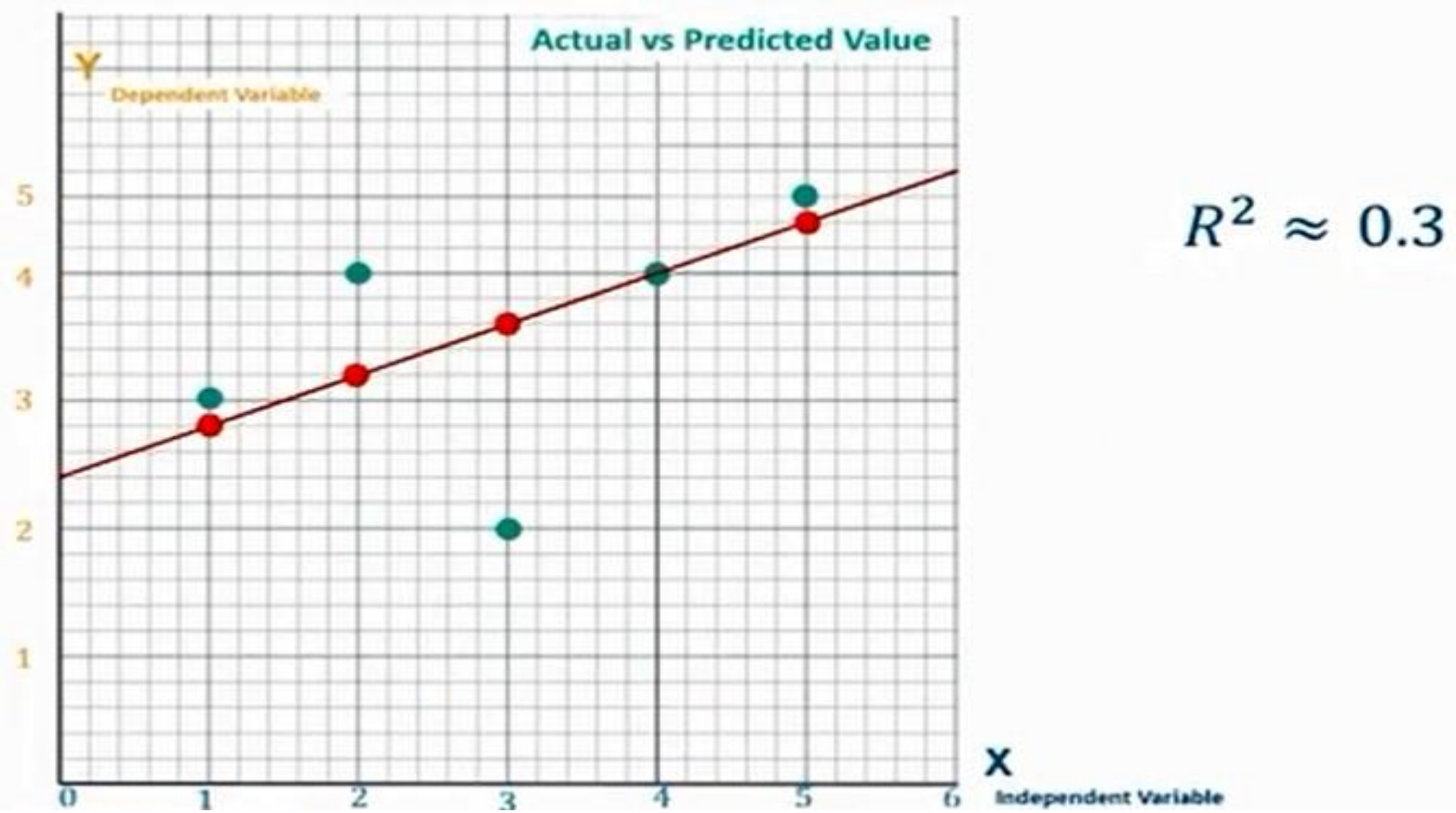


x	y	$y - y$	$(y-y)^2$	y_p	$(y_p - y)$	$(y_p - y)^2$
1	3	- 0.6	0.36	2.8	-0.8	0.64
2	4	0.4	0.16	3.2	-0.4	0.16
3	2	-1.6	2.56	3.6	0	0
4	4	0.4	0.16	4.0	0.4	0.16
5	5	1.4	1.96	4.4	0.8	0.64
mean y	3.6		$\sum$ 5.2			$\sum$ 1.6

$$R^2 = \frac{1.6}{5.2} = \frac{\sum \left((y_p - \bar{y}\right)^2}{\sum (y - \bar{y})^2}$$

Lets check the goodness of fit

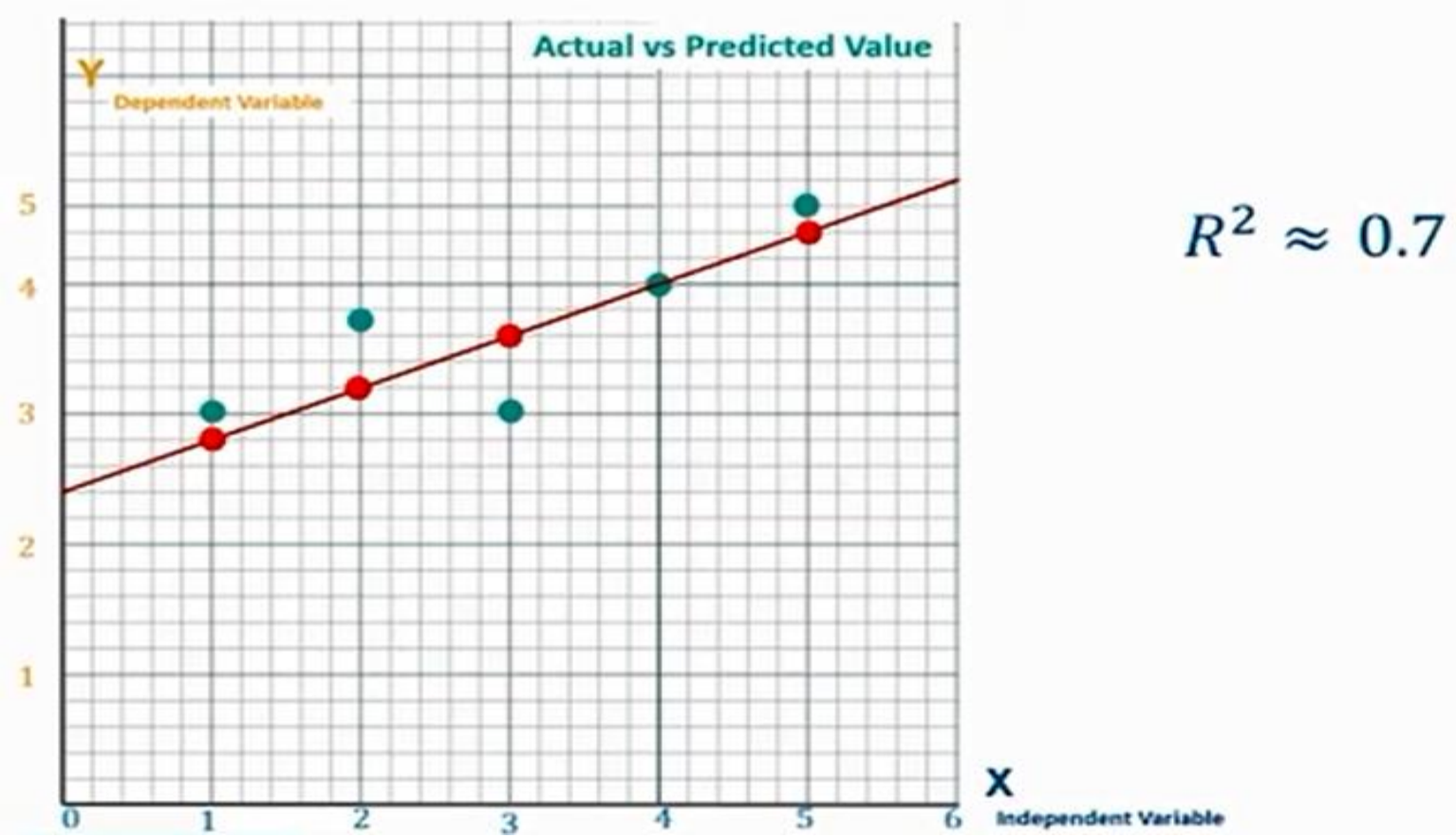
R-Squared method



- **Check how good is the model performing**
- **A "good" R-Squared value depends on the context.**

Lets check the goodness of fit

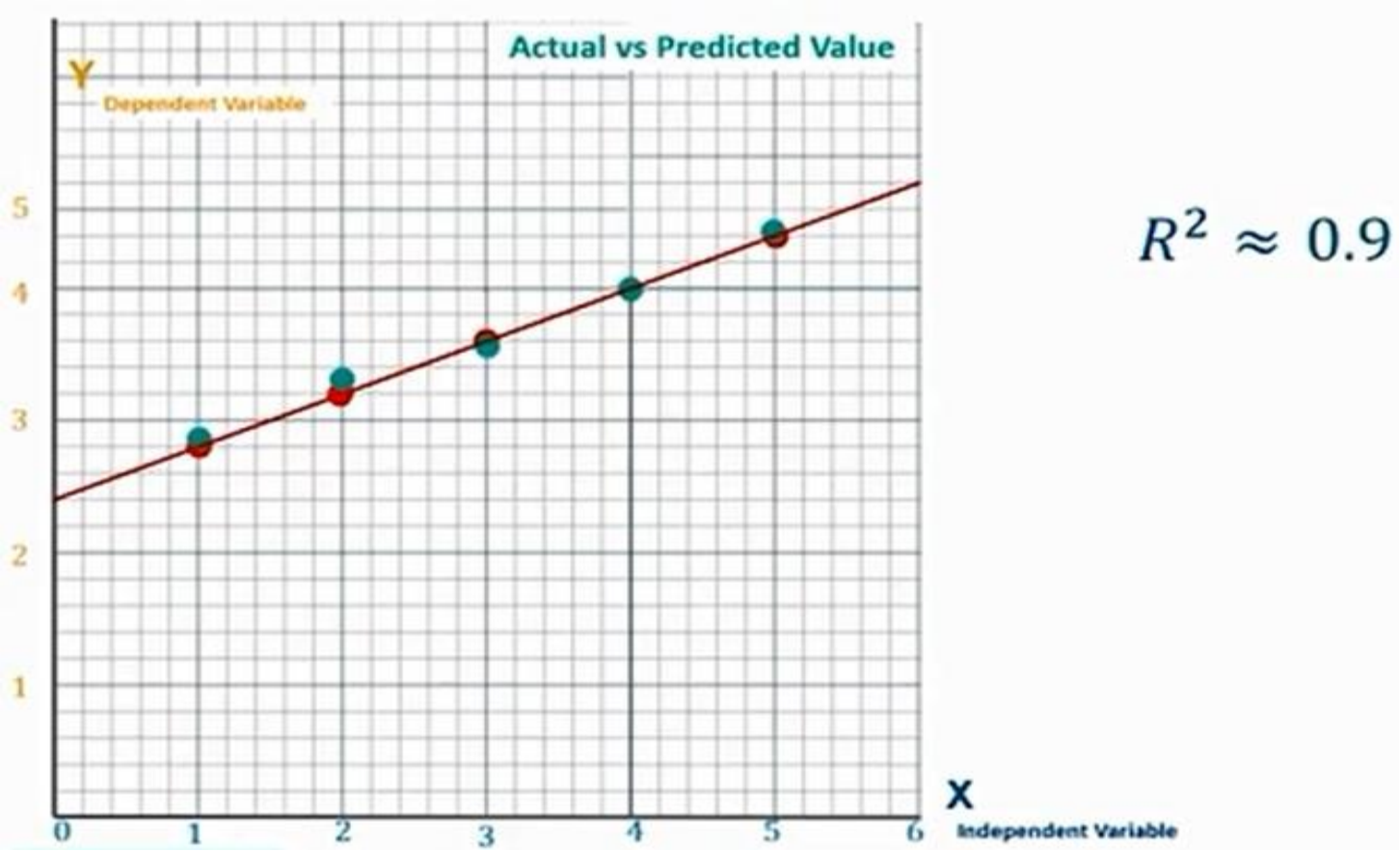
R-Squared method



$$R^2 \approx 0.7$$

Lets check the goodness of fit

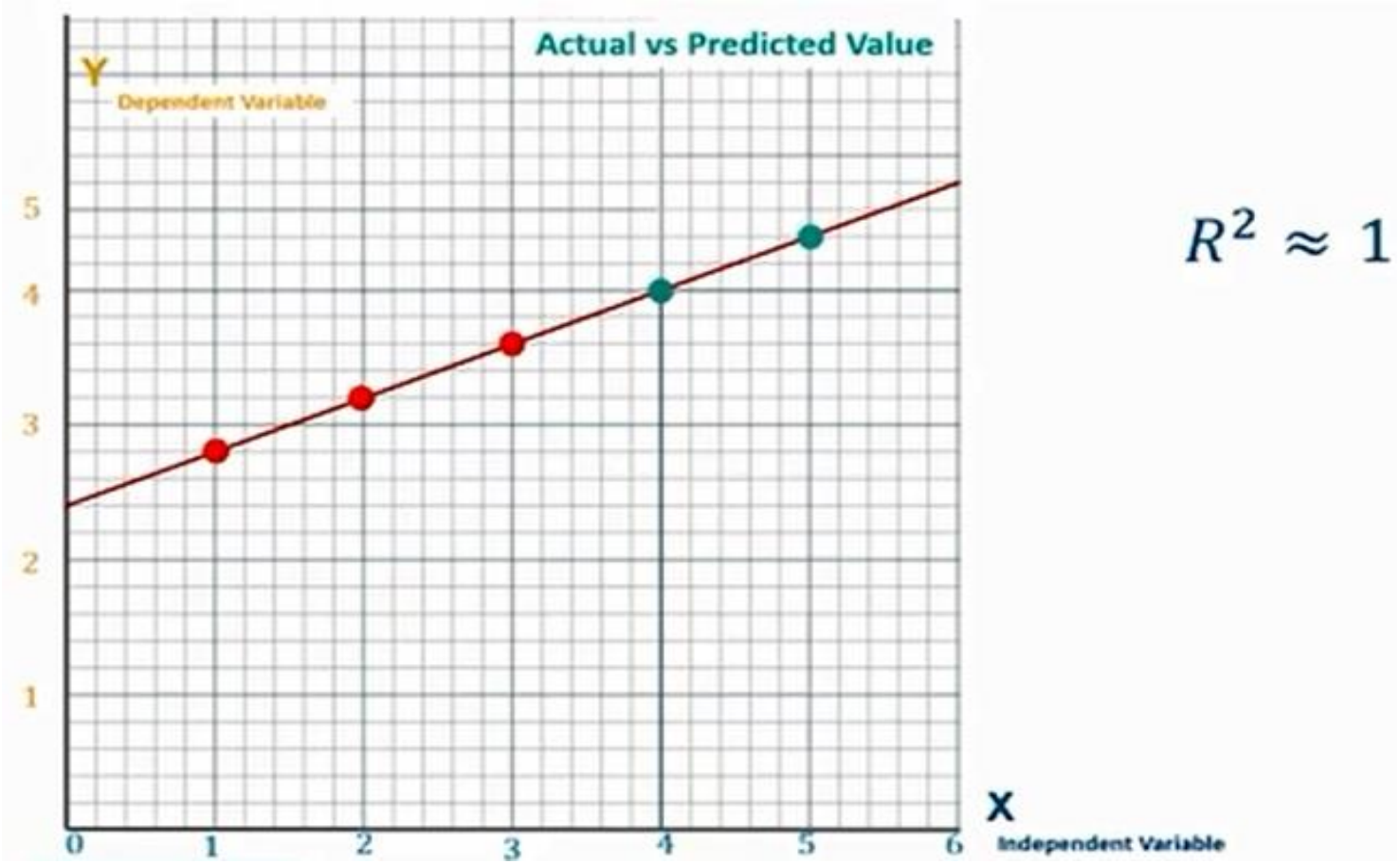
R-Squared method



$R^2 \approx 0.9$

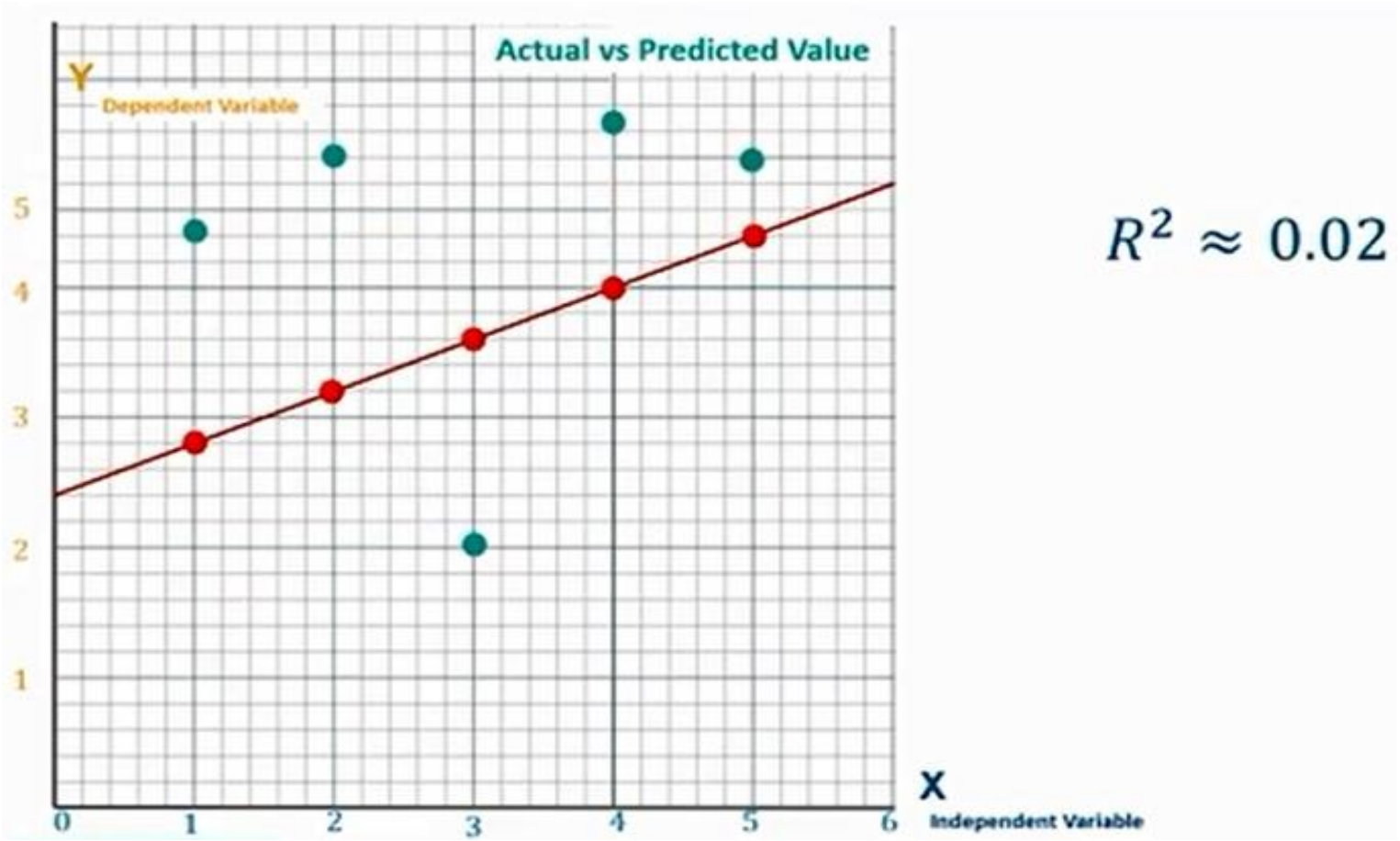
Lets check the goodness of fit

R-Squared method



Lets check the goodness of fit

R-Squared method



Lets check the goodness of fit

R-Squared method

- R-Squared is a statistical measure of fit that indicates how much variation of a dependent variable is explained by the independent variable(s) in a regression model.
- An R-squared of 100% means that all movements of a security (or other dependent variables) are completely explained by movements in the index (or the independent variable(s) you are interested in).
- A "good" R-Squared value will depend on the context.
- In some fields, such as the social sciences, even a relatively low R-Squared such as 0.5 could be considered relatively strong.
- In other fields, the standards for a good R-Squared reading can be much higher, such as 0.9 or above.

Logistic Regression-What and Why

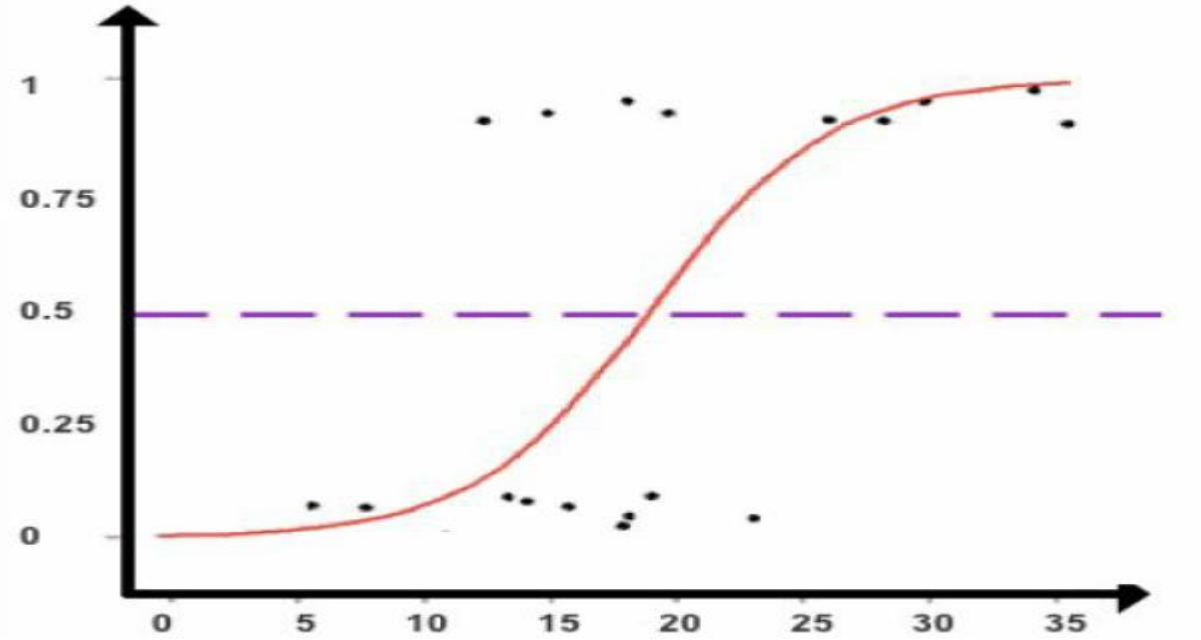
Logistic Regression produces results in a binary format which is used to predict the outcome of a categorical dependent variable.

So the outcome should be discrete/categorical such as.

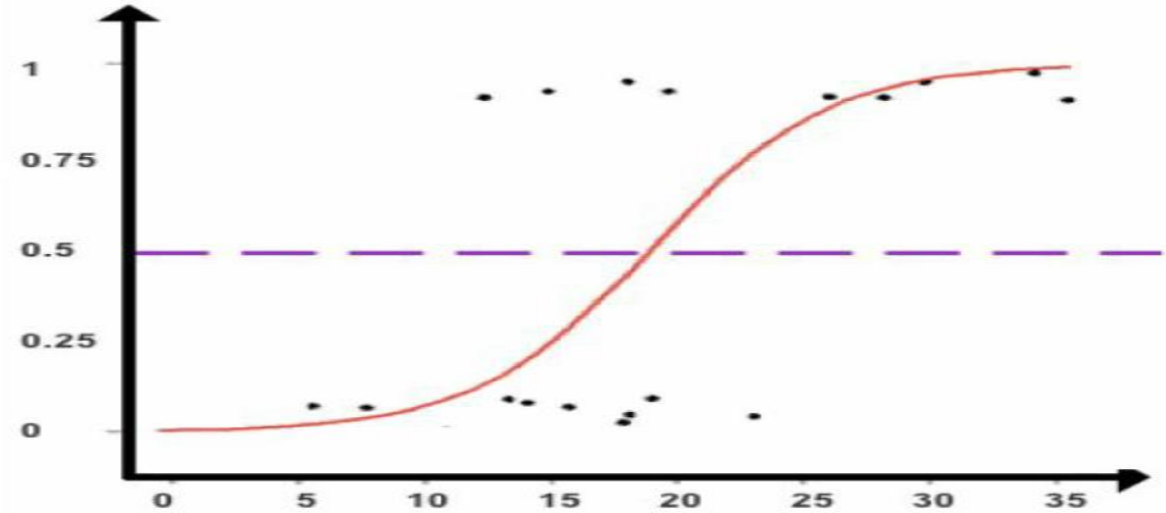
Logistic Regression-What and Why

- It uses ***Sigmoid Function***.
- The sigmoid function produces an S-shaped curve that can convert any number and map it into a numerical value between 0 and 1.
- Logistic regression adopts the sigmoid function to analyze data and predict discrete classes that exist in a dataset.
- Logistic regression is typically used for binary classification to predict two discrete classes.

$$y = \frac{1}{1+e^{-x}}$$



Logistic Regression-What and Why



$$y = \frac{1}{1+e^{-x}}$$

- The sigmoid function, also called logistic function gives an 'S' shaped curve that can take any real-valued number and map it into a value between 0 and 1.
- If the curve goes to positive infinity, y predicted will become 1.
- If the curve goes to negative infinity, y predicted will become 0.
- If the output of the sigmoid function is more than 0.5, we can classify the outcome as 1 or YES.
- If it is less than 0.5, we can classify it as 0 or NO.

Different Types of Logistic Regression

Three different types of Logistic Regression are as follows:

1. Binary Logistic Regression: In this, the target variable has only two 2 possible outcomes. For Example, 0 and 1, or pass and fail or true and false.

2. Multinomial Logistic Regression: In this, the target variable can have three or more possible values without any order. For Example, Predicting preference of food i.e. Veg, Non-Veg, Vegan.

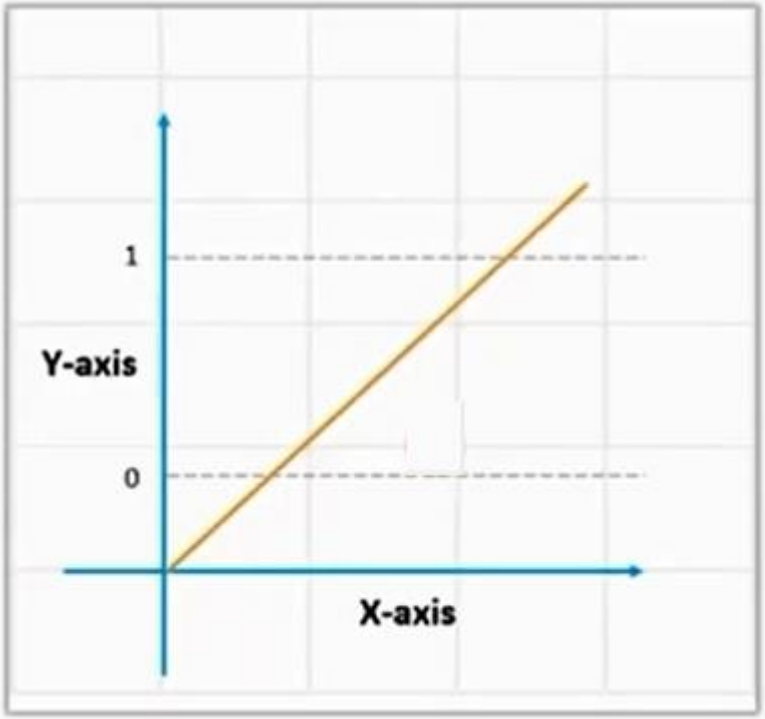
3. Ordinal Logistic Regression: In this, the target variable can have three or more values with ordering. For Example, Movie rating from 1 to 5.

Impact of Outliers on Logistic Regression

The estimates of the Logistic Regression are sensitive to unusual observations such as outliers, high leverage, and influential observations.

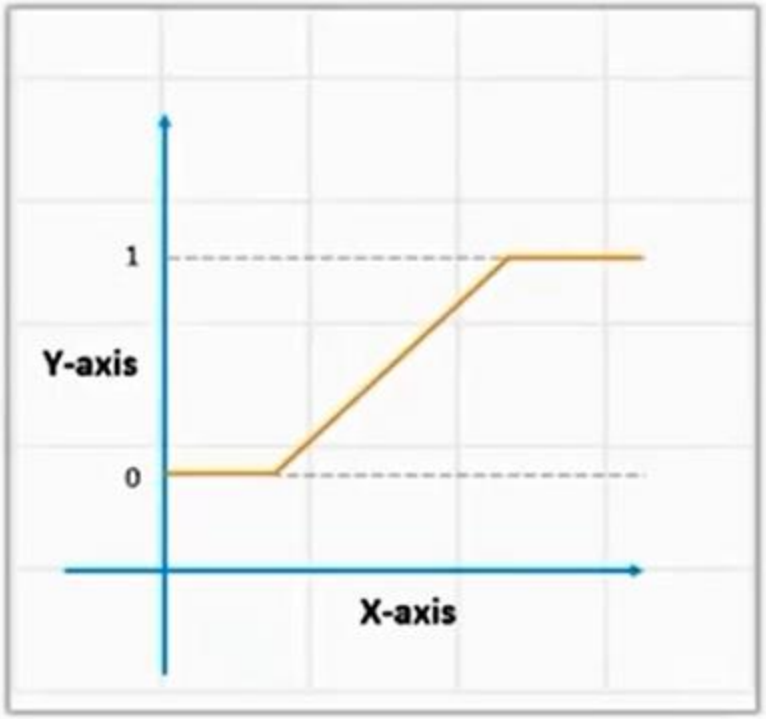
Therefore, **to solve the problem of outliers**, **a sigmoid function** is used in Logistic Regression.

Why not Logistic Regression



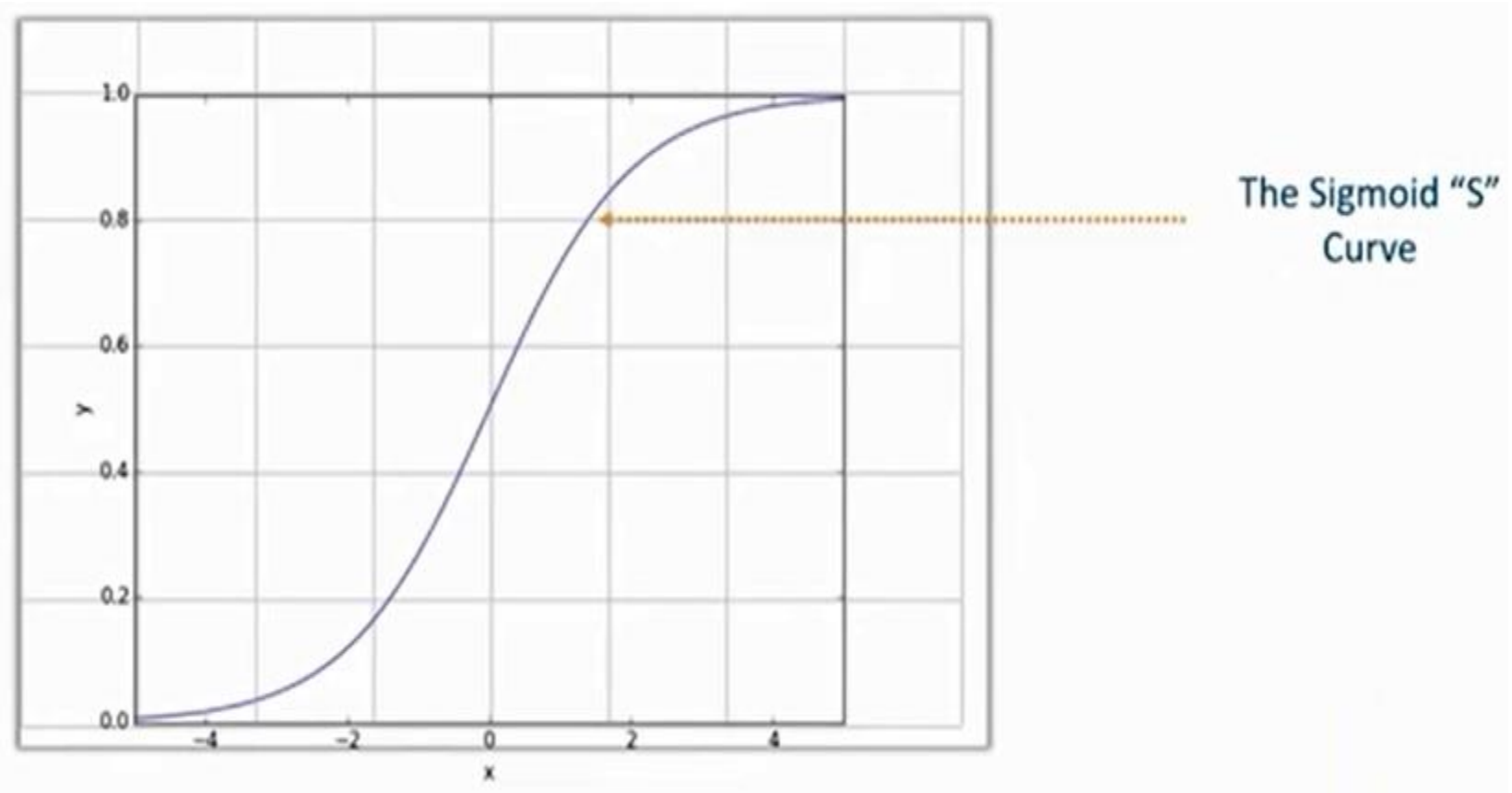
Since our value of Y will be between 0 and 1, the linear line has to be clipped at 0 and 1.

Why not Logistic Regression

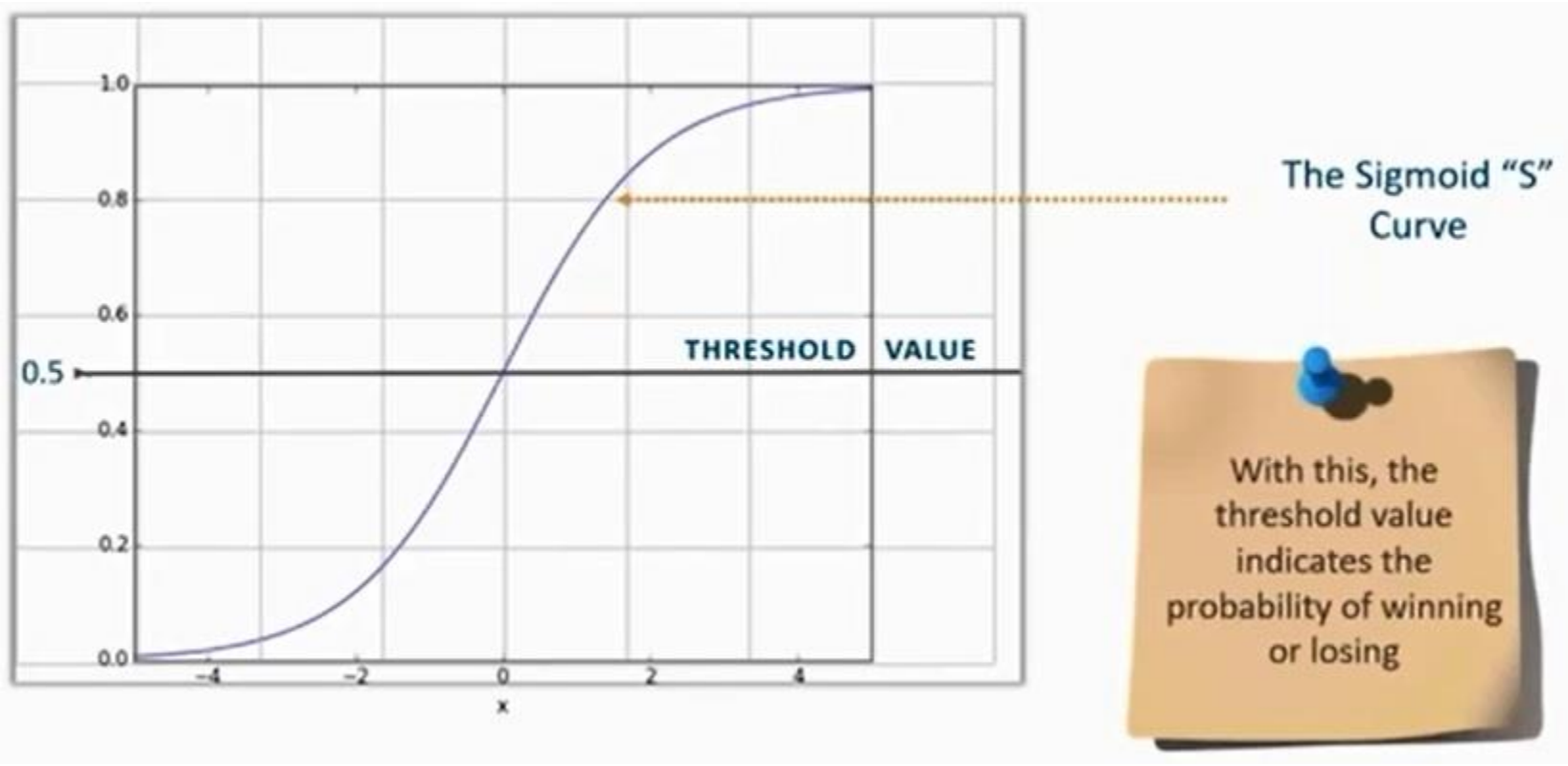


With this, our resulting curve cannot be formulated into a single formula. Hence we came up with **Logistic**!

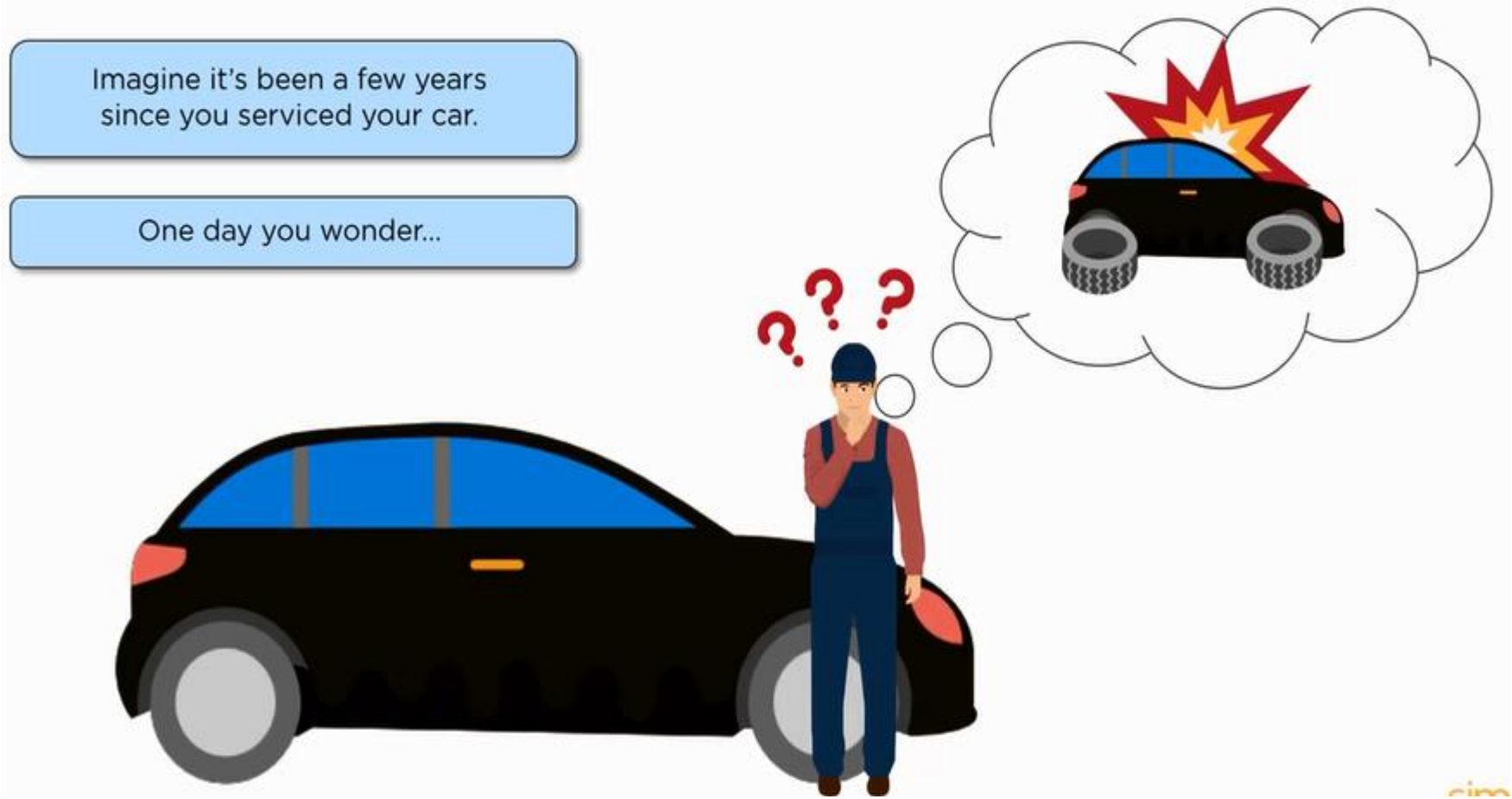
Logistic Regression Curve



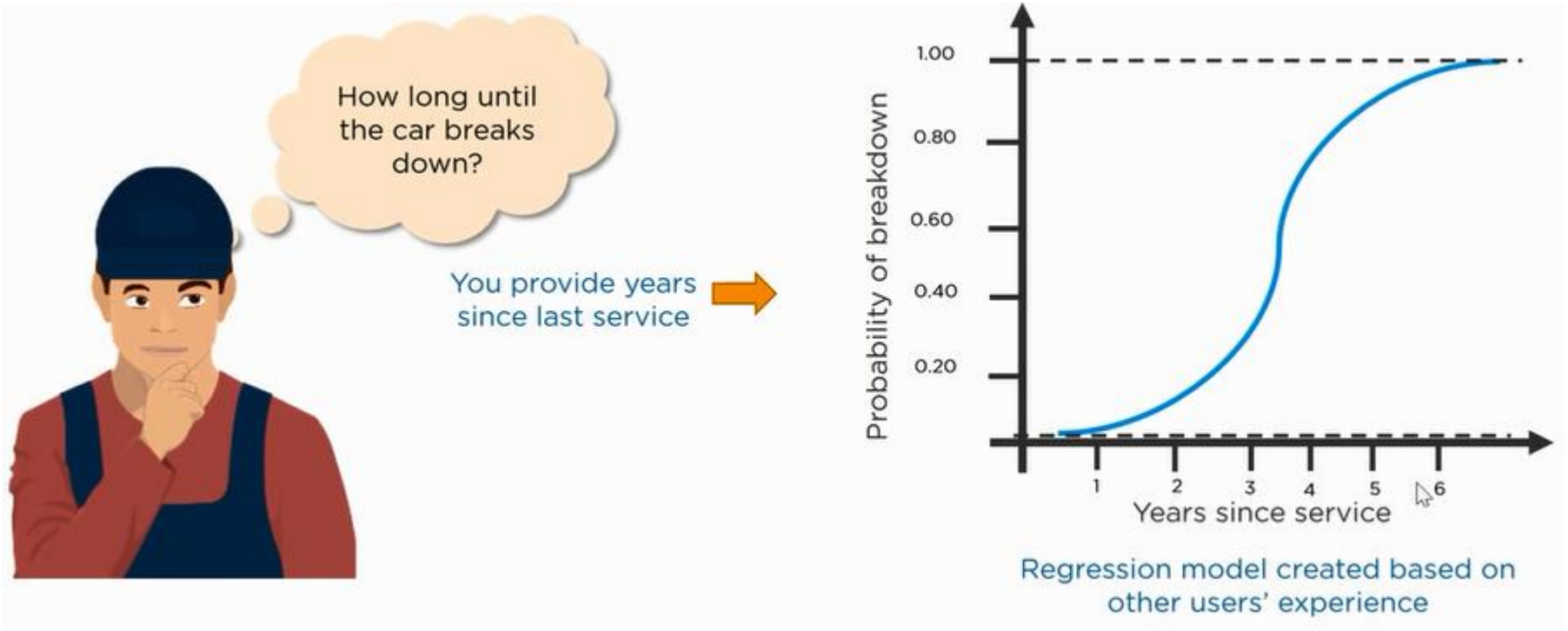
Logistic Regression Curve



Logistic Regression Example

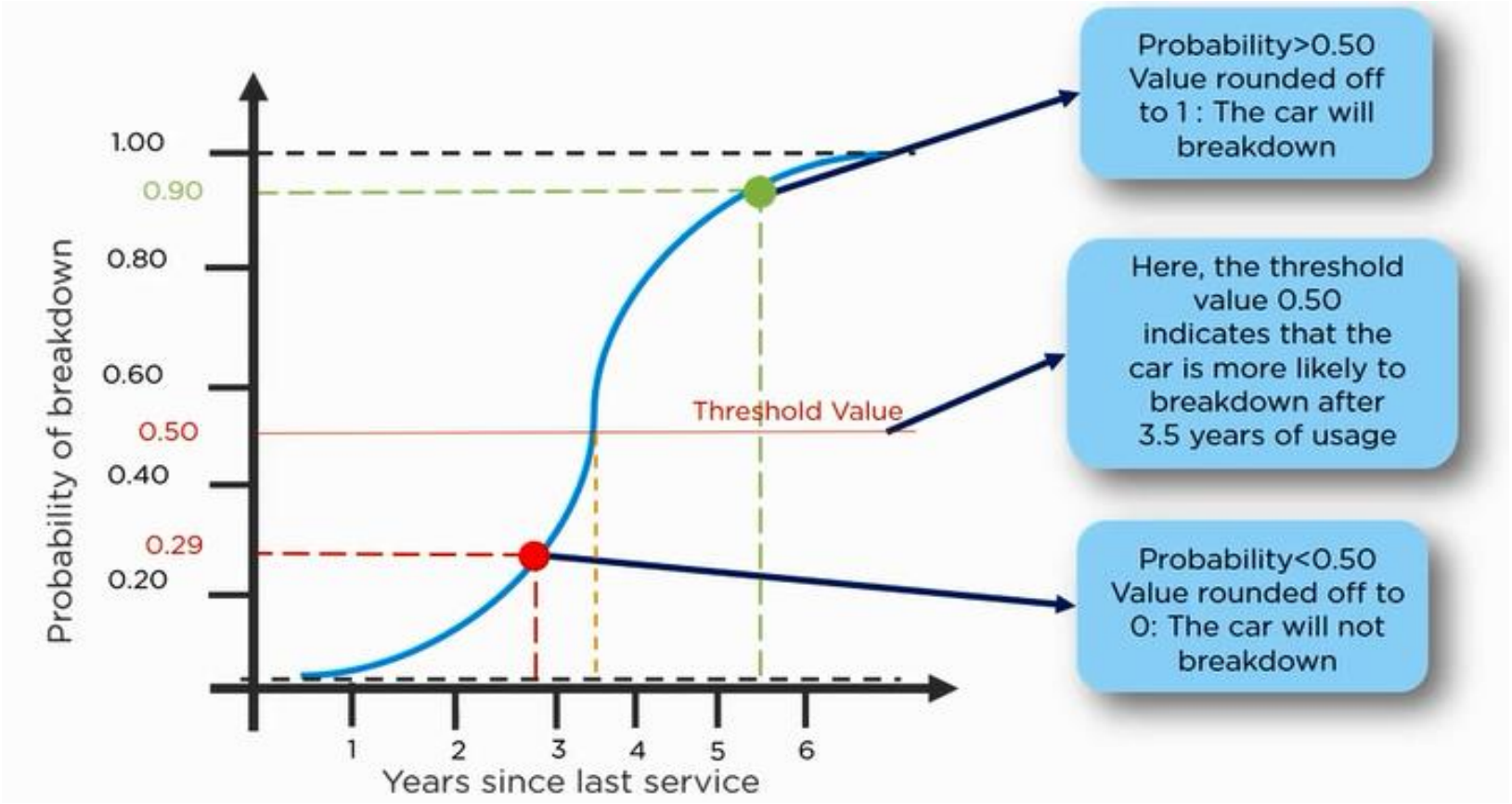


Logistic Regression Example



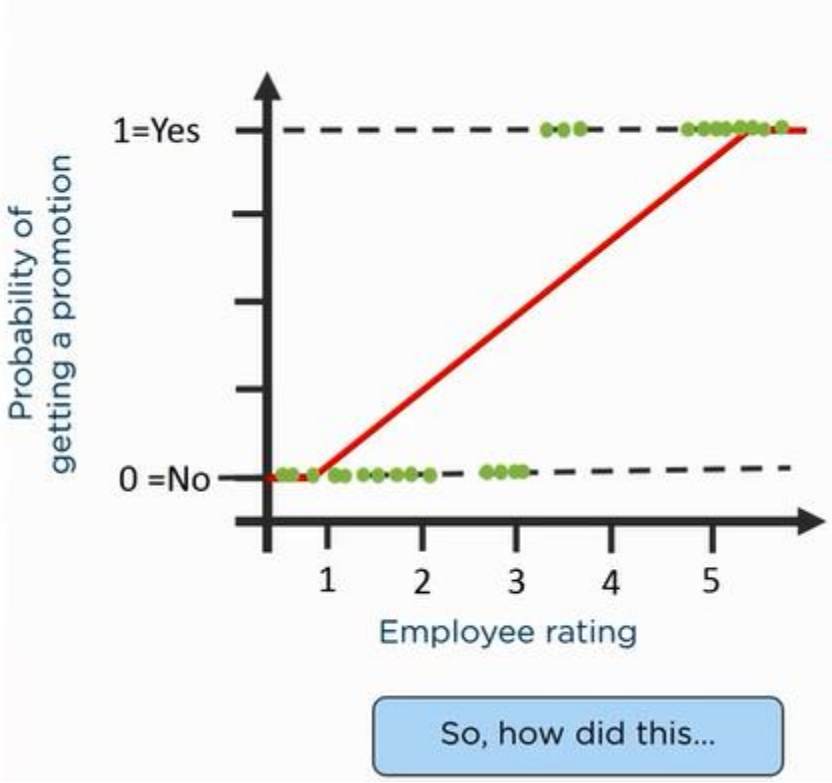
It's a Classification Algorithm , used to predict binary outcomes for a given set of Independent Variables. The dependent variables outcome is discrete.

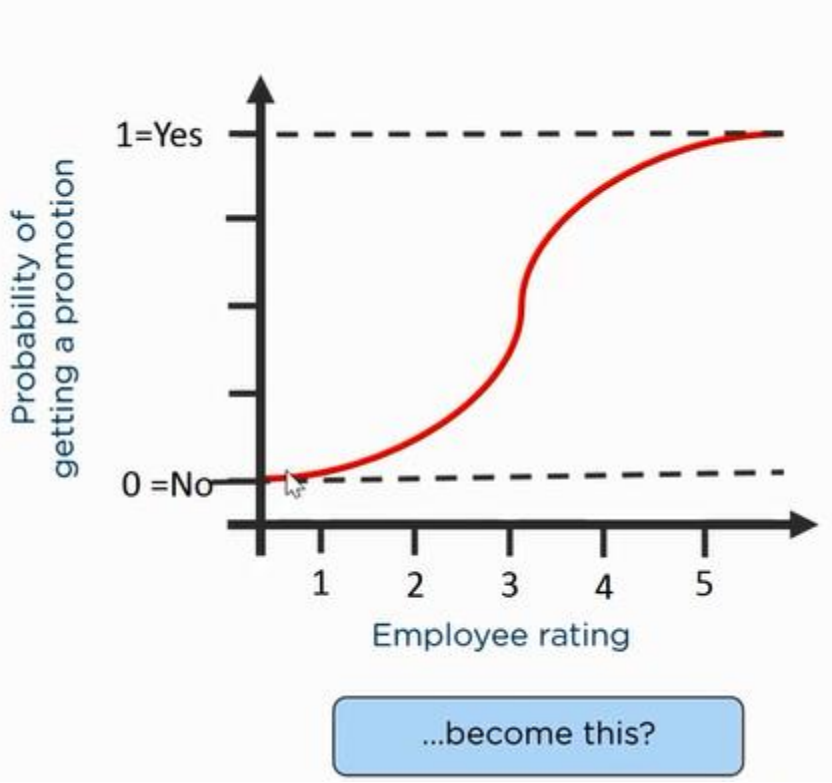
Logistic Regression Example



Model Makes Prediction

The Math behind Logistic Regression





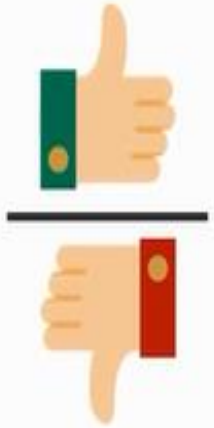
The Math behind Logistic Regression

What are the odds?

Odds are defined as the ratio of the probability of an event occurring to the probability of the event not occurring.

To understand Logistic Regression, let's talk about the odds of success

$$\text{Odds } (\theta) = \frac{\text{Probability of an event happening}}{\text{Probability of an event not happening}} \quad or \quad \theta = \frac{p}{1-p}$$



The values of odds range from 0 to ∞
The values of probability change from 0 to 1

Logistic Regression-Odds

Let's begin with probability. Probabilities range between 0 and 1. Let's say that the probability of success is .8, thus **p = .8**

Then the probability of failure is **q = 1 – p = .2**

Odds are determined from probabilities and range between **0 and infinity. Odds are defined as the ratio of the probability of success and the probability of failure.** The **odds of success are**

odds(success) = p/(1-p) or p/q = .8/.2 = 4,

that is, the odds of success are 4 to 1. The odds of failure would be

odds(failure) = q/p = .2/.8 = .25.

This looks a little strange but it is really saying that the odds of failure are 1 to 4. The **odds of success** and the **odds of failure** are just **reciprocals of one another**, i.e., 1/4 = .25 and 1/.25 = 4.

Logistic Regression-Odds

For the customer service data, the proportion of customers who would recommend the service in the sample of customers is p =0.84. **Find the odds of recommending the service department.**

Soln-The proportion of customers who would not recommend the service department =1- p =1- 0.84=0.16

Therefore, the odds of recommending the service department are

odds = p /(1-p) =0.86/0.16 =5.25

Logistic Regression-an odds ratio

This example was given by Pedhazur (1997). Suppose that seven out of 10 males are admitted to an engineering school while three of 10 females are admitted. **Calculate Odds ratio for Admission.**

The probabilities for admitting a male are,

p = 7/10 = .7 q = 1 – .7 = .3

If you are male, the probability of being admitted is 0.7 and the probability of not being admitted is 0.3. Here are the same probabilities for females,

p = 3/10 = .3 q = 1 – .3 = .7

If you are female it is just the opposite, the probability of being admitted is 0.3 and the probability of not being admitted is 0.7. Now we can use the probabilities to compute the odds of admission for both males and females,

odds(male) = .7/.3 = 2.33333 odds(female) = .3/.7 = .42857

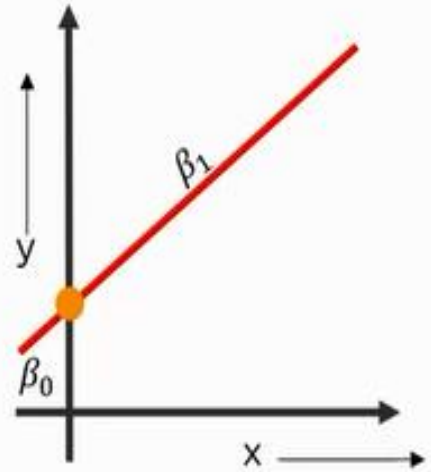
Next, we compute the odds ratio for admission,

Odds Ratio = 2.3333/.42857 = 5.44

Thus, **for a male, the odds of being admitted are 5.44 times as large as the odds for a female being admitted.**

The Math behind Logistic Regression

Take the equation of the straight line



Here, β_0 is the y-intercept
β_1 is the slope of the line
x is the value of the x co-ordinate
y is the value of the prediction

The equation would be: $y = \beta_0 + \beta_1 x$

The Math behind Logistic Regression

The **Logistic Regression Equation** is derived from the Straight Line Equation

Equation of a Straight Line

$P=B_0+B_1X+\ldots\ldots\ldots$ Range is from Infinity to Infinity

Let's try to derive the Logistic Regression Equation from the Straight Line Equation

$P=B_0+B_1X+\ldots\ldots\ldots$ In Logistic Equation Y can be only from 0 to 1

Now, to get the range of Y between 0 and Infinity, Lets transform P

(P/1-P)

P=0 then 0

P=1 then infinity Now, the range is between 0 to infinity

Let us transform it further, to get range between (Infinity and Infinity)

Log(P/1-P) $\longrightarrow$ **$P=B_0+B_1X+\ldots\ldots\ldots$**Final Logistic Regression Equation

The Math behind Logistic Regression

Now, we predict the odds of success

$$\log\left(\frac{p(x)}{1-P(x)}\right) = \beta_0 + \beta_1 x$$

Exponentiating both sides:

$$e^{ln}\left(\frac{p(x)}{1-p(x)}\right) = e^{\beta_0+\beta_1 x}$$

$$\left(\frac{p(x)}{1-p(x)}\right) = e^{\beta_0+\beta_1 x}$$

Let $Y = e^{\beta_0+\beta_1 x}$

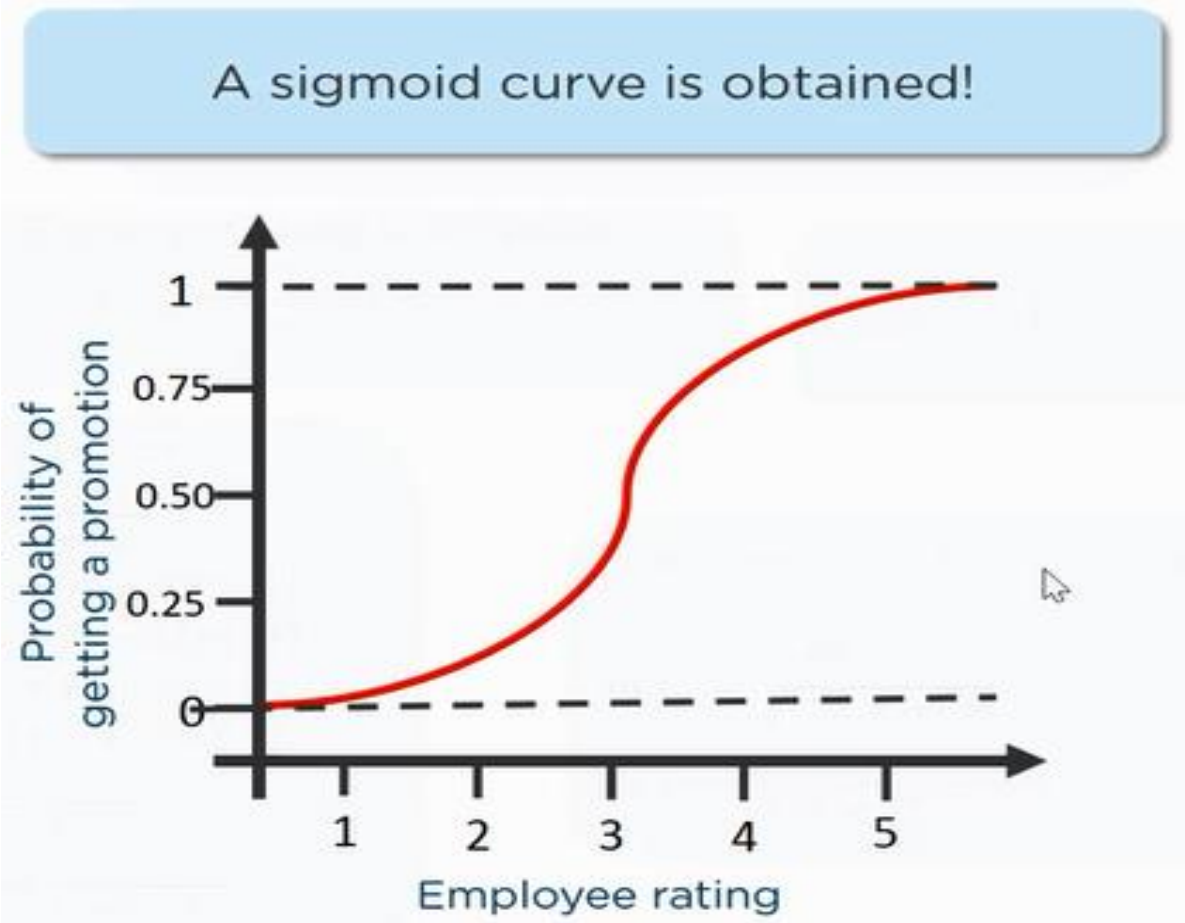
Then $\frac{p(x)}{1-p(x)} = Y$

$$p(x) = Y(1-p(x))$$
$$p(x) = Y - Y(p(x))$$
$$p(x) + Y(p(x)) = Y$$
$$p(x)(1+Y) = Y$$
$$p(x) = \frac{Y}{1+Y}$$
$$p(x) = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}}$$

The equation of a sigmoid function:

$$p(x) = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}}$$
$$p(x) = \frac{1}{1+e^{-(\beta_0+\beta_1 x)}}$$

The Math behind Logistic Regression



Logistic Regression Use cases

Weather Prediction

Helps determine the kind of Weather that can be expected

Logistic Regression Use cases

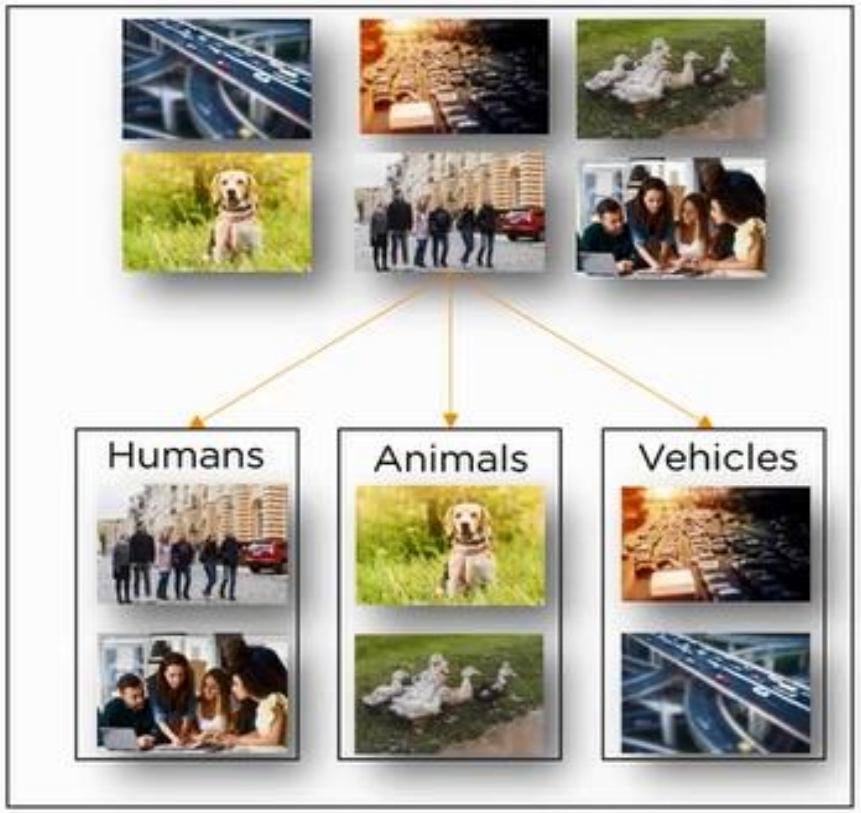


Image Categorization

Identifies the different components
That are present in the image, and
Helps categorize them

Logistic Regression Use Cases

Linear Regression Vs Logistic Regression

Basic	Linear Regression	Logistic Regression
Core Concept	The data is modelled using Straight line	The probability of some obtained event is represented as a linear function of a combination of Predictor variables –S-Curve
Used with	Continuous Variable	Categorical Variables
Output/Prediction	Value of the variable	Probability of occurrence of event
	Solves Regression Problems	Solves Classification Problems
Accuracy and Goodness of FIT	Measured by Loss, R squared, Adjusted R squared etc.	Accuracy, Precision, Recall, F1 Score, ROC curve, Confusion Matrix etc.
Estimated using	Least Squares (OLS) method	Maximum Likelihood Estimation (MLE) method

Advantages of Logistic Regression?

1. Logistic Regression is very easy to understand.
2. It requires less training.
3. It performs well for simple datasets as well as when the data set is linearly separable.
4. It doesn’t make any assumptions about the distributions of classes in feature space.
5. A Logistic Regression model is less likely to be overfitted but it can ovefit in high dimensional datasets. To avoid overfitting these scenarios, One may consider regularization.
6. They are easier to implement, interpret, and very efficient to train.

Disadvantages of Logistic Regression?

1. Sometimes a lot of **Feature Engineering** is required.

2. If the independent features are correlated with each other it may affect the performance of the classifier.

3. It is quite sensitive to **noise** and **overfitting**.

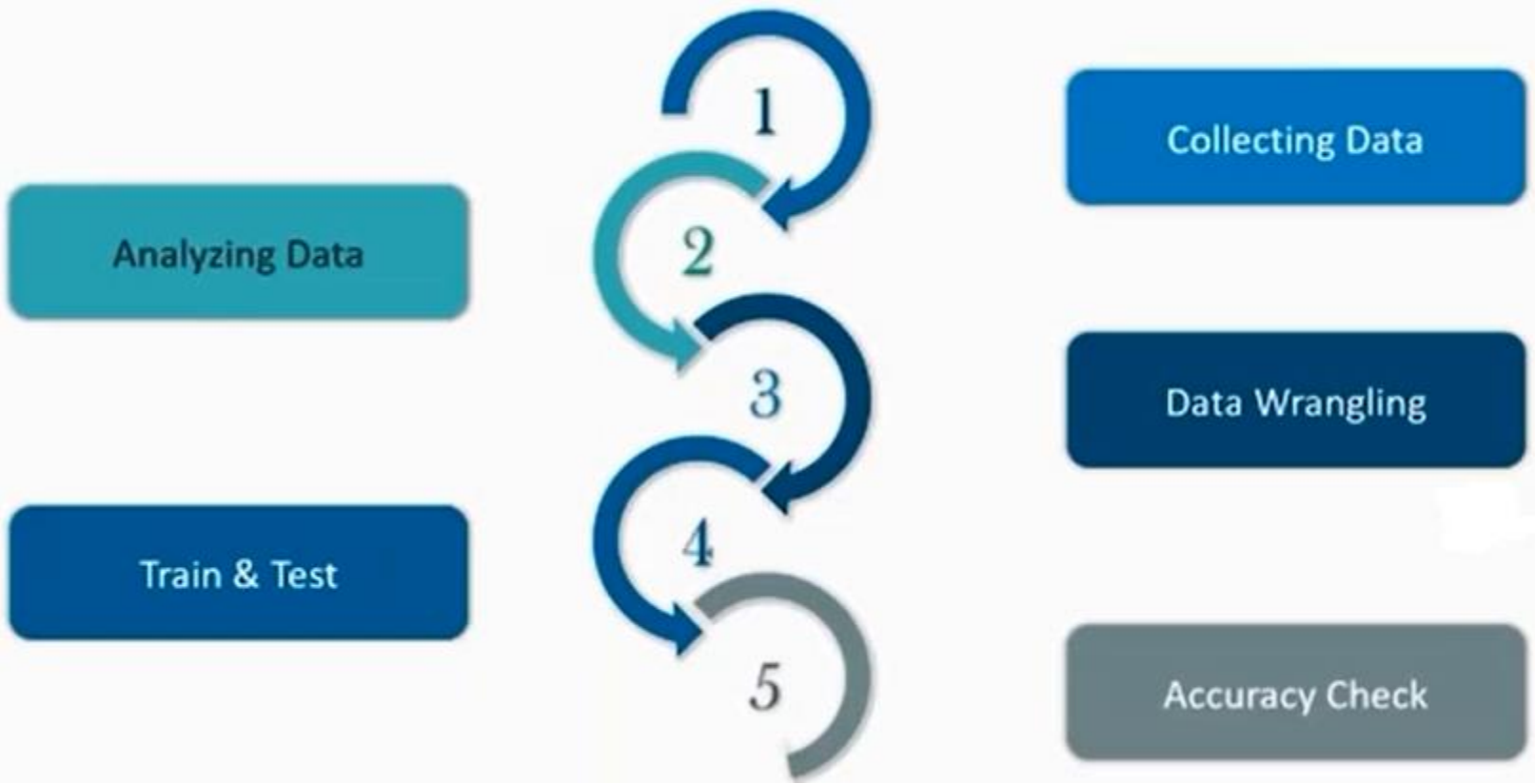
4. Logistic Regression should not be used if the number of observations is lesser than the number of features, otherwise, it may lead to overfitting.

5. By using Logistic Regression, non-linear problems can't be solved because it has a linear decision surface. But in real-world scenarios, the linearly separable data is rarely found.

6. By using Logistic Regression, it is tough to obtain complex relationships. Some algorithms such as neural networks, which are more powerful, and compact can easily outperform Logistic Regression algorithms.

7. In Linear Regression, there is a linear relationship between independent and dependent variables but in Logistic Regression, independent variables are linearly related to the log odds (log(p/(1-p)).

Implement Logistic Regression



What is Classification

Classification is the process of dividing the datasets into different categories or group by adding labels.

Note-It adds the data point to a particular labelled group on the basis of some condition.

Introduction to Classification

Classification is a **Supervised Learning Method**.

How will you separate an apple from a sweet lemon?

To classify fruits, first an idea of the fruits involved-apple and sweet lemon.

Required parameters such as colour, size, and shape.

Thus a dataset needs to be constructed to train a classifier on how an apple looks.

These attributes are called input features, attributes, or independent variables.

The input also includes labels of known instances.

So a classifier takes a set of features as input and learns what features characterize an apple or a sweet lemon.

Then it takes an unknown instance (which needs to be classified) and assigns the label 'apple" or "sweet lemon'.

This label is a dependent variable and this process is called **Classification, Prediction, or Recognition.**

Introduction to Classification

Classification process involves the following two phases

1. Training Phase **2. Testing Phase**

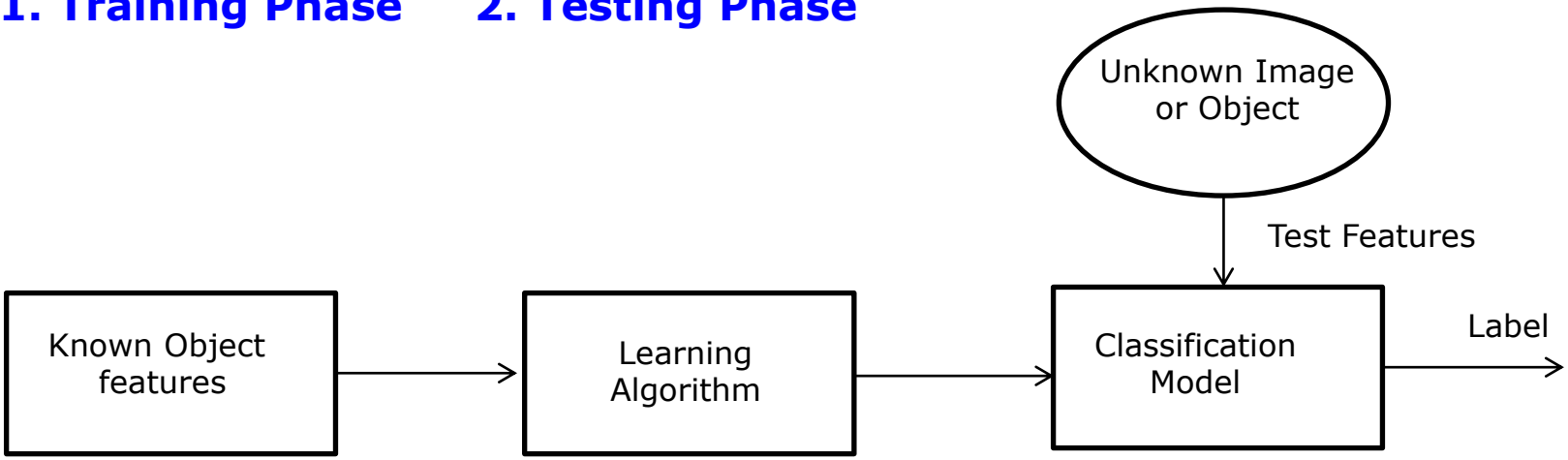


Fig. Sample Classification Scheme

1. **Training Phase-**The classifier algorithm is fed with a large set of known data. This dataset is called training data or labelled data.
2. **Testing Phase-**The Constructed model is tested and evaluated with unknown test data.

The Classification task is both **Descriptive as well as Predictive,** that is if the model can explain its classification decisions, it is called Descriptive.
Decision Tree based Classifiers are **descriptive in nature**, where as **Neural Network-based Classification** are **Predictive based**.

Factor affecting Classifier Performance

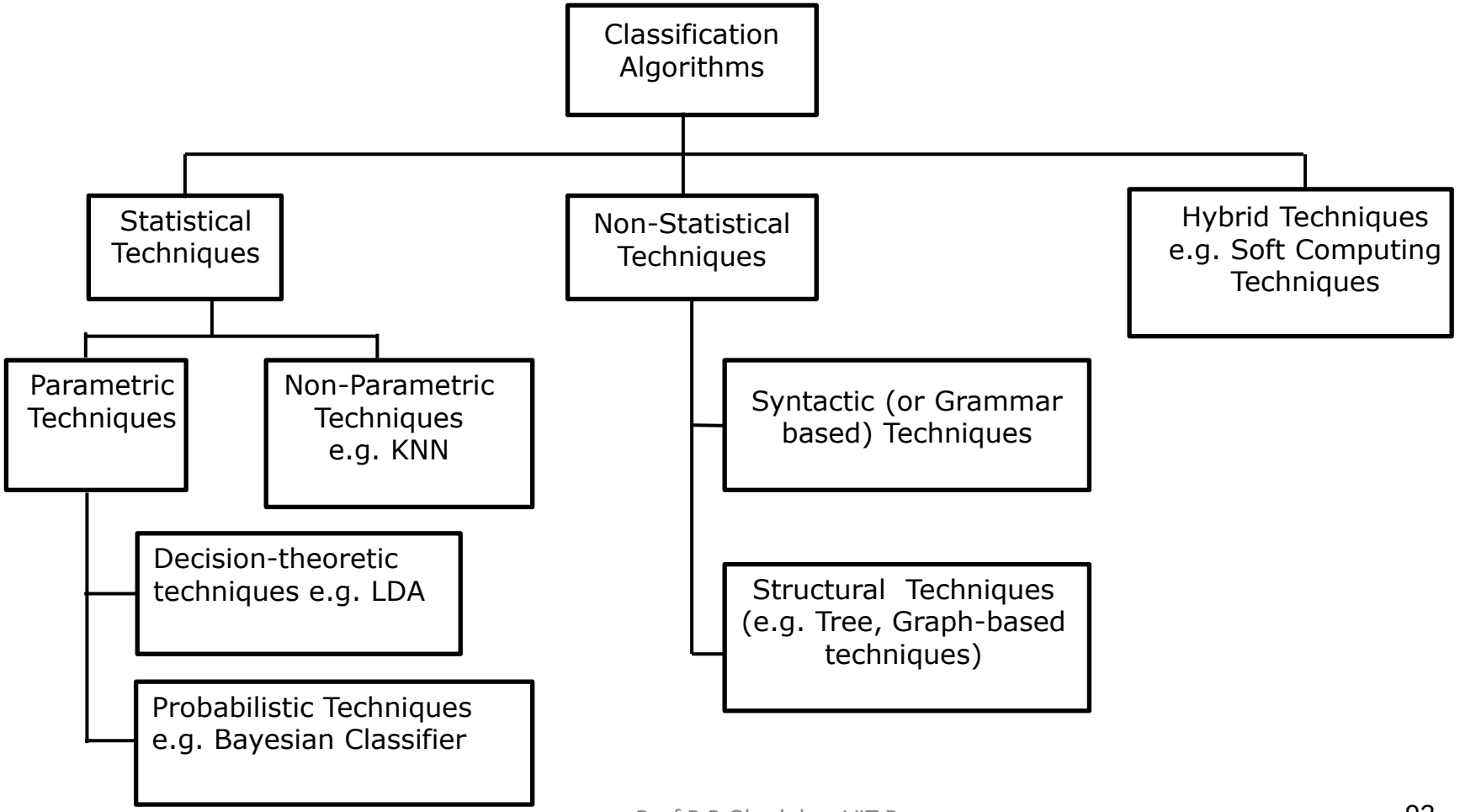
1. **Nature of Data-**Availability of Good Quality training data.
2. **Nature of Learning-**The learning process should not take more data than necessary (Known as Over-fitting of the model)

❖ **Classification Schemes**

1. Pixel based techniques
2. Feature based techniques.

Different types of Classification Algorithms

- There are many ways to design a classifier. Some of the popular Design Techniques are shown in figure.



Statistical Techniques

- It uses statistical principles for deriving models from the given training dataset using Statistical learning techniques. Two types
- **Parametric Techniques**
- It takes a set of training data and construct a classification model.
- It is assumed that the Probability Distribution Function or Density Function of the data is known for each class.
- The parameters are estimated from the data itself by assuming a distribution for a given data. Ex-Bayesian Classifier.
- **Non parametric Techniques**
- When nothing is known about the densities of the data, no assumption can be made. Ex-K-Nearest Neighbour (KNN classifier)

Parameter Techniques

- **Two types-Decision-theoretic techniques, Probabilistic techniques.**
- **Decision-theoretic techniques-**
- It is also called as discriminant function analysis.
- The idea is to design a decision function or a discriminant function that responds differently for each class.
- Example-Linear Discriminant Analysis (LDA) & Template Matching.
- **Probabilistic techniques**
- Probability plays an important role in prediction.
- Bayesian classifier calculates the prior probability and conditional probability and uses these values to assign a label to the unknown instance.
- Example-Bayesian Classifier.

Decision Theoretic Method

- It is also called as Linear Discriminant Analysis (LDA).
- The idea is to design a decision function or a Discriminant function that responds differently for each class(i.e. unique response for each class)
- This decision function can be integrated with the decision rule for classification.
- To classify the 2-feature object, the two features (x and y) are plotted as a point in a 2D graph called the feature space.
- For objects having multiple features the graph is multidimensional.
- It is used to design decision boundaries or discriminating functions to separate the feature vector cluster in the feature space.

Decision Theoretic Method

- The idea of LDA is to use the decision function to discriminate the input features.
- For the two features X_1, and X_2 , the decision boundary would be $d(X)=X_1-mX_2-c$.
- The key idea is to design a decision surface such that $d(X)>0$ would identify a particular feature and $d(X) < 0$ would identify the another feature.
- All the points at the decision boundary would satisfy the condition $d(X) = 0$.
- This idea can also be extended for multiple features.
- Let $x=(x_1,x_2,x_3.....x_n)^T$ represent the n-dimensional vector.
- Let the number of classifiers be k.

Decision Theoretic Method

- Designing k Decision function $d_1(x)$, $d_2(x)$,….$d_k(x)$. The instance is classified as **class i and not j if**

$$d_i(x) > d_j(x); \quad i \neq j \text{ for } i, j = 1,2,\ldots..,k$$

Then the **decision boundary** separating two classes i and j is given as follow

$$d_i(x) - d_j(x) = 0$$

A decision rule can be designed as follow:

Assign the instance to the class w_i if $d_{ij} > 0$ and assign the instance to w_j if $d_{ij} < 0$

Template Matching

- **Template Matching or Matched Filtering**
  - Template matching is one of the simplest technique in **Object Recognition.**
  - The template is superimposed on and correlated with the image.
  - At every pixel, the degree of similarity is evaluated.
  - The correlation between the template and the image replaces the centre pixel of the mask in the resultant image.
  - It is then moved to the adjacent position. This process is repeated till we cover all pixels of the image.
  - At the end of this process the Maximum value indicates the Best match.
  - The correlation is high when there is a perfect match between the template and the image.
  - Based on the highest correlation value, the degree of match can be determined.
  - The Biggest disadvantage of this scheme is that no variation in scale or orientation is permitted.

Template Matching-Example

Example-Consider the template and image array shown in figure a. b. & c. Perform template matching and show the result.

Solution-The resultant correlation array is shown in following Fig.

1	1	1
1	1	1
1	1	1

(a) Template

1	1	1	0	0
1	1	1	0	0
1	1	1	1	1
0	0	1	1	1
0	0	1	1	1

(b) Image Array

9	7	5
7	7	7
5	7	9

(c) Resultant Correlation array

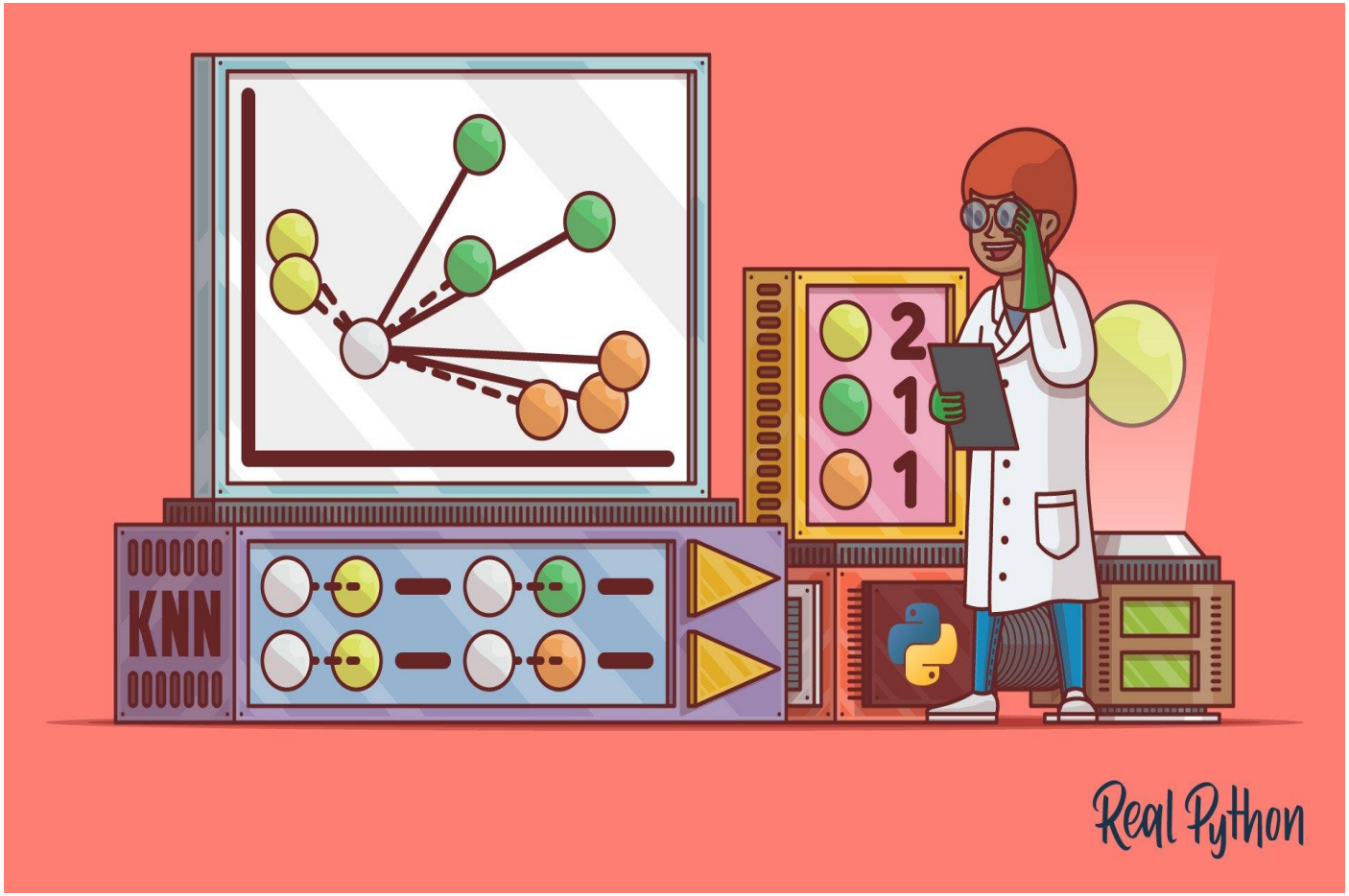
It can be observed that each element of the result is obtained by counting the number of similar values between the template and the image array.

The highest value (9 in this case) indicates the position where there is perfect match between the template and the image.

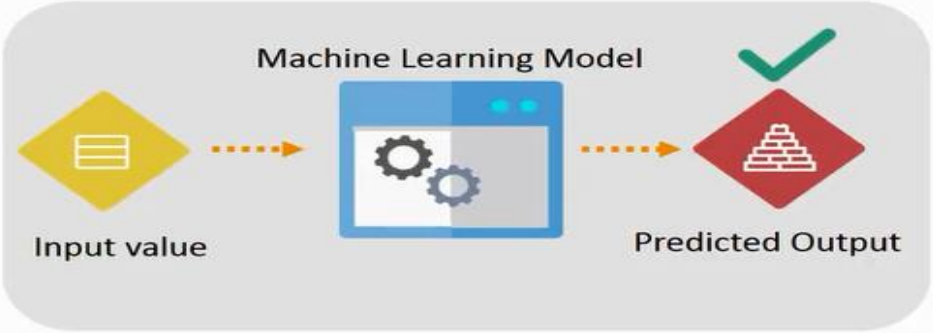
Drawback-The biggest disadvantage of template matching is that no variation in scale or orientation is permitted.

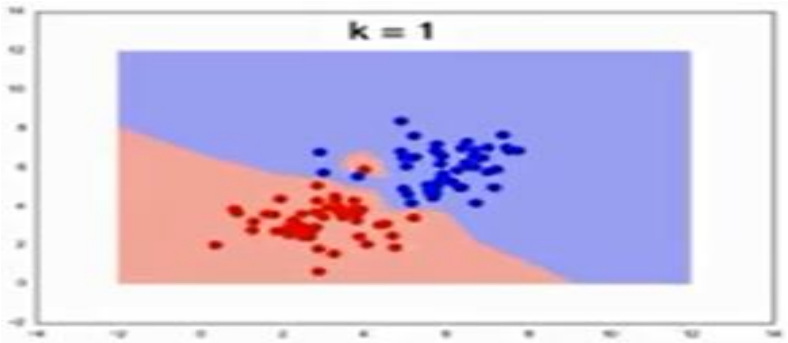
K Nearest Neighbor Classifier

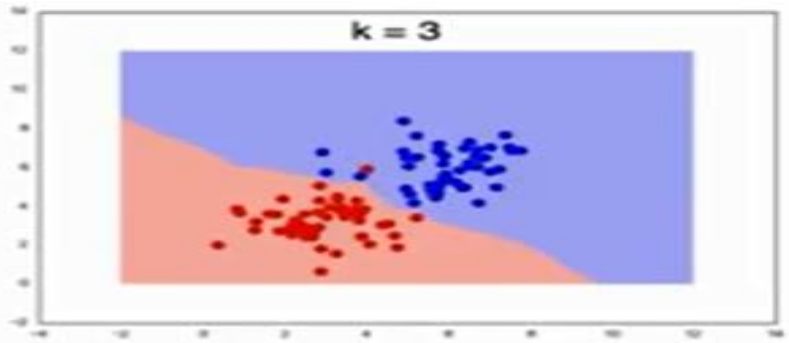
K Nearest Neighbour is a simple Algorithm that stores all the available cases and classifies the new data or case based on a similarity measure.

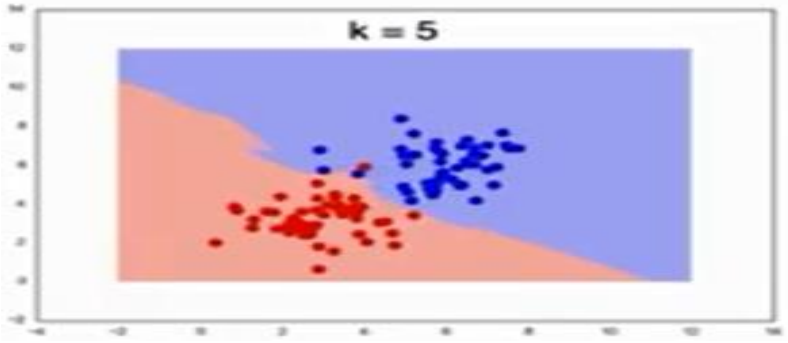


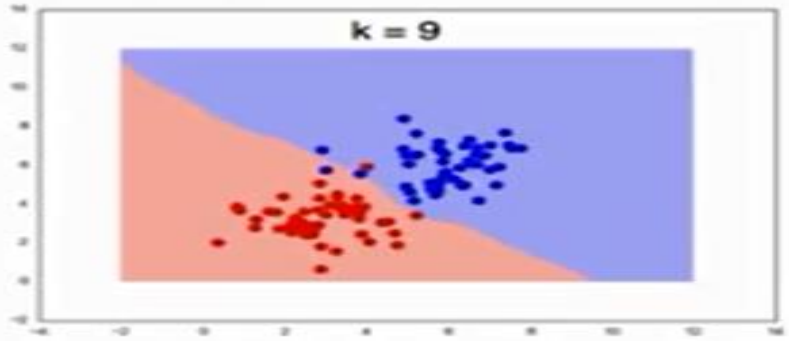
Why K Nearest Neighbor Classifier











K Nearest Neighbor Classifier Example

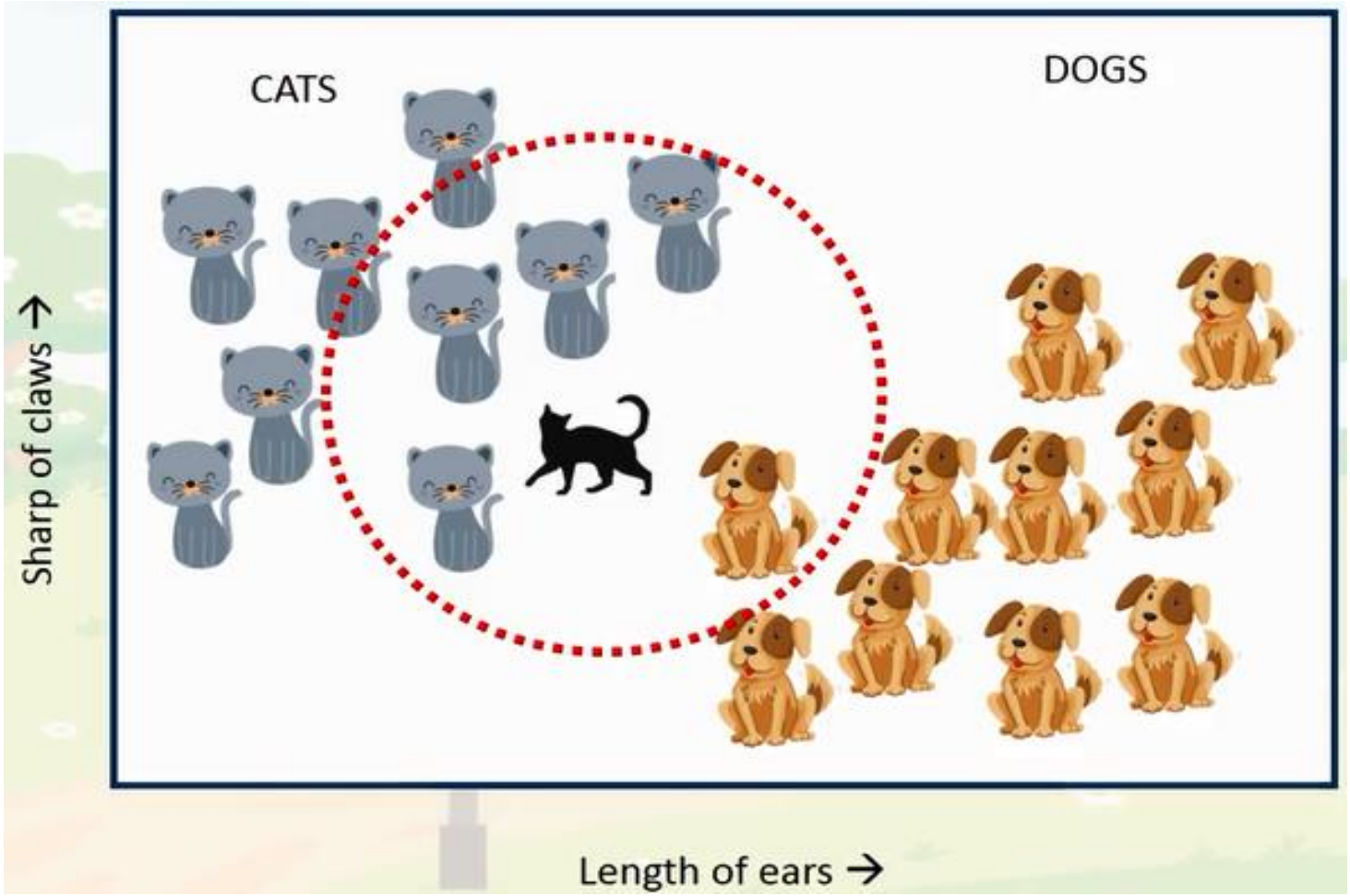
K Nearest Neighbor Classifier Example

K Nearest Neighbor Classifier Example

CATS	DOGS
	
Sharp Claws, uses to climb	Dull Claws
Smaller length of ears	Bigger length of ears
Meows and purrs	Barks
Doesn't love to play around	Loves to run around

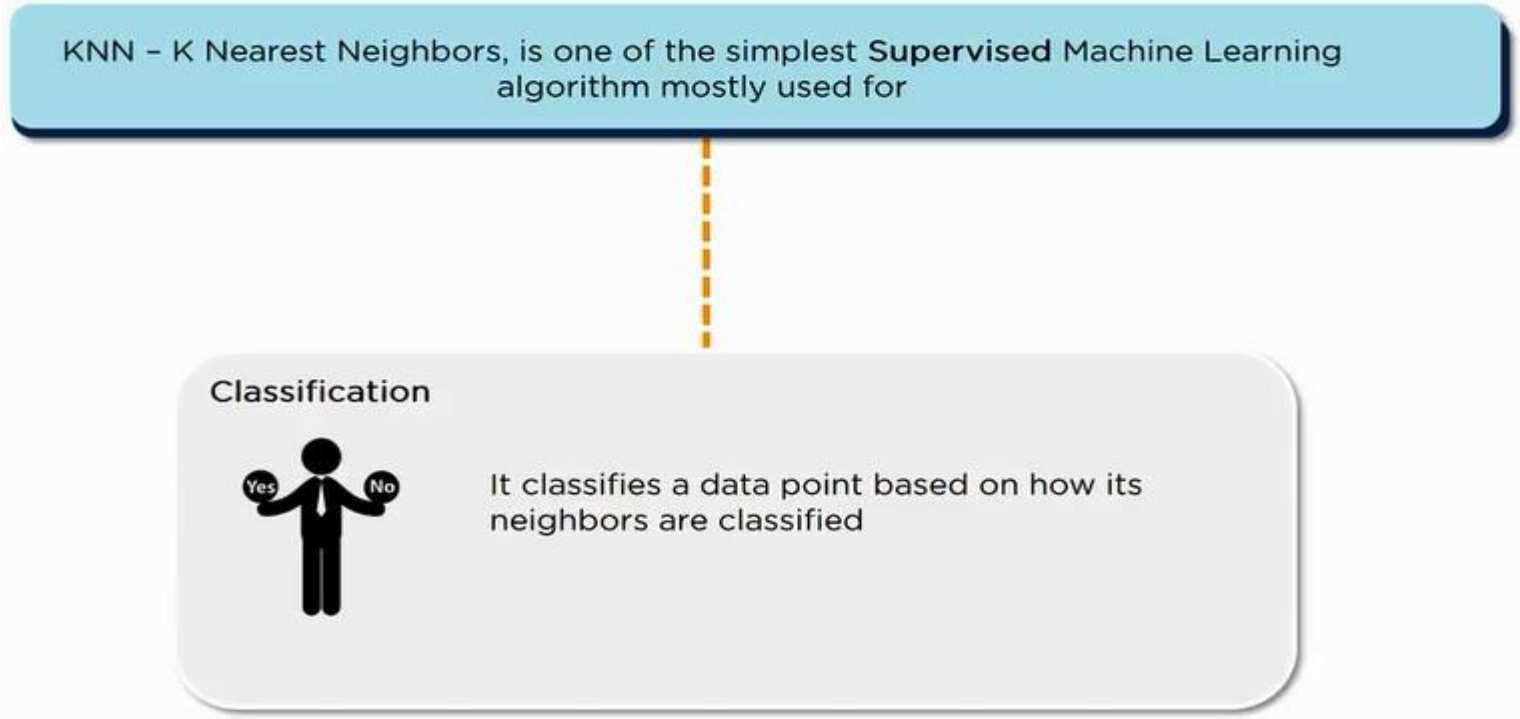
K Nearest Neighbor Classifier Example

K Nearest Neighbor Classifier Example



K Nearest Neighbor Classifier Example

What is KNN Algorithm



KNN Store all available cases and Classifies new cases based a on similarity measure

K Nearest Neighbor Classifier

K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm. It is used for both classification as well as regression predictive problems.

The following two properties would define KNN well:

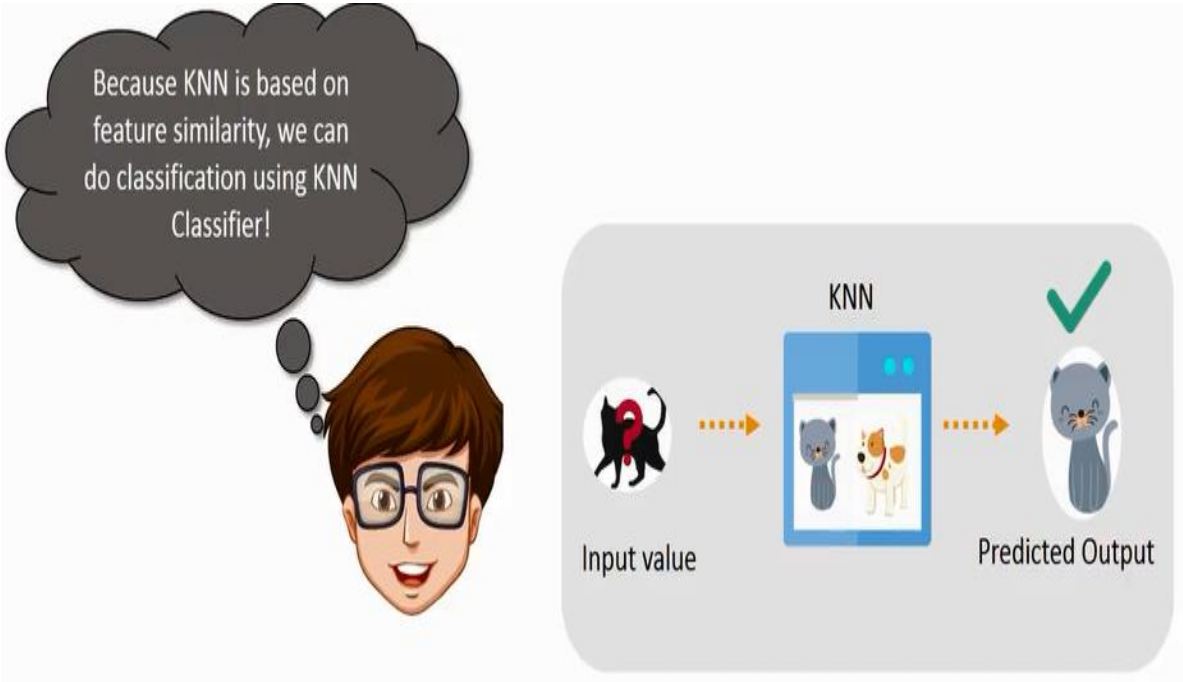
Lazy learning algorithm: KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification.

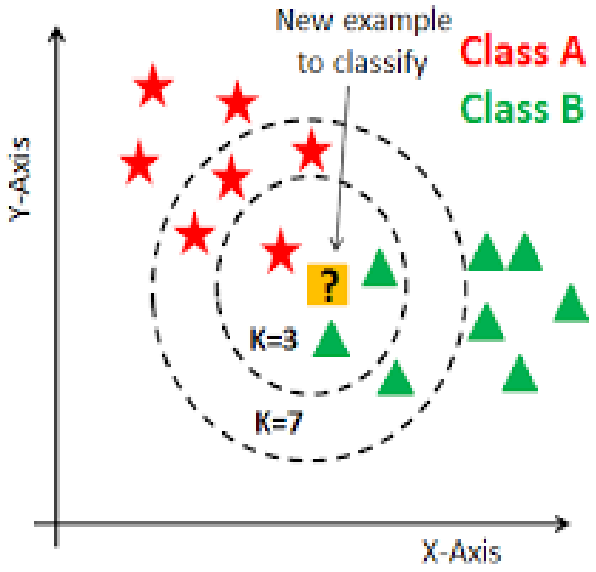
A lazy learning algorithm is simply an algorithm where the algorithm generalizes the data after a query is made.

Non-parametric learning algorithm: KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data.

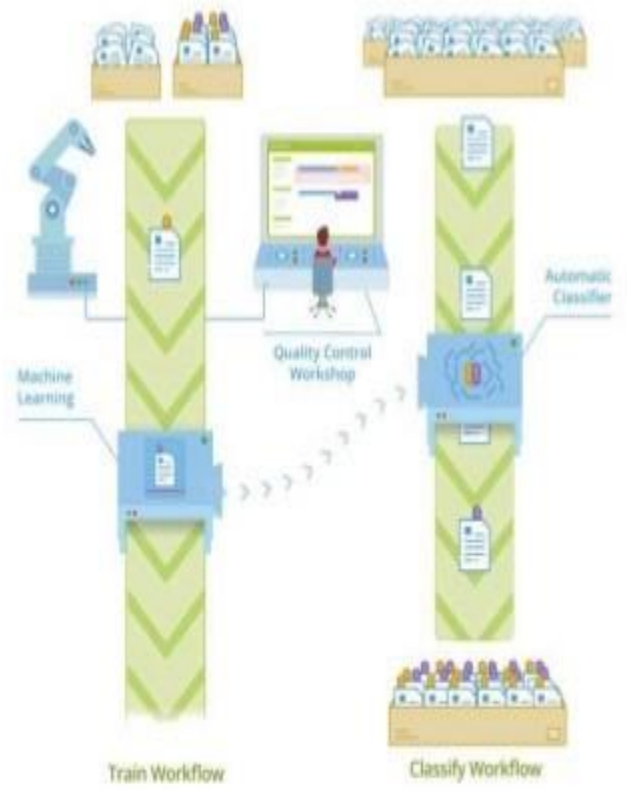
K Nearest Neighbor Classifier

The "K" in the KNN algorithm is the Nearest Neighbor we wish to take vote from.





K Nearest Neighbor Classifier



KNN Classifier

Similarity measures can be used to determine 'alikeness' of different tuples in the databases.

In this, a representative of every class is selected.

The classification is performed by assigning each tuples to the class to which it is more similar.

Let us assume that the classes are $\{C_1, C_2, \ldots\ldots C_n)$ and the training dataset D has $\{t_1, t_2, \ldots.. t_n)$ tuples.

The idea is to assign unknown instance t_i to class C_j such that the similarity measure of (t, C_i) is greater than or equal to the similarity measure of (t, C_j), where C_i is not equal to C_j.

The similarity can be obtained using **Distance Measures**.

KNN Classifier

Algorithm

1. Choose the representative of the class. Normally, the center or centroid of the class is chosen as the representative of the class.
2. Compare the test tuple and the center of the class
3. Classify the tuple to the appropriate class.

This procedure can be generalized as **KNN Algorithm**. Here 'K' is an integer. **KNN Algorithm** is as follows

1. Pick a suitable value for K.
2. Identify the **K-neighbours** for the unknown instance that needs to be classified.
3. Take the majority class of the **K-neighbours** as the class of the target unknown instance.

So the logic is that the class of the unknown instance is the majority class of the closest neighbours as chosen by K.

KNN Classifier

Algorithm

Step1: For implementing any algorithm, we need dataset. So during the first step of KNN, we must load the training as well as test data.

Step2: Next, we need to choose the value of K i.e. the nearest data points. K can be any integer.

Step3: For each point in the test data do the following:

3.1: Calculate the distance between test data and each row of training data with the help of any of the method namely: Euclidean, Manhattan or Hamming distance. The most commonly used method to calculate distance is Euclidean.

3.2: Now, based on the distance value, sort them in ascending order.

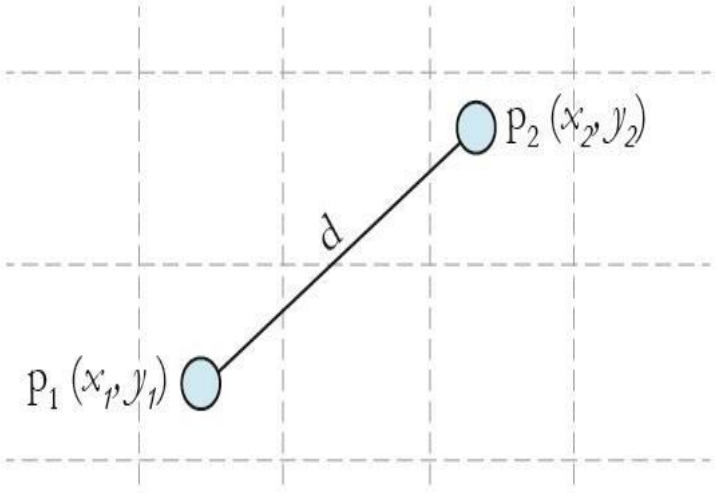
3.3: Next, it will choose the top K rows from the sorted array.

3.4: Now, it will assign a class to the test point based on most frequent class of these rows.

Step4: End

Euclidean Distance

- The **Euclidean distance** or **Euclidean metric** is the "ordinary" (i.e. straight-line) distance between two points in Euclidean space.
- The **Euclidean distance** between points **p** and **q** is the length of the line segment connecting them ($\overline{pq}$).



Euclidean distance (d) $= \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2}$$

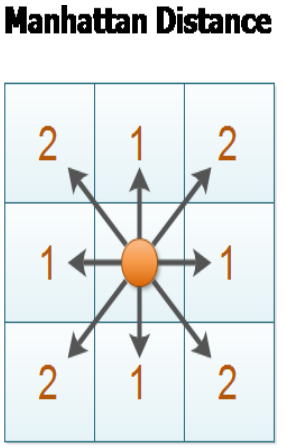
$$= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

Manhattan Distance or City block distance

- **Taxicab geometry** is a form of geometry in which the usual metric of Euclidean geometry is replaced by a new metric in which the distance between two points is the sum of the (absolute) differences of their coordinates.
- **The Manhattan distance**, also known as **rectilinear distance, city block distance, taxicab metric** is defined as the sum of the lengths of the projections of the line segment between the points onto the coordinate axes.

$$d = \sum_{i=1}^{n} |x_i - y_i|$$

- In chess, the distance between squares on the chessboard for rooks is measured in Manhattan distance.



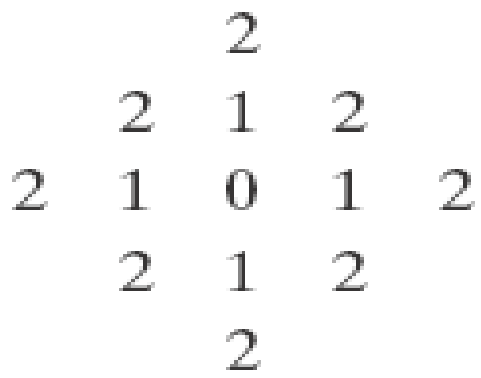
$$|x_1 - x_2| + |y_1 - y_2|$$

Manhattan Distance or City block distance

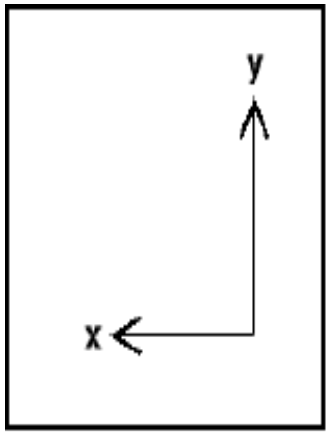
D_4 distance or City Block (Manhattan) Distance.

$$D_4(p,q) = |x-s| + |y-t|$$

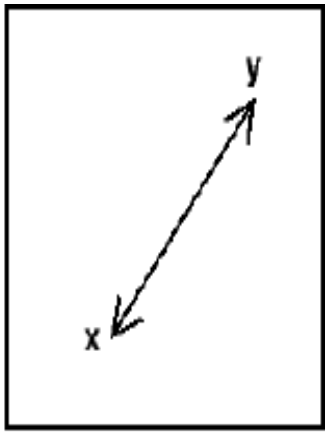
Points having City Block distance from p less than or equal to r from diamond centered at p.



Euclidean vs. Manhattan Distance



Manhattan



Euclidean

Minkowski Distance

Euclidean Distance Measurement Formula

$$d = ((p_1 - q_1)^2 + (p_2 - q_2)^2)^{1/2}$$

We can generalize this for an n-dimensional space as:

$$D_e = \left(\sum_{i=1}^{n} (p_i - q_i)^2 \right)^{1/2}$$

Minkowski Distance

Minkowski Distance is the generalized form of Euclidean and Manhattan Distance.

The formula for Minkowski Distance is given as:

$$D = \left(\sum_{i=1}^{n} |p_i - q_i|^p \right)^{1/p}$$

Here, **p** represents the order of the norm. When the **order(p) is 1**, it will represent **Manhattan Distance** and when the **order in the above formula is 2**, it will represent **Euclidean Distance**.

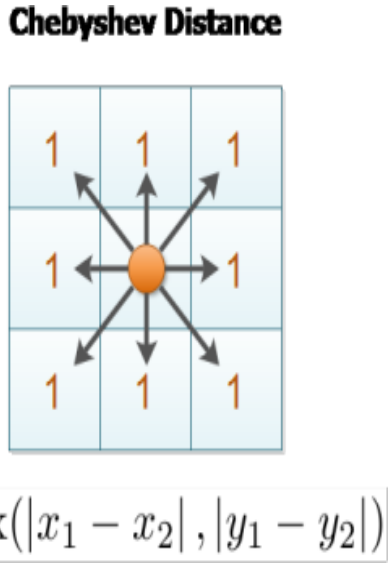
Chebyshev or Chessboard Distance

- The Chebyshev distance between two vectors or points p and q, with standard coordinates p_i and q_i respectively, is :

$$D_{\text{Chebyshev}}(p,q) := \max_i(|p_i - q_i|).$$

- It is also known as chessboard distance, since in the game of chess the minimum number of moves needed by a king to go from one square on a chessboard to another equals the Chebyshev distance between the centers of the squares

	a	b	c	d	e	f	g	h	
8	5	4	3	2	2	2	2	2	8
7	5	4	3	2	1	1	1	2	7
6	5	4	3	2	1	♔	1	2	6
5	5	4	3	2	1	1	1	2	5
4	5	4	3	2	2	2	2	2	4
3	5	4	3	3	3	3	3	3	3
2	5	4	4	4	4	4	4	4	2
1	5	5	5	5	5	5	5	5	1
	a	b	c	d	e	f	g	h	



$$\max(|x_1 - x_2|, |y_1 - y_2|)$$

Chebyshev or Chessboard Distance

D_8 distance or Chess Board distance is defined as

$$\mathbf{D_8(p,q)= \max(|x-s|,|y-t|)}$$

$S= \{q|D_8(p,q) \leq r\}$ form a square centered at p.

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

Numerical

Example 3.1 Let $V = \{0, 1\}$. Compute the D_e, D_4, D_8, and D_m distances between two pixels p and q. Let the pixel coordinates of p and q be (3, 0) and (2, 3), respectively, for the image shown in Fig. 3.10.

Find the distance measures.

	0	1	2	3
0	0	1	1	1
1	1	0	0	1
2	1	1	1	1 (q)
3	1	1	1	1
	(p)			

Fig. 3.10 Sample image

Solution The Euclidean distance is

$$D_e = \sqrt{(x-s)^2 + (y-t)^2} = \sqrt{(3-2)^2 + (0-3)^2}$$

$$= \sqrt{1+9} = \sqrt{10}$$

$$D_4 = |x-s| + |y-t| = |3-2| + |0-3|$$

$$= 1 + 3 = 4$$

$$D_8 = \max\left(|x-s|, |y-t|\right) = \max\left(|3-2|, |0-3|\right)$$

$$= \max(1, 3) = 3$$

Hamming Distance

- The Hamming distance between two strings of equal length is the number of positions at which the corresponding symbols are different.
- In another way, it measures the minimum number of substitutions required to change one string into the other.

- **Example : The Hamming distance between:**
  - "karolin" and "kathrin" is 3.
  - "karolin" and "kerstin" is 3.
  - **1011101** and **1001001** is 2.
  - **2173896** and **2233796** is 3.

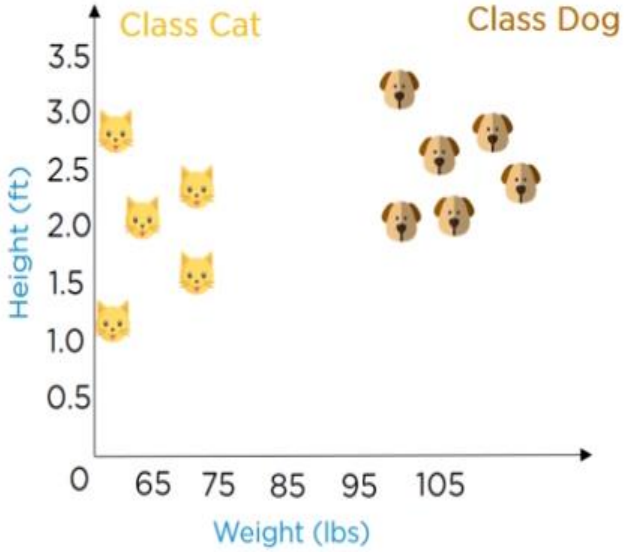
- It is used in telecommunication to count the number of flipped bits in a fixed-length binary word as an estimate of error, and therefore is sometimes called the signal distance.

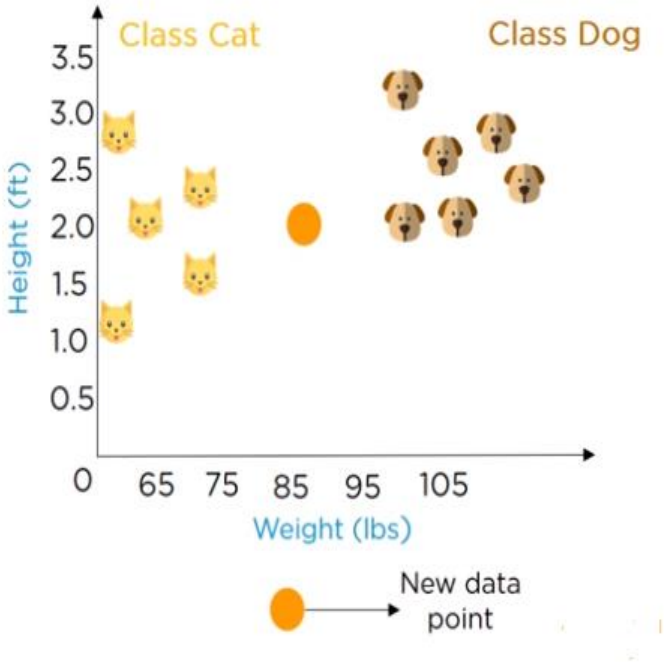
KNN Classifier

K Nearest Neighbors (KNN)

K Nearest Neighbors: KNN is a Classification algorithm generally used to predict categorical values.

To find if a new data point is a  or a  ?

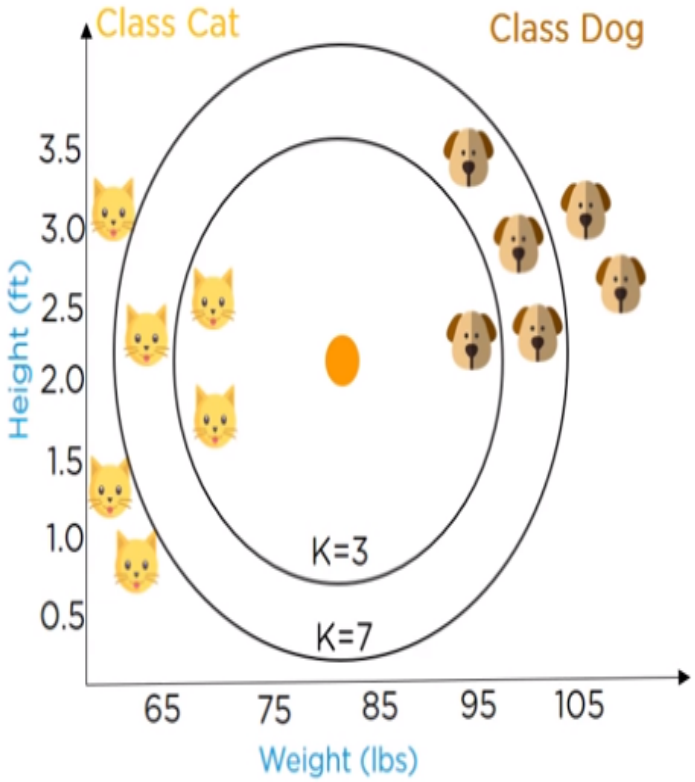




KNN Classifier

K Nearest Neighbors

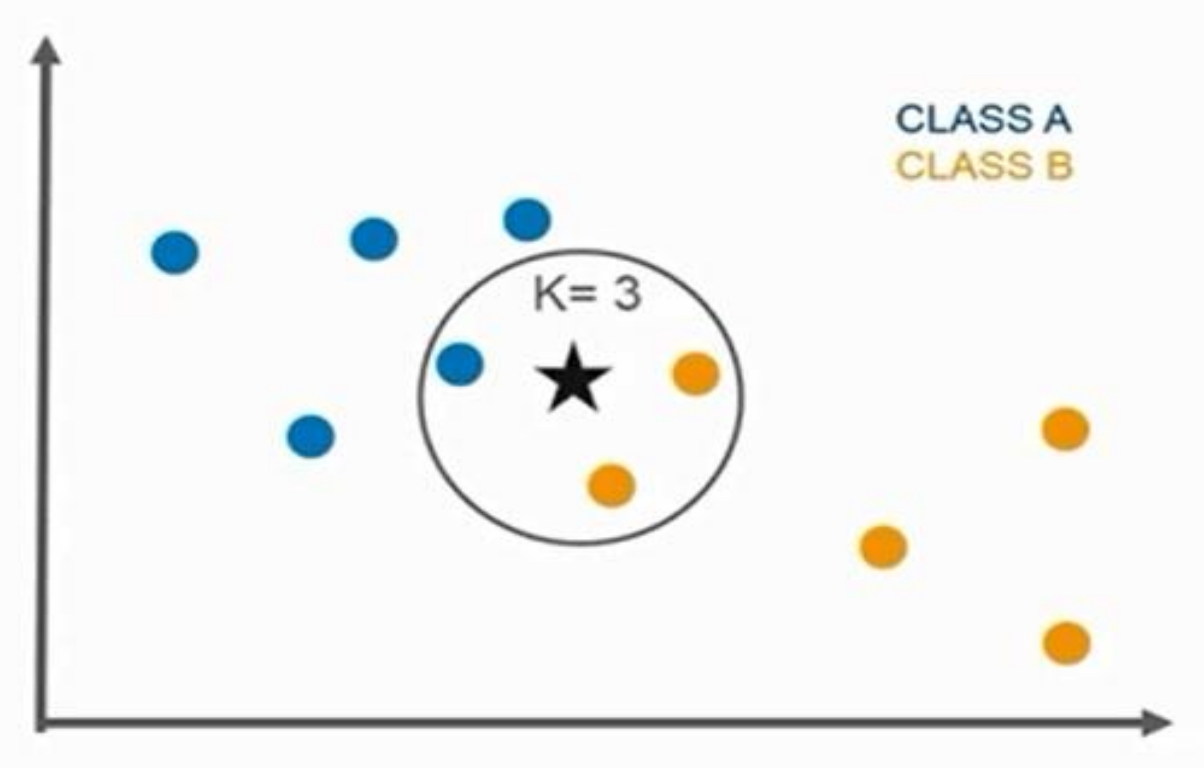
Choosing a K will define what class a new data point is assigned to:



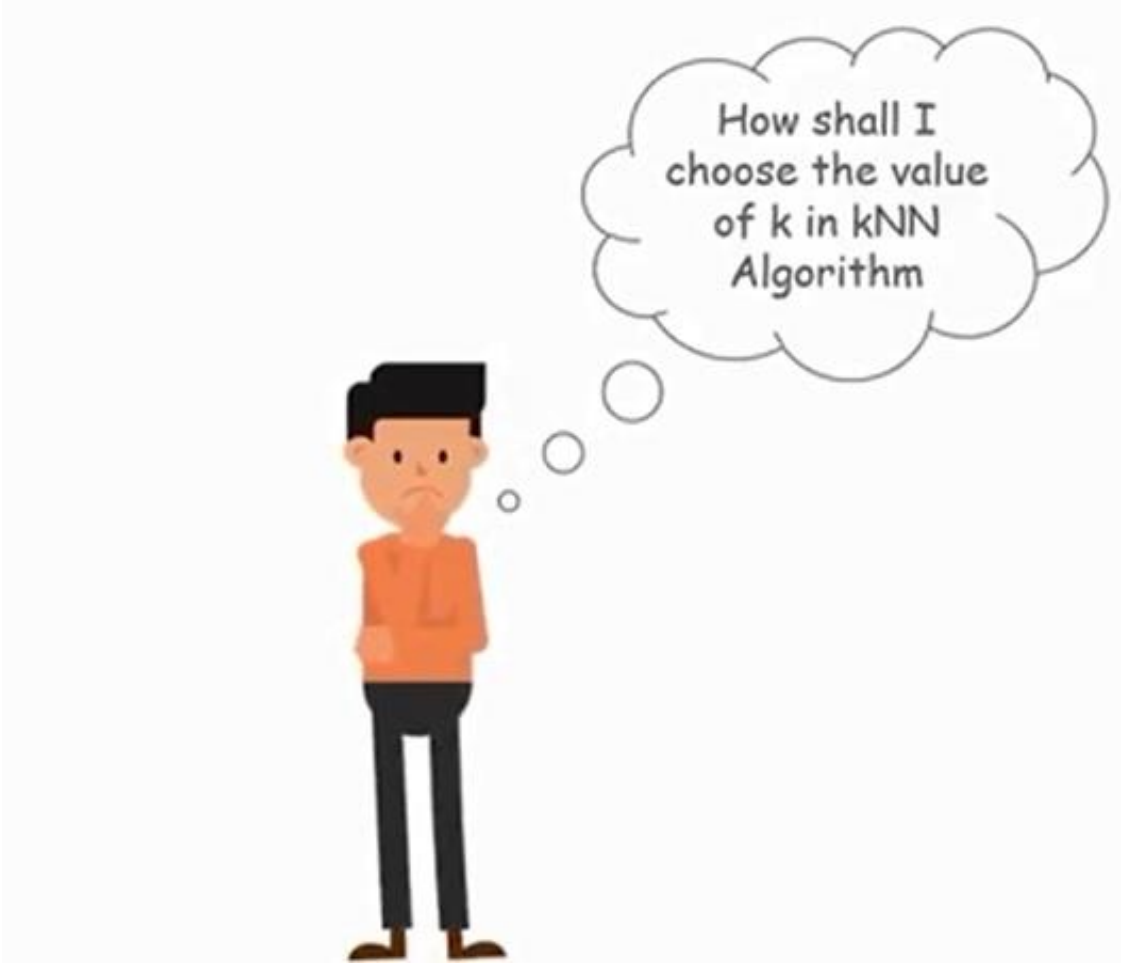
If K=3, the new data point belongs to class Cat

If K=7, the new data point belongs to class Dog

How does K Nearest Neighbor Classifier works



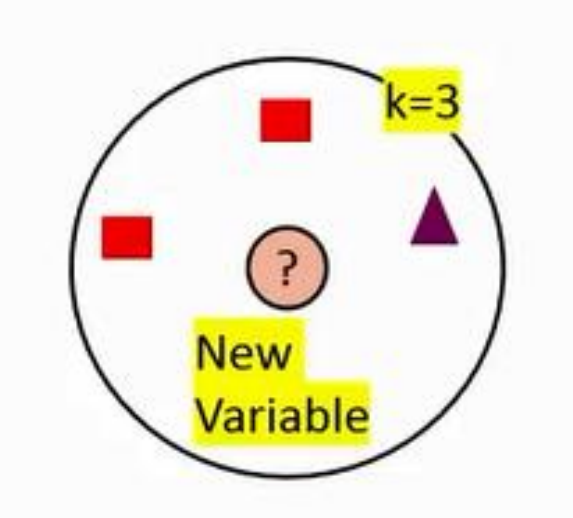
How to choose the value of K in KNN Algorithm



How to choose the value of K in KNN Algorithm

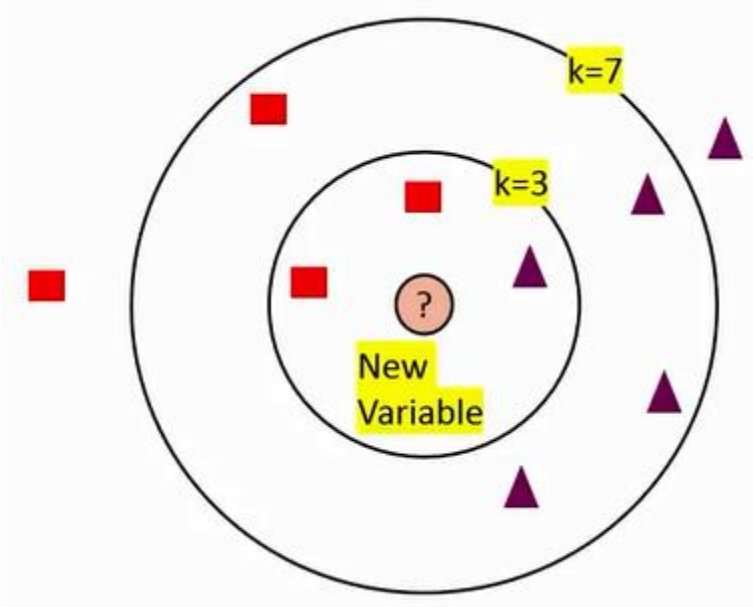
- KNN Algorithm based on feature similarity: Choosing the right value of K is a process of **parameter tuning,** and its **important for better accuracy**

Example:-



So at K=3, We can Classify "? " As Square instead of triangle

How to choose the value of K in KNN Algorithm

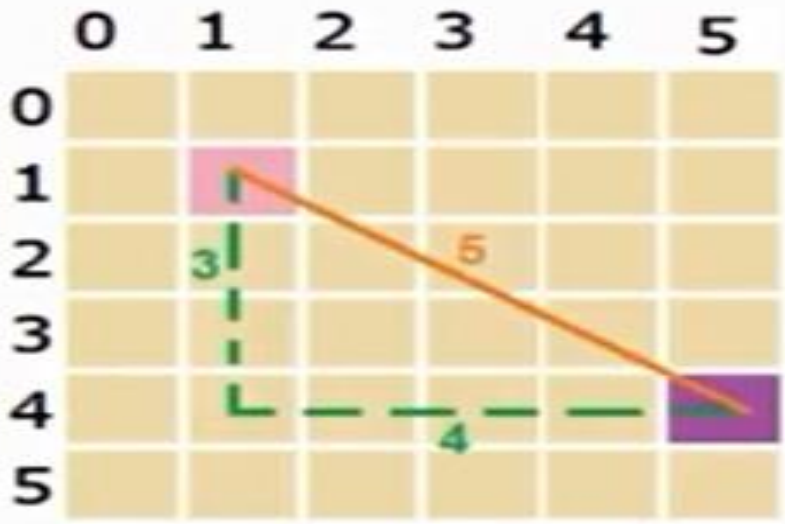


So at K=7, We can Classify "? " As triangle instead of Square

How to choose the value of K in KNN Algorithm

- There are no pre-defined statistical methods to find the most favorable value of K.
- Initialize a random K value and start computing.
- Odd value of K is selected to avoid confusion between two classes of data
- Choosing a small value of K leads to unstable decision boundaries.
- The substantial K value is better for classification as it leads to smoothening the decision boundaries.
- **Derive a plot between error rate and K denoting values in a defined range.**
- **Then choose the K value as having a minimum error rate.**

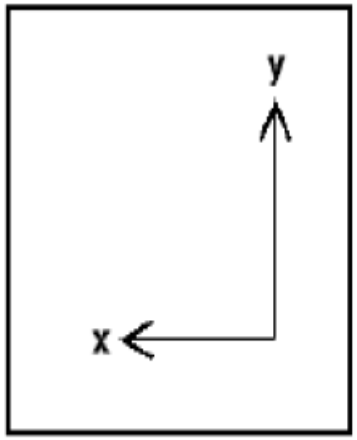
How things are predicted using KNN Algorithm



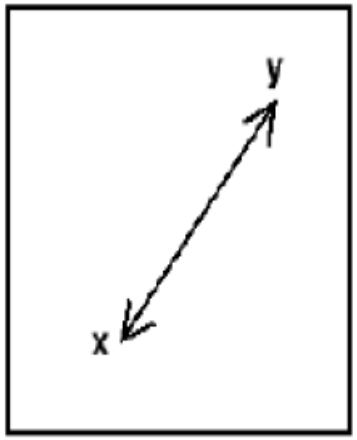
Euclidean distance = $\sqrt{(5-1)^2 + (4-1)^2} = 5$

Manhattan distance = $|5-1| + |4-1| = 7$

How things are predicted using KNN Algorithm



Manhattan



Euclidean

KNN Classifier-Numerical

Find the Nearest Neighbour of the following

a. 25 in the list {1 3 15 20 40 50 } in 1D

b. (1,3) in the list {(1,1),(3,3),(5,5)} in 2D

Soln- The neighbour can be determined by the distance between the elements.

(a) The element 25 is closer to 20 as its distance is less than the other elements. So 20 is the closest 1-neighbour of 25. The three closest neighbours of 25 are 20,15 and 40.

(b) Any distance measure can be used. If Euclidean distance is used, then the Euclidean distance between (1,3) and {(1,1), (3,3),(5,5)} are 2,2 and 3 respectively.

(c) So the nearest neighbours of (1,3) are {(1,1), (3,3)}

KNN Classifier-Numerical

2. The following points, coordinates, and classes are given

Point	Coordinates	Class
x1	(2,0)	Class1
x2	(4,2)	Class1
x3	(2,3)	Class1
x4	(-1,2)	Class2
x5	(-2,3)	Class2

Classify the point(1,1) using nearest neighbour technique with K=1.

Solution-Find Euclidean distance between the given features and (1,1)

De=$\sqrt{(x1 - x2)^2 + (y1-y2)^2}$ Where (x1,y1) and (x2,y2) are the two points between which we have to calculate the distance.

(2,0) (1,1)=De=$\sqrt{(2 - 1)^2 + (0-1)^2}$ =$\sqrt{2}$=1.414

(4,2) (1,1)=De=$\sqrt{(4 - 1)^2 + (2-1)^2}$ =$\sqrt{10}$=3.1623

KNN Classifier-Numerical

Solution-

(2,3) (1,1)=De=$\sqrt{(2-1)^2+(3-1)^2}$ =$\sqrt{5}$=2.236

(-1,2) (1,1)=De=$\sqrt{(-1-1)^2+(2-1)^2}$ =$\sqrt{5}$=2.236

(-2,3) (1,1)=De=$\sqrt{(-2-1)^2+(3-1)^2}$ =$\sqrt{13}$=3.605

The minimum distance is between (1,1) and (2,0) whose class is 1. Therefore, class of (1,1) is 1.

3. If k=3, the there nearest points are computed and its majority class is assigned to the point.(Home work)

How does KNN Algorithm Works

Consider a dataset having two variables: height (cm) & weight (kg) and each point is classified as Normal or Underweight

Weight(x2)	Height(y2)	Class
51	167	Underweight
62	182	Normal
69	176	Normal
64	173	Normal
65	172	Normal
56	174	Underweight
58	169	Normal
57	173	Normal
55	170	Normal

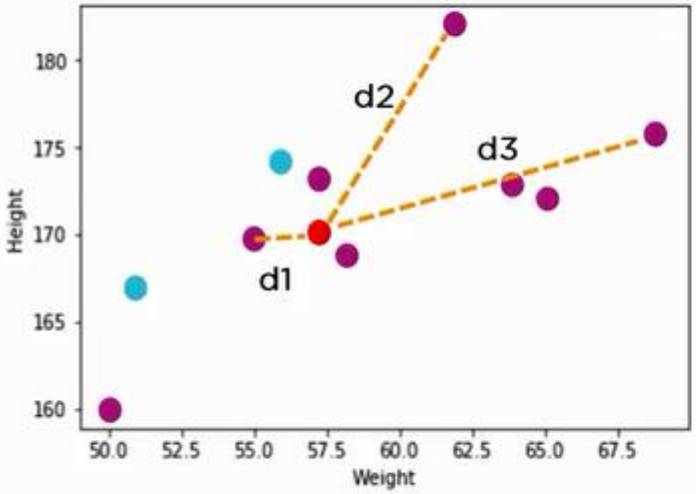
How does KNN Algorithm Works

On the basis of the given data we have to classify the below set as Normal or Underweight using KNN

57 kg	170 cm	?

How does KNN Algorithm Works

Let's calculate it to understand clearly:



dist(**d1**)= √(170-167)² + (57-51)² ~= 6.7

dist(**d2**)= √(170-182)² + (57-62)² ~= 13

dist(**d3**)= √(170-176)² + (57-69)² ~= 13.4

Similarly, we will calculate Euclidean distance of unknown data point from all the points in the dataset

How does KNN Algorithm Works

Hence, we have calculated the Euclidean distance of unknown data point from all the points as shown:

Where (x1, y1) = (57, 170) whose class we have to classify

Weight(x2)	Height(y2)	Class	Euclidean Distance
51	167	Underweight	6.7
62	182	Normal	13
69	176	Normal	13.4
64	173	Normal	7.6
65	172	Normal	8.2
56	174	Underweight	4.1
58	169	Normal	1.4
57	173	Normal	3
55	170	Normal	2

How does KNN Algorithm Works

Now, lets calculate the nearest neighbor at k=3

Weight(x2)	Height(y2)	Class	Euclidean Distance
51	167	Underweight	6.7
62	182	Normal	13
69	176	Normal	13.4
64	173	Normal	7.6
65	172	Normal	8.2
56	174	Underweight	4.1
58	169	Normal	1.4
57	173	Normal	3
55	170	Normal	2

57 kg	170 cm	?

KNN Algorithm Example

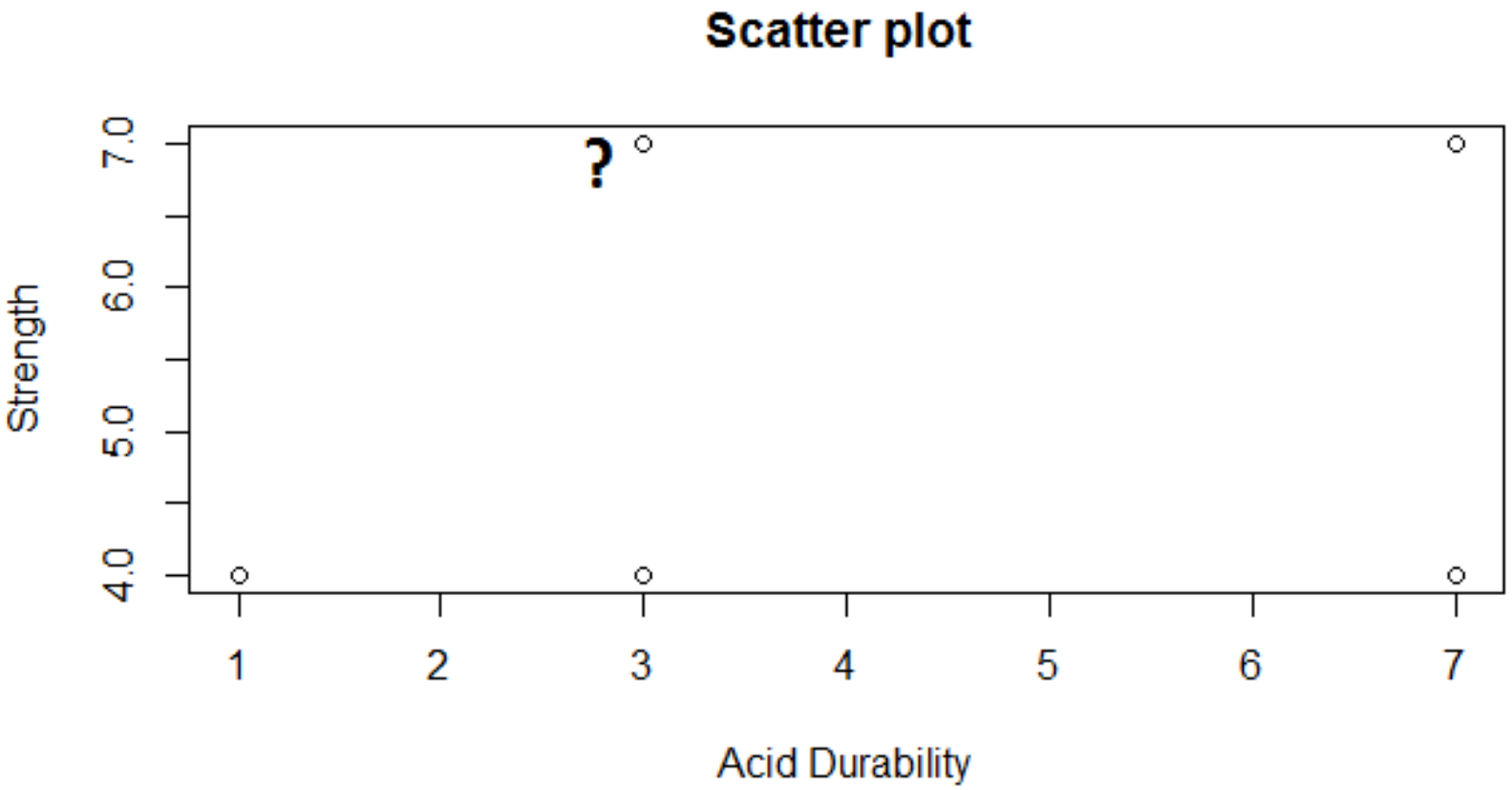
Q1 We have data from the questionnaires survey (to ask people opinion) and objective testing with two attributes (acid durability and strength) to classify whether a special paper tissue is good or not. Here is four training samples

Points	X1 (Acid Durability)	X2(strength)	Y=Classification
P1	7	7	BAD
P2	7	4	BAD
P3	3	4	GOOD
P4	1	4	GOOD

KNN Algorithm Example

Points	X1(Acid Durability)	X2(Strength)	Y(Classification)
P1	7	7	BAD
P2	7	4	BAD
P3	3	4	GOOD
P4	1	4	GOOD
P5	3	7	?

KNN Algorithm Example



Euclidean Distance From Each Point

KNN				
Euclidean Distance of P5(3,7) from	P1	P2	P3	P4
	(7,7)	(7,4)	(3,4)	(1,4)
	Sqrt((7-3) ^{2} + (7-7)^2) = $\sqrt{16}$ = 4	Sqrt((7-3) ^{2} + (4-7)^2) = $\sqrt{25}$ = 5	Sqrt((3-3) ^{2} + (4-7)^2) = $\sqrt{9}$ = 3	Sqrt((1-3) ^{2} + (4-7)^2) = $\sqrt{13}$ = 3.60

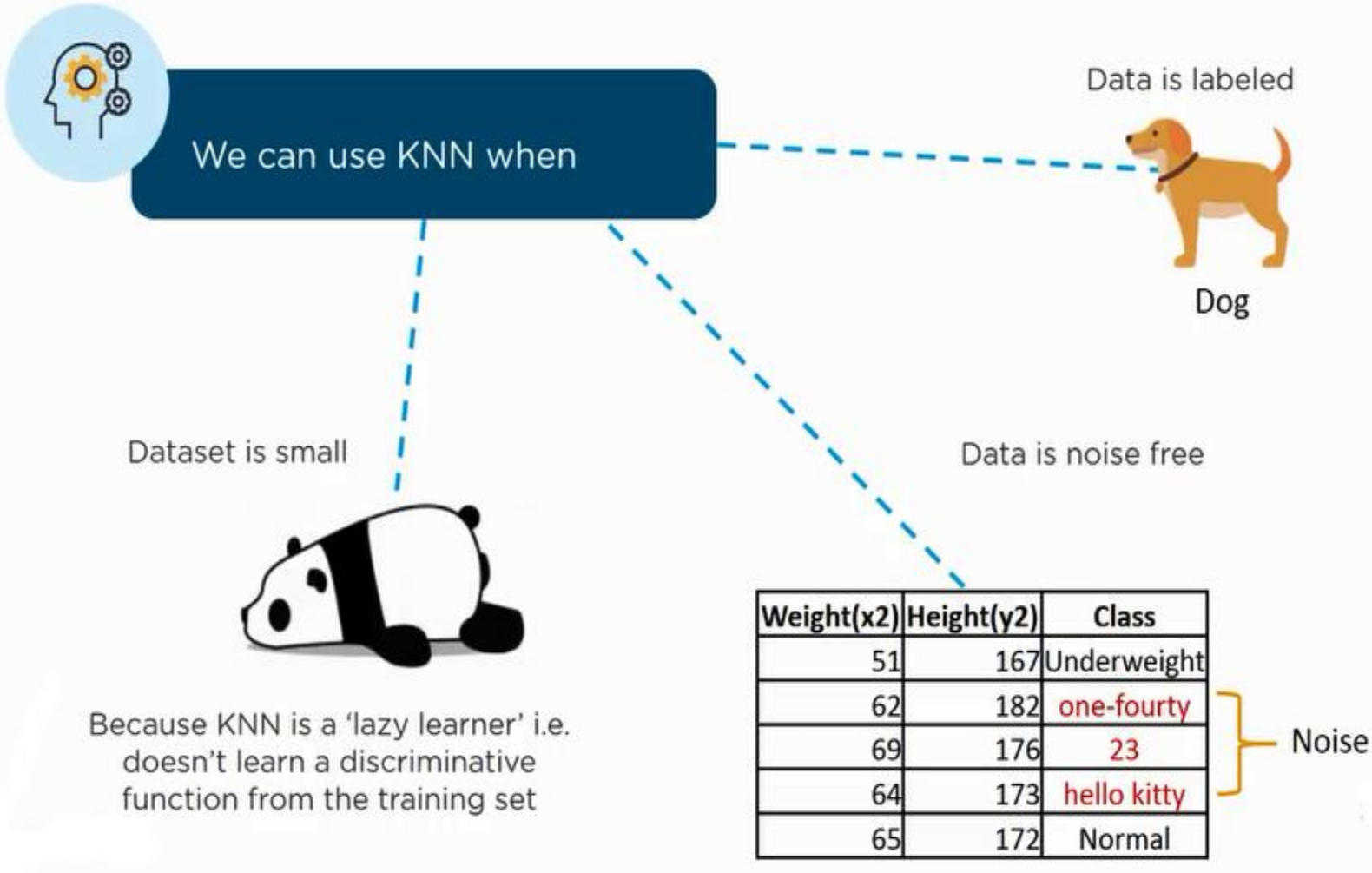
3 Nearest Neighbour

Euclidean Distance of P5(3,7) from	P1	P2	P3	P4
	(7,7)	(7,4)	(3,4)	(1,4)
	Sqrt((7-3) ^{2} + (7-7)^2) = $\sqrt{16}$ = 4	Sqrt((7-3) ^{2} + (4-7)^2) = $\sqrt{25}$ = 5	Sqrt((3-3) ^{2} + (4-7)^2) = $\sqrt{9}$ = 3	Sqrt((1-3) ^{2} + (4-7)^2) = $\sqrt{13}$ = 3.60
Class	BAD	BAD	GOOD	GOOD

KNN Classification

Points	X1(Durability)	X2(Strength)	Y(Classification)
P1	7	7	BAD
P2	7	4	BAD
P3	3	4	GOOD
P4	1	4	GOOD
P5	3	7	GOOD

When do we use KNN Algorithm



Weight(x2)	Height(y2)	Class
51	167	Underweight
62	182	one-fourty
69	176	23
64	173	hello kitty
65	172	Normal

KNN

Pros

It is very simple algorithm to understand and interpret.

It is very useful for nonlinear data because there is no assumption about data in this algorithm.

It is a versatile algorithm as we can use it for classification as well as regression.

It has relatively high accuracy but there are much better supervised learning models than KNN.

Cons

It is computationally a bit expensive algorithm because it stores all the training data.

High memory storage required as compared to other supervised learning algorithms.

Prediction is slow in case of big N.

It is very sensitive to the scale of data as well as irrelevant features.

Applications of KNN

The following are some of the areas in which KNN can be applied successfully –

Banking System
KNN can be used in banking system to predict weather an individual is fit for loan approval? Does that individual have the characteristics similar to the defaulters one?

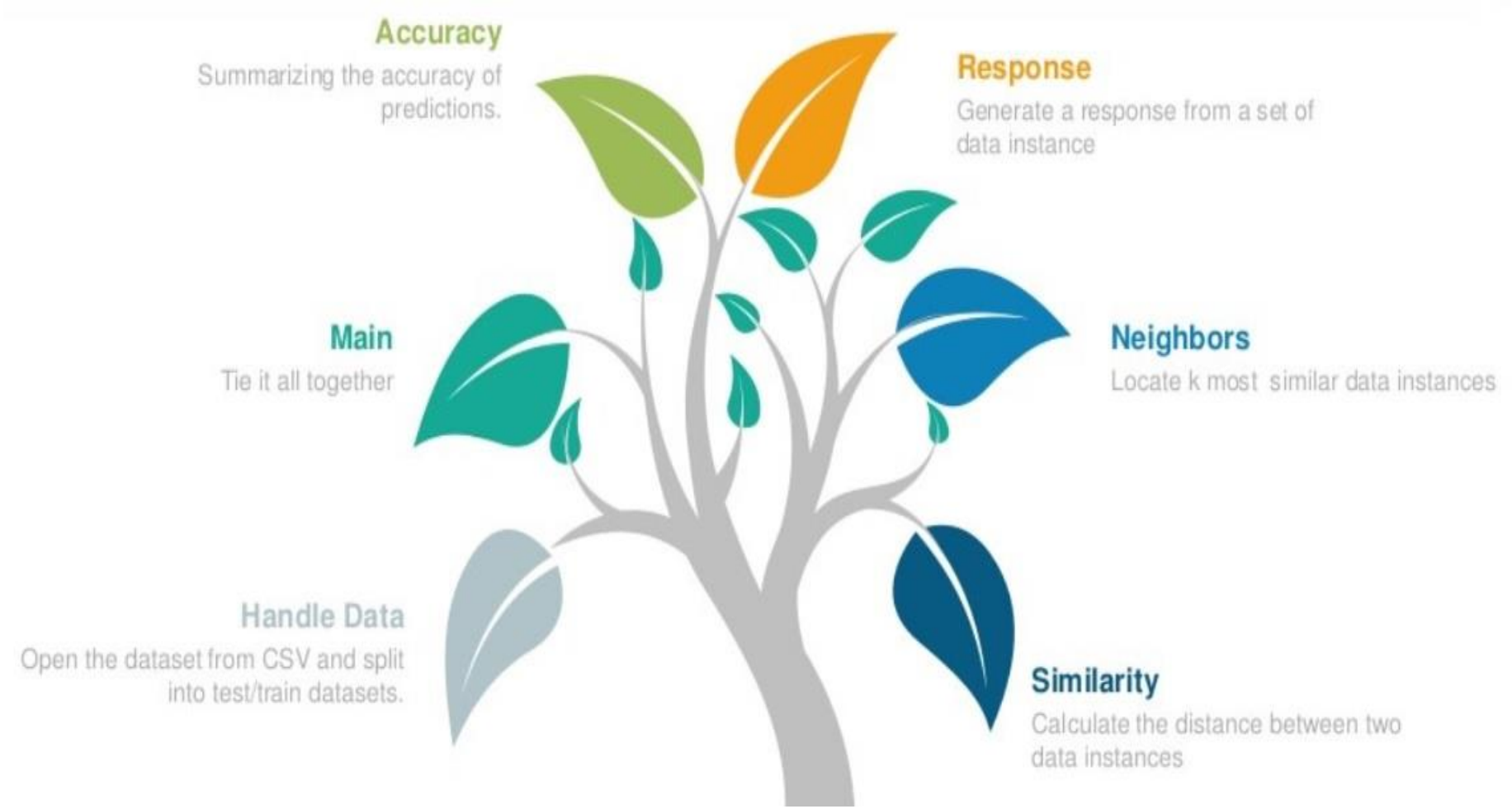
Calculating Credit Ratings
KNN algorithms can be used to find an individual's credit rating by comparing with the persons having similar traits.

Politics
With the help of KNN algorithms, we can classify a potential voter into various classes like "Will Vote", "Will not Vote", "Will Vote to Party 'Congress', "Will Vote to Party 'BJP'.

Other areas in which KNN algorithm can be used are **Speech Recognition, Handwriting Detection, Image Recognition and Video Recognition.**

Let us Code-KNN

Iris Data Set



Let us Code-KNN

Iris Data Set

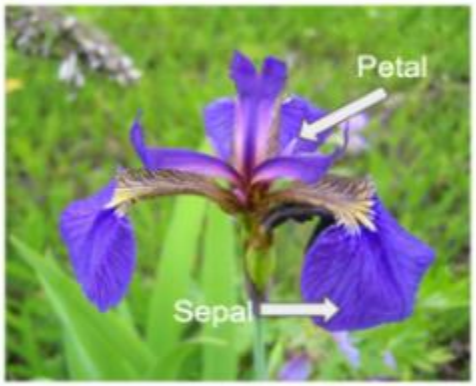
1. The k-nearest neighbor algorithm is imported from the scikit-learn package.
2. Create feature and target variables.
3. Split data into training and test data.
4. Generate a k-NN model using neighbors value.
5. Train or fit the data into the model.
6. Predict the future.

Let us Code-KNN

Iris Flower Data Set

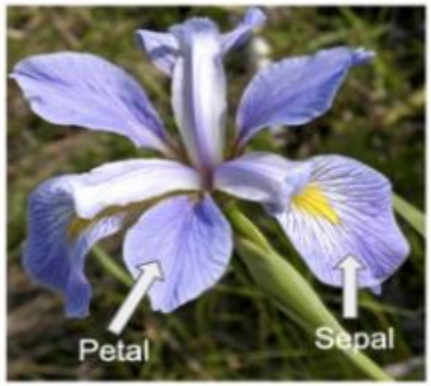
In this dataset, there are **4 features sepal length, sepal width, petal length and petal width** and the **target variable has 3 classes namely 'setosa', 'versicolor', and 'virginica'.** Objective for a multiclass classifier is to predict the target class given the values for the four features.

Iris setosa



Iris versicolor

Iris virginica



Training & Test data

Features | Labels

Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	Iris setosa
4.9	3.0	1.4	0.2	Iris setosa
7.0	3.2	4.7	1.4	Iris versicolor
6.4	3.2	4.5	1.5	Iris versicolor
6.3	3.3	6.0	2.5	Iris virginica
5.8	3.3	6.0	2.5	Iris virginica

Characteristics of this dataset:

150 samples, with 4 attributes (same units, all numeric), Balanced class distribution (50 samples for each class), No missing data,

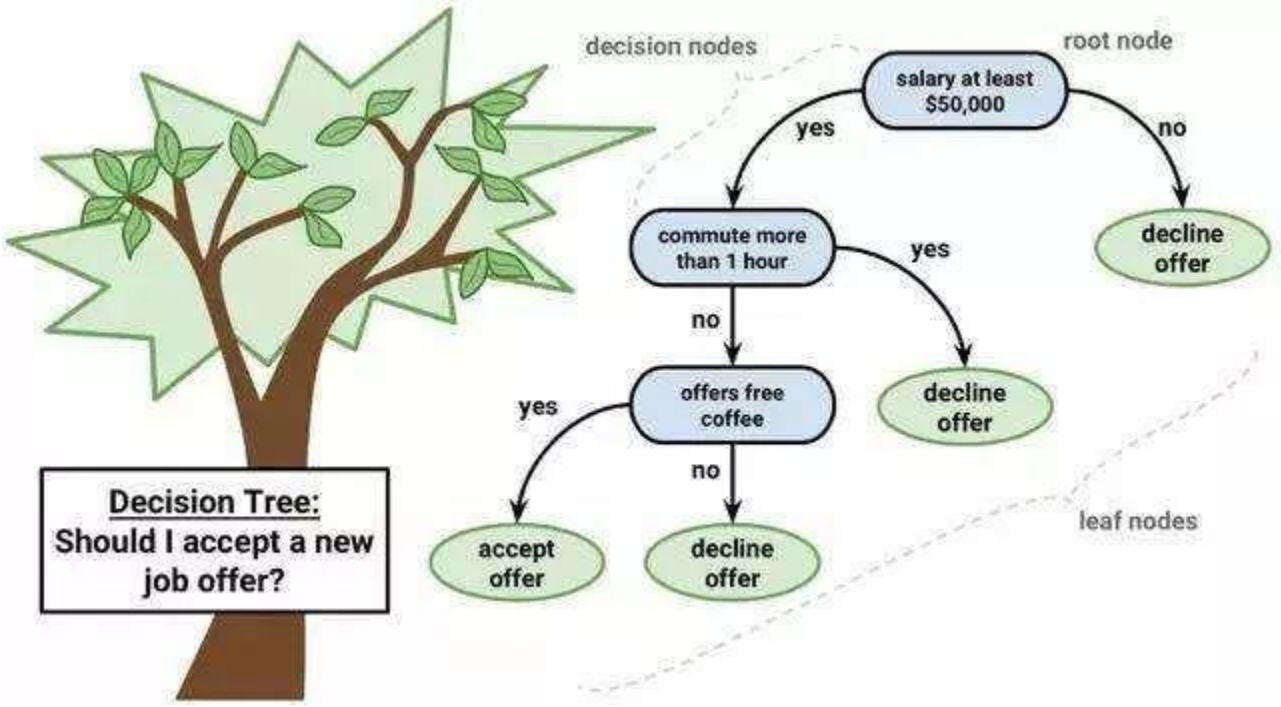
Print target vector iris species: 0 = setosa, 1 = versicolor, 2 = virginica

Decision Tree

- Graphical Representation of all the possible solutions to a decision
- Decisions are based on some conditions
- Decision made can be easily explained

What is Decision Tree

- **A decision tree** is a graphical representation of all the possible solutions to a decision based on a conditions.



Decision tree is one of the predictive modeling approaches used in **statistics, data mining and machine learning.**

Internal nodes represent the features of a dataset, branches represent the Decision rules, and **each Leaf node represents the outcome.**

What is Decision Tree

- **Decision trees** are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions.
- It is one of the most widely used and practical methods for **Supervised Learning.**
- **Decision Trees** are **a non-parametric supervised learning method** used for **both classification and regression tasks.**
- **Non-Parametric-**Algorithms that do not make strong assumptions about the form of the mapping function
- **Nonparametric methods are good when you have a lot of data.**
- **No prior knowledge,**
- **When you don't want to worry too much about choosing just the right features.**

Decision Tree

- Tree models where the target variable can take a discrete set of values are called **Classification Trees.**
- **Decision trees** where the target variable can take **continuous values (typically real numbers) are called regression trees.**
- **Classification And Regression Tree (CART)** is general term for this.
- **Information gain** is used to decide which feature to split on at each step in building the tree.

Decision Tree

- In Decision Tree
- At each step we should choose the split that results in the purest daughter nodes.
- A commonly used measure of **purity** is called **information.**
- For each node of the tree, the information value measures how much information a feature gives us about the class.
- The **split** with **the highest information gain** will be taken as the **first split** and **the process will continue until all children nodes are pure, or until the information gain is 0.**
- Decision tree uses the tree representation to solve the problem in which each **Leaf node** corresponds to **a class label** and **attributes** are represented on the **internal node of the tree.**

Decision Tree-Gini Impurity

Pure

Pure means, in a selected sample of dataset all data belongs to same class (PURE).

Impure

Impure means, data is mixture of different classes.

Definition of Gini Impurity

- **Gini Impurity** is a **measurement of the likelihood of an incorrect classification** of a new instance of a random variable, if that new instance were randomly classified according to the distribution of class labels from the data set.
- If our dataset is Pure then likelihood of incorrect classification is 0.
- If our sample is mixture of different classes then likelihood of incorrect classification will be high.

Decision Tree-Algorithm

Steps for Making Decision Tree

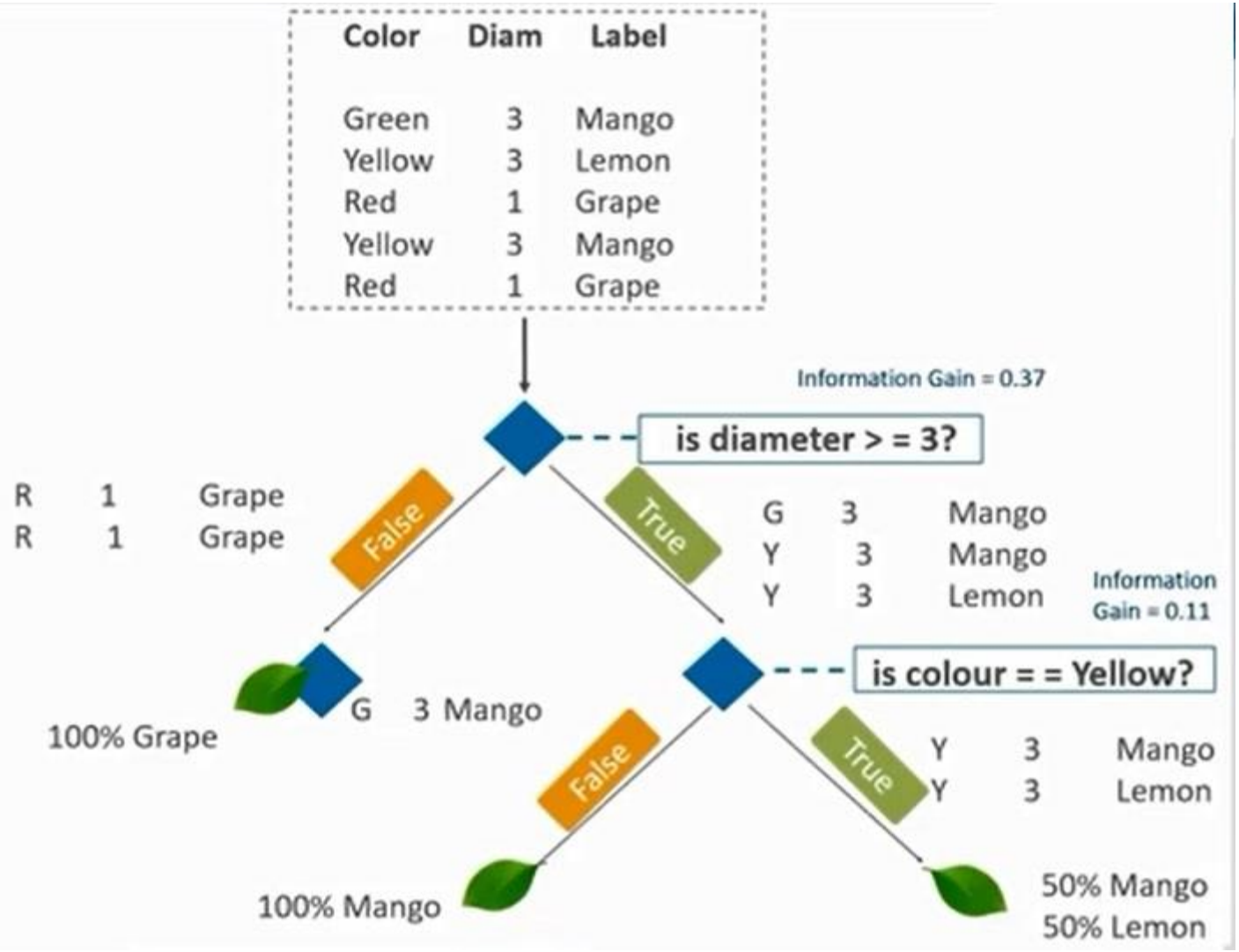
1. Get **list of rows (dataset)** which are taken into consideration for making decision tree (recursively at each nodes).
2. **Calculate uncertainty** of our dataset or Gini impurity or how much our data is mixed up etc.
3. **Generate list of all question** which needs to be asked at that node.
4. **Partition rows** into **True rows and False rows** based on each question asked.
5. Calculate **information gain OR gini impurity** and partition of data from previous step.
6. Update **highest information gain** based on each question asked.
7. Update **best question based on information gain** (higher information gain).
8. **Divide the node on best question.**
9. **Repeat again** from step 1 again **until we get pure node (leaf nodes).**

Understanding a Decision Tree

- **Dataset**

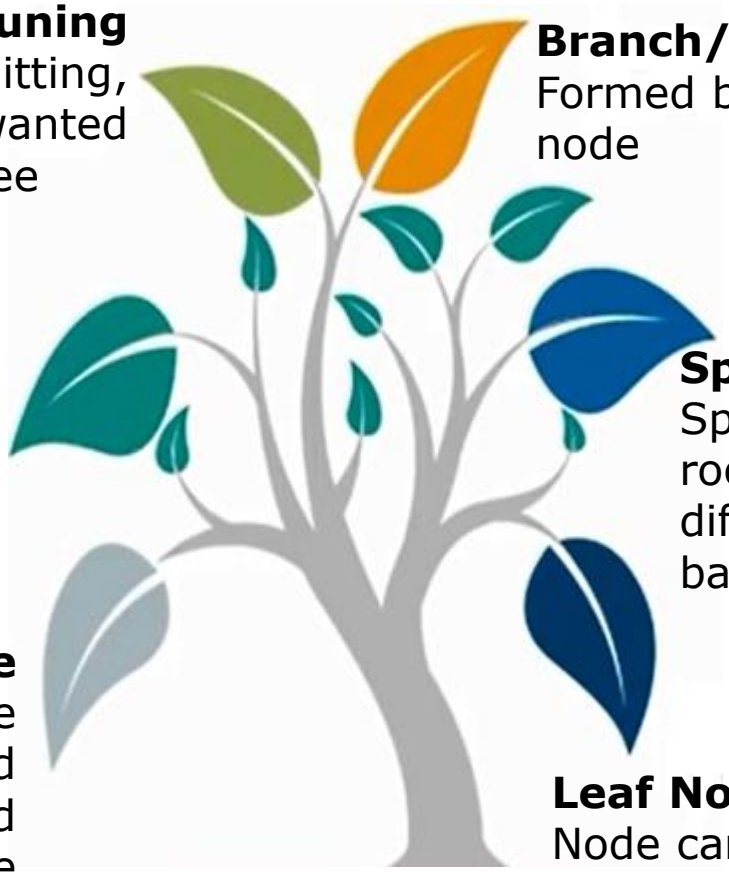
Colour	Diameter	Label
Green	3	Mango
Yellow	3	Mango
Red	1	Grape
Red	1	Grape
Yellow	3	Lemon

Understanding a Decision Tree



Decision Tree Terminologies

Pruning
Opposite of Splitting, basically Moving unwanted branches from the Tree



Branch/Sub Tree
Formed by Splitting the Tree/ node

Parent/Child Node
Root Node is the parent node and All the other nodes branched from it is Known as child node

Splitting
Splitting is dividing the root node/sub–Node Into different parts on the basis of some condition

Root Node
It represents the entire population or Sample and this further gets divided into two Or more homogenous sets

Leaf Node
Node cannot be further Segregated Into further nodes

CART Algorithm

- Lets first visualize the Decision Tree

Which Question to ask and When?

CART which stands for **Classification And Regression Trees**

What are Splitting Measures?

- With more than one attribute taking part in the decision-making process, it is **necessary to decide the relevance and importance of each of the attributes.**
- Thus, **placing the most relevant at the root node** and **further traversing down by splitting the nodes.**
- As we move further down the tree, **the level of impurity or uncertainty decreases,** thus leading to a better classification or best split at every node.
- To decide the same, splitting measures such as Information Gain, Gini Index, etc. are used.

Learn about Decision Tree

But how we do choose the best attribute
Or
How does a tree decide where to Split?

Information Gain

The **information gain is the decrease in entropy after a dataset is split on the basis of an attribute**. Constructing a decision tree is about finding attribute that returns the highest information gain

Gini Index

The **measure of impurity (or purity)** used in building decision tree in CART is Gini Index

Chi Square

It is an algorithm to find out the statistical significance the differences between the sub-node and parent node

Reduction in Variance

Reduction in Variance is an algorithm used for continuous target variables (Regression problems). The **split with lower variance is selected as the criteria to split the population**

Gini Index

Gini Index: It is calculated by subtracting the sum of squared probabilities of each class from one. It favors larger partitions and easy to implement whereas information gain favors smaller partitions with distinct values.

$$Gini\ Index = 1 - \sum (P(x=k))^2$$

A feature with a lower Gini index is chosen for a split.

Gini index or Gini impurity measures **the degree or probability of a particular variable being wrongly classified when it is randomly chosen.**

But what is actually meant by 'impurity'?

If all the elements belong to a single class, then it can be called pure. The degree of Gini index varies between 0 and 1, Where-

0 denotes that all elements belong to a certain class or **if there exists only one class or Pure data**, and

1 denotes that **the elements are randomly distributed across various classes or impure data.**

A Gini Index of 0.5 denotes equally distributed elements into some classes.

Gini Index Example

Calculate the Gini Index for Past Trend

Past Trend	Open Interest	Trading Volume	Return
Positive	Low	High	Up
Negative	High	Low	Down
Positive	Low	High	Up
Positive	High	High	Up
Negative	Low	High	Down
Positive	Low	Low	Down
Negative	High	High	Down
Negative	Low	High	Down
Positive	Low	Low	Down
Positive	High	High	Up

Gini Index Example

Calculate the Gini Index for Past Trend

P(Past Trend=Positive): 6/10

P(Past Trend=Negative): 4/10

If (Past Trend = Positive & Return = Up), probability = 4/6

If (Past Trend = Positive & Return = Down), probability = 2/6

Gini index = 1 - ((4/6)^2 + (2/6)^2) = 0.45

If (Past Trend = Negative & Return = Up), probability = 0

If (Past Trend = Negative & Return = Down), probability = 4/4

Gini index = 1 - ((0)^2 + (4/4)^2) = 0

Weighted sum of the Gini Indices can be calculated as follows:

Gini Index for Past Trend = (6/10)0.45 + (4/10)0 = 0.27

Gini Index Example

Calculate the Gini Index for Open Interest

P(Open Interest=High): 4/10

P(Open Interest=Low): 6/10

If (Open Interest = High & Return = Up), probability = 2/4

If (Open Interest = High & Return = Down), probability = 2/4

Gini index = 1 - ((2/4)^2 + (2/4)^2) = 0.5

If (Open Interest = Low & Return = Up), probability = 2/6

If (Open Interest = Low & Return = Down), probability = 4/6

Gini index = 1 - ((2/6)^2 + (4/6)^2) = 0.45

Weighted sum of the Gini Indices can be calculated as follows:

Gini Index for Open Interest = (4/10)0.5 + (6/10)0.45 = 0.47

Research & Null Hypothesis

- **The Research Hypothesis (H1):-**It proposes that the two variables are related in the Population.

- **The Null Hypothesis (H0):-** It states that **no association exists between the two variables** in the population, and therefore the **variables are statistically independent.**

The Null Hypothesis defines **Expected and Observed Frequencies.**

Chi Square Formula

Chi Square Formula

- The Chi-square formula is **used in the Chi-square test to compare two statistical data sets.**
- Chi-Square is one of the most useful **non-parametric statistics**.
- The Chi-Square test is used in data consist of people distributed across categories, and to know whether that distribution is different from what would expect by chance.
- A very small Chi-Square test statistic means that your observed data fits your expected data extremely well.
- A very large Chi-Square test statistic means that the data does not fit very well.
- If the chi-square value is large, you can reject the null hypothesis.
- Chi-Square is one way to show a relationship between two categorical variables.
- There are two types of variables in statistics: numerical variables and non-numerical variables.
- The value can be calculated by using the given observed frequency and expected frequency.
- The Chi-Square is denoted by χ^2 and the formula is:

$$\chi^2 = \sum (O - E)^2 / E$$

Where, O = Observed frequency, E = Expected frequency,
$\sum$ = Summation χ^2 = Chi-Square value

Chi Square Formula

Chi Square Formula

$$\chi^2 = \sum (O - E)^2 / E$$

- **Expected Frequencies(E)** is the cell frequencies that would be expected in a table **if the two tables were statistically independent.**
- **Observed Frequencies(O)** is the **cell frequencies actually observed in a table.**

$$\text{Expected Value} = \frac{(Row\ Total) * (Column\ Total)}{Total\ Number\ of\ Observatiosn}$$

To obtain the expected frequencies for any cell in which the two variables are assumed independent, multiply the row and column totals for that cell and divide the product by the total number of cases in the table

Chi Square Formula

- **Chi-square test** is an Inferential Statistics technique designed to test for significant relationships between two variables.
- A **chi-square test is a statistical test** used to **compare observed results with expected results.**
- The purpose of this test is **to determine if a difference between observed data and expected data is due to chance, or if it is due to a relationship** between the variables you are studying.
- **Chi-square** requires no assumptions about the shape of the population distribution from which a sample is drawn.
- However, like all inferential techniques it assumes random sampling.

Chi Square Example

Chi Square Example

- Calculate the chi-square value for the following data

	Male	Female
Full Stop	6(observed) 6.24 (expected)	6 (observed) 5.76 (expected)
Rolling Stop	16 (observed) 16.12 (expected)	15 (observed) 14.88 (expected)
No Stop	4 (observed) 3.64 (expected)	3 (observed) 3.36 (expected)

Solution:
Now calculate Chi Square using the following formula:

$$\chi^2 = \Sigma\ (O - E)^2 / E$$

Calculate this formula for each cell, one at a time. For example, cell #1 (Male/Full Stop):Observed number is: 6, Expected number is: 6.24
Therefore, $\chi^2 = (6 - 6.24)^2 / 6.24 = 0.0092$

- Continue doing this for the rest of the cells, and add the final numbers for each cell together to get the final Chi-Square number.
- There are **6 total cells**, so at the end, you **should be adding six numbers together for your final Chi-Square number.**

Limitations of The Chi Square Test

- **The Chi-square test does not give us much information about the strength of the relationship or its substantive significance in the population.**

- The Chi-square test is **sensitive to sample size**.
- The **size of the calculated chi-square is directly proportional to the size of the sample**, independent of the strength of the relationship between the variables.

Example

- **The below table gives the relationship between handedness and gender in the U.S.A. This data is also known as HANES data.**

	Men	Women	Total
Right-Handed	934	1070	2004
Left-Handed	113	92	205
Ambidextrous	20	08	28
Total	1067	1170	2237

$$\text{Expected Value} = \frac{(Row\ Total) * (Column\ Total)}{Total\ Number\ of\ Observations}$$

$$= \frac{1067 * 2004}{2237} = 956$$

Similarly, calculate expected values for Left-Handed & Ambidextrous

Example

- **The below table gives the relationship between handedness and gender in the U.S.A. This data is also known as HANES data.**

	Men	Women	Total
Right-Handed	934	1070	2004
Left-Handed	113	92	205
Ambidextrous	20	08	28
Total	1067	1170	2237

$$x^2 = sum\ of\ \frac{(observed\ frequency - expected\ frequency)^2}{expected\ frequency}$$

$$\chi^2 = \Sigma\ (O - E)^2 / E$$

Example

$$\chi^2 = \Sigma\,(O - E)^2 / E$$

$$\frac{(934-956)^2}{956} + \frac{(1{,}070-1{,}048)^2}{1{,}048} + \frac{(113-98)^2}{98}$$

$$+ \frac{(92-107)^2}{107} + \frac{(20-13)^2}{13} + \frac{(8-15)^2}{15}$$

$$\chi^2 = 12$$

Important Terms of Decision Tree

ENTROPY

ENTROPY IS THE MEASURE OF RANDOMNESS OR UNPREDICTABILITY IN THE DATASET

EXAMPLE

HIGH ENTROPY

THIS DATASET HAS A VERY HIGH ENTROPY

Entropy and Information Gain

Entropy

Entropy measures the impurity or uncertainty present in the data.

$$H(S) = -\sum_{i=1}^{N} p_i \log_2 p_i$$

where:

- ***S – set of all instances in the dataset***
- ***N – number of distinct class values***
- ***pi – event probability***

Information Gain (IG)

IG indicates how much "information" a particular feature/ variable gives us about the final outcome.

$$Gain(A,S) = H(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} \cdot H(S_j) = H(S) - H(A,S)$$

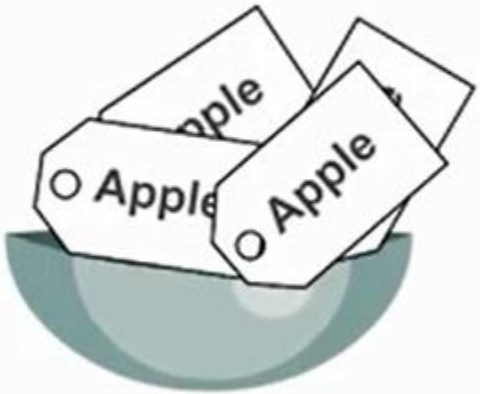
where:

H(S) – entropy of the whole dataset S

- *|Sj| – number of instance with j value of an attribute A*
- *|S| – total number of instances in dataset S*
- *v – set of distinct values of an attribute A*
- *H(Sj) – entropy of subset of instances for attribute A*
- *H(A, S) – entropy of an attribute A*

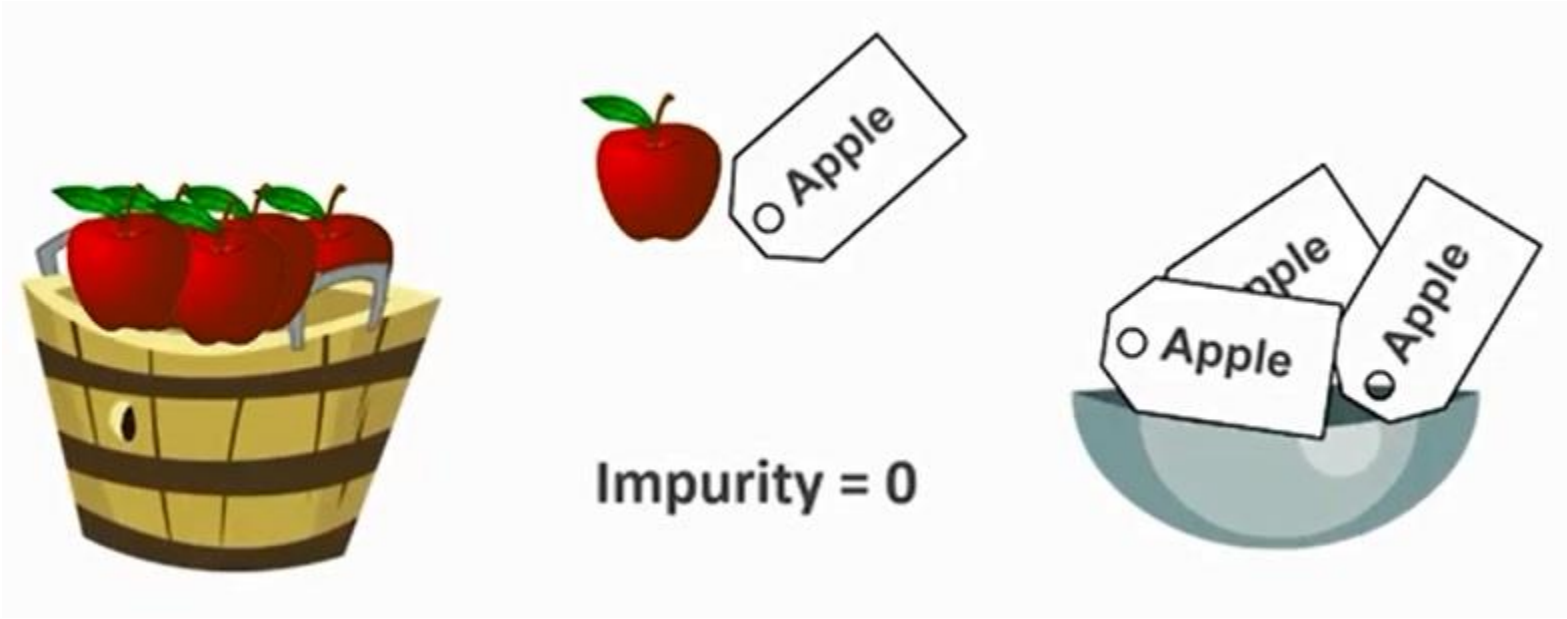
Entropy

Let us first understand What is impurity



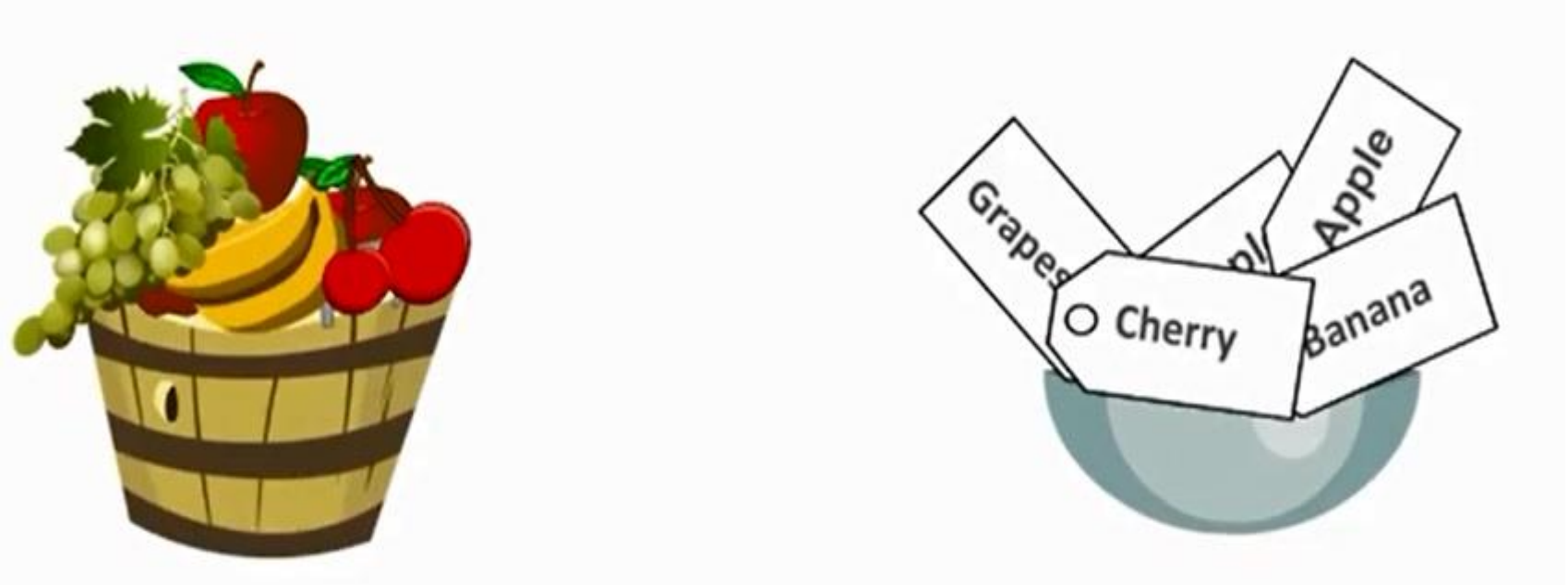
Entropy

Let us first understand What is impurity



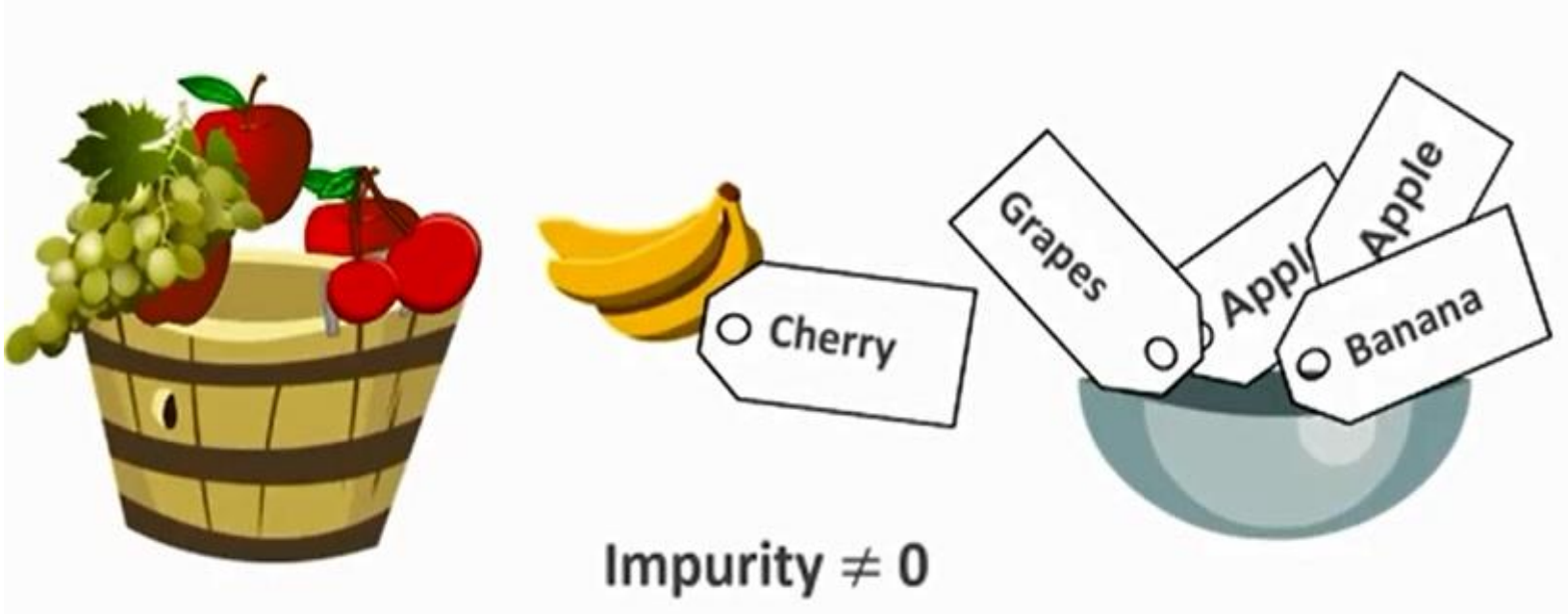
Entropy

Let us first understand What is impurity



Entropy

Let us first understand What is impurity

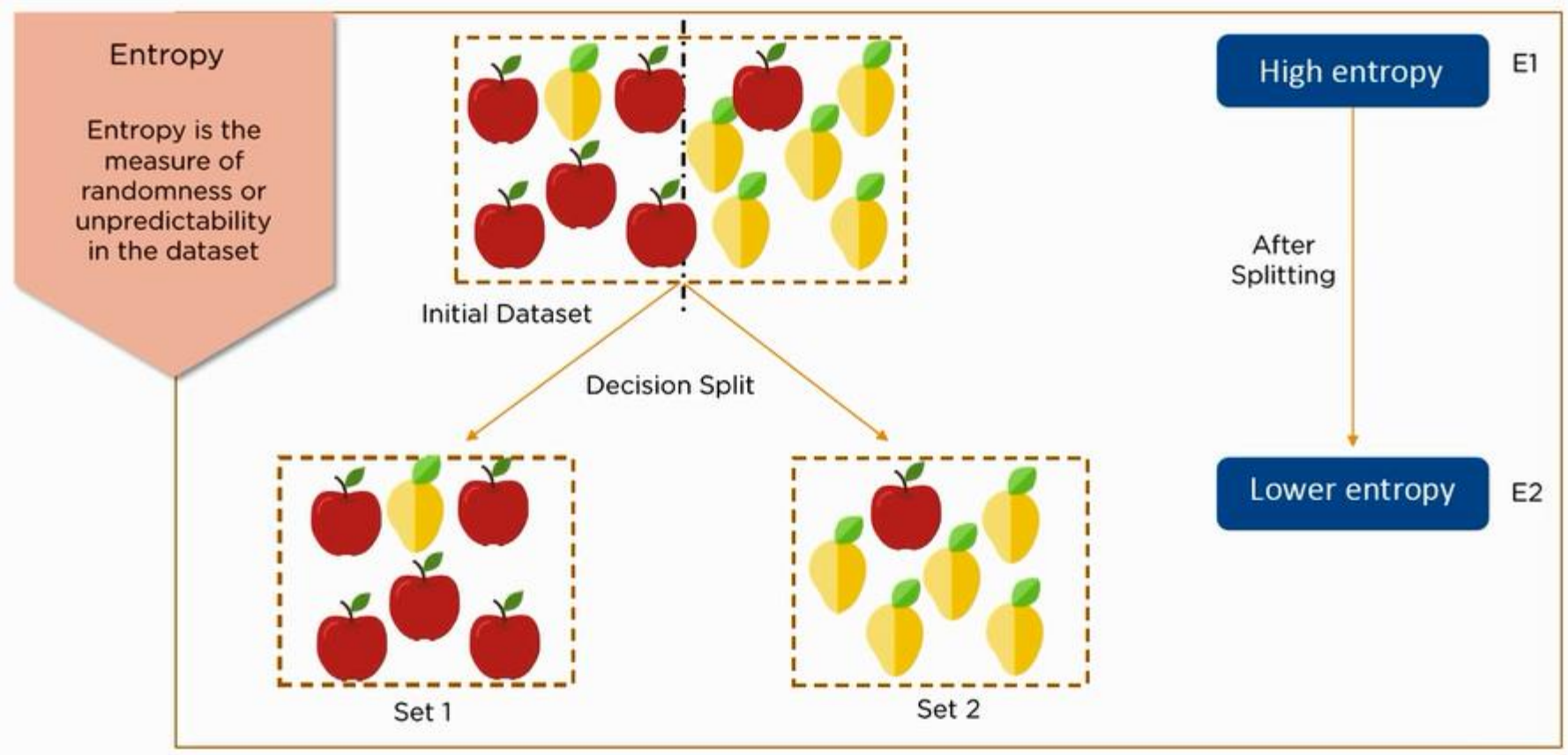


What is Entropy

Defines randomness in the data

Entropy is just a metric which measures the impurity or

The first step to solve the problem of a Decision Tree.



Entropy Coding-Example

$\log_2 20 = \log_{10} 20/\log_{10} 2 = \log_{10} 20/.3010$

Example 7.3 Calculate the entropy for the symbols shown in Table 7.2.

Table 7.2 Symbols and their distribution

Symbol	1	2	3	4	5	6
Probability	0.4	0.2	0.2	0.1	0.05	0.05

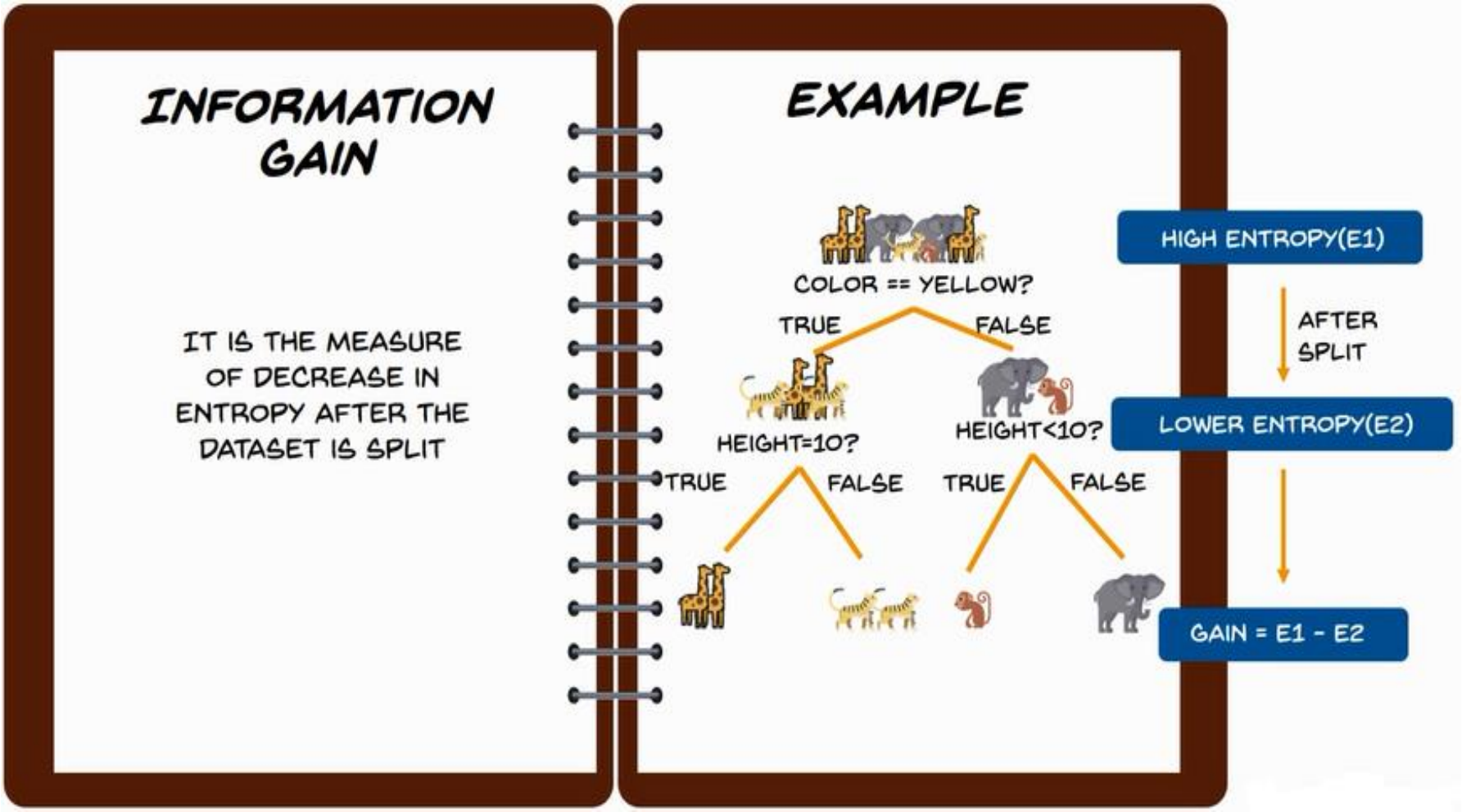
Solution Entropy $= -\Sigma p_i \times \log_2 p_i$; as $\log_2 x = \log_{10} x/\log_{10} 2$

$= -[0.4 \times (\log10(0.4)/\log10(2)) + 0.2 \times (\log10(0.2)/\log10(2)) + 0.2 \times (\log10(0.2)/\log10(2)) + 0.1 \times (\log10(0.1)/\log10(2)) + 0.05 \times (\log10(0.05)/\log10(2)) + 0.05 \times (\log10(0.05)/\log10(2))]$

$= -[-0.5288 - 0.4644 - 0.4644 - 0.3322 - 0.2161 - 0.2161]$

$= 2.22$

Important Terms of Decision Tree

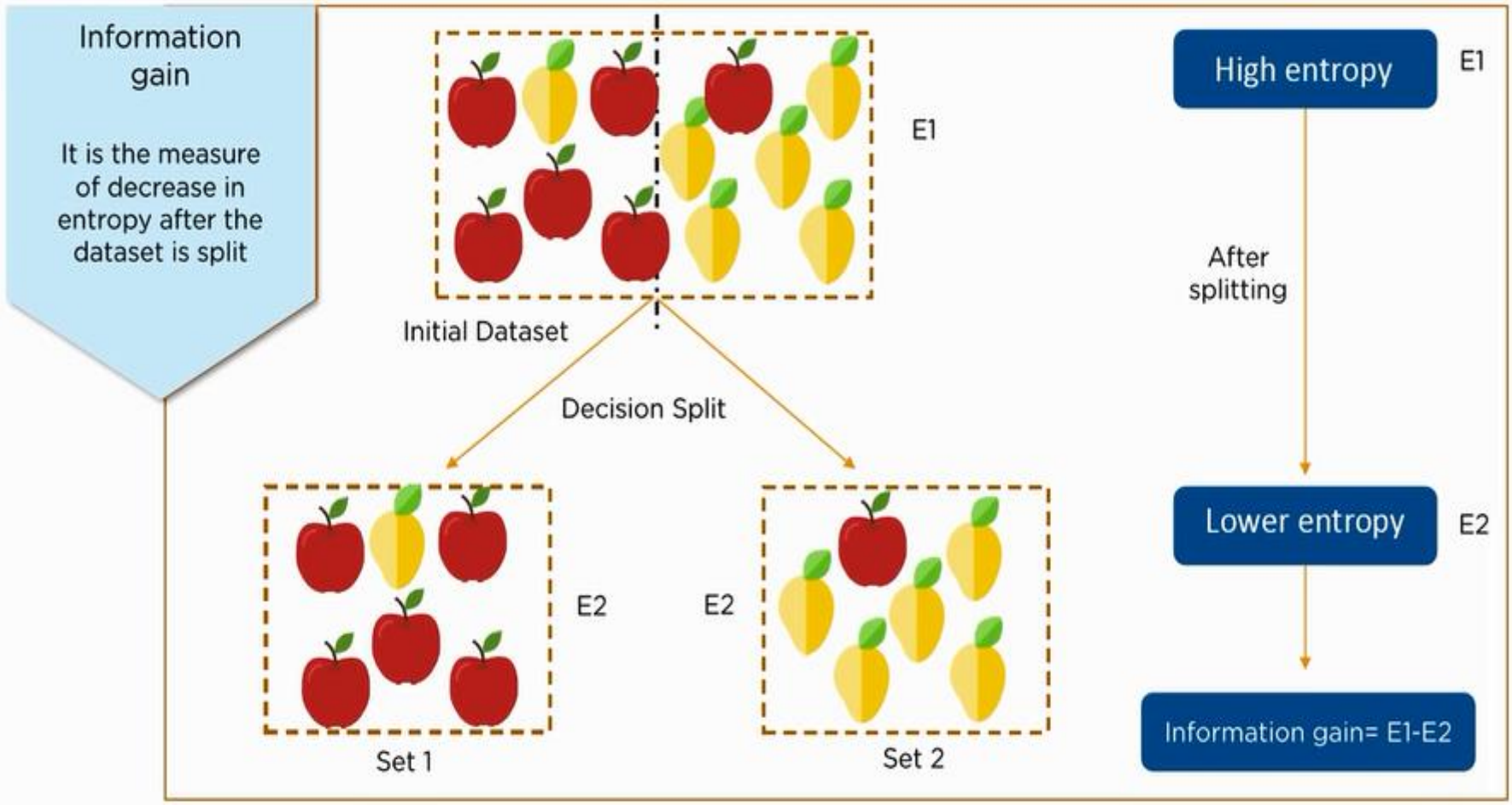


What is Information Gain

Measures the reduction in Entropy
Decides which attribute should be selected as the Decision Node

If S is our total Collection
Information Gain =Entropy(S)- [(Weighted Average) X Entropy (Each feature)]

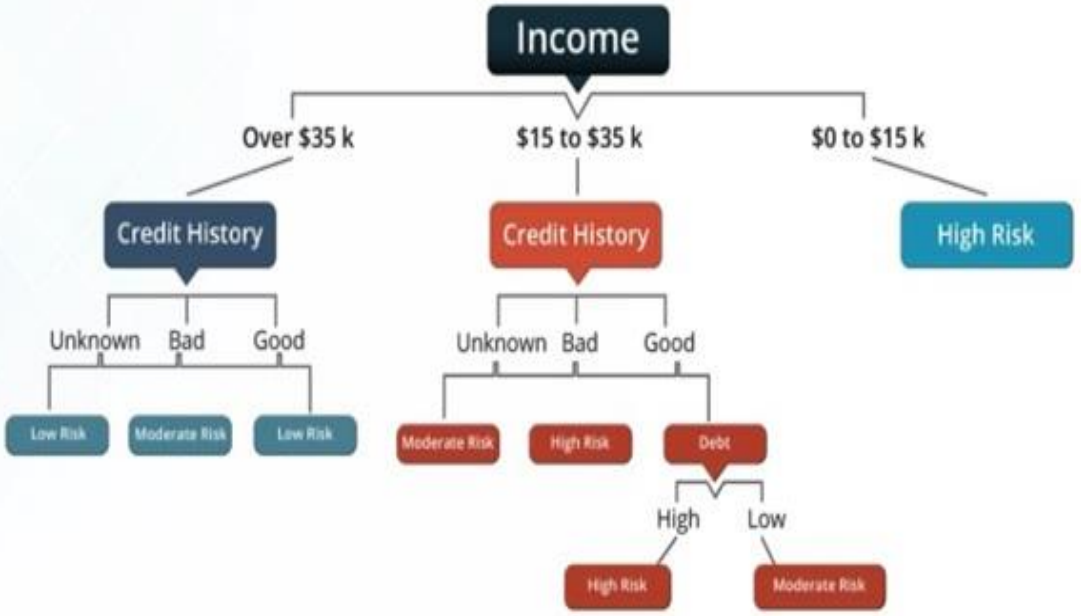
Information Gain



What is Information Gain

Use case-Credit Risk Analysis

- To minimize loss, the bank needs a decision rule to predict whom to give approval of the loan.
- An applicant's demographic (income, debts, credit history) and socio-economic profiles are considered.
- Data science can help banks recognize behavior patterns and provide a complete view of individual customers.



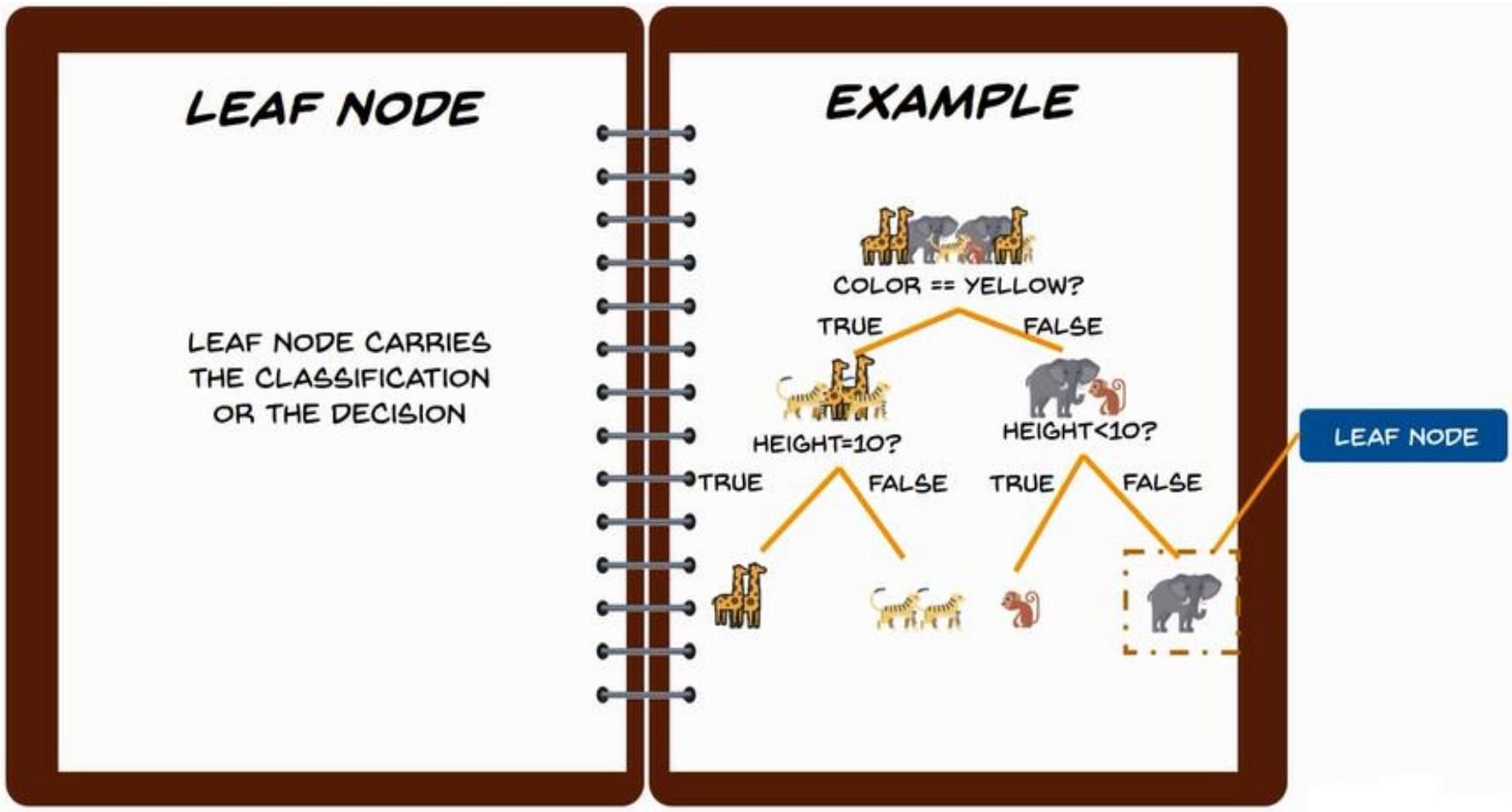
What is Information Gain

Use case-Credit Risk Analysis

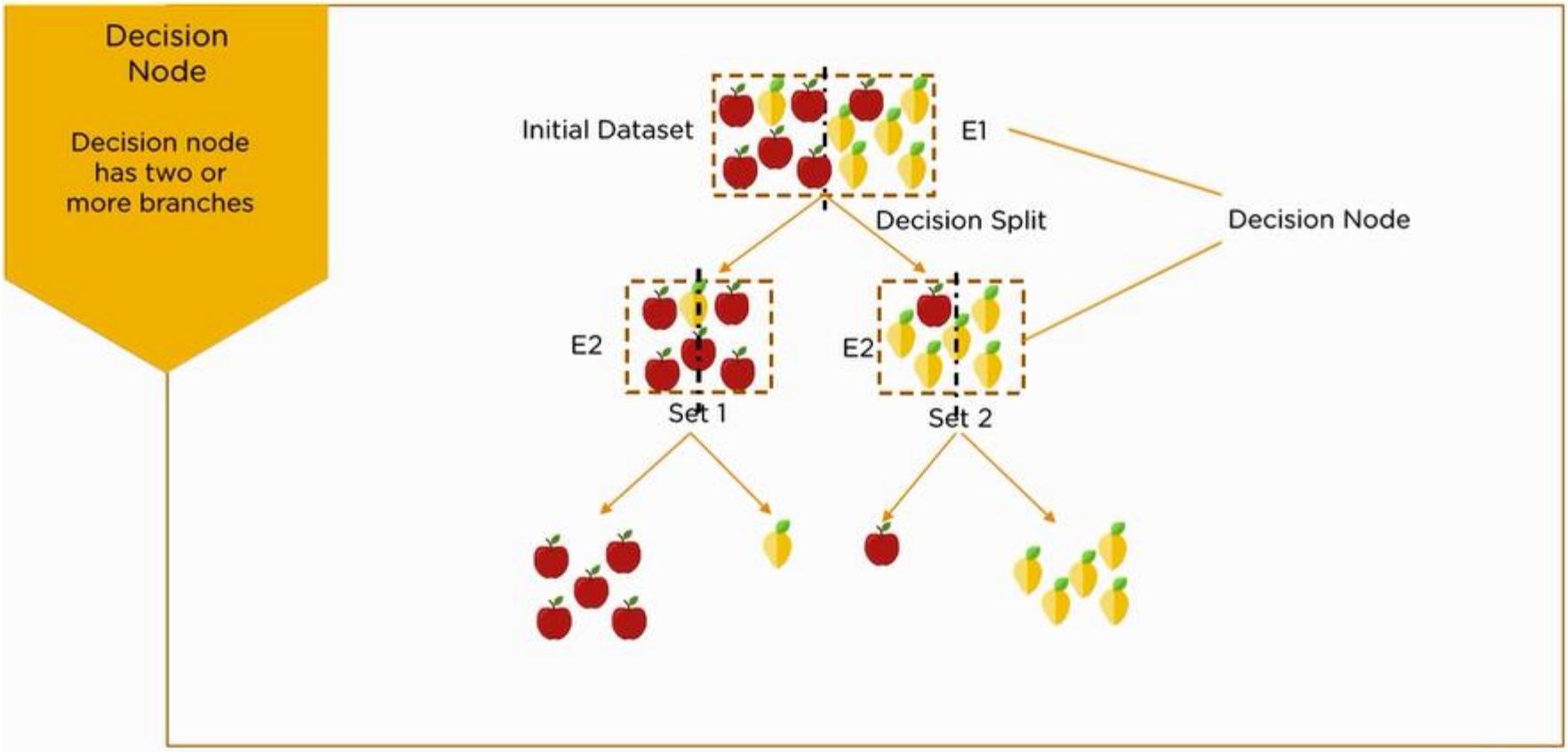
- To minimize loss, the bank needs a decision rule to predict whom to give approval of the loan.
- An applicant's demographic (income, debts, credit history) and socio-economic profiles are considered.
- Data science can help banks recognize behavior patterns and provide a complete view of individual customers.

Variable	Measurement
Marital Status	Married, Not Married
Gender	Male, Female
Age	Varied
Status	Default, Non Default
Time of Payment	Varied
Employment	Employed, Unemployed
Homeownership	With Home, Without Home
Education Level	Secondary and above, Below secondary

Important Terms of Decision Tree-Leaf Node



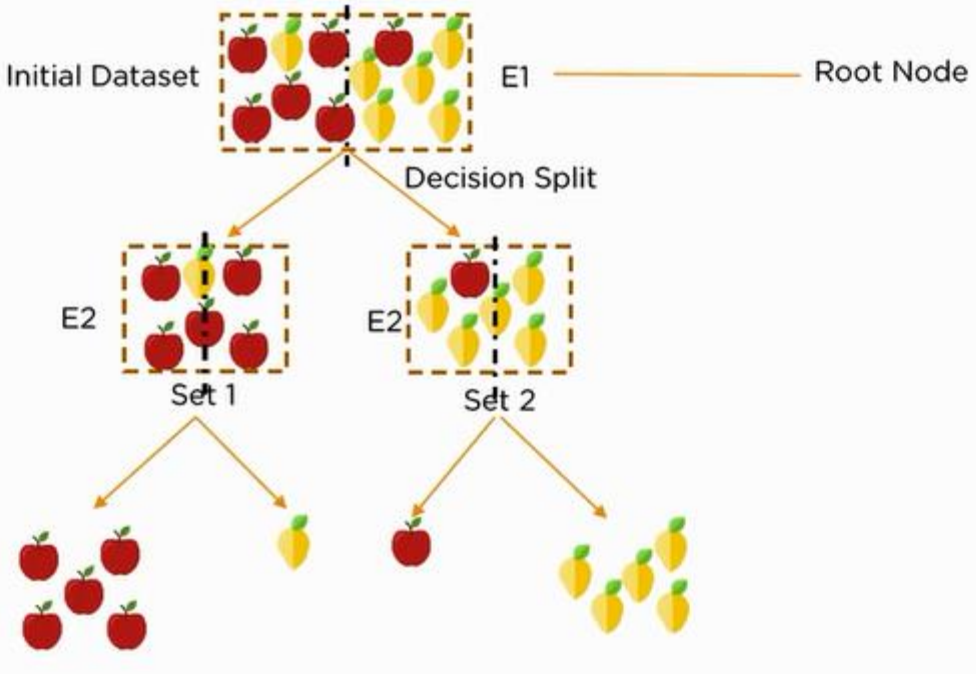
Important Terms of Decision Tree-Decision Node



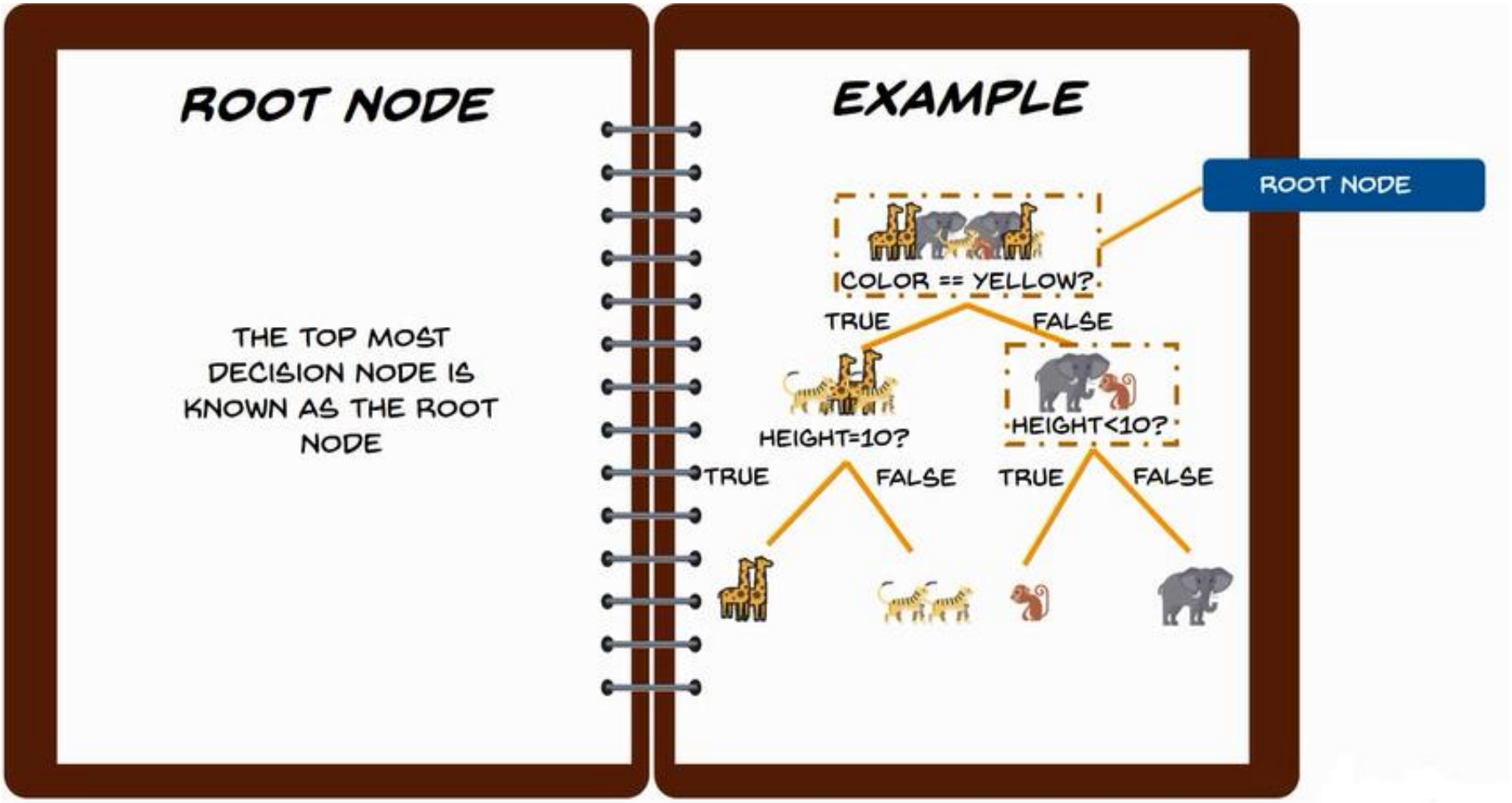
Important Terms of Decision Tree-Root Node

Root Node

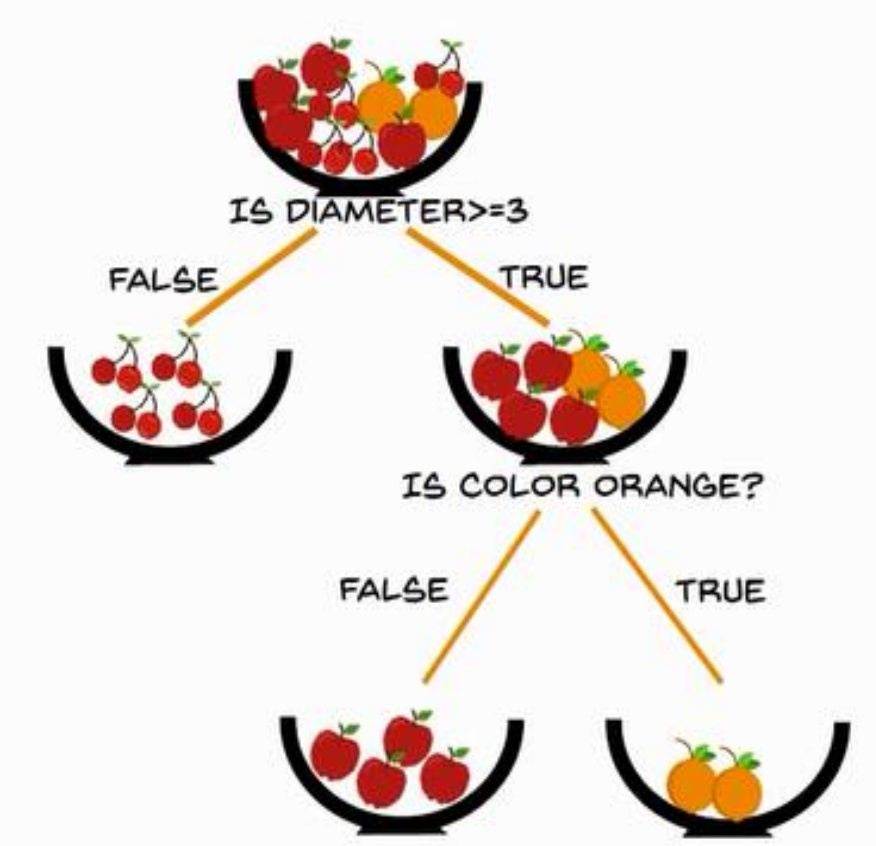
The top most Decision node is known as the Root node



Important Terms of Decision Tree-Root Node



Decision Tree-Example



Example-How to decide Root node and other nodes

Consider whether a dataset based on which we will determine **whether to play football or not.**

Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

There are for independent variables to determine the dependent variable. The **independent variables are Outlook, Temperature, Humidity, and Wind.** The **dependent variable is whether to play football or not.**

How to decide Root node and other nodes

Find the entropy of the class variable.

E(S) = -[(9/14)log(9/14) + (5/14)log(5/14)] = 0.94

Here total there are 14 yes/no. Out of which 9 yes and 5 no. Based on it we calculated probability above.

From the above data for **outlook** we can arrive at the following table easily

		play		
		yes	no	total
Outlook	sunny	2	3	5
	overcast	4	0	4
	rainy	3	2	5
				14

Now we have to calculate average weighted entropy. ie, we have found the total of weights of each feature multiplied by probabilities.

E(S, outlook) = (5/14)*E(2,3) + (4/14)*E(4,0) + (5/14)*E(3,2)

= (5/14)(-(3/5)log(3/5)-(2/5)log(2/5))+ (4/14)(0) + (5/14)(-(2/5)log(2/5)-(3/5)log(3/5))

= 0.693

How to decide Root node and other nodes

The next step is to find the information gain. It is the difference between parent entropy and average weighted entropy we found above.

IG(S, outlook) = 0.94 - 0.693 = 0.247

Similarly find Information gain for Temperature, Humidity, and Windy.

IG(S, Temperature) = 0.940 - 0.911 = 0.029

IG(S, Humidity) = 0.940 - 0.788 = 0.152

IG(S, Windy) = 0.940 - 0.8932 = 0.048

Now select the feature having the largest entropy gain. Here it is Outlook. So, it forms the first node(root node) of our decision tree.

How to decide Root node and other nodes

Now our data look as follows

Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes

Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Overcast	Hot	High	Weak	Yes
Overcast	Cool	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes

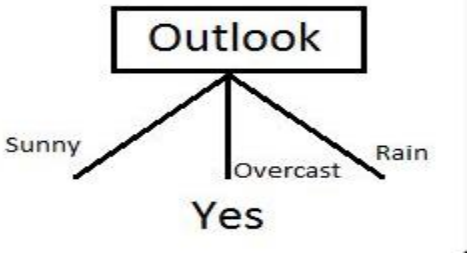
Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Rain	Mild	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Since **overcast contains only examples of class 'Yes'** we can set it as **yes**. **That means**

If outlook is overcast football will be played.

How to decide Root node and other nodes

Since overcast contains only examples of class 'Yes' we can set it as yes. That means If outlook is overcast football will be played. Now our decision tree looks as follows.



The next step is to find the next node in our decision tree. Now we will find one under sunny. We have to determine which of the following Temperature, Humidity or Wind has higher information gain.

Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes

Calculate parent entropy E(sunny)

E(sunny) = (-(3/5)log(3/5)-(2/5)log(2/5)) = 0.971.

How to decide Root node and other nodes

Now **Calculate the information gain of Temperature.** IG(sunny, Temperature)

		play		
		yes	no	total
Temperature	hot	0	2	2
	cool	1	0	2
	mild	1	1	1
				5

E(sunny, Temperature) = (2/5)*E(0,2) + (2/5)*E(1,1) + (1/5)*E(1,0)=2/5=0.4

Now calculate information gain.

IG(sunny, Temperature) = 0.971–0.4 =0.571, **Similarly we get**

IG(sunny, Humidity) = **0.971**

IG(sunny, Windy) = 0.020

Here, **IG(sunny, Humidity) is the largest value. So, Humidity is the node that comes under sunny.**

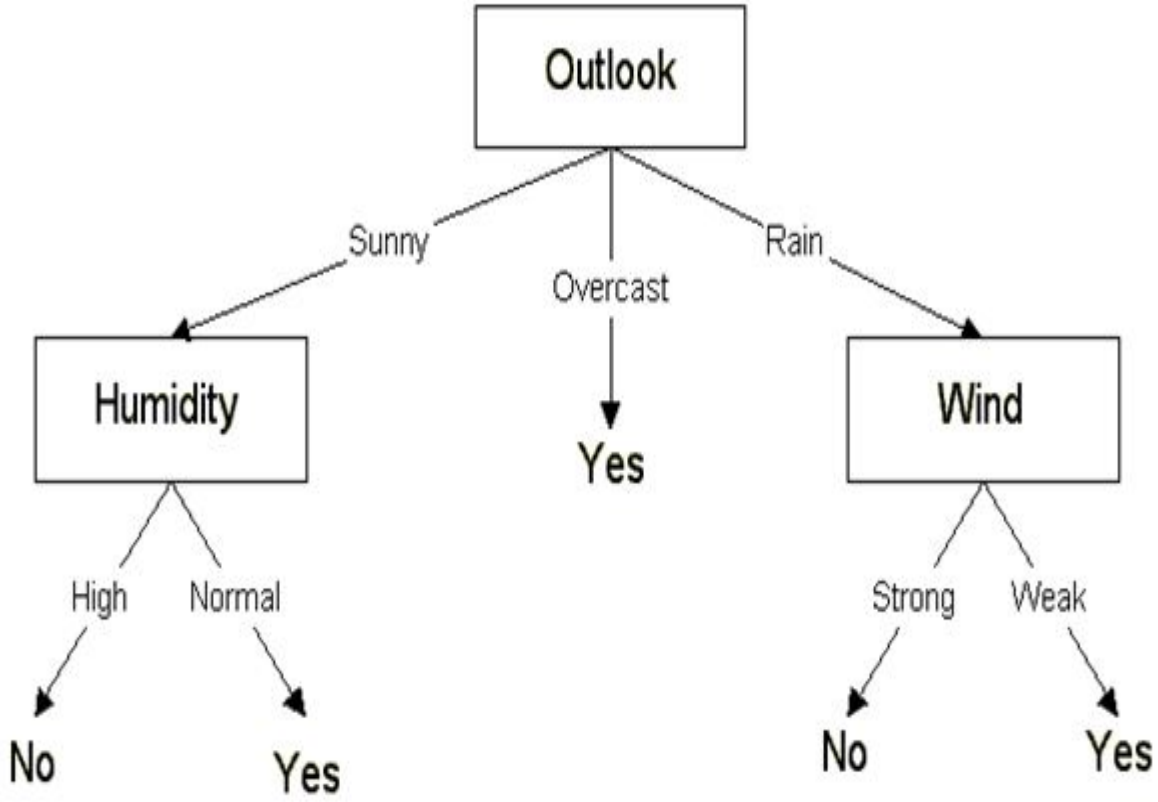
	play	
Humidity	yes	no
high	0	3
normal	2	0

For humidity from the above table, we can say that **play will occur if humidity is normal** and **will not occur if it is high. Similarly, find the nodes under rainy.**

Note: A branch with entropy more than 0 needs further splitting.

How to decide Root node and other nodes

Finally, our decision tree will look as below:



Classification using CART algorithm is similar to it. But, instead of Entropy, we use **Gini Impurity.**

Decision Tree Example

Problem Statement

To classify the different types of Fruits in the Bowl Based on Different Features

Decision Tree Example

The Datasets (Bowl) is looking quite Messy and the Entropy is very high in this case

Decision Tree Example

Training Dataset

Colour	Diameter	Label
Red	3	Apple
Yellow	3	Lemon
Purple	1	Grapes
Red	3	Apple
Yellow	3	Lemon
Purple	1	Grapes

Decision Tree Example

How to Split the data

We have to frame the Conditions that Split the data in such a way that the **Information Gain is Highest.** **Gain** is the measure of decrease in Entropy after Splitting

- Now we will try to choose a condition that gives us the Highest Gain.
- We will do that by Splitting the data using each condition and checking the gain that we get out them.

Decision Tree Example

Training Dataset

Colour	Diameter	Label
Red	3	Apple
Yellow	3	Lemon
Purple	1	Grapes
Red	3	Apple
Yellow	3	Lemon
Purple	1	Grapes

Conditions

Colour Purple ?

Diameter 1

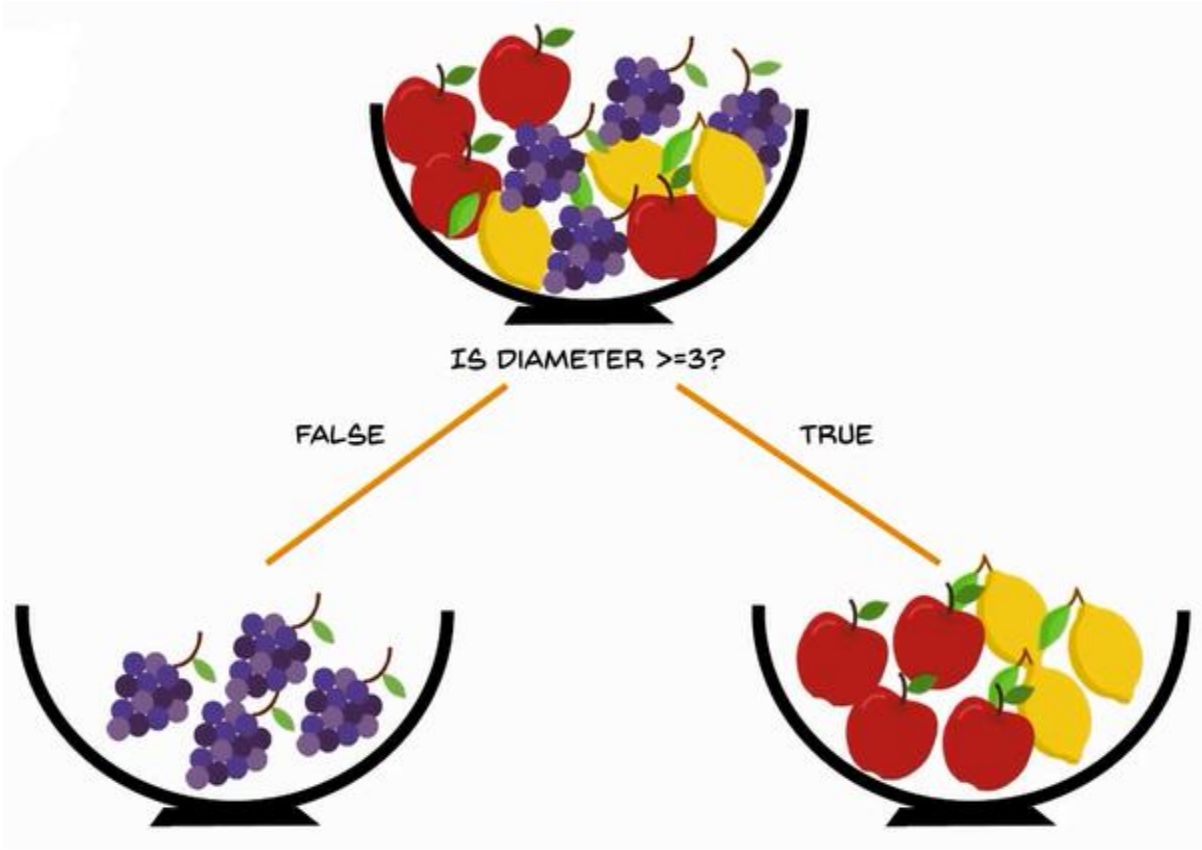
Colour Red ?

Colour Yellow?

Diameter 3

Decision Tree Example

The Entropy after Splitting has decreased considerably

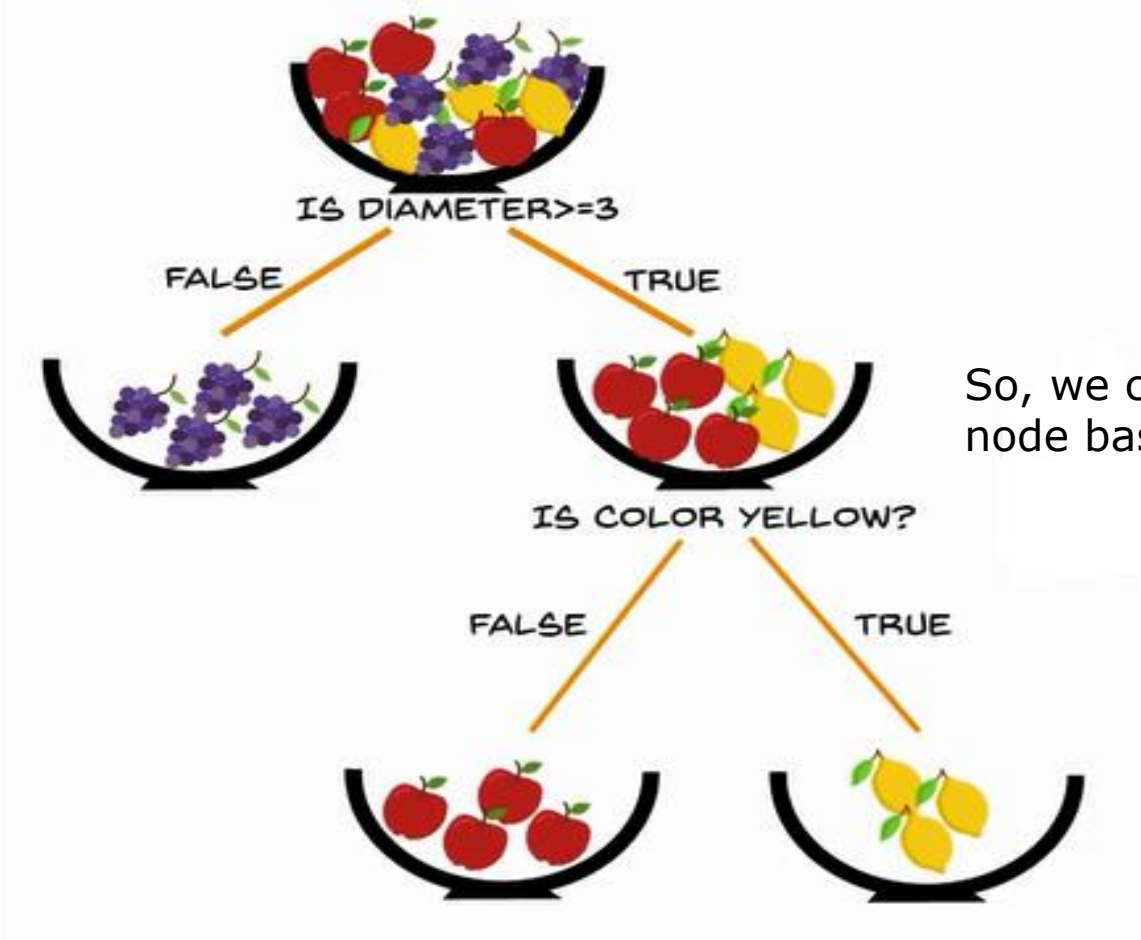


So No further Splitting is required for this node

However this node still requires a split to decrease the entropy further

Decision Tree Example

The Entropy after Splitting has decreased considerably



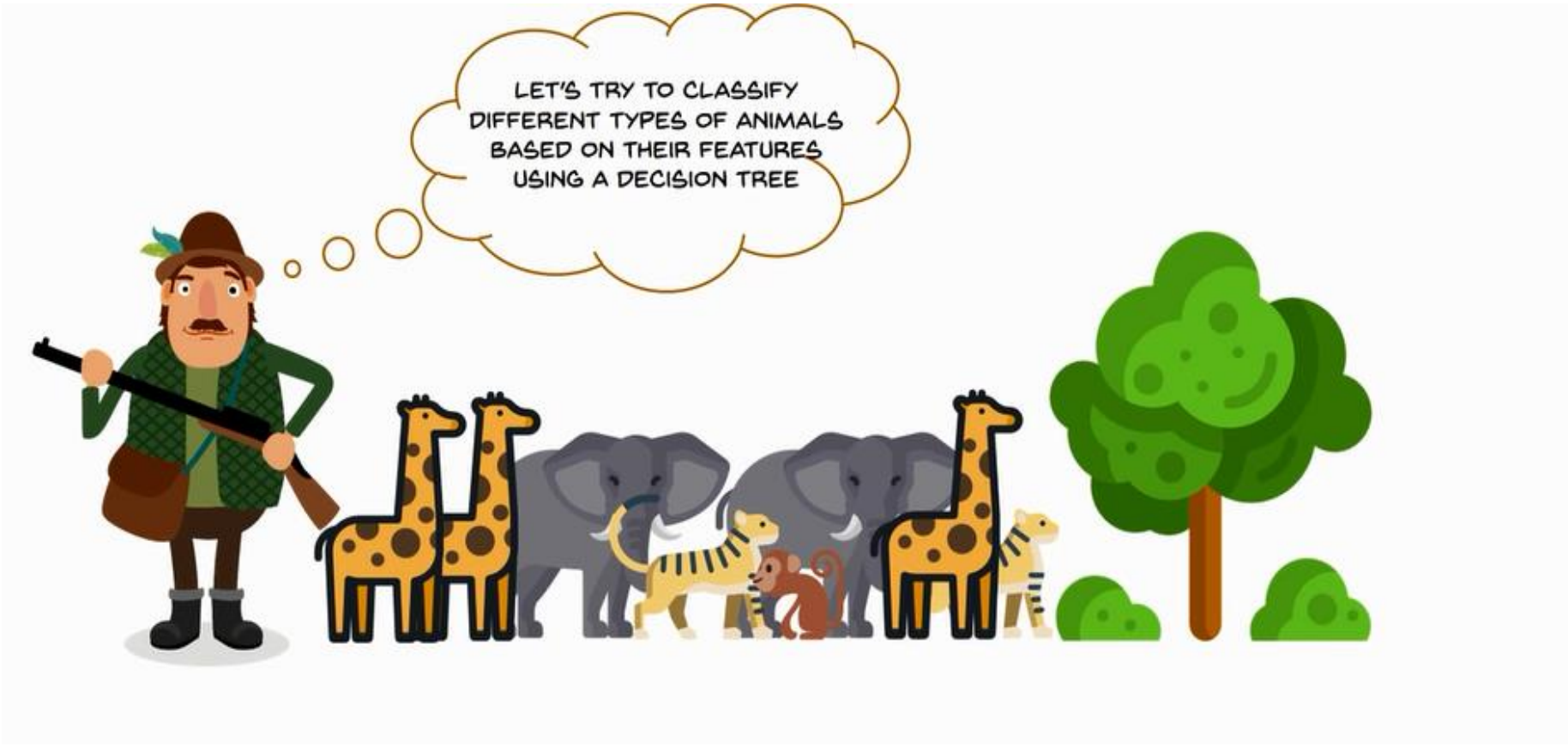
So, we can Split the right node based on Colour

Now the Entropy in this case is Zero

We can predict Apple, Lemon and Grapes with 100% Accuracy

Decision Tree Example

Decision Tree Example



Decision Tree Example

PROBLEM STATEMENT

TO CLASSIFY THE DIFFERENT TYPES OF ANIMALS BASED ON THEIR FEATURES USING DECISION TREE

THE DATASET IS LOOKING QUITE MESSY AND THE ENTROPY IS HIGH IN THIS CASE

TRAINING DATASET

COLOR	HEIGHT	LABEL
GREY	10	ELEPHANT
YELLOW	10	GIRAFFE
BROWN	3	MONKEY
GREY	10	ELEPHANT
YELLOW	4	TIGER

Decision Tree Example

HOW TO SPLIT THE DATA

WE HAVE TO FRAME THE CONDITIONS THAT SPLIT THE DATA IN SUCH A WAY THAT THE INFORMATION GAIN IS THE HIGHEST

NOTE

GAIN IS THE MEASURE OF DECREASE IN ENTROPY AFTER SPLITTING

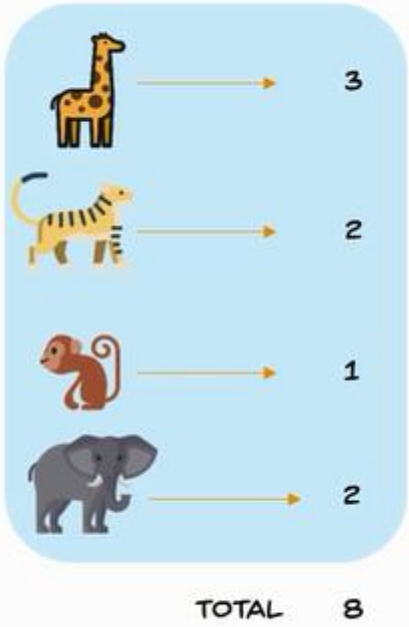
Decision Tree Example

FORMULA FOR ENTROPY

$$\sum_{i=1}^{k} P(valuei).log_2(P(value_i)$$

LET'S TRY TO CALCULATE THE ENTROPY FOR THE CURRENT DATASET

Decision Tree Example



Decision Tree Example

LET'S USE THE FORMULA

$$\sum_{i=1}^{k} P(valuei).log_2(P(value_i))$$

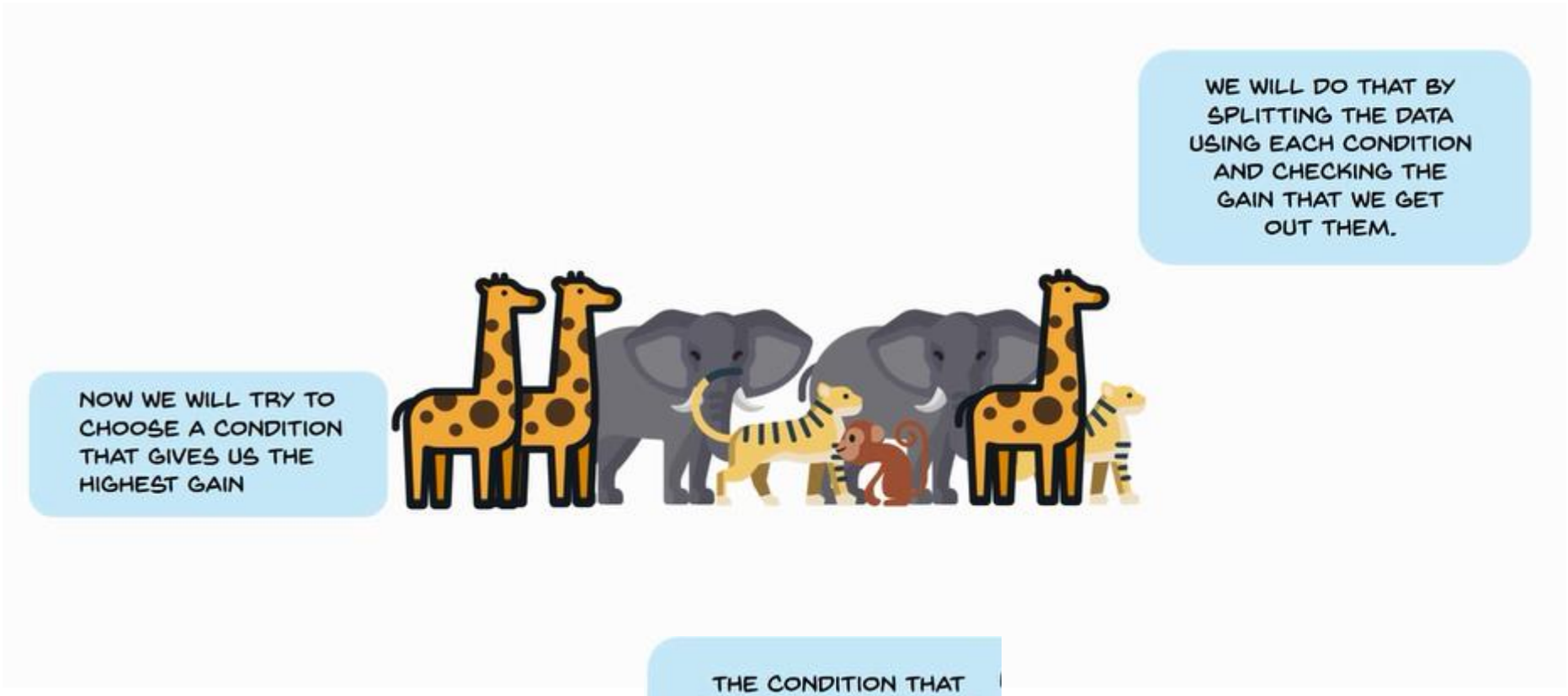
ENTROPY = $(\frac{3}{8})\ log_2(\frac{3}{8}) + (\frac{2}{8})\ log_2(\frac{2}{8}) + (\frac{1}{8})\ log_2(\frac{1}{8}) + (\frac{2}{8})\ log_2(\frac{2}{8})$

ENTROPY=0.571

WE WILL CALCULATE THE ENTROPY OF THE DATASET SIMILARLY AFTER EVERY SPLIT TO CALCULATE THE GAIN

GAIN CAN BE CALCULATED BY FINDING THE DIFFERENCE OF THE SUBSEQUENT ENTROPY VALUES AFTER SPLIT

Decision Tree Example



Decision Tree Example

CONDITIONS

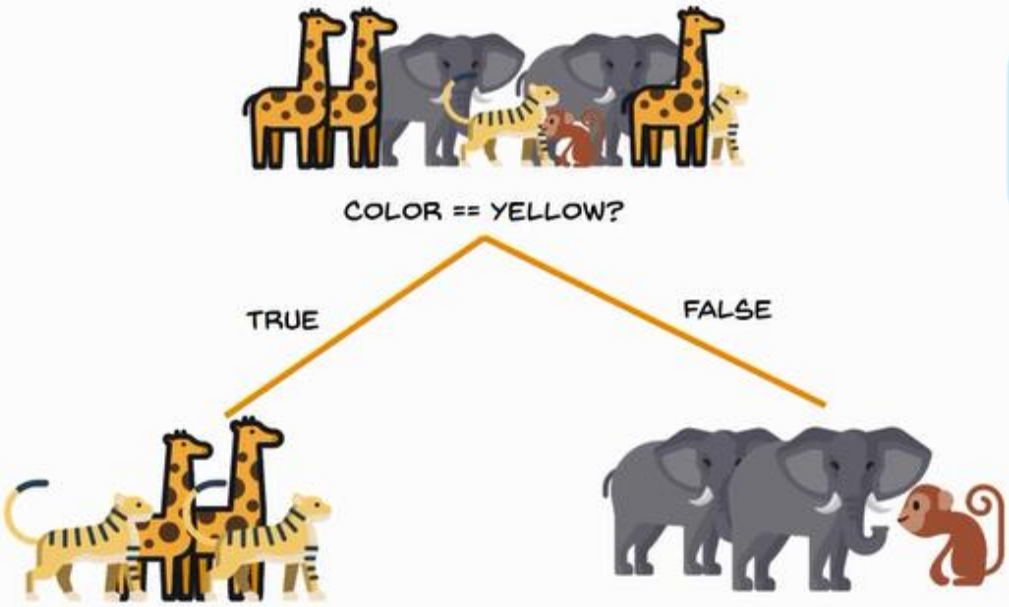
- COLOR== YELLOW?
- HEIGHT>=10
- COLOR== BROWN?
- COLOR==GREY
- DIAMETER<10

TRAINING DATASET

COLOR	HEIGHT	LABEL
GREY	10	ELEPHANT
YELLOW	10	GIRAFFE
BROWN	3	MONKEY
GREY	10	ELEPHANT
YELLOW	4	TIGER

Decision Tree Example

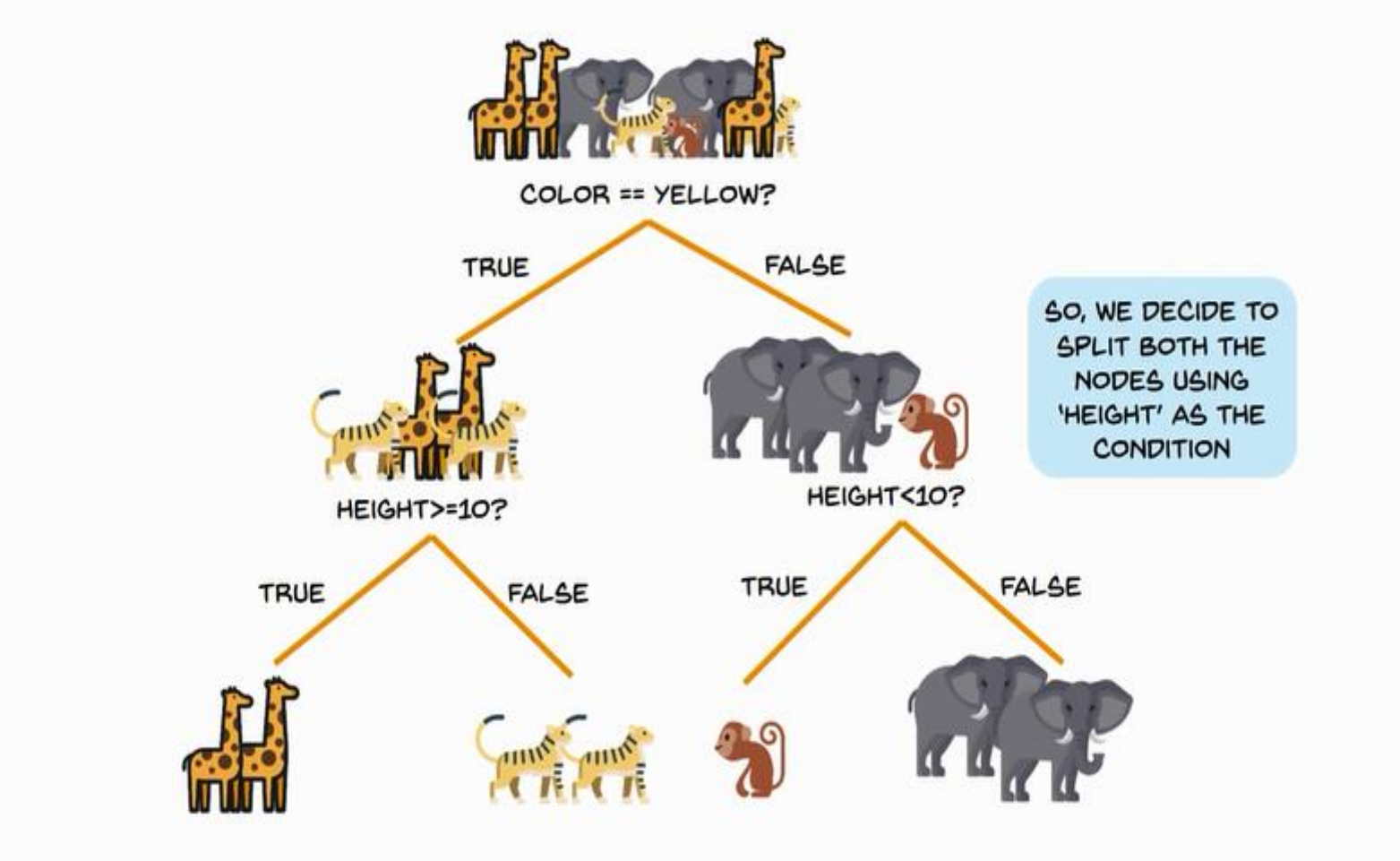
WE SPLIT THE DATA



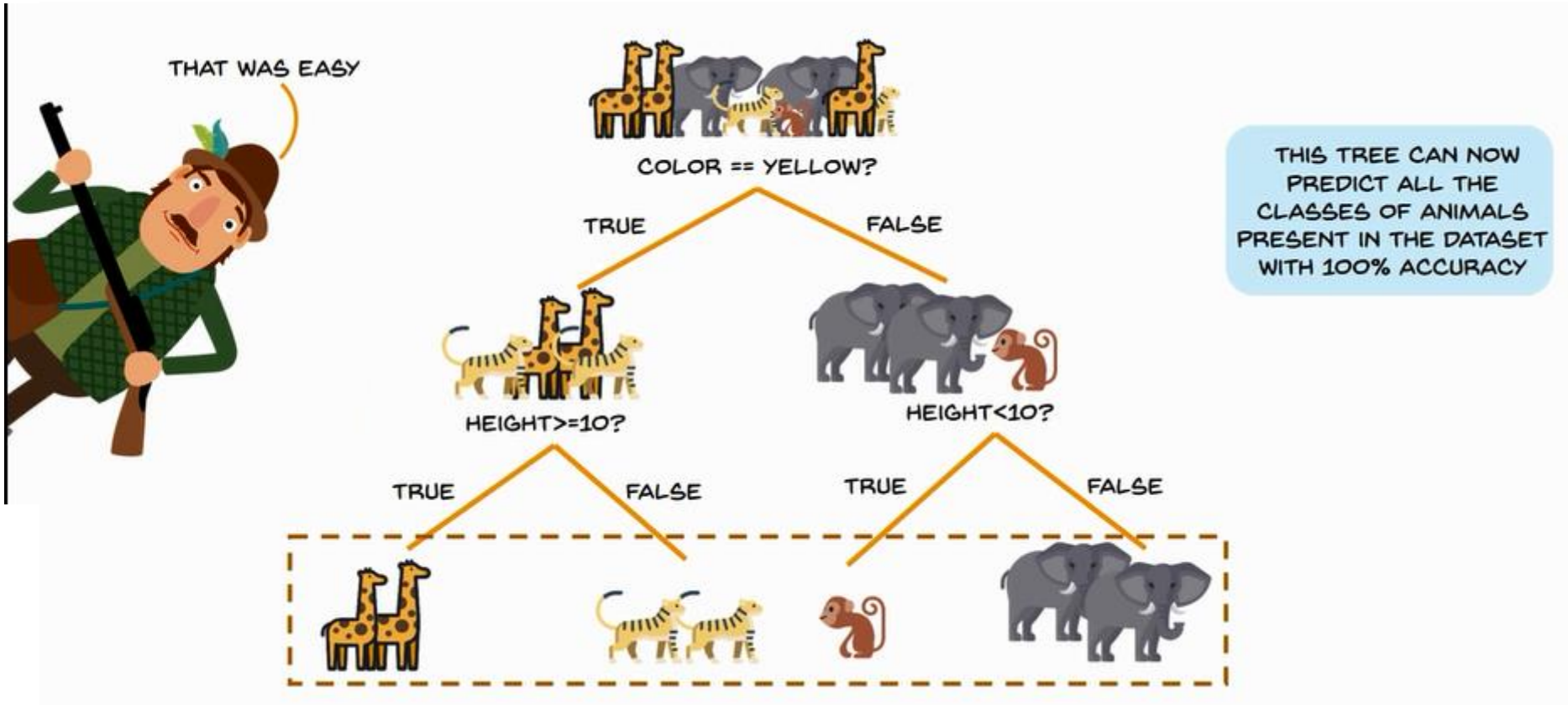
THE ENTROPY AFTER SPLITTING HAS DECREASED CONSIDERABLY

HOWEVER WE STILL NEED SOME SPLITTING AT BOTH THE BRANCHES TO ATTAIN AN ENTROPY VALUE EQUAL TO ZERO

Decision Tree Example



Decision Tree Example



How Does the Decision Tree Algorithm works ?

The basic idea behind any decision tree algorithm is as follows:

1. Select the best Feature using Attribute Selection Measures(ASM) to split the records.

2. Make that attribute/feature a decision node and break the dataset into smaller subsets.

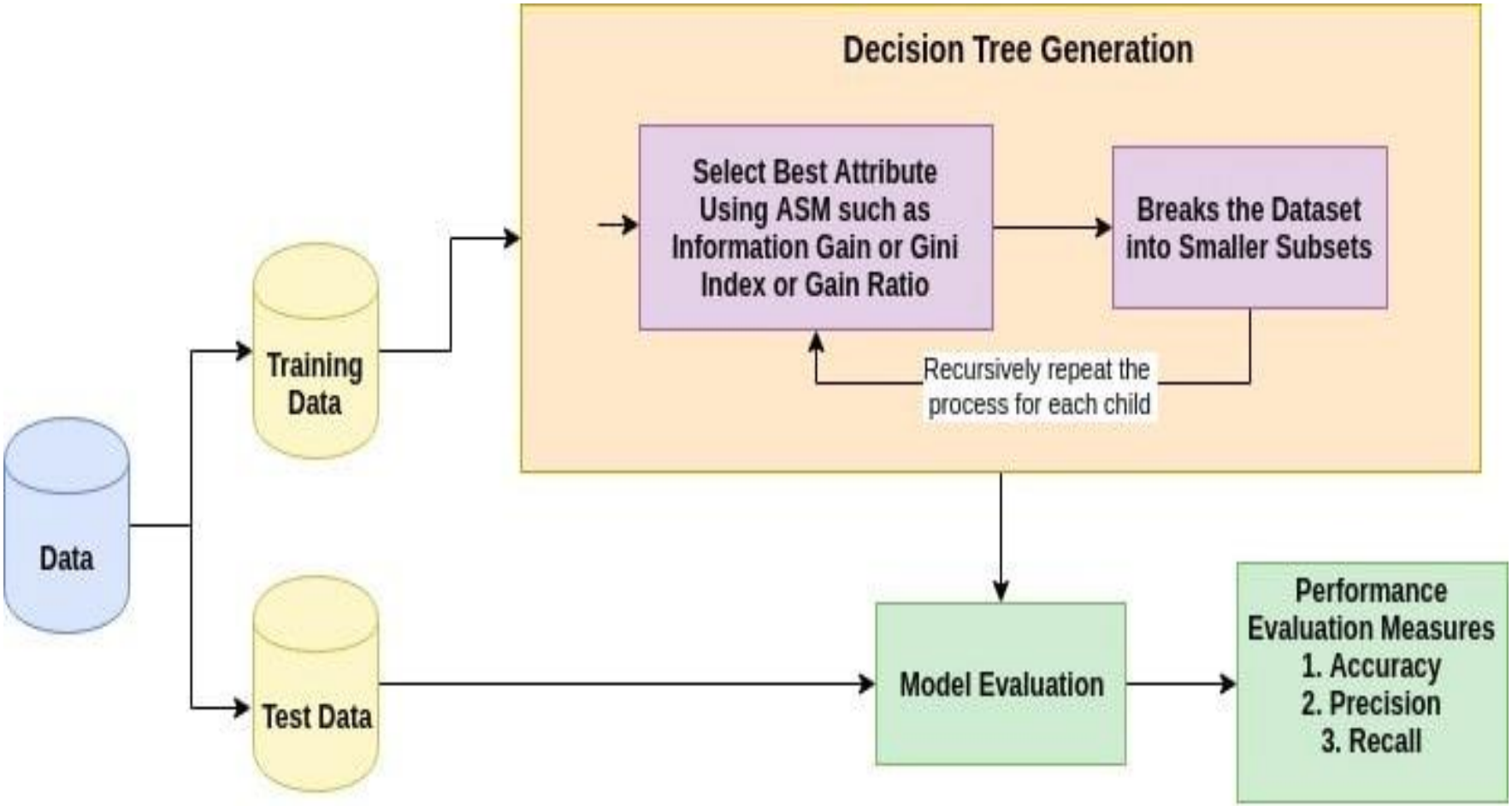
3 Start the tree-building process by repeating this process recursively for each child until one of the following condition is being achieved :

a) All tuples belonging to the same attribute value.

b) There are no more of the attributes remaining.

c) There are no more instances remaining.

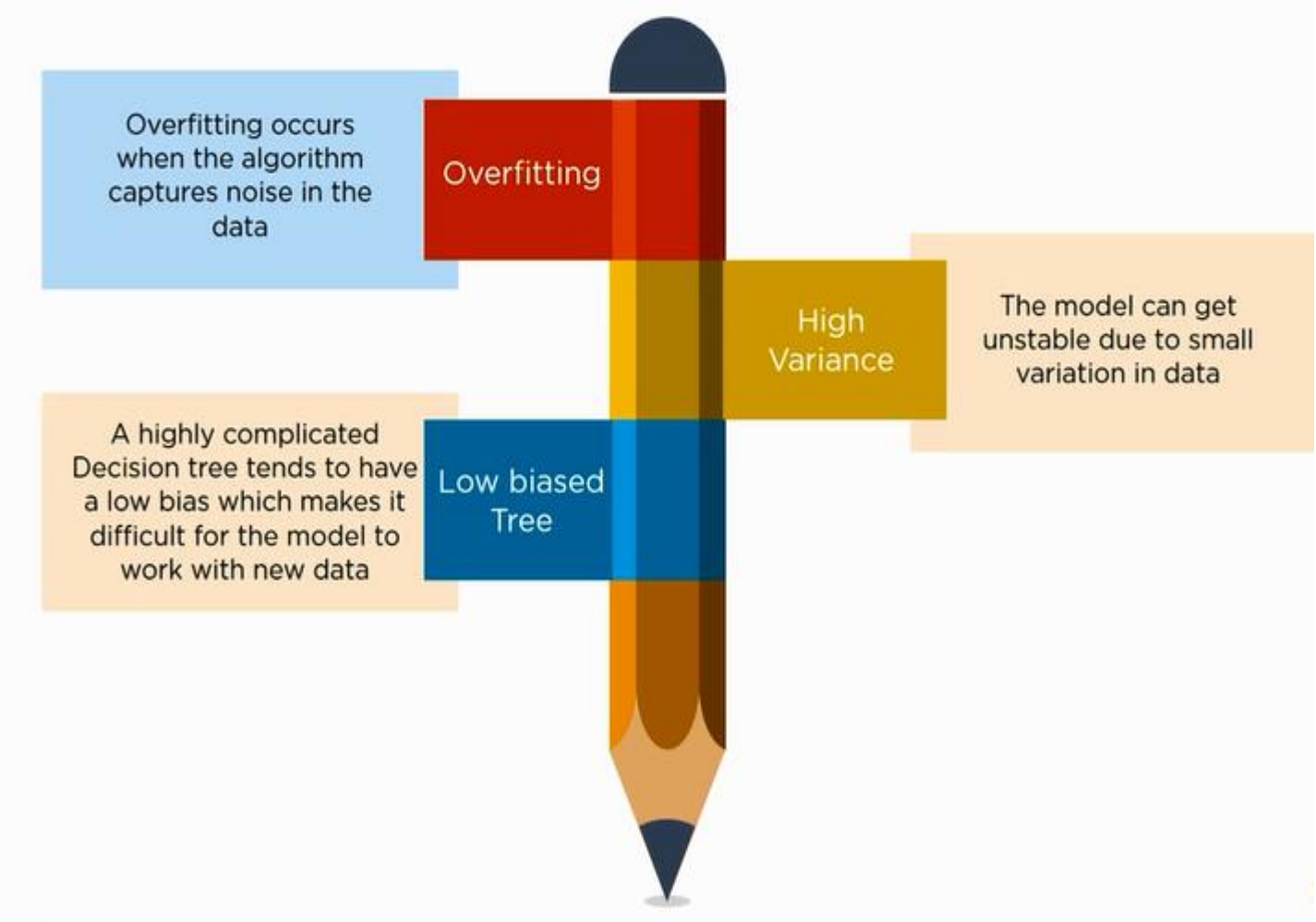
How Does the Decision Tree Algorithm works?



Advantages of Decision Trees

1. It is **simple to implement** and it **follows a flow chart type structure** that resembles human-like decision making.
2. **Little effort required for data preparation**- There is very little need for data cleaning in decision trees compared to other Machine Learning algorithms
3. Can handle **numerical and categorical data.**
4. **Non linear parameters** don't effect its performance.
5. It proves to be **very useful for decision-related problems.**
6. It helps **to find all of the possible outcomes** for a given problem.

Disadvantages of Decision Tree



- The **decision tree contains lots of layers, which makes it complex.**
- For **more class labels, the computational complexity of the decision tree may increase**.

Decision Tree Applications

- Assessing **prospective growth opportunities** for Businesses.
- Serve as a support tool in several fields.
- The use of **a decision tree support tool can help lenders to** evaluate a customer’s credit worthiness to prevent losses.
- Decision trees **can also be used in operations research** in **planning logistics and strategic management.**

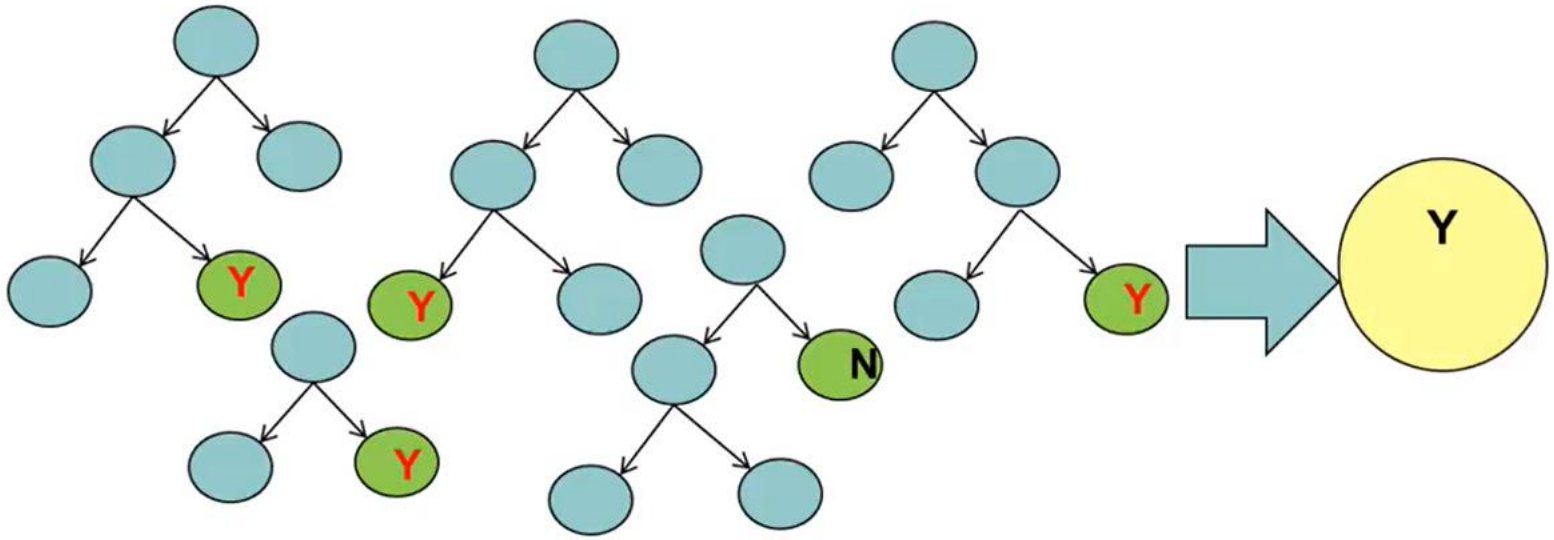
Ensemble Learning

- Ensemble means Group
- In **Ensemble learning**, individual models come together and bring forth a model that is more accurate.
- It consist of two important terms

1. Bagging
2. Boosting

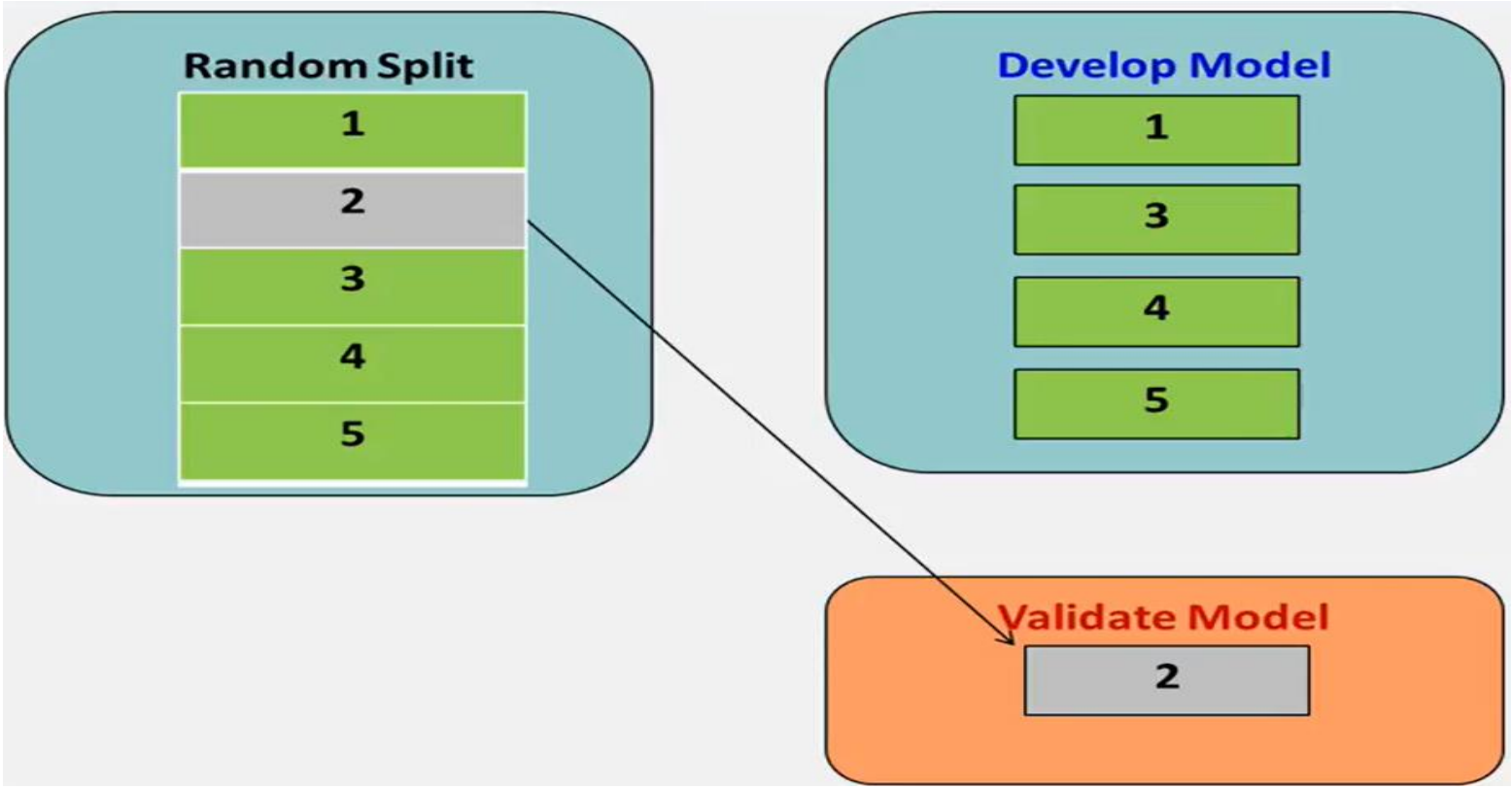
Bagging

- Various models are built in parallel on various samples and then the various models vote to give the final model and hence prediction.

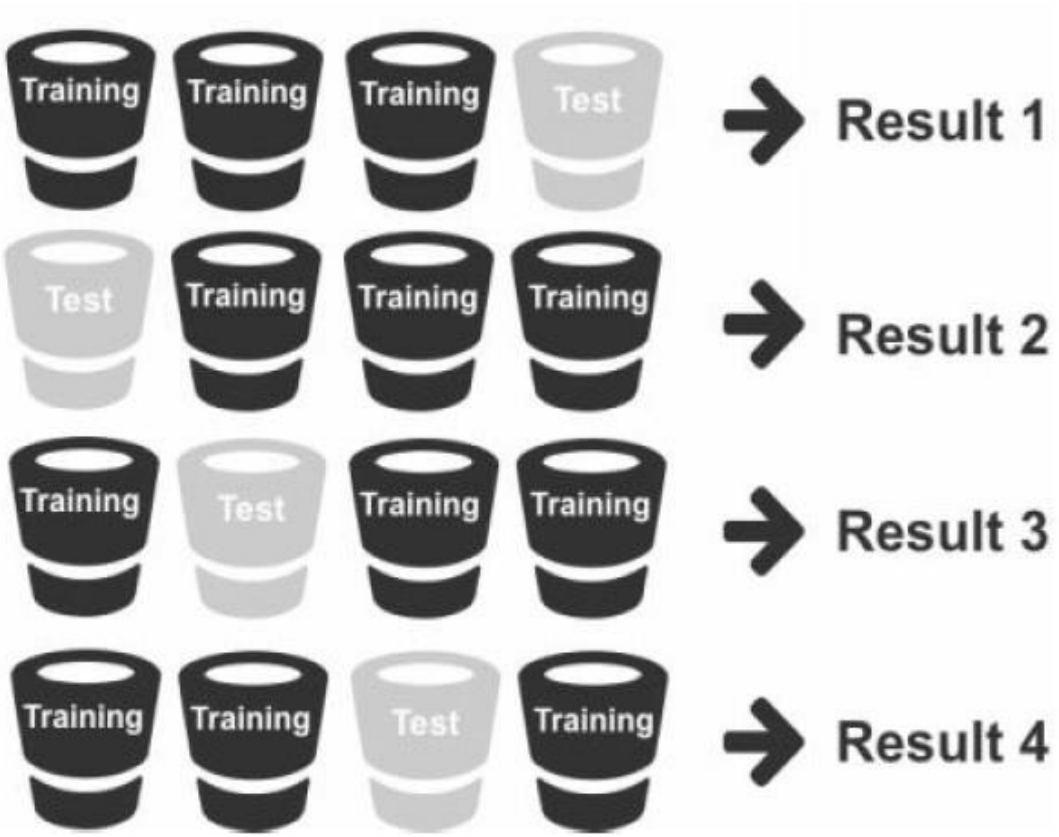


Boosting

- In Boosting, the models are built in series. In each successive model, the weights are adjusted based on the learning of previous model.



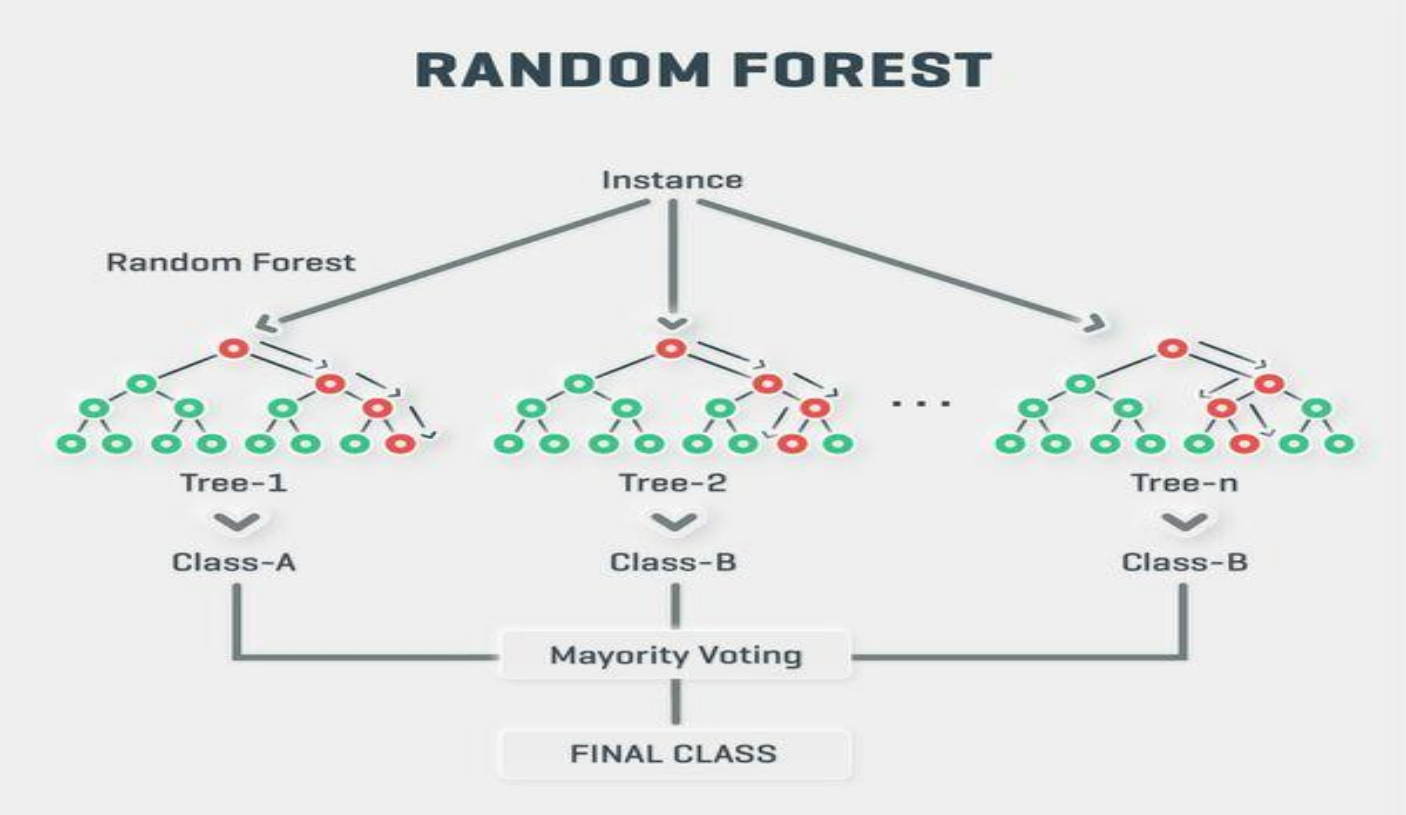
K-fold Validation



- To perform k-fold validation, data are first randomly assigned to k number of equal sized buckets.
- One bucket is then reserved as the test bucket and is used to measure and evaluate the performance of the remaining (k-1) buckets.

Bagging-Random Forest

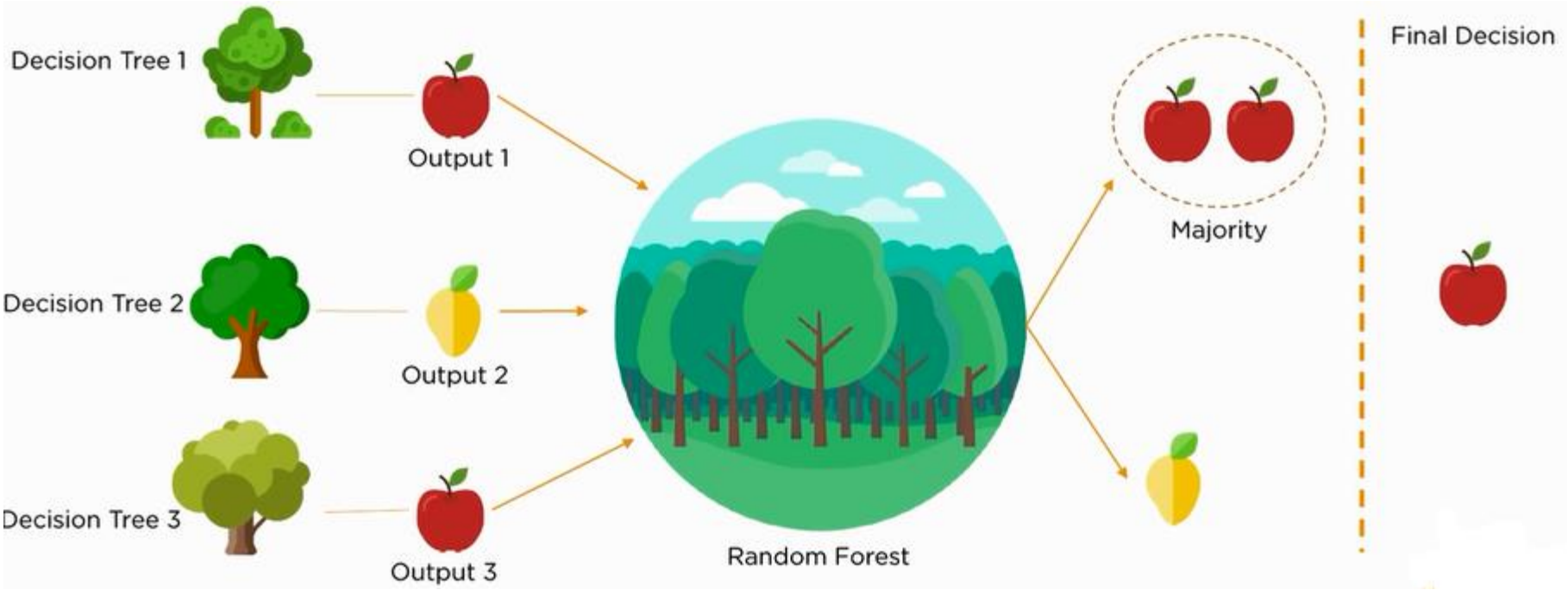
- Build **multiple decision trees** and **merges them together.**
- More **accurate and stable prediction.**
- Random Decision Forest correct for decision trees' habit of **overfitting** to their training set.
- Trained with the "bagging" method



Random Forest

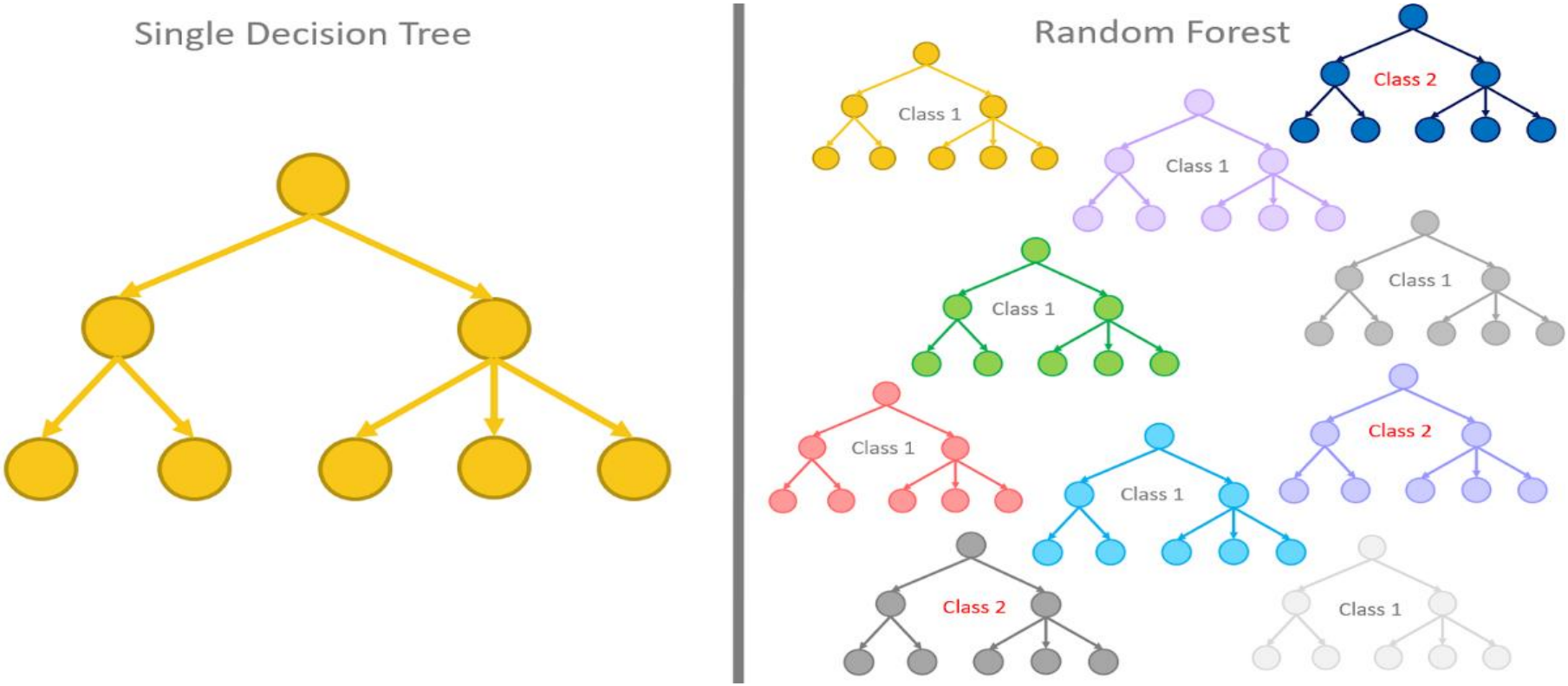
- Random Forest or Random Decision Forest is a method that operates by Constructing multiple Decision Trees during training phase.

The **Decision of majority of the majority of Trees** is chosen by the Random Forest as a Final Decision.



Decision Tree and Random Forest

- **Decision trees** are the Machine Learning models used to make predictions by going through each and every feature in the data set, one-by-one.
- **Random forests** on the other hand are a collection of decision trees being grouped together and trained together that use random orders of the features in the given data sets.



Why Random Forest

No overfitting

Use of multiple trees reduce the risk of overfitting

Training time is less

High accuracy

Runs efficiently on large database

For large data, it produces highly accurate predictions

Estimates missing data

Random Forest can maintain accuracy when a large proportion of data is missing

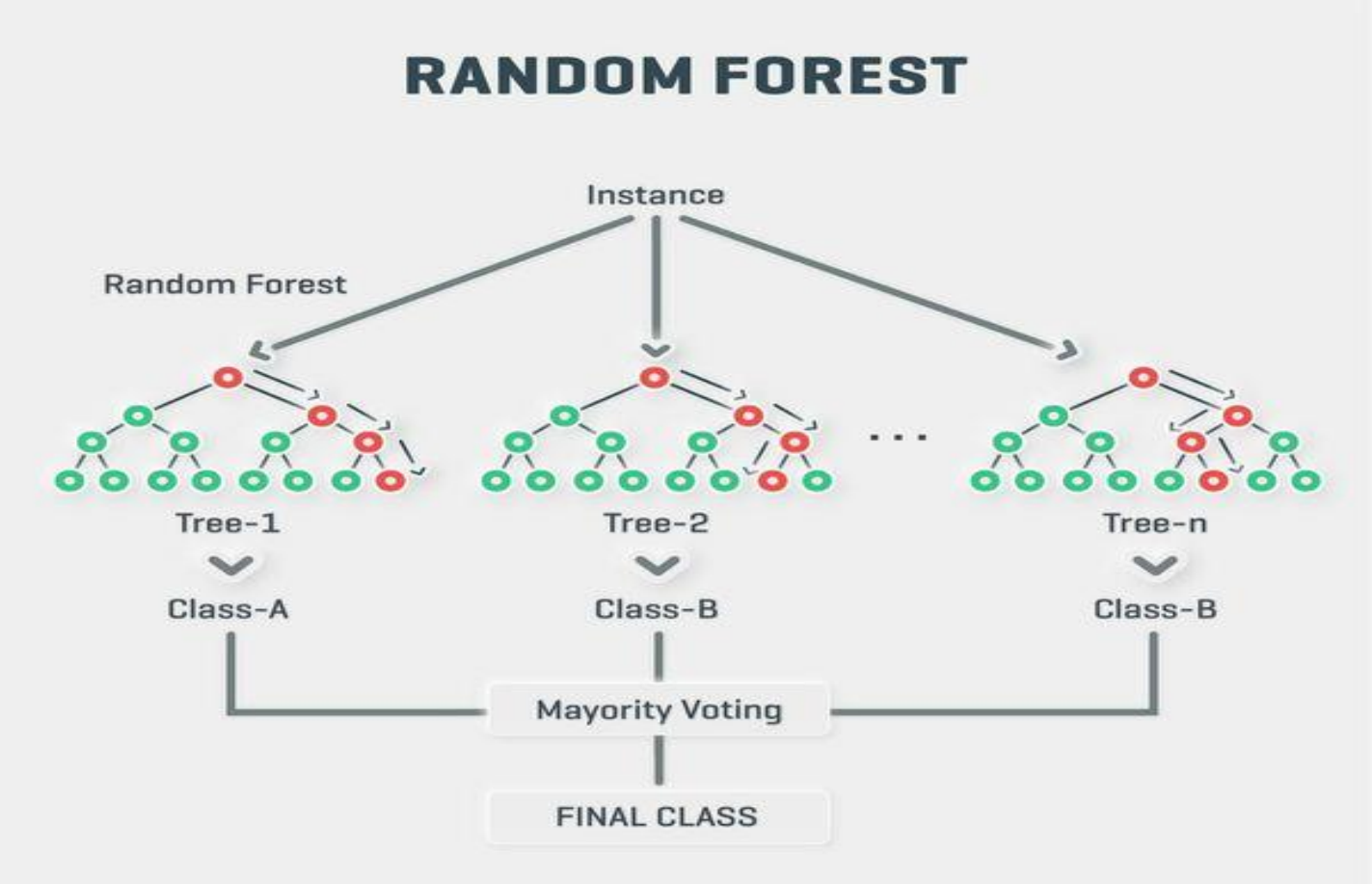
What is Random Forest

Random Forest is versatile Algorithm capable of performing both

1. Regression
2. Classification

It is a type of ensemble learning method

Commonly used predictive modeling and Machine Learning Technique



Random Forest Example

Random Forest Example-

Let's say you want to decide if to watch movie **"Edge of Tomorrow"** or not

So you will decide based on following two actions

1. You can **ask your best friend**
2. You can **ask bunch of friends**

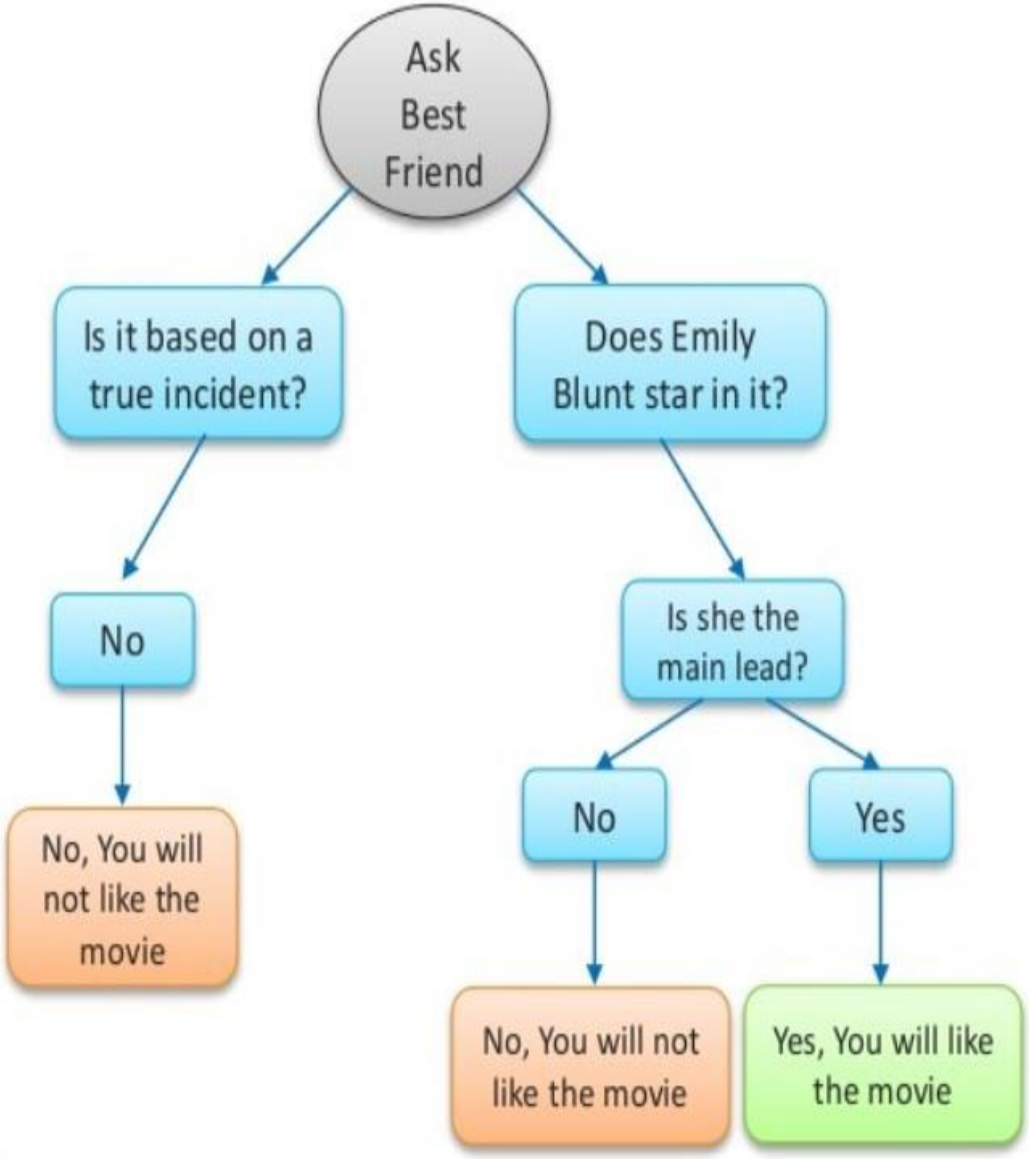
Random Forest Example

Random Forest Example

To figure out if you will like "Edge of Tomorrow" or not, Your friend will analyze a few thing as

1. If you like adventure and Action
2. If you like Emily Blunt

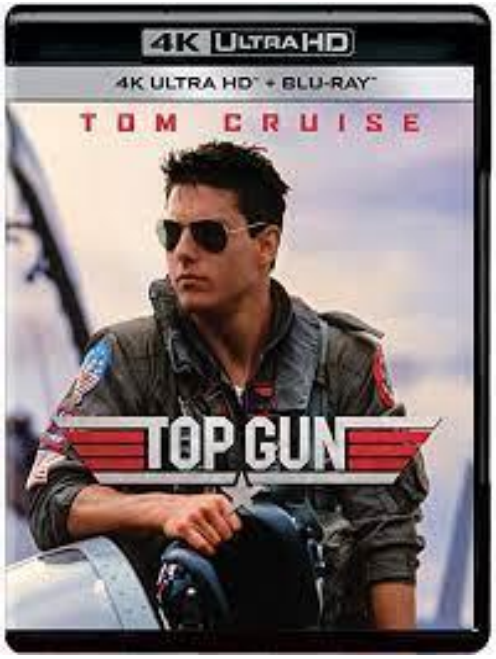
Thus, **a decision tree** created by your best friend.

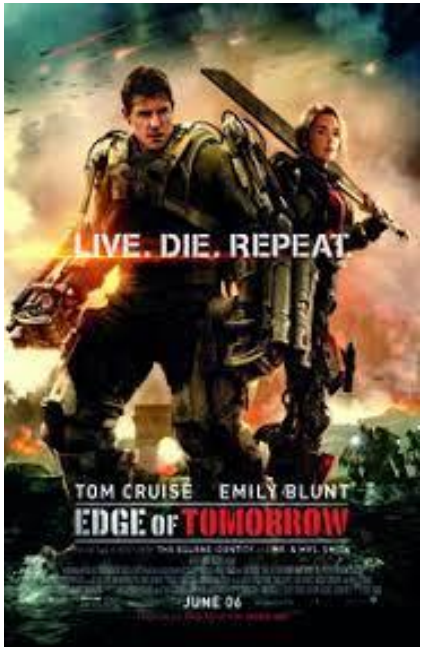


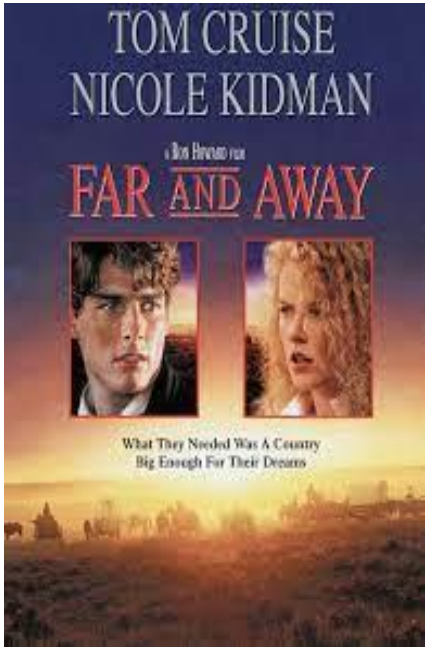
Random Forest Example

Random Forest Example

In order to get more accurate recommendation, you will have to ask bunch of friends, Say Friend1, Freind2, and Friend3 and consider their vote.
Each one of them may take movies of different type and further decide.
The majority of votes will decide the final outcomes.
Thus, you build Random Forest of group of friends.

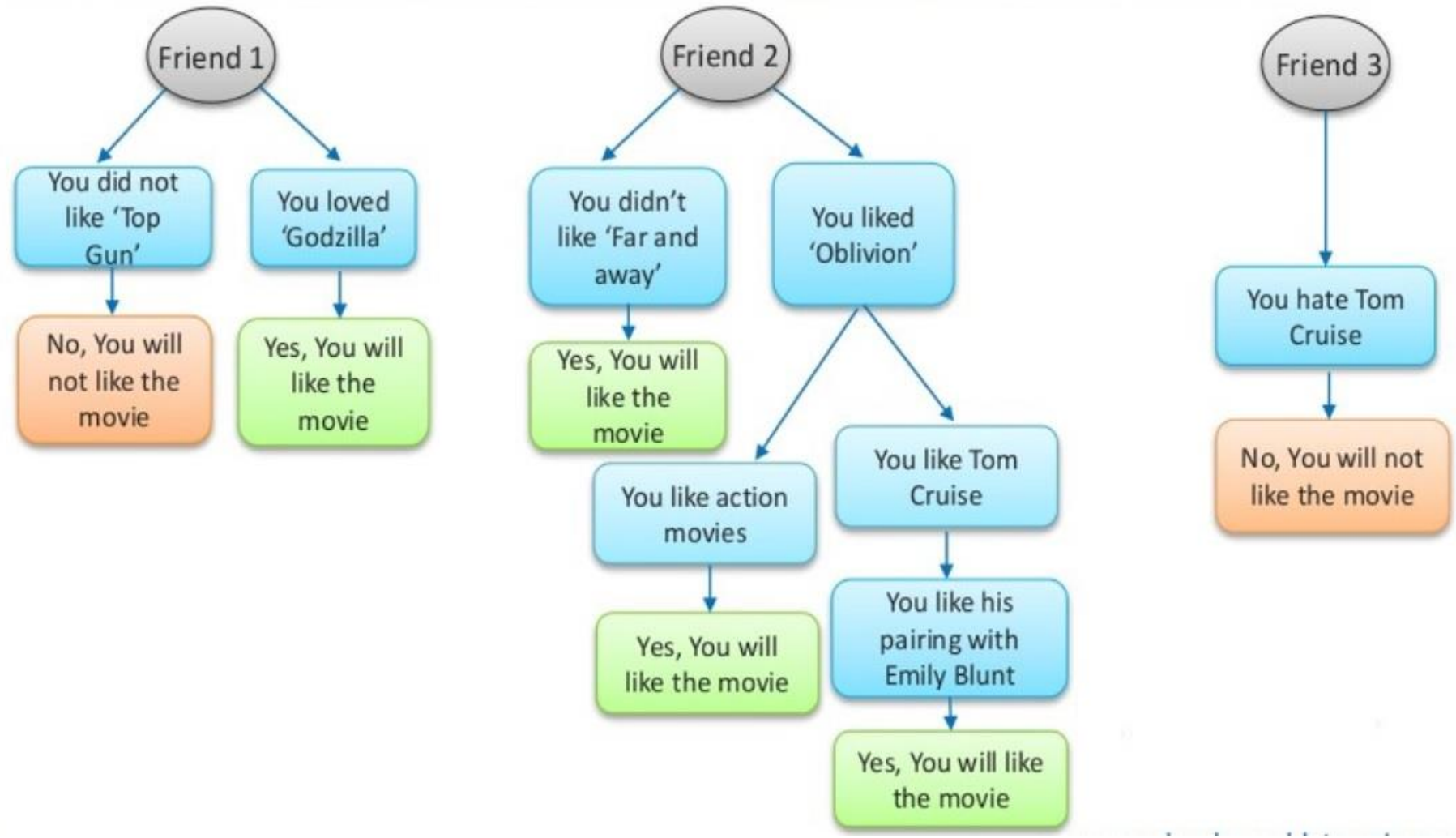






Random Forest Example

Random Forest Example



Random Forest Use Cases

Banking- Identification of loan risk applicants by their probability of defaulting Payments

Medicine- Identification of at risk patients and disease trends

Land Use- Identification of areas of similar Land use

Marketing-Identifying Customer churn.

Random Forest Algorithm

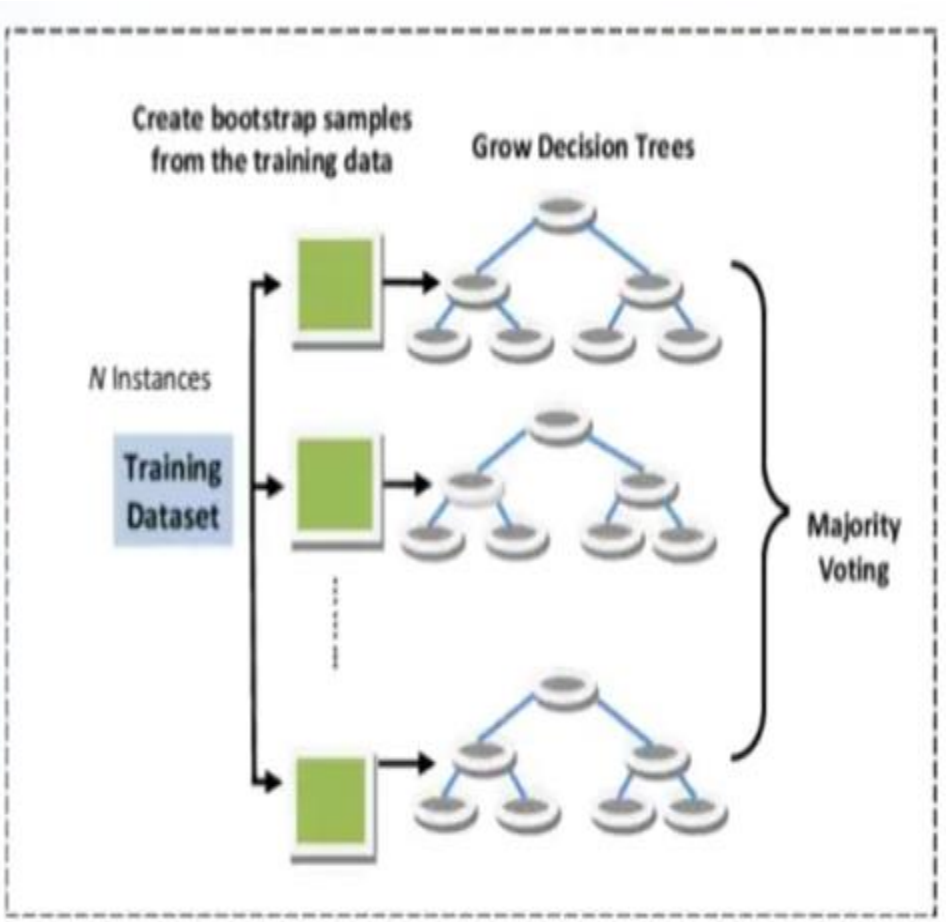
1. Randomly Select m features From T, where m<<T

2. For node d, Calculate the best split point among the m feature

3. Split the node into two daughter Nodes using the best split

4. Repeat First three steps until **n** number of nodes has been reached

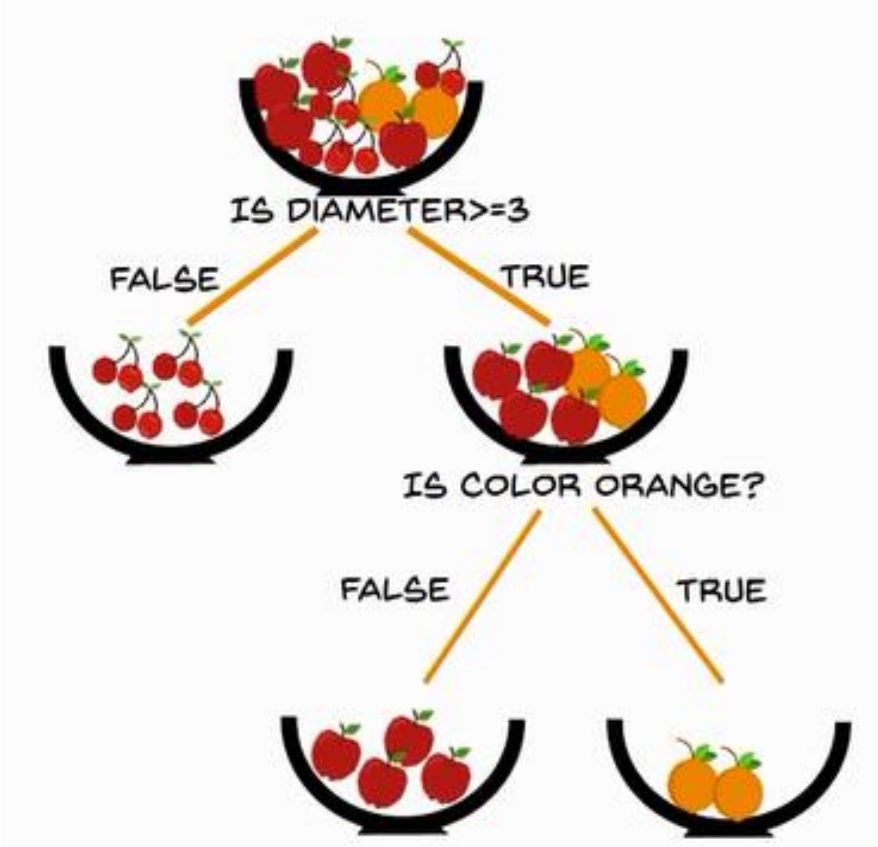
Build your Forest by repeating steps 1-4 for **D** Number of times



- ✓ T: number of features
- ✓ *D*: number of trees to be constructed
- ✓ *V*: Output: the class with the highest vote

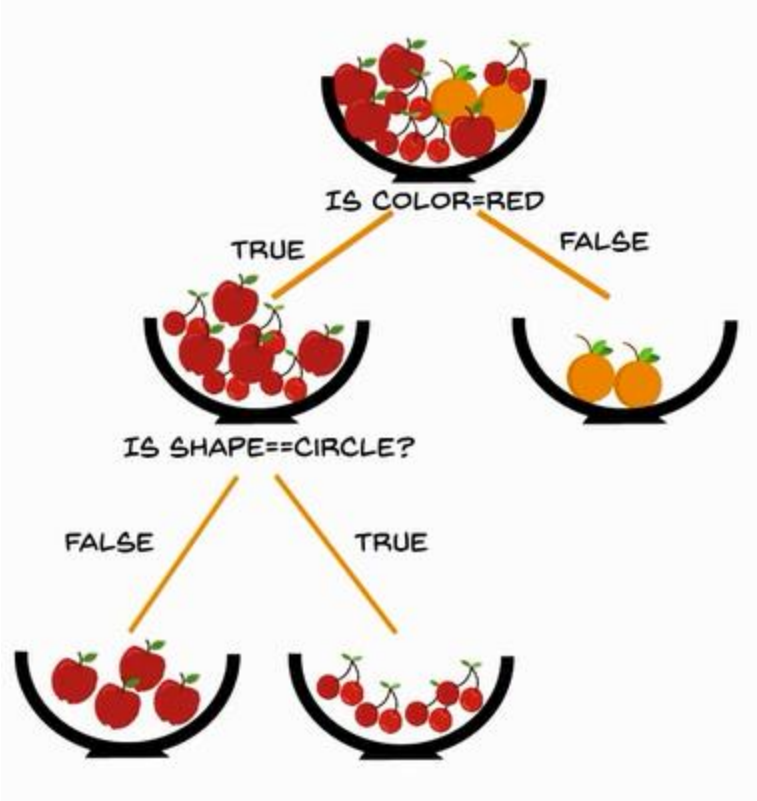
Random Forest Example

Let this be a tree 1



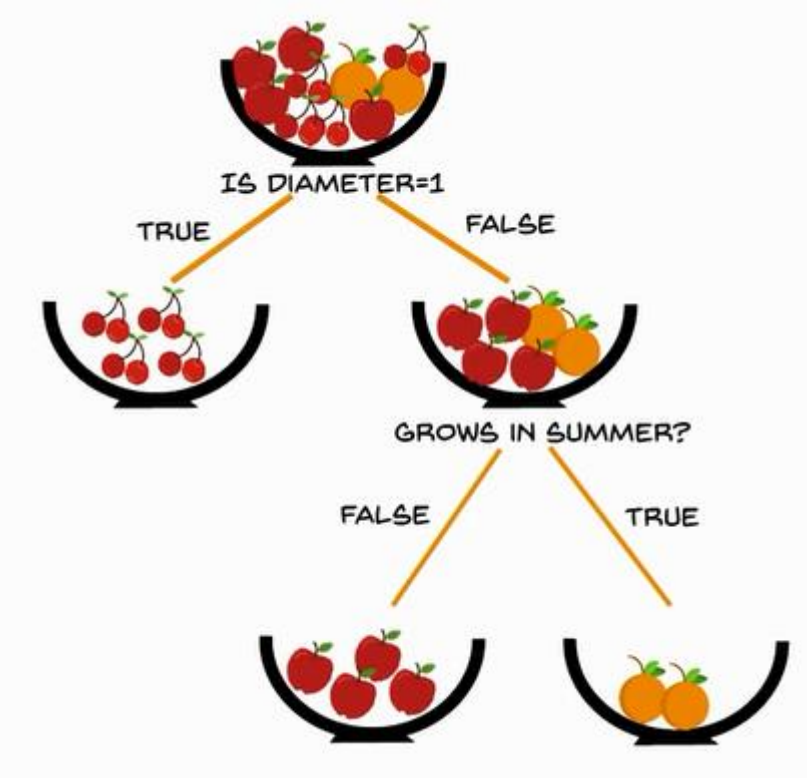
Random Forest Example

Let this be a tree 2

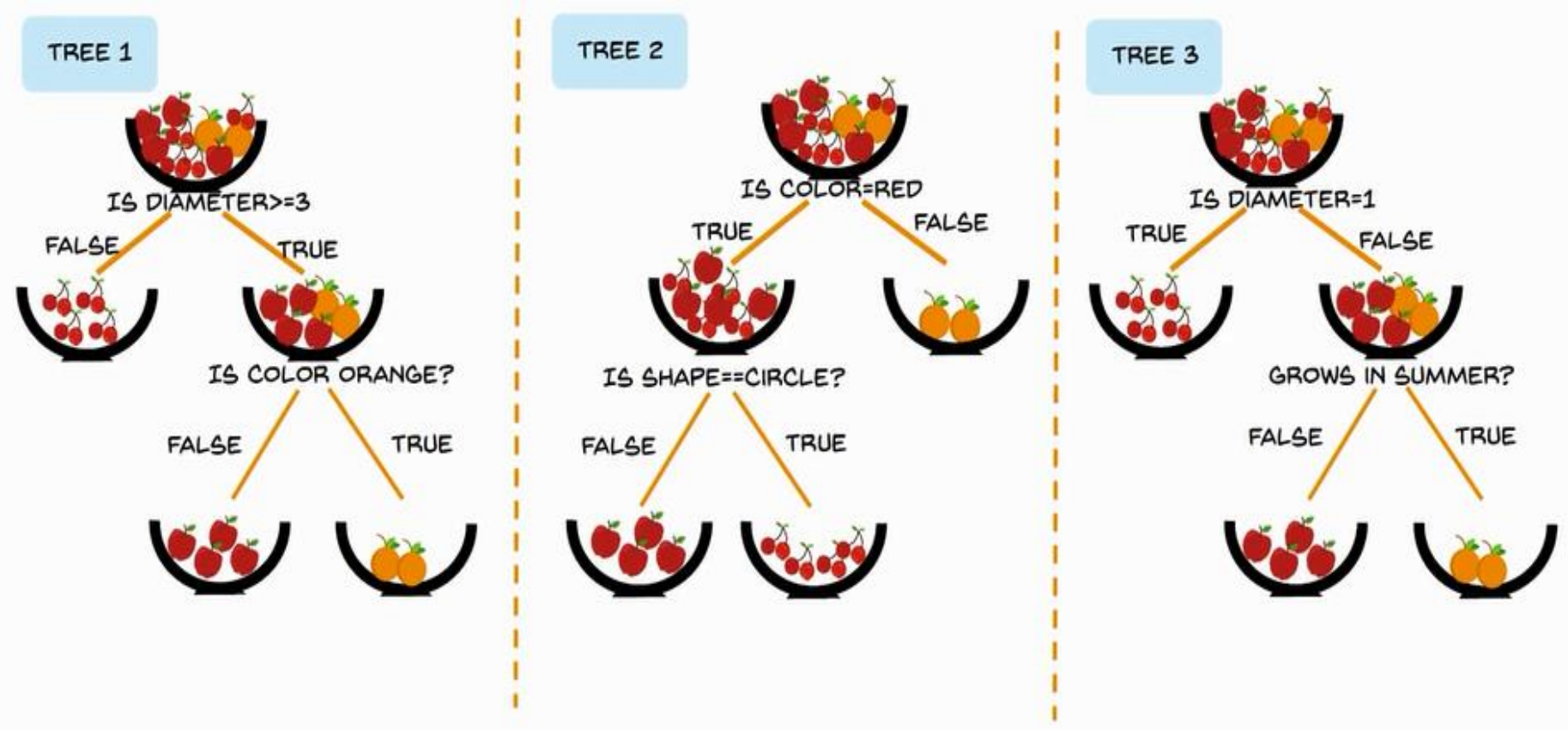


Random Forest Example

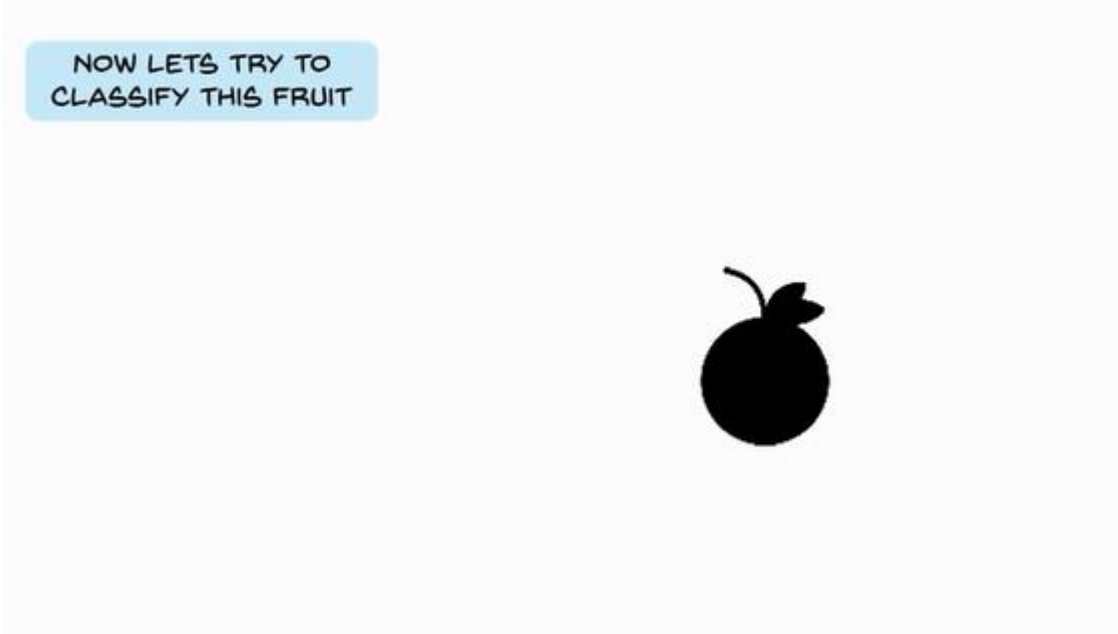
Let this be a tree 3



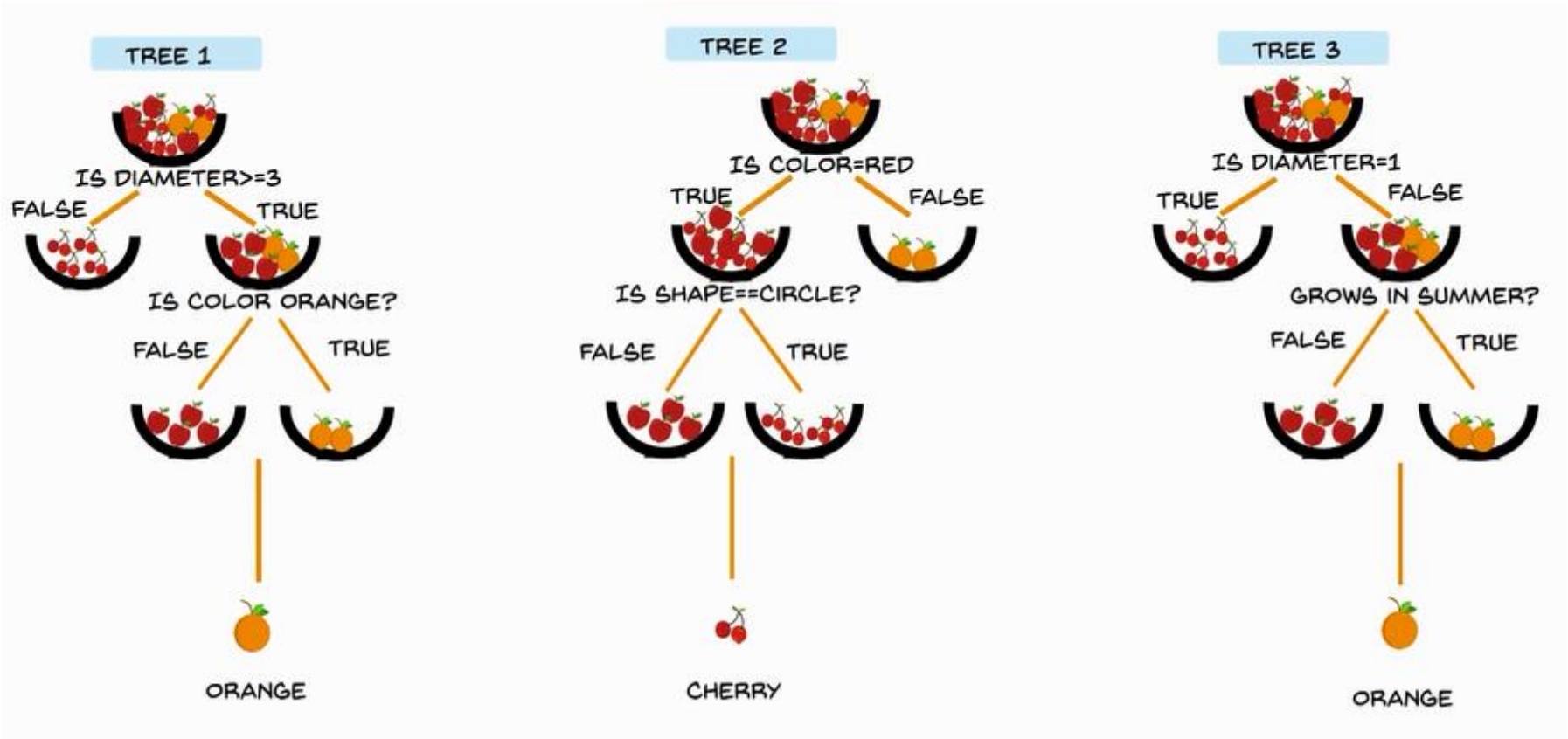
Random Forest Example



Random Forest Example



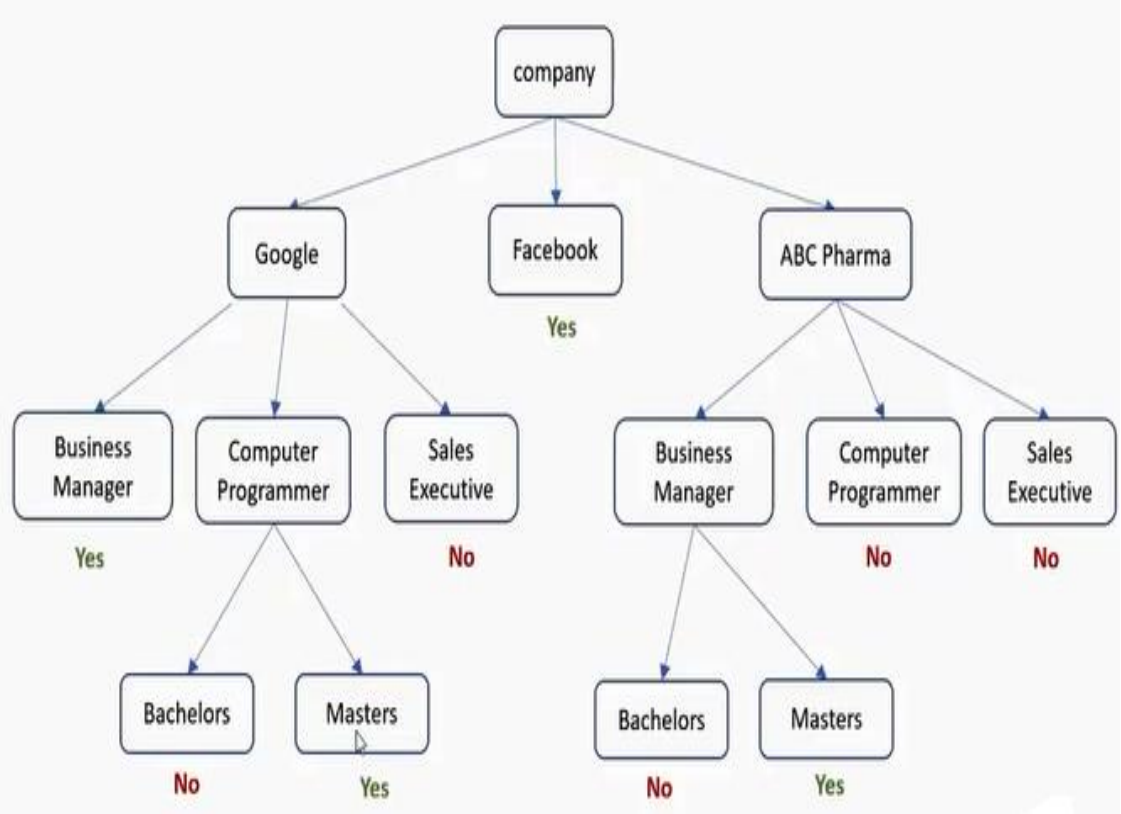
Random Forest Example



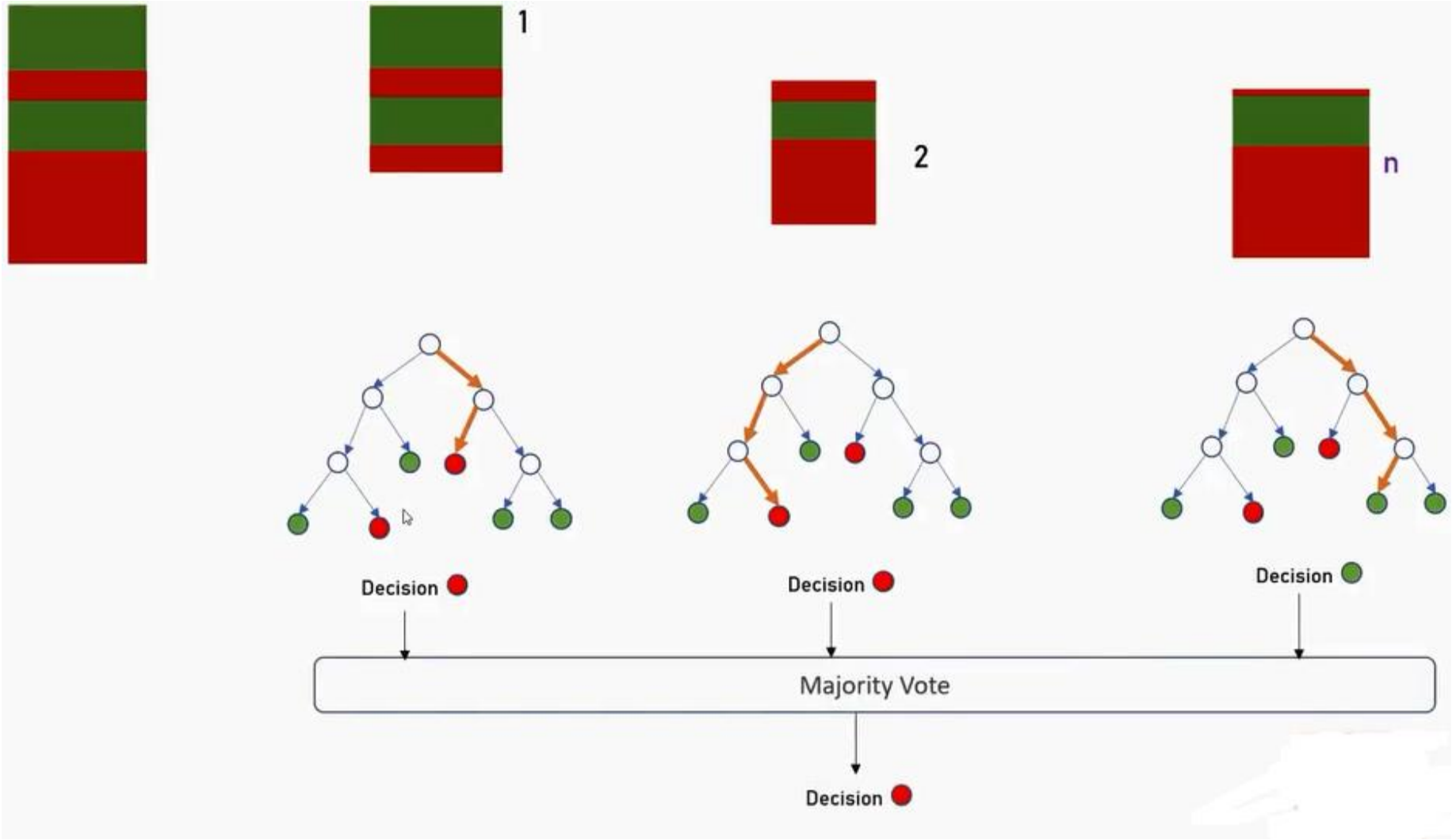
Random Forest Example

Random Forest-Example- Salary >100K $

Company	Job	Degree	Salary_more_then_100k
google	sales executive	bachelors	0
google	sales executive	masters	0
google	business manager	bachelors	1
google	business manager	masters	1
google	computer programmer	bachelors	0
google	computer programmer	masters	1
abc pharma	sales executive	masters	0
abc pharma	computer programmer	bachelors	0
abc pharma	business manager	bachelors	0
abc pharma	business manager	masters	1
facebook	sales executive	bachelors	1
facebook	sales executive	masters	1
facebook	business manager	bachelors	1
facebook	business manager	masters	1
facebook	computer programmer	bachelors	1
facebook	computer programmer	masters	1

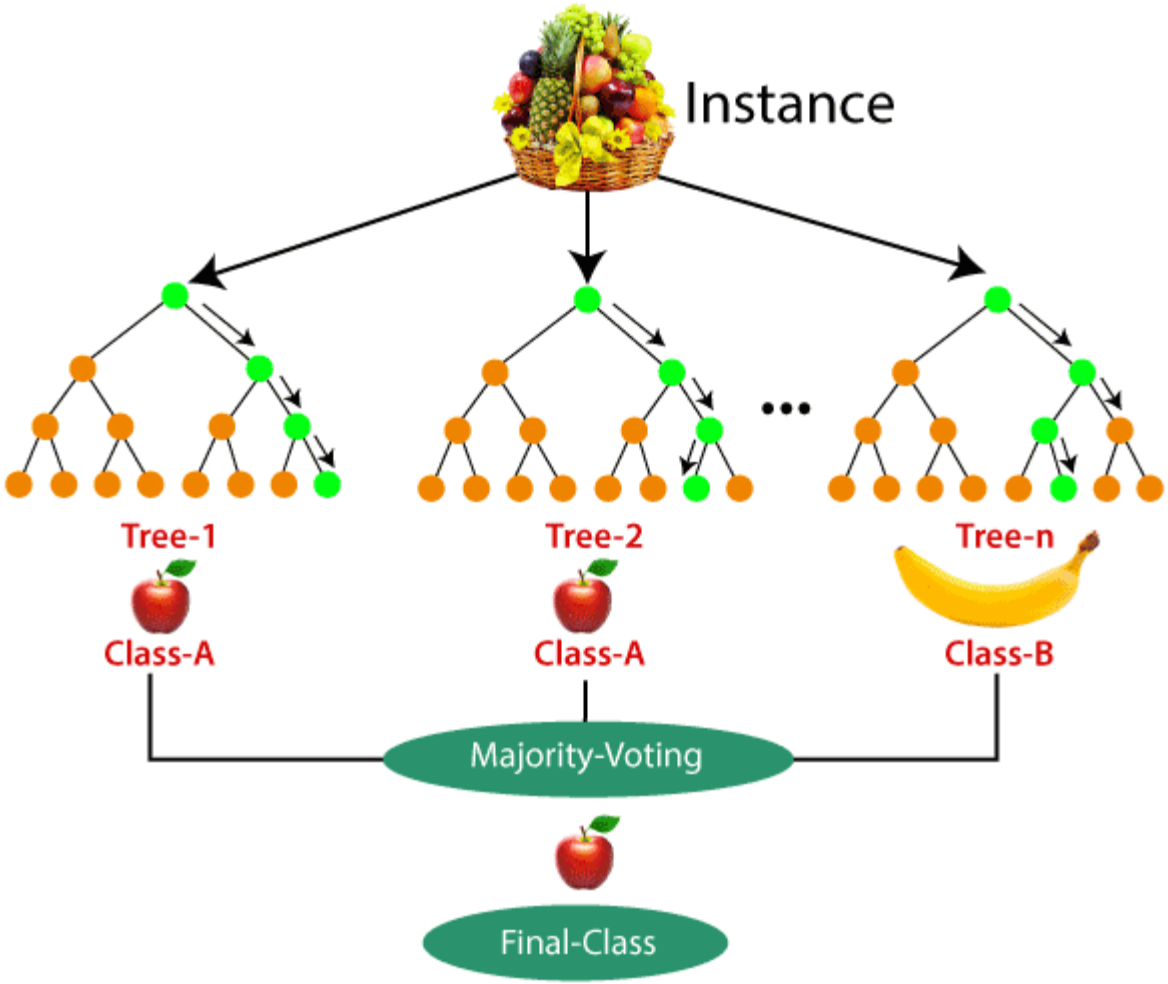


Random Forest-Example- Salary >100K $



Random Forest Example

The majority of the decision trees have chosen *apple* as their prediction. This makes the classifier choose *apple* as the final prediction.



Random Forest Features

The strengths of decision tree

1. Decision trees are able to generate understandable rules.
2. Decision trees perform classification without requiring much computation.
3. Decision trees are able to handle both continuous and categorical variables.
4. Decision trees provide a clear indication of which fields are most important for prediction or classification.

The weaknesses of decision tree

1. Decision trees are **less appropriate** for estimation tasks where the goal is **to predict the value of a continuous attribute.**
2. Decision trees are **prone to errors in classification** problems with many class and **relatively small number of training examples.**
3. Decision tree can be **computationally expensive to train**.

- The process of **growing a decision tree is computationally expensive.**
- **Pruning algorithms** can also be **expensive** since many candidate sub-trees must be formed and compared.

Random Forest Applications

Remote Sensing

Used in ETM devices to acquire images of the earth's surface.

Accuracy is higher and training time is less

Object Detection

Multiclass object detection is done using Random Forest algorithms

Provides better detection in complicated environments

Kinect

Random Forest is used in a game console called Kinect

Tracks body movements and recreates it in the game

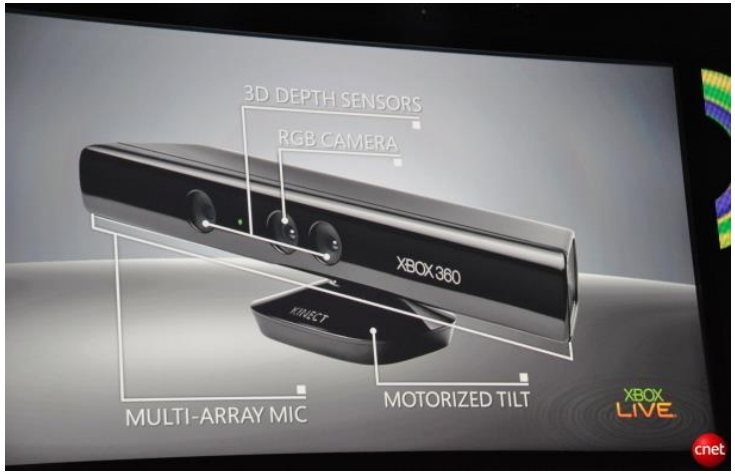
The **Enhanced Thematic Mapper Plus (ETM+) Instrument** is a fixed "whisk-broom", eight-band, multispectral scanning radiometer capable of providing high-resolution imaging information of the Earth's surface.

Kinect

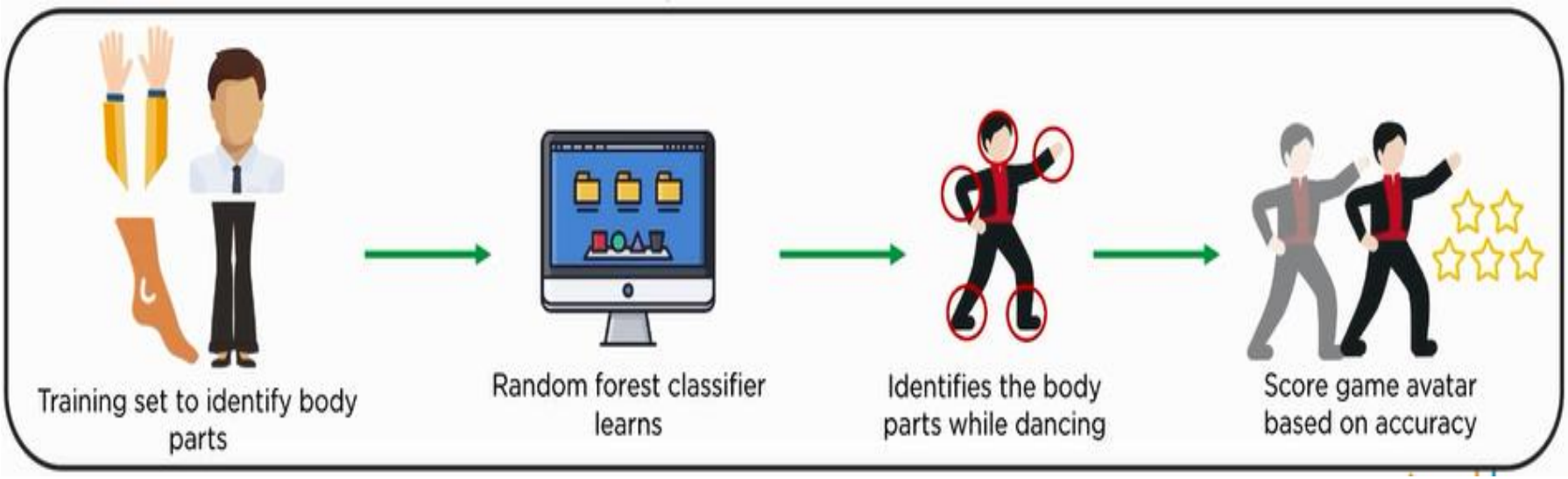
Kinect is a line of motion sensing input devices.
Used to perform real-time gesture recognition, speech recognition and body skeletal detection

Kinect

- Object Recognition: http://www.youtube.com/watch?feature=iv&v=fQ59dXOo63o
- Mario: http://www.youtube.com/watch?v=8CTJL5lUjHg
- 3D: http://www.youtube.com/watch?v=7QrnwoO1-8A
- Robot: http://www.youtube.com/watch?v=w8BmgtMKFbY
- Kinect: https://www.youtube.com/watch?v=ELrEJJT_eng



Random Forest Applications



Random Forest Advantages and Disadvantages

Advantages

- It can perform both regression and classification tasks.
- A random forest produces good predictions that can be understood easily.
- It can handle large datasets efficiently.
- The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm.

Disadvantages

- When using a random forest, more resources are required for computation.
- It consumes more time compared to a decision tree algorithm.

Random Forest Conclusion

- The random forest algorithm is a machine learning algorithm that is easy to use and flexible.
- It uses ensemble learning, which enables organizations to solve regression and classification problems.
- This is an ideal algorithm for developers because it solves the problem of overfitting of datasets.
- It’s a very resourceful tool for making accurate predictions needed in strategic decision making in organizations.


ThankYou
Merci
Salamat
Kop
Hvala
Tack
Xie
kun
Gra
Dankie
Kruthagnathalu
Arigatou
Dhonnobaad
Asante
Hain
Dhan
daa
Shukriya
Khopjai
Dhanyavad
Gomapsupnida
Kun
Euxaristo
Danke
dank
Terima
Enkosi
Fyrir
Blagodariya
Gracias
Nandree
Grazzi
raibh
krap
Dakujem
Daw
Waad
Dhanyavadaalu
Dank
Takk
Gamsahapnida
Te ekkür
Dekuju/Dekujeme
Diolch
Kiitos
Sheun
umesc
Shnorhakalutiun
Kasih
Mamnoon
Todah
Shokriya
Ngiyabonga
Cam
Dziękuję
Shokrun
Spaas
Mul
Ačı
or
al
Dhanyavaad
Go
Grazie
Falemiderit
Or
ederim
Shukriya