IT 307 ML
UNIT 1
Introduction to ML

Presented by Dr. Ankush Sawarkar
Email: adsawarkar@sggs.ac.in

FACULTY PROFILE

- Qualifications
  - Ph D in Machine Learning & Deep Learning
  - M.Tech in Computer Science & Engineering
  - B.E in Computer Science & Engineering
- 10 year of experience
  - Data Science Researcher
  - Data Science Consultant
  - Software Developer
  - Assistant Professor
  - Corporate Trainer
- Area of specialization
  - Data Science
  - AI - Machine Learning, Deep Learning
  - Natural Language Processing
  - Bioinformatics
  - Computer Vision
  - Visualization - Tableau

https://scholar.google.com/citations?hl=en&user=Y3CY1-wAAAAJ

Dr. Ankush D. Sawarkar (Ph.D. VNIT Nagpur)
Assistant Professor, *Dept. Information Technology,*
Email| adsawarkar@sggs.ac.in / ankush1sawarkar@gmail.com
https://orcid.org/my-orcid?orcid=0000-0001-7099-1987
https://www.webofscience.com/wos/author/record/ABE-6640-2020
https://www.scopus.com/authid/detail.uri?authorId=56669766500
https://www.researchgate.net/profile/Ankush-Sawarkar
https://sites.google.com/view/ankushsawarkar/home

Books:

1. Kevin Murphy, **Machine Learning: A Probabilistic Perspective**, MIT Press, 2012.
2. **Machine Learning For Dummies** by John Mueller and Luca Massaron
3. Trevor Hastie, Robert Tibshirani, Jerome Friedman, **The Elements of Statistical Learning**, Springer 2009 (freely available online).
4. Christopher Bishop, **Pattern Recognition and Machine Learning**, Springer, 2007

At the end of this course

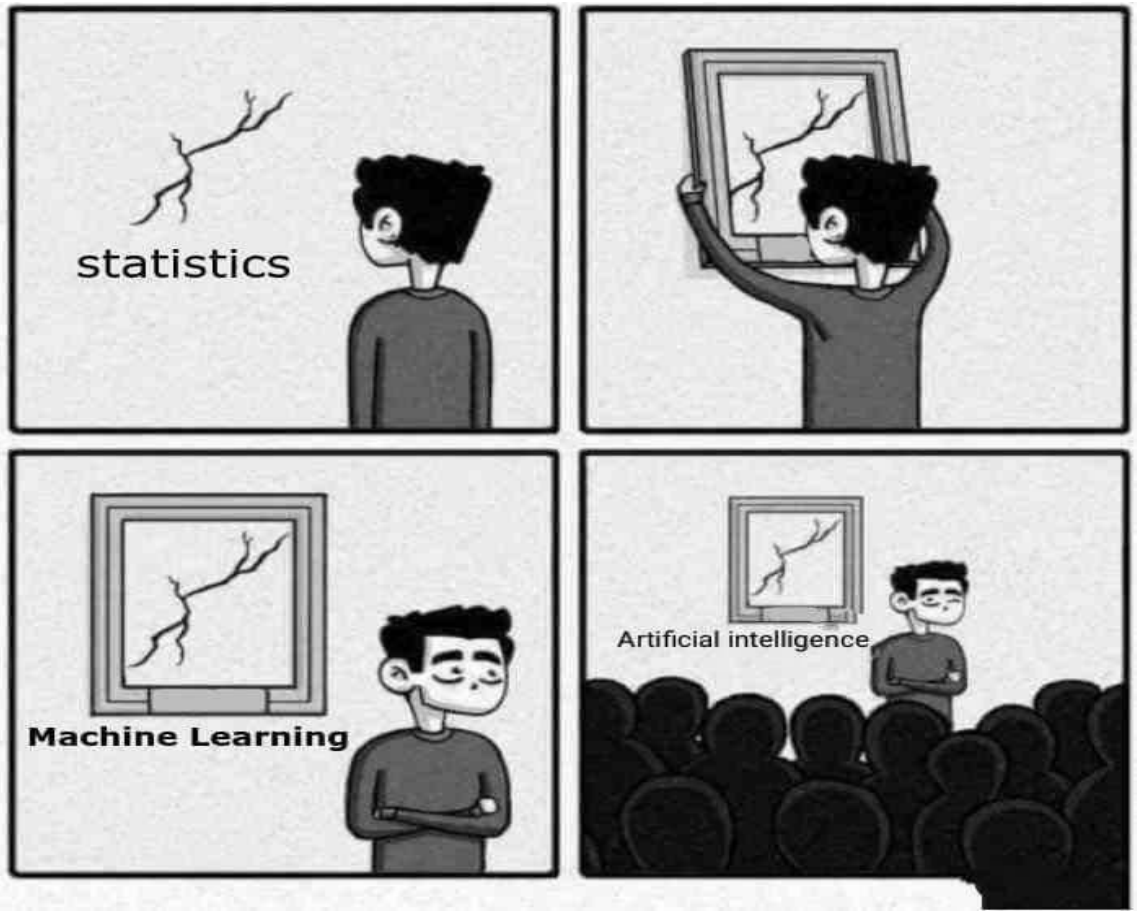
- Machine Learning, Types of ML
- Regression/ Classification / Clustering
- Overfitting & Underfitting
- Model Selection - MAE/ RMSE/ Regularization
- Evaluation Metrics - Confusion Matrix (TP/TN/FP/FN)/ Accuracy/ Precision/ Recall/ F1Score
- Logistics Regression/ Sigmoid/ Limitation of Logistics Reg
- Decision Tree - Gini Index/ Information Gain
- K Fold Cross Validation/ K- Nearest Neighbors/ Random Forest
- Support Vector Machine (SVM) Tuning Parameter
- Clustering - Kmeans/ Elbow method/ Performance measures

ML Techniques Overview

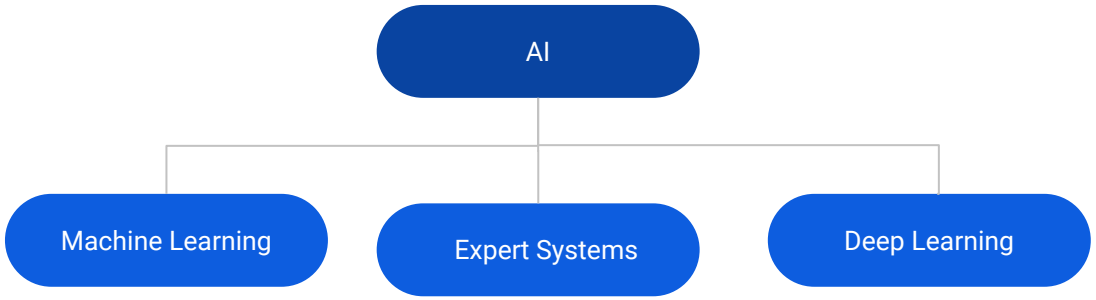
- “The field of study that gives computers the ability to learn without being explicitly programmed.” - Arthur Samuel

- “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E”. - Tom Mitchell

AI Landscape



AI Landscape…



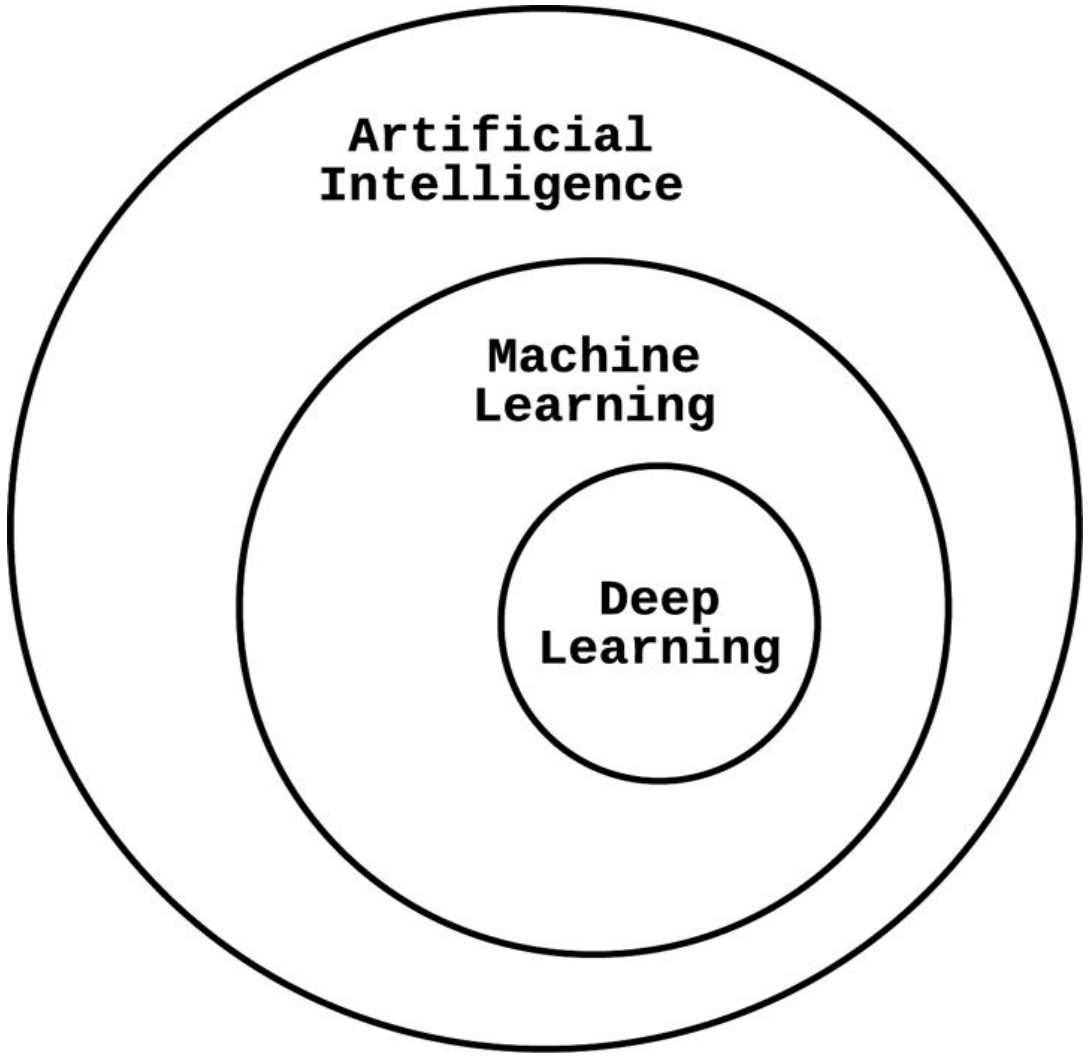
Stats + Algorithms

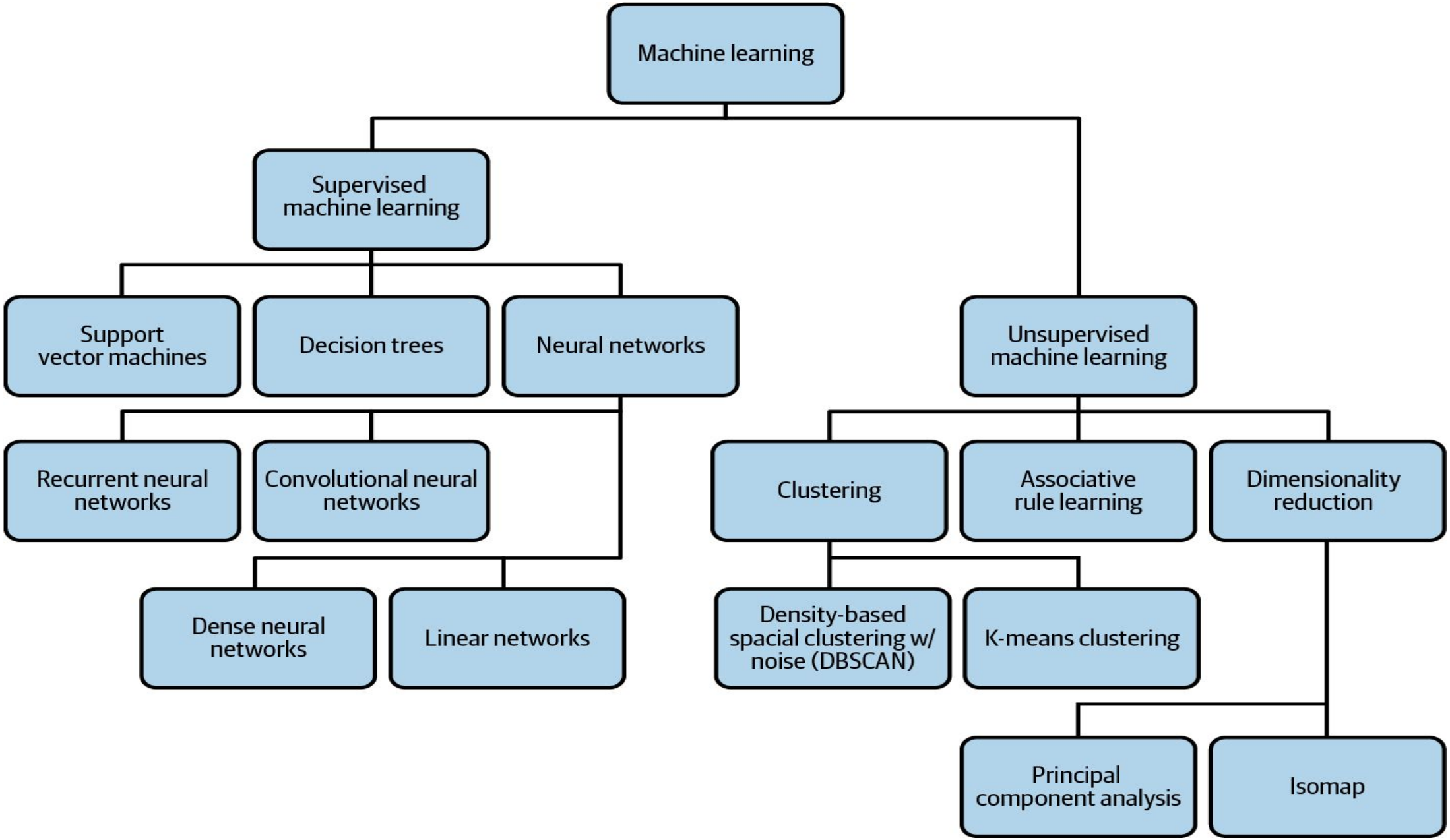
Rule based systems to solve specific problems

ML driven by Neural Networks

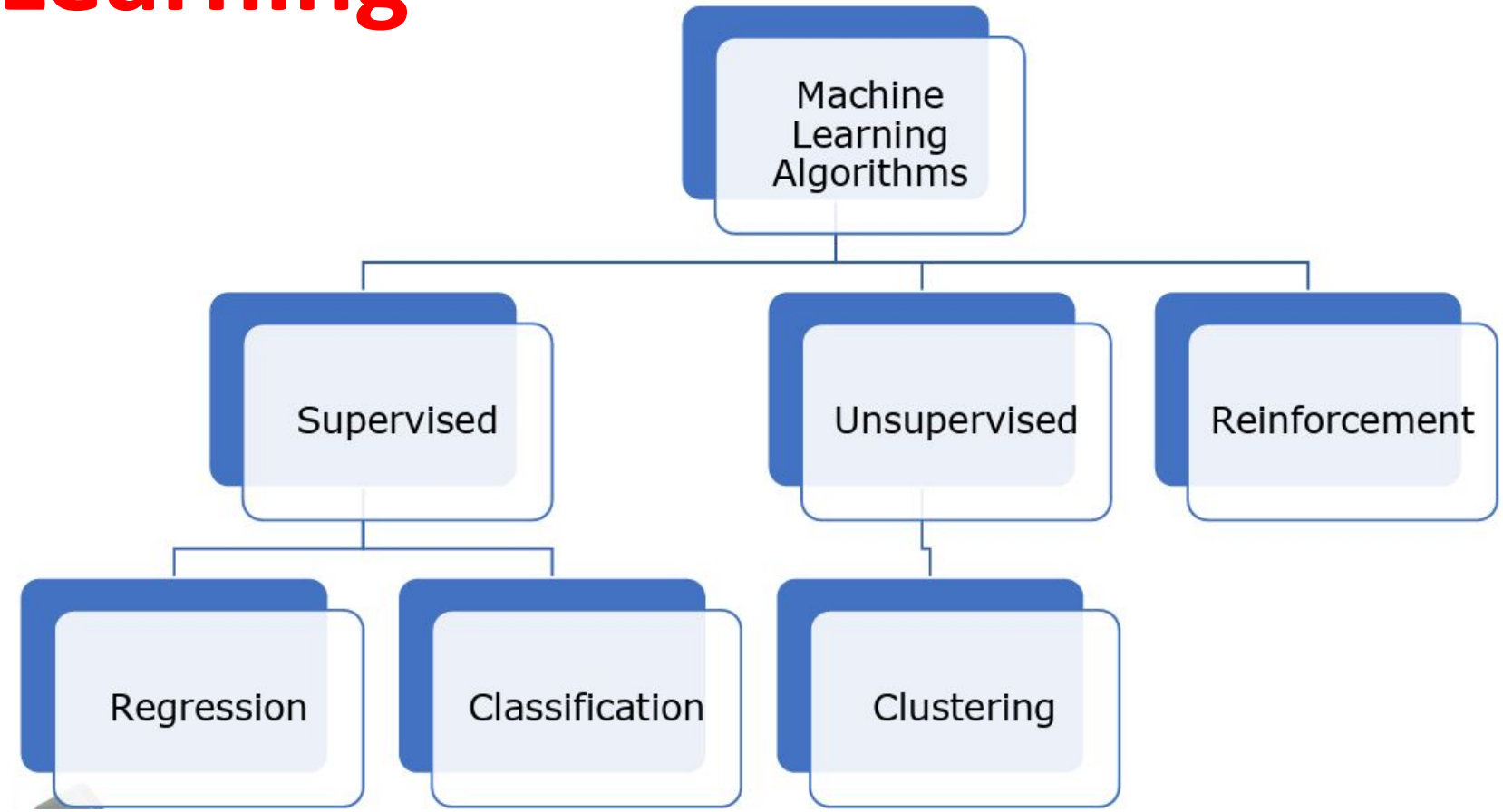
What are the factors driving the development of AI ?

Another way of looking




Machine learning
Supervised machine learning
Support vector machines
Decision trees
Neural networks
Recurrent neural networks
Convolutional neural networks
Dense neural networks
Linear networks
Unsupervised machine learning
Clustering
Associative rule learning
Dimensionality reduction
Density-based spacial clustering w/ noise (DBSCAN)
K-means clustering
Principal component analysis
Isomap

Machine Learning



Supervised Learning

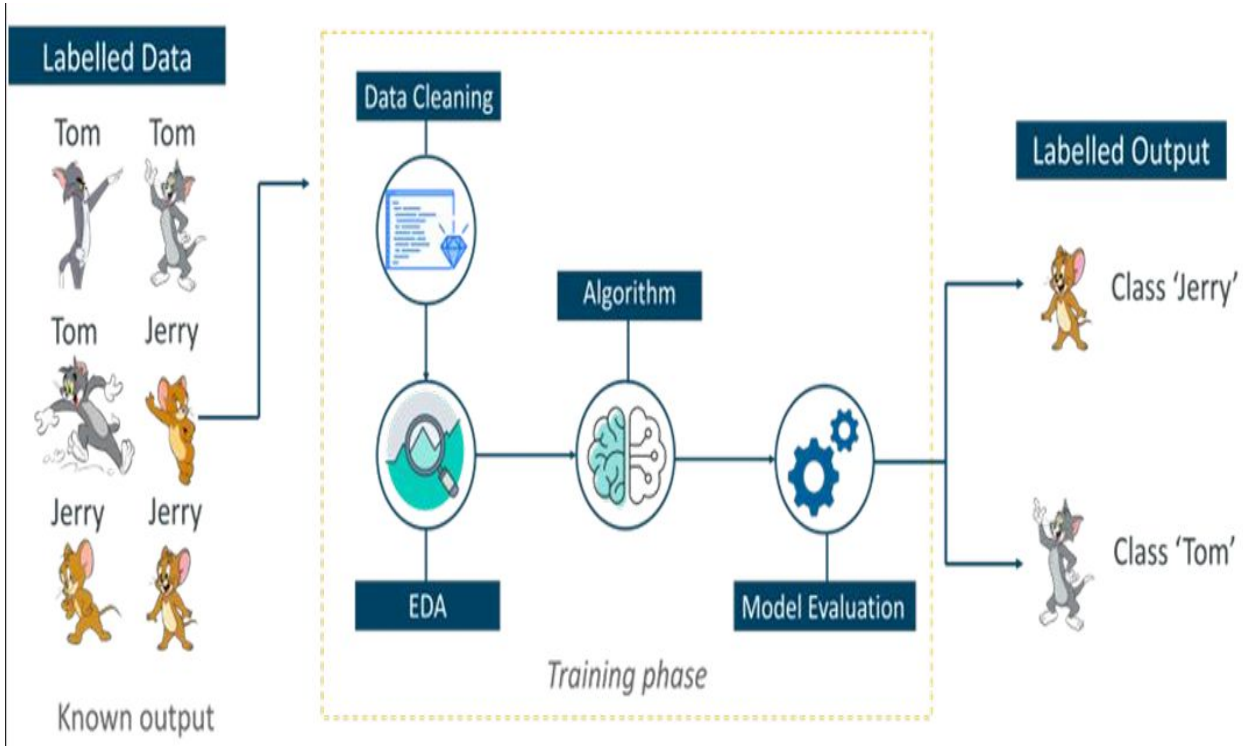


Fig. 2 Supervised Learning REF[1]

Unsupervised Learning

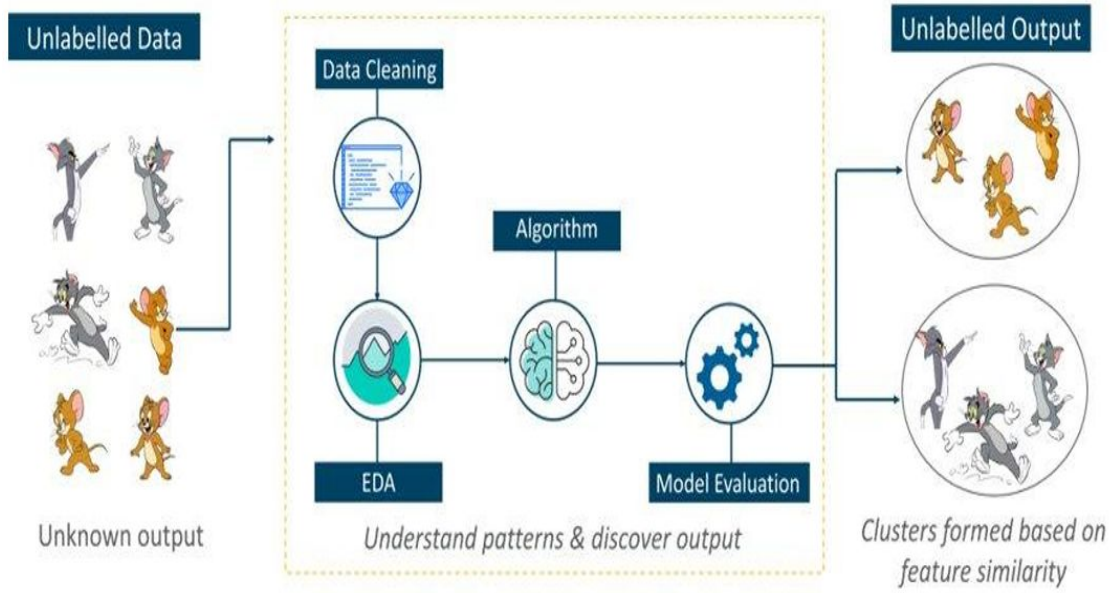


Fig. 3 Unsupervised Learning REF[1]

Supervised Learning

There are two categories of supervised learning:

- Classification task
- Regression task

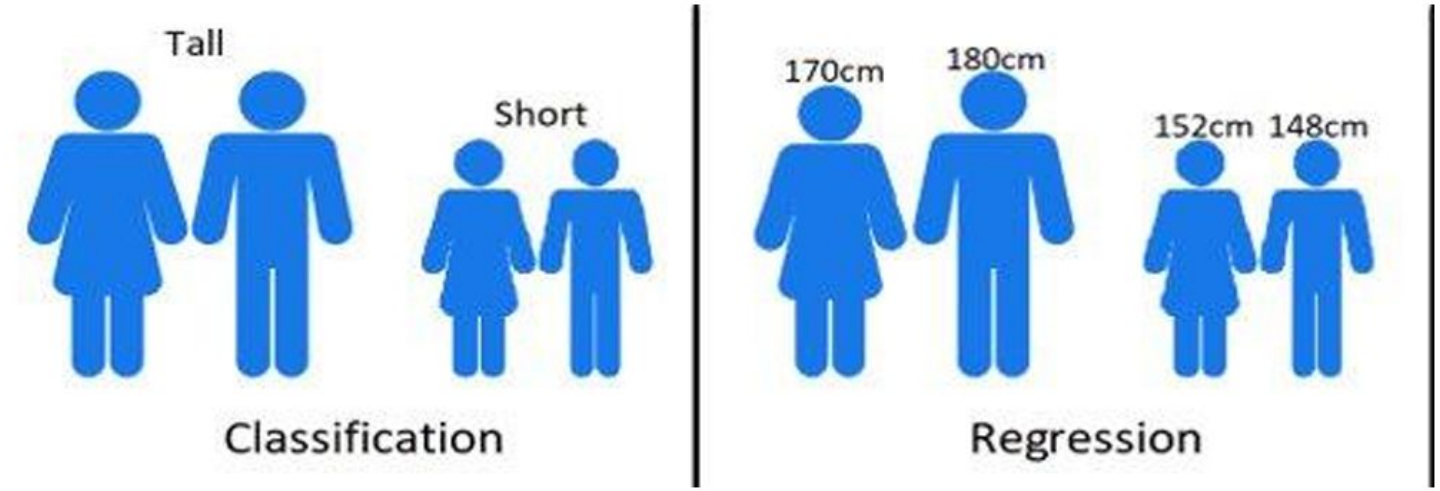


Fig. 5 Supervised Learning- Classification and Regression REF[3]

Unsupervised Learning- Clustering

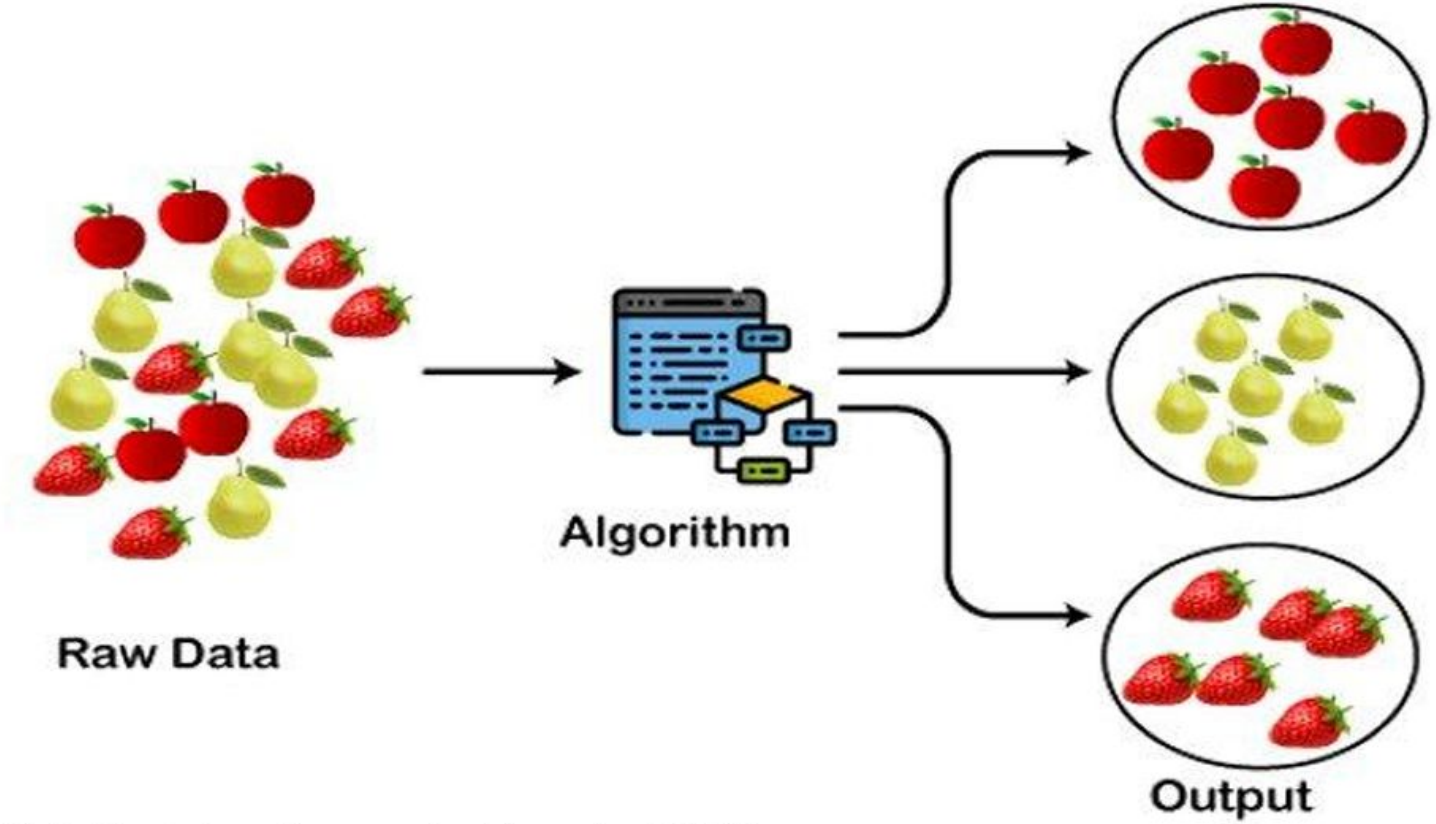


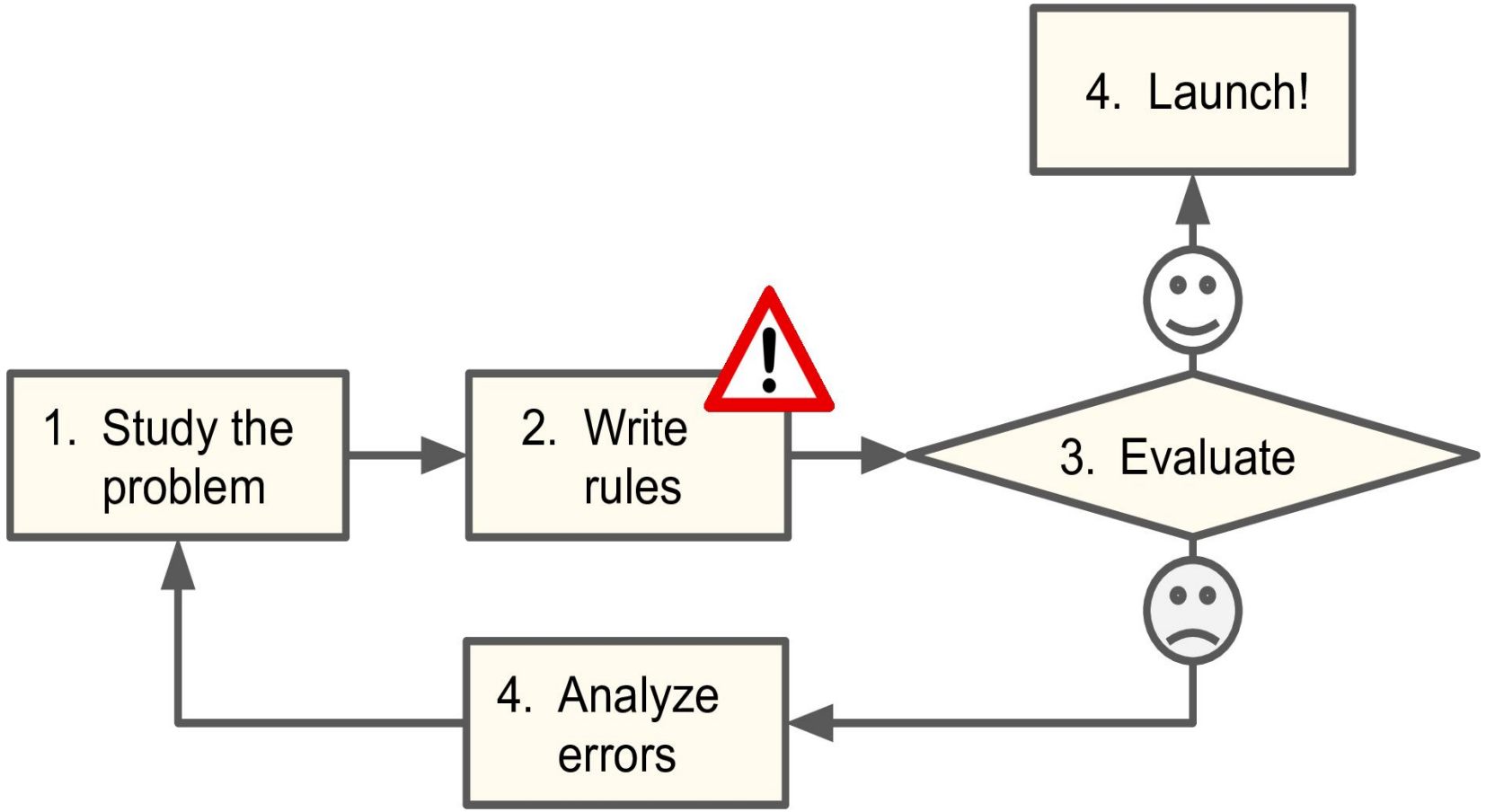
Fig.6 Clustering- Unsupervised Learning REF[2]

Characteristics Comparison

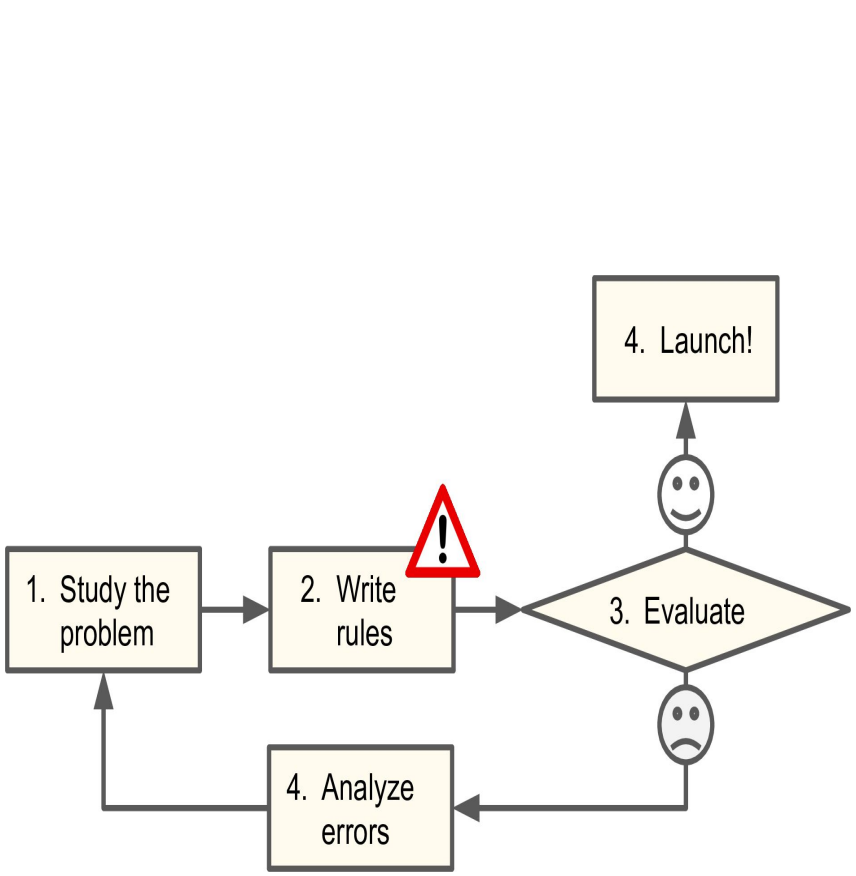
Regression	Classification	Clustering
• Supervised Learning	• Supervised Learning	• Unsupervised Learning
• Output is a continuous quantity	• Output is a categorical quantity	• Assigns data points into clusters
• Main aim is to forecast or predict	• Main aim is to compute the category of the data	• Main aim is to group similar items clusters
• Eg: Predict stock market price	• Eg: Classify emails as spam or non-spam	• Eg: Find all transactions which are fraudulent in nature
• Algorithm: Linear Regression	• Algorithm: Logistic Regression	• Algorithm: K-means

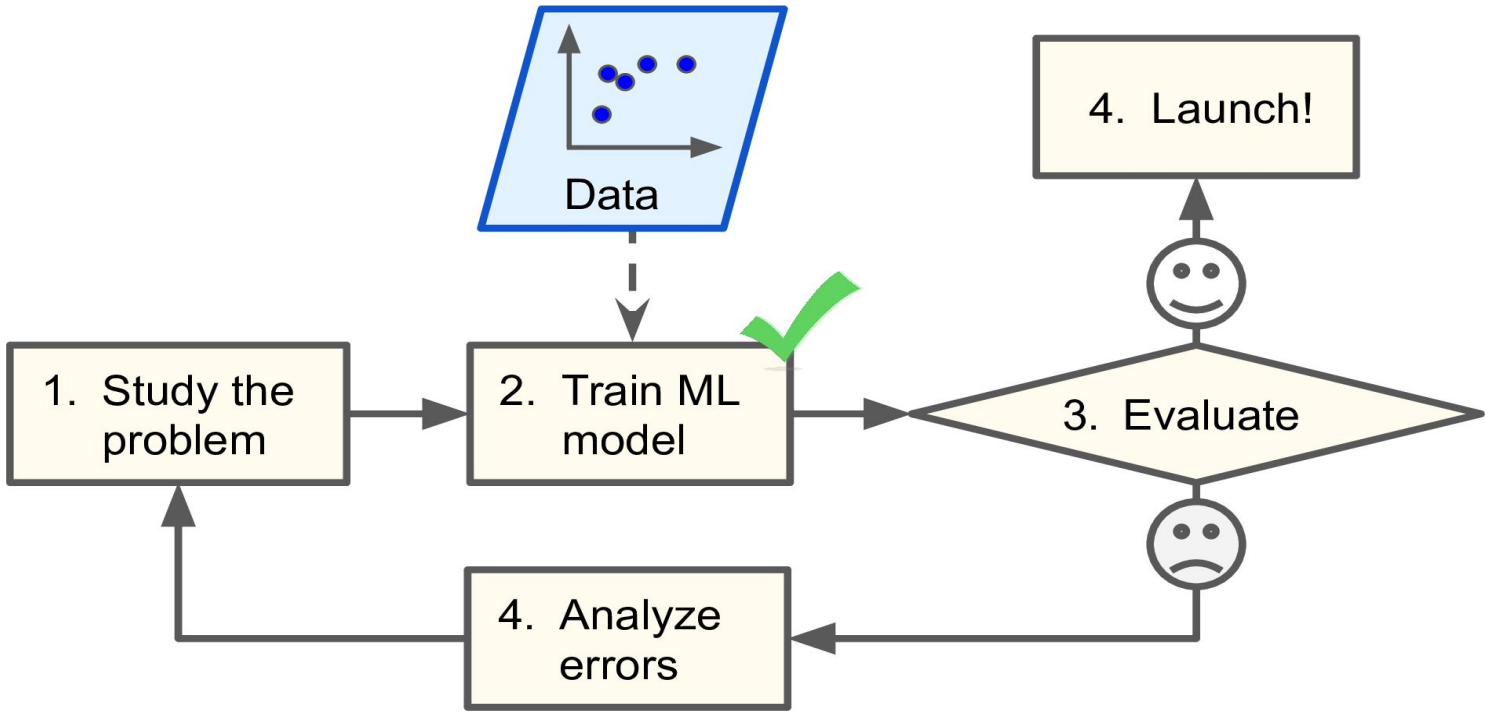
Fig. 7 Characteristic Comparison REF[1]

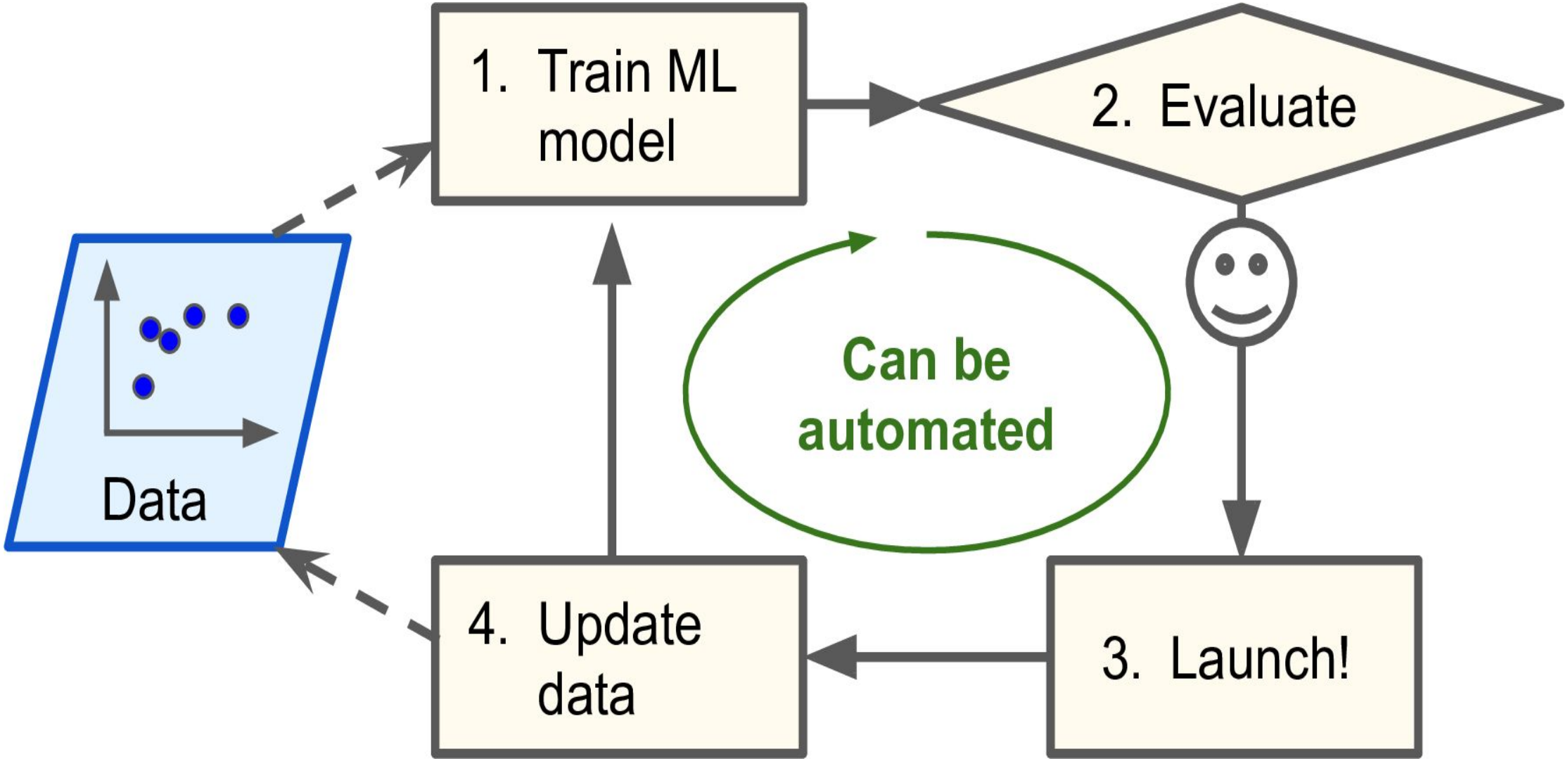
Why do we need ML?
The Email Spam Problem



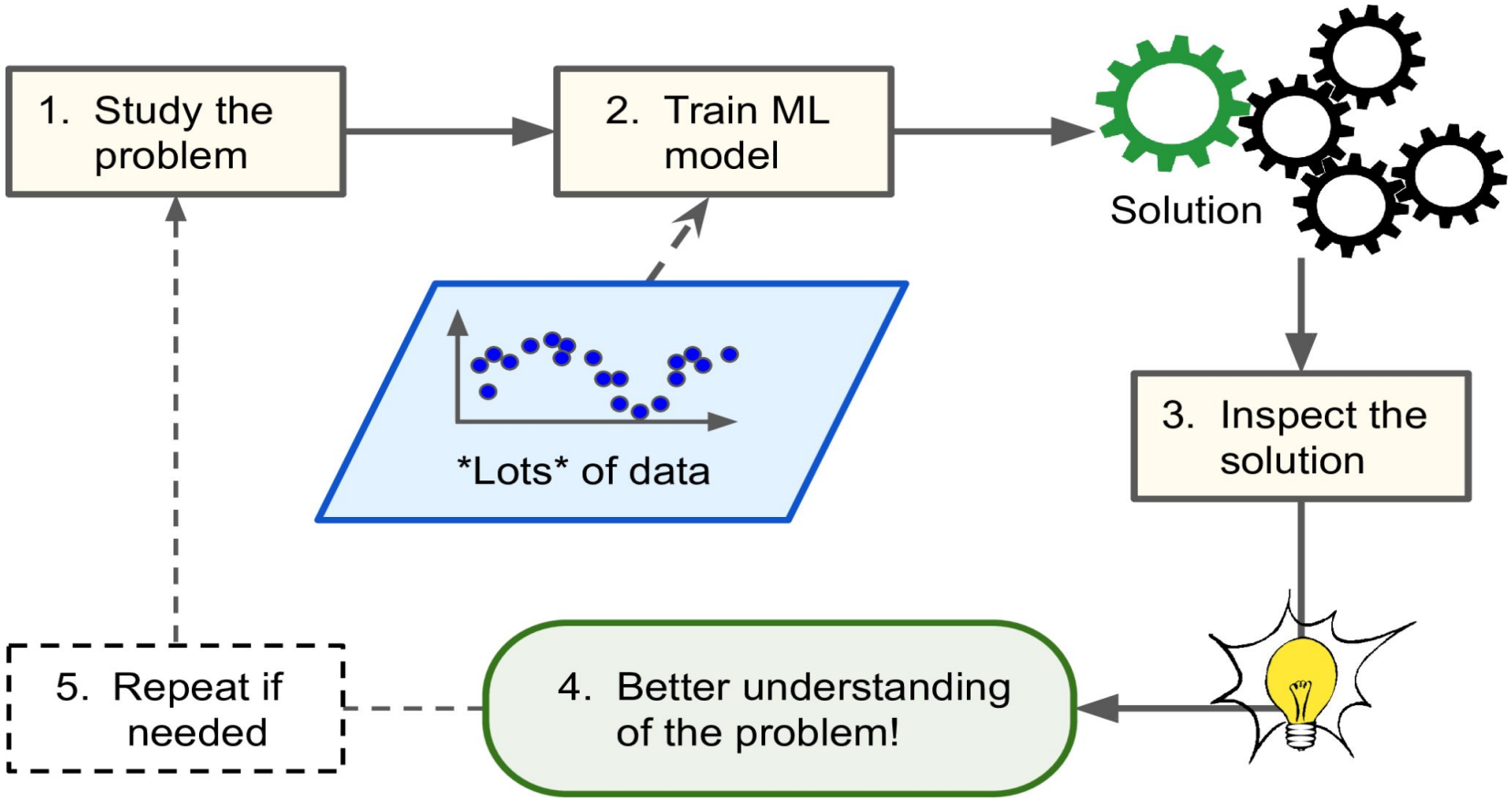
Why do we need ML?
The Email Spam Problem






1. Train ML model
2. Evaluate
Data
Can be automated
4. Update data
3. Launch!

Help Humans Learn



For instance, once a spam filter has been trained on enough spam, it can easily be inspected to reveal the list of words and combinations of words that it believes are the best predictors of spam. Sometimes this will reveal unsuspected correlations or new trends, and thereby lead to a better understanding of the problem.

Where to use ML?

- Problems for which existing solutions require a lot of fine-tuning or long lists of rules: one Machine Learning model can often simplify code and perform better than the traditional approach.
- Complex problems for which using a traditional approach yields no good solution: the best Machine Learning techniques can perhaps find a solution.
- Fluctuating environments: a Machine Learning system can easily be retrained on new data, always keeping it up to date.
- Getting insights about complex problems and large amounts of data.

Some Typical Applications

- Analyzing images of products on a production line to automatically classify them
- Detecting tumors in brain scans
- Forecasting your company's revenue next year, based on many performance metrics
- Segmenting clients based on their purchases so that you can design a different marketing strategy for each segment
- Recommending a product that a client may be interested in, based on past purchases

Questions to be kept in mind when designing an ML System?

What question(s) am I trying to answer? Do I think the data collected can answer that question?

What is the best way to phrase my question(s) as a machine learning problem?

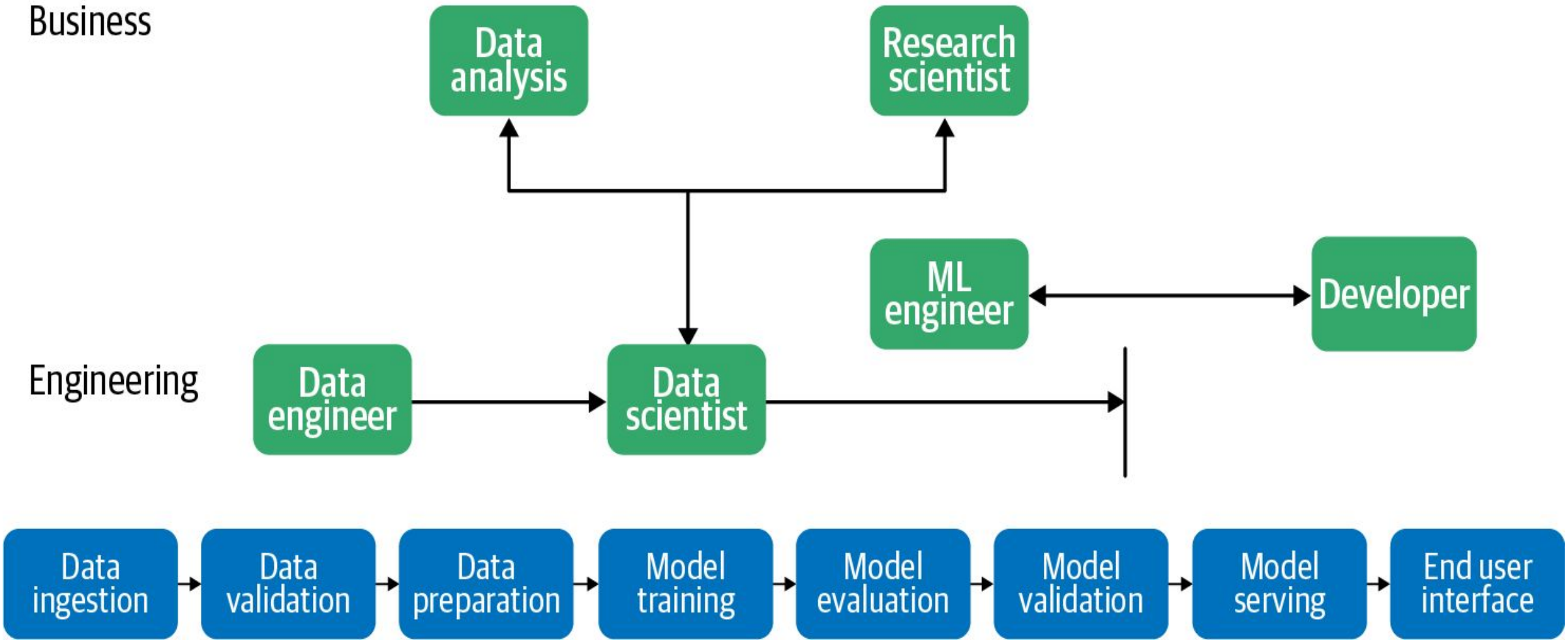
Have I collected enough data to represent the problem I want to solve?

What features of the data did I extract, and will these enable the right predictions?

How will I measure success in my application?

How will the machine learning solution interact with other parts of my research or business product?

Different Roles in an Organization's ML Model Development Process



Machine Learning Process

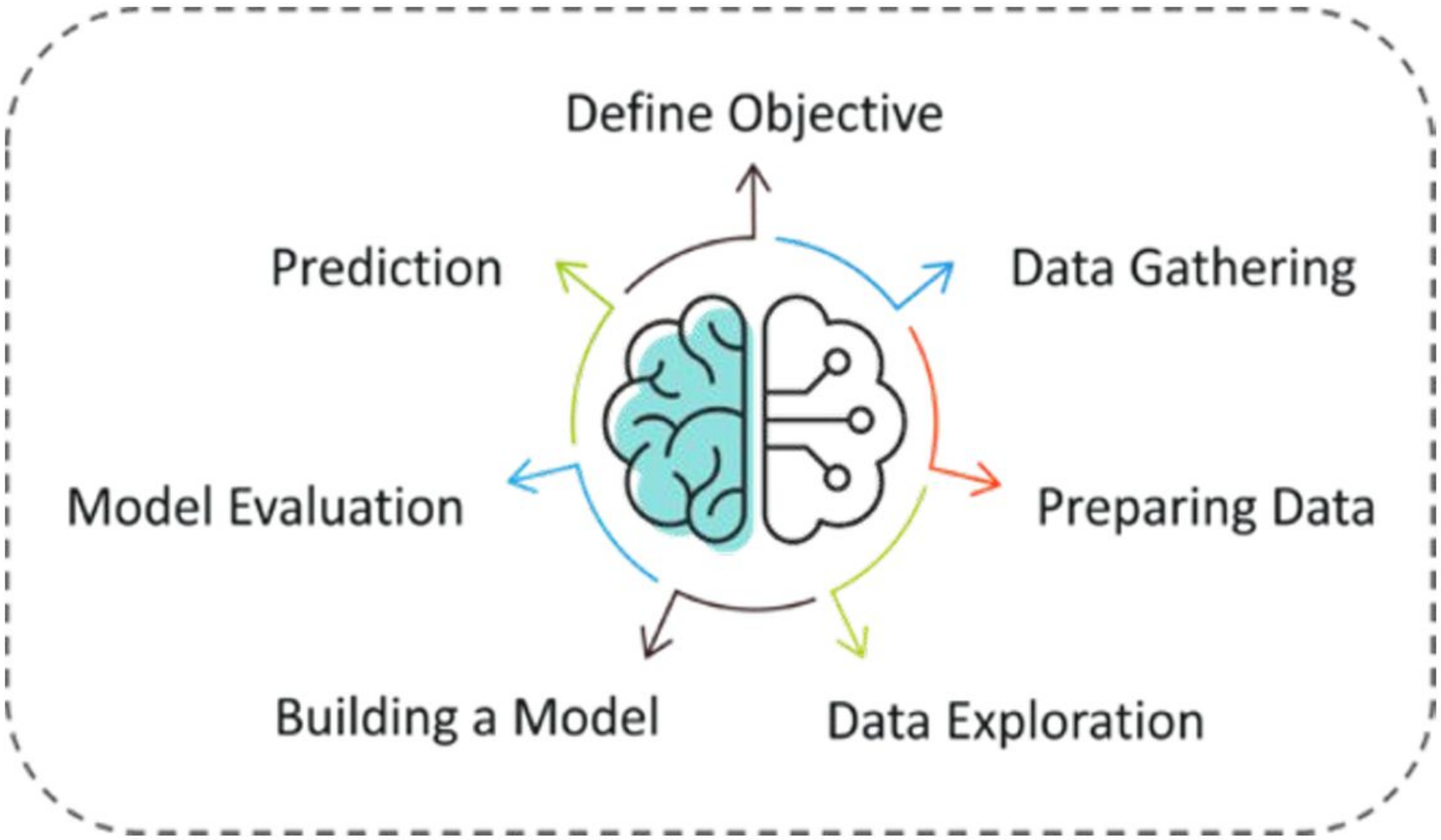


Fig. 1 Machine Learning process REF[1]

Machine Learning Key Terms

Algorithm: (Logic) A Machine Learning algorithm is a set of rules and statistical techniques

Model: (Main component of Machine Learning) Algorithm maps how the decision is taken

Machine Learning Key Terms

- **Predictor Variable**: It is a feature(s)/ input of the data that can be used to predict the output.
- **Response Variable**: It is the feature or the output variable that needs to be predicted by using the predictor variable(s).
- **Training Data:** The machine learning model is built using the training data (Learning).
- **Testing Data:** After the model is trained, it must be tested to evaluate how accurately it can predict an outcome.

Machine Learning Key Terms

Hours- **Independent Variable □ X□ Predictor**

Scores – Dependent Variable□ Y□ Response

student_score.csv

Hours	Scores
2.5	21
5.1	47
3.2	27
8.5	75
3.5	30
1.5	20
9.2	88
5.5	60
8.3	81
2.7	25
7.7	85
5.9	62
4.5	41
3.3	42
1.1	17
8.9	95
2.5	30
1.9	24
6.1	67
7.4	69
2.7	30
4.8	54
3.8	35
6.9	76
7.8	86

Machine Learning Key Terms

Country, Age, Salary

Independent Variable □ X□ Predictor

Purchased

Dependent Variable□ Y□ Response

Country	Age	Salary	Purchased
France	44	72000	No
Spain	27	48000	Yes
Germany	30	54000	No
Spain	38	61000	No
Germany	40		Yes
France	35	58000	Yes
Spain		52000	No
France	48	79000	Yes
Germany	50	83000	No
France	37	67000	Yes

Machine Learning Key Terms

Algorithm: (Logic) A Machine Learning algorithm is a set of rules and statistical techniques

Model: (Main component of Machine Learning) Algorithm maps how the decision is taken

Machine Learning Key Terms

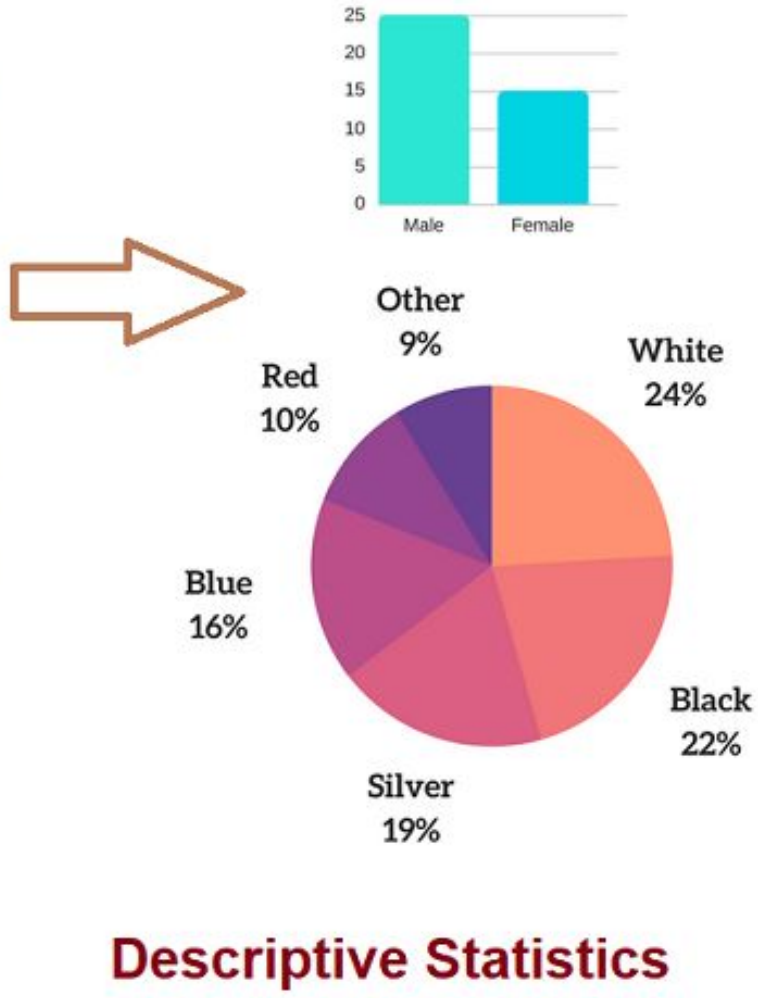
- **Predictor Variable**: It is a feature(s)/ input of the data that can be used to predict the output.
- **Response Variable**: It is the feature or the output variable that needs to be predicted by using the predictor variable(s).
- **Training Data:** The machine learning model is built using the training data (Learning).
- **Testing Data:** After the model is trained, it must be tested to evaluate how accurately it can predict an outcome.

Descriptive Statistics Examples

- Descriptive statistics about a class involve the average score of marks obtained by students for different courses including Maths, Physics, Chemistry.
- Analyzing favorite color of CAR liked by students in a class and representing it in the form of graph or pie chart.

	A	B	C	D
1	Respondent Number	Age	Gender	Favorite Car Color
2	1	22	M	White
3	2	37	F	Silver
4	3	45	F	Black
5	4	62	F	Gray
6	5	28	M	Red
7	6	45	M	Green
8	7	88	F	Brown
9	8	61	M	White
10	9	95	M	Black
11	10	27	M	White
12	11	39	F	Green
13	12	43	M	Brown
14	13	55	F	Black
15	14	59	F	White

RAW DATA



Descriptive Statistics

Descriptive Statistics Types

- Measures of Central Tendency
- Measures of Dispersion or Variation

	A	B	C	D	E
1	Student	Scores			
2	Student 1	72			
3	Student 2	91			
4	Student 3	77			
5	Student 4	80			
6	Student 5	72			
7	Student 6	46			
8	Student 7	81			
9	Student 8	54			
10	Student 9	83			
11	Student 10	46			
12					
13					
14					
15					
16					

Scores	
Mean	70.2
Standard Error	5.050412524
Median	74.5
Mode	72
Standard Deviation	15.9708067
Sample Variance	255.0666667
Kurtosis	-1.005270166
Skewness	-0.646893849
Range	45
Minimum	46
Maximum	91
Sum	702
Count	10

Descriptive Statistics Types

- Measures of Central Tendency
  - Mean:
  - Formula 1: Sum of all values/ No. of samples= Σ (x) /N

Student	Scores
Student 1	72
Student 2	91
Student 3	77
Student 4	80
Student 5	72
Student 6	46
Student 7	81
Student 8	54
Student 9	83
Student 10	46

Sum= 702, N=10, then what is Mean?

Descriptive Statistics Types

Student	Scores
Student 1	72
Student 2	91
Student 3	77
Student 4	80
Student 5	72
Student 6	46
Student 7	81
Student 8	54
Student 9	83
Student 10	46

- Measures of Central Tendency
  - Mean:
  - Formula 1: Sum of all values/ No. of samples= Σ (x) /N
  - If values are repetitive, grouped data and getting frequency, Mean is expressed as

    Formula 2: Mean = $\sum(x_i . f_{x_i}) / N$

x	f_x	$x . f_x$
72	2	=72*2 = 144
91	1	91
77	1	77
80	1	80
46	2	= 46*2 = 92
81	1	81
54	1	54
83	1	83

  - Notations:
    - For Population- μ
    - For sample- $\bar{x}$

Population: The group we are interested in studying
Sample: A subset of Population

Descriptive Statistics Types

Student	Scores
Student 1	72
Student 2	91
Student 3	77
Student 4	80
Student 5	72
Student 6	46
Student 7	81
Student 8	54
Student 9	83
Student 10	46

- Measures of Central Tendency
  - Median: Middle observation
  - Arrange in ascending order: 46, 46, 54, 72, 72, 77, 80, 81, 83, 91
  - If *N* is even- Median is mean of [*N/2 th and (N/2+1) th] elements*

    46, 46, 54, 72, 72, 77, 80, 81, 83, 91

  - If *N* is odd- Median is median is *(N+1)/2th element*

    46, 46, 54, 72, 72, 77, 80, 81, 83

- Measures of Central Tendency
  - Mode: Most repetitive value
  - If there is clear such unique value

Descriptive Statistics Types

- Measures of Central Tendency
  - Mode: Most repetitive value

Student	Scores
Student 1	72
Student 2	91
Student 3	77
Student 4	80
Student 5	72
Student 6	46
Student 7	81
Student 8	54
Student 9	83
Student 10	46

If there is no such clear unique value

$$\text{Mode formula} = L + h\,\frac{(f_m - f_1)}{(f_m - f_1) + (f_m - f_2)}$$

where,

- 'L' is the lower limit of the modal class.
- 'h' is the size of the class interval.
- 'f_m' is the frequency of the modal class.
- 'f_1' is the frequency of the class preceding the modal class.
- 'f_2' is the frequency of the class succeeding the modal class.

Class L to U	f_{Class}
30-40	0
40-50	2
50-60	1
60-70	0
70-80	**4**
80-90	2
90-100	1

Central Tendency Question

- The marks scored by students are given here:

Marks scored Xi	No. of people scoring it Freq of Xi, as F_{Xi}
1	10
2	12
3	20
4	15
5	10

- Compute Mean, Median and mode F_{Xi}

Descriptive Statistics Types

- Measures of Central Tendency
- Measures of Dispersion or Variation

	A	B	C	D	E
1	Student	Scores			
2	Student 1	72			
3	Student 2	91			
4	Student 3	77			
5	Student 4	80			
6	Student 5	72			
7	Student 6	46			
8	Student 7	81			
9	Student 8	54			
10	Student 9	83			
11	Student 10	46			
12					
13					
14					
15					
16					

	Scores
Mean	70.2
Standard Error	5.050412524
Median	74.5
Mode	72
Standard Deviation	15.9708067
Sample Variance	255.0666667
Kurtosis	-1.005270166
Skewness	-0.646893849
Range	45
Minimum	46
Maximum	91
Sum	702
Count	10

Measures of Dispersion or Variation

- Dispersion in statistics describes the **spread** of the data values in a given dataset.
- It reveals the extent to which the values of the individual items differ in a data set.
- **Minimum : 46**
- **Maximum:91**
- **Sum:702**
- **Count:10**
- **Range: Highest-lowest : 45**

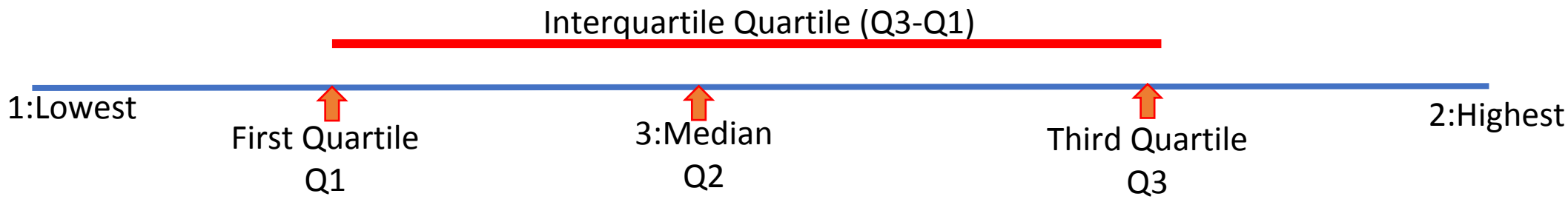
$$\text{Sample standard error} = \frac{\sigma}{\sqrt{n}}, \text{if } \sigma \text{ is known}$$

	A	B	C	D	E
1	Student	Scores			
2	Student 1	72			
3	Student 2	91			
4	Student 3	77			
5	Student 4	80			
6	Student 5	72			
7	Student 6	46			
8	Student 7	81			
9	Student 8	54			
10	Student 9	83			
11	Student 10	46			
12					
13					
14					
15					
16					

Scores	
Mean	70.2
Standard Error	5.050412524
Median	74.5
Mode	72
Standard Deviation	15.9708067
Sample Variance	255.0666667
Kurtosis	-1.005270166
Skewness	-0.646893849
Range	45
Minimum	46
Maximum	91
Sum	702
Count	10

Measures of Dispersion or Variation

- **Quartiles**: The quartile measures the spread of values above and below the median by dividing the distribution into **four groups**.



  - It is a measure of variability around the median.
  - A quartile divides data into three points—a lower quartile (Q1), median (Q2), and upper quartile (Q3)—to form four groups of the dataset.

- **Interquartile range: Q3-Q1**

Find out Q1,Q2 and Q3 for given dataset

45 100 54 72 91 80 77 81 46 83 72

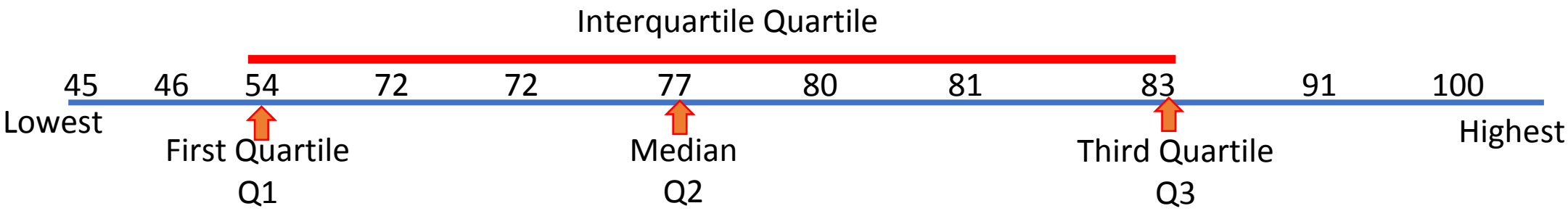
Measures of Dispersion or Variation

- Steps to find Quartile:

Find out Q1,Q2 and Q3 for given dataset

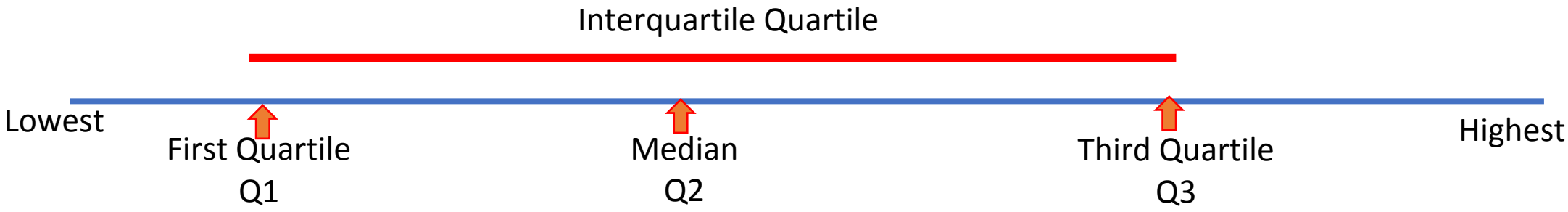
45 100 54 72 91 80 77 81 46 83 72

- 1. Arrange data in order from lowest to highest.
- 2. Find Median (Q2) = (n+1)*50%= (n+1)*0.5 = (n+1)/2 = (11+1)/2= 6th
- 3. Find the median of the data values that fall below Q2, that gives Q1= (n+1)*25%= (n+1)*0.25= (n+1)/4 = (11+1)/4= 3rd
- 4. Find the median of the data values that fall above Q2, that gives Q3 = (n+1)*75%= (n+1)*0.75= (n+1)*(3/4)= (11+1)*(3/4)=9th



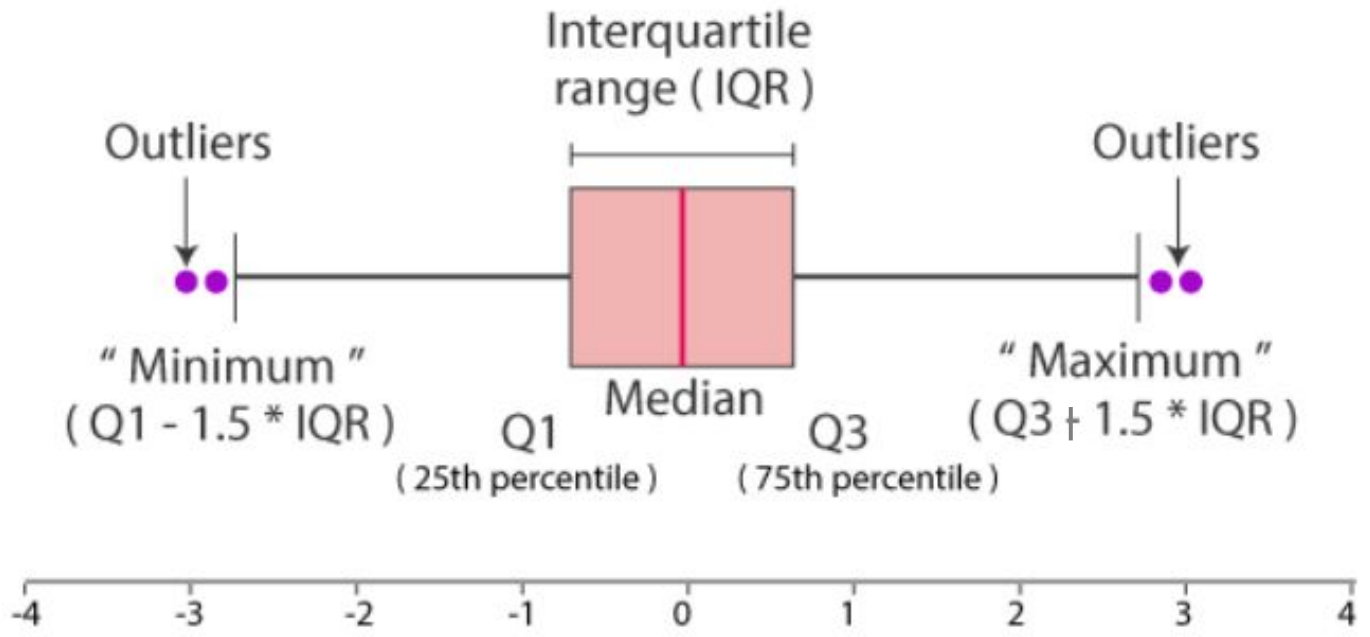
Measures of Dispersion or Variation

- Steps to find Quartile:
- 1. Arrange data in order from lowest to highest.
- 2. Find Median (Q2)
- 3. Find the median of the data values that fall below Q2, that gives Q1
- 4. Find the median of the data values that fall above Q2, that gives Q3
- [1,2,3,4,4,5,6,8,9,10,12]



Boxplot

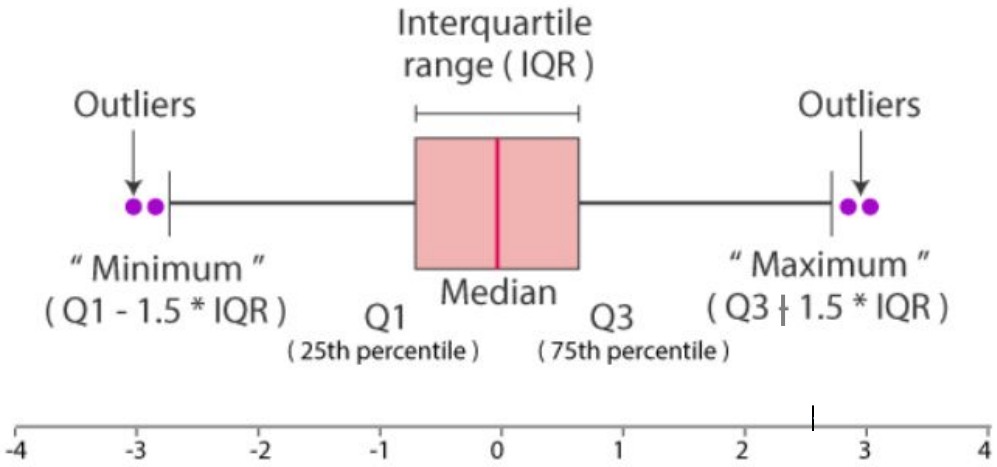
- It is a method for graphically demonstrating the locality, spread and skewness groups of numerical data through their quartiles.
- Box plot (called whiskers) indicating variability outside the upper and lower quartiles.



Different parts of boxplot

Boxplot for detecting Outliers

- **Minimum**: The minimum value in the given dataset
- **First Quartile (Q1)**: The first quartile is **the median of the lower half** of the data set.
- **Median:** The median is the middle value of the dataset, which divides the given dataset into two equal parts. The median is considered as the second quartile.
- **Third Quartile (Q3):** The third quartile is the median of the upper half of the data.
- **Maximum:** The maximum value in the given dataset.



Different parts of boxplot

- **Interquartile Range (IQR):** The difference between the third quartile and first quartile is known as the interquartile range. (i.e.)

$$IQR = Q3-Q1$$

- **Outlier:** The data that falls on the extreme left or right side of the ordered data is called outliers.
- Generally, the outliers fall more than the specified distance from the first and third quartile.
- Outliers are greater than Q3+(1.5 ×IQR) or less than Q1-(1.5× IQR).

Measures of Dispersion or Variation

Measures of Dispersion with RANGE

- Standard Deviation:
  - It provides information on how much variation from the mean exists.
  - However, the standard deviation shows how each value in a dataset varies from the mean.
- Variance: Square of standard deviation

Sample Standard Deviation, s:

$$s = \sqrt{\frac{\Sigma(x_i - \bar{x})^2}{n-1}}$$

Population Standard Deviation, σ:

$$\sigma = \sqrt{\frac{\Sigma(x_i - \mu)^2}{n}}$$

Sample Variance, s^2 :

$$s^2 = \frac{\Sigma(x_i - \bar{x})^2}{n-1}$$

Population Variance, σ^2 :

$$\sigma^2 = \frac{\Sigma(x_i - \mu)^2}{n}$$

Variance and Standard Deviation

Find the population variance for the following set of numbers: 26, 29, 30, 38, 32.

1. Find the Mean $\mu = (26 + 29 + 30 + 38 + 32) / 5.0 = 155/5 = 31$
2. Complete the table.

26	*-5*	*25*
29	*-2*	*4*
30	*-1*	*1*
38	*7*	*49*
32	*1*	*1*

Population Variance, σ^2 :

$$\sigma^2 = \frac{\Sigma(x_i - \mu)^2}{n}$$

Population Standard Deviation, σ:

$$\sigma = \sqrt{\frac{\Sigma(x_i - \mu)^2}{n}}$$

3. Add all numbers of column 3: 25 + 4 + 1 + 49 + 1 = 80
4. Divide it by the number of items in your data set: 80 / 5 = 16. Thus 16 is the population variance, σ^2 for this set of data.
5. Standard Deviation, $\sigma\ is\ square\ root\ of\ variance$: $\sigma = \sqrt{16} = 4.0$

Covariance and Correlation

- Covariance: Shows how the two variables differ

- Correlation: Shows how the two variables are related

Covariance and Correlation

- Covariance: Shows how the two variables differ

$$s_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

Range: -Inf to +Inf
Covariance of *x* with itself is Variance of *x*

- The numerical value of covariance does not have any significance however if it is positive then both variables vary in the same direction else if it is negative then they vary in the opposite direction.

- Correlation: Shows how the two variables are related

- $Corr(x,y) = \frac{Cov(x,y)}{\sigma_x . \sigma_y} = \frac{S\_xy}{\sigma_x . \sigma_y}$

Range: -1 to +1
If value is -1/ + 1: Linear relationship
If value is 0: No linear relationship

Covariance Computation

$$s_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

For given data, let us find Covariance.

	S&P 500	ABC Corp.
2013	1,692	68
2014	1,978	102
2015	1,884	110
2016	2,151	112
2017	2,519	154

1. Calculate the mean (average) prices for each asset.
   Mean(S&P), m1 = 2044.80,
   Mean(ABC Corp.) m2 = 109.20,
2. For each security, find the difference between each value and mean price.

Each entry in **column a** is entry in (column S&P 500) – m1 and

Each entry in **column b** is entry in (column S&P 500) – m2

	S&P 500	ABC Corp.	a	b	a x b
2013	1,692	68	-352.80	-41.20	14,535.36
2014	1,978	102	-66.80	-7.20	480.96
2015	1,884	110	-160.80	0.80	-128.64
2016	2,151	112	106.20	2.80	297.36
2017	2,519	154	474.20	44.80	21,244.16
Mean	**2,044.80**	**109.20**	**Sum**		**36,429.20**

Covariance Computation

$$s_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

For given data, let us find Covariance.

	S&P 500	ABC Corp.
2013	1,692	68
2014	1,978	102
2015	1,884	110
2016	2,151	112
2017	2,519	154

3. Multiply the results obtained in the previous step as a×b.
4. Take the sum of all entries in column a×b.
5. Using the number calculated in step 4, find the covariance.

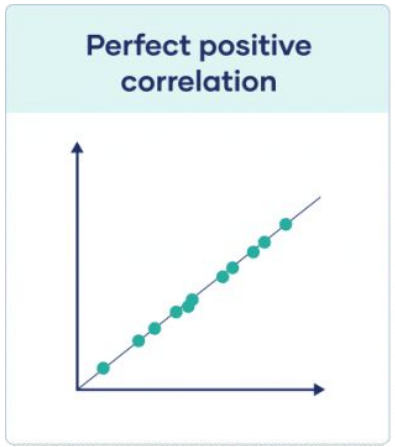
	S&P 500	ABC Corp.	a	b	a x b
2013	1,692	68	-352.80	-41.20	14,535.36
2014	1,978	102	-66.80	-7.20	480.96
2015	1,884	110	-160.80	0.80	-128.64
2016	2,151	112	106.20	2.80	297.36
2017	2,519	154	474.20	44.80	21,244.16
Mean	**2,044.80**	**109.20**	**Sum**		**36,429.20**

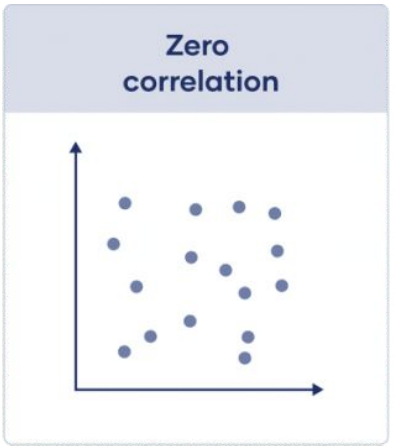
$$\text{Cov}(\text{S\&P 500}, \text{ABC Corp.}) = \frac{36{,}429.20}{5-1} = 9{,}107.30$$

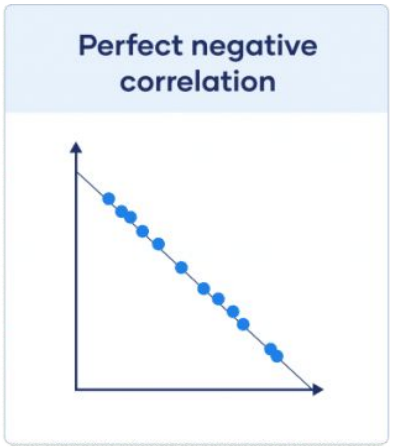
The positive value of covariance indicates that the price of the stock and the S&P 500 tend to move in the same direction.

Correlation Positive/Negative

- The more time a student spends watching TV, the lower their exam scores tend to be. In other words, the variable time spent watching TV and the variable exam score have a negative correlation. As time spent watching TV increases, exam scores decrease.
- The more time an individual spends running, the lower their body fat tends to be. In other words, the variable running time and the variable body fat have a negative correlation. As time spent running increases, body fat decreases.
- The height and weight of people have positive correlation with each other.
- The correlation between the temperature and total ice cream sales is positive. In other words, when it's hotter outside the total ice cream sales of companies tends to be higher since more people buy ice cream when it's hot out.

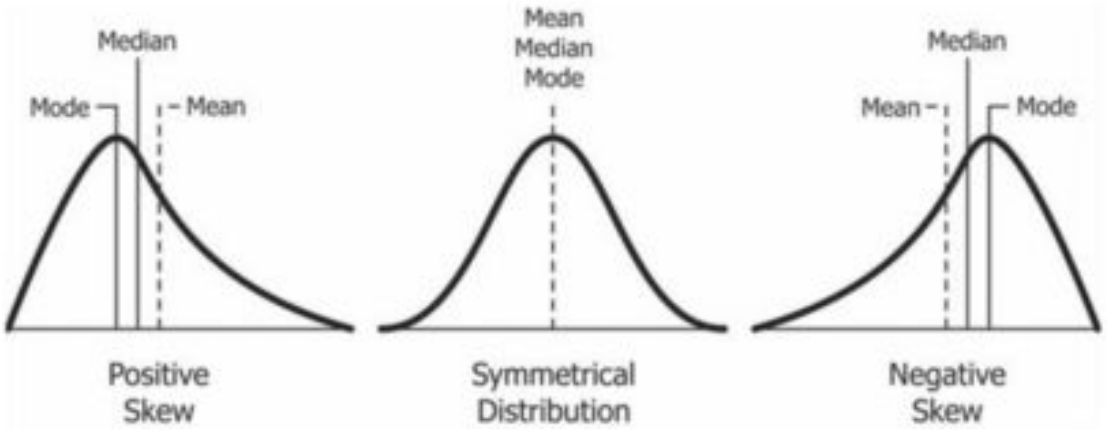






Skewness

- It is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean.
- The skewness value can be positive, zero, negative, or undefined.
- For Unimodal Distribution:
  - **Symmetric**: Even on either side of mean (mean= median= mode)
  - **Positive skew**: tail is on the right side of the distribution (mean > median)
  - **Negative skew**: tail is on the left side of the distribution (mean < median)



	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891
5	2.9	56642
6	3.0	60150
7	3.2	54445
8	3.2	64445
9	3.7	57189
10	3.9	63218
11	4.0	55794
12	4.0	56957
13	4.1	57081
14	4.5	61111
15	4.9	67938
16	5.1	66029
17	5.3	83088
18	5.9	81363
19	6.0	93940
20	6.8	91738
21	7.1	98273
22	7.9	101302
23	8.2	113812
24	8.7	109431
25	9.0	105582
26	9.5	116969
27	9.6	112635
28	10.3	122391
29	10.5	121872

Questions:

For the following given data set (Only for Y_test), you have to calculate mean, std deviation, variance, MAE, MSE, RMSE and R2 Score.

```
df=pd.DataFrame({'Actual':y_test,'Predicted':y_pred})
print(df)
```

	Actual	Predicted
0	37731	40748.961841
1	122391	122699.622956
2	57081	64961.657170
3	63218	63099.142145
4	116969	115249.562855
5	109431	107799.502753

Linear Regression

- Linearity
- Linear Regression estimates real values (cost of houses, number of calls, total sales , etc.)
- Base is continuous variables.
- Establish a relationship between the independent and dependent variables by fitting the best line.
- This best-fit line is known as the regression line and is represented by a linear equation **Y= b1X + b0.**

Linear Regression : Example

- Linear equation Y= b1X + b0.
- Remember your childhood
- Teacher assigns a task- Arrange all the children in the class height-wise
- What you as a child would have done?
- Look (visually analyze) at the height and build of people and arrange them using a combination of these visible parameters

Linear Regression : Example

- This is a linear regression in real life!
- The child has figured out that height and build would be correlated to the weight by a relationship, which looks like the equation above.
- Y – Dependent Variable
- a – Slope
- X – Independent variable
- b – Intercept (the predicted value of Y when the X is 0)

Linear Regression : Example

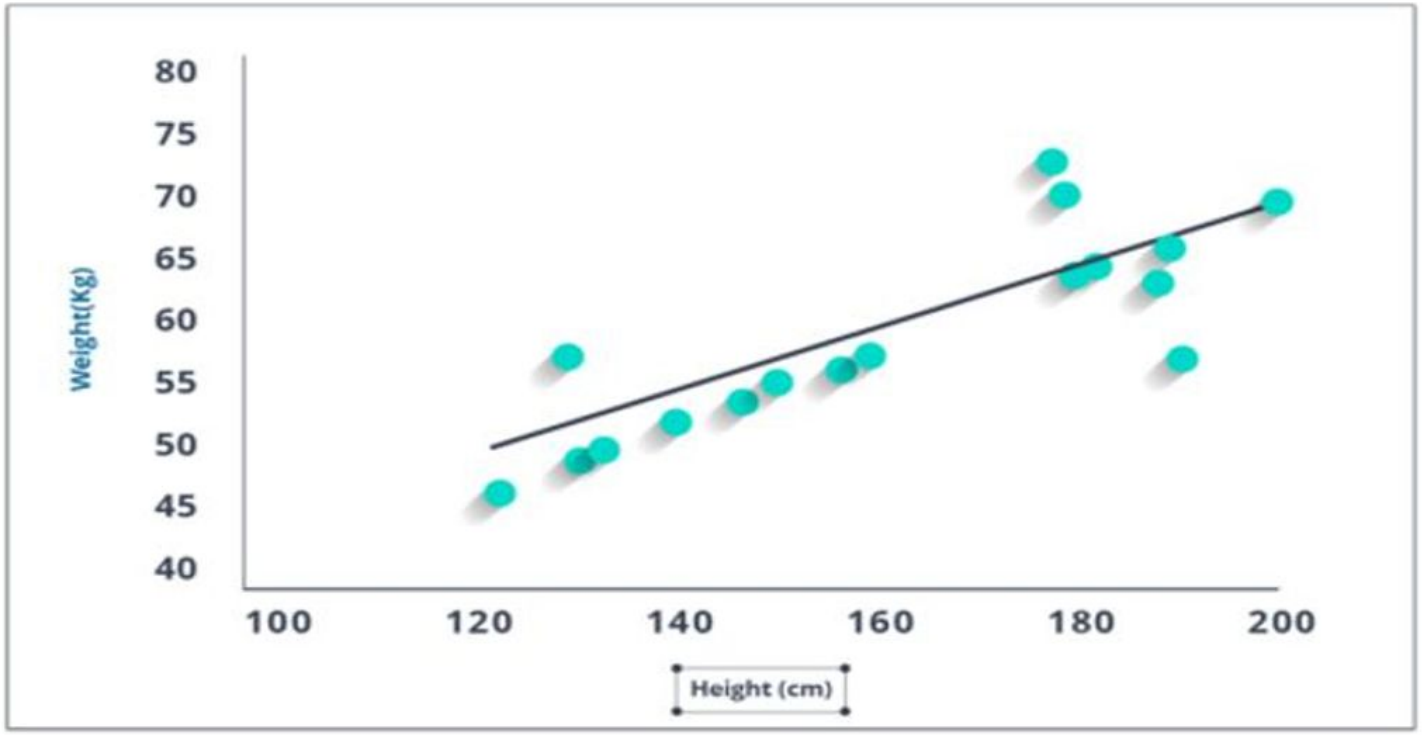


Fig. 8 Linear Regression Plot REF[4]

Use Case: Score Prediction

Problem Statement

Build a model to predict salary based on the number of years of experience.

Data

Use the Salary Data dataset and analyze the relationship between YearsExperience and Salary variables using a linear regression

Machine Learning Key Terms

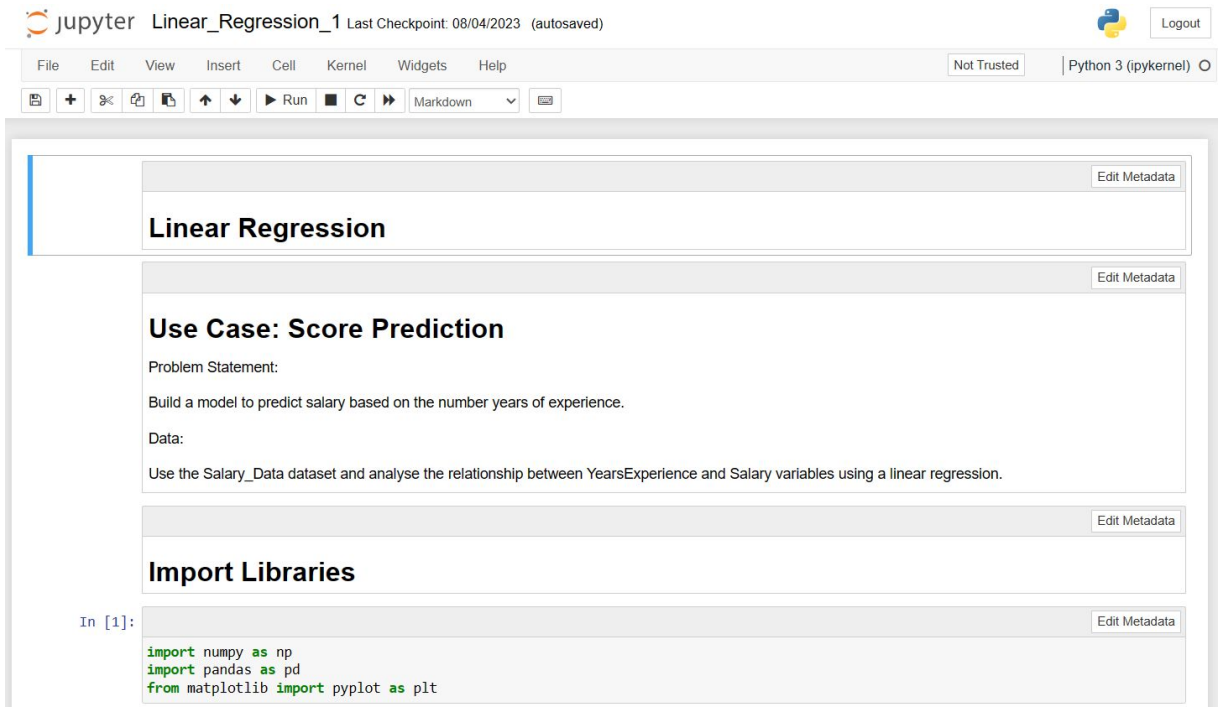
Hours- Independent Variable □ X□ Predictor

Scores – Dependent Variable□ Y□ Response

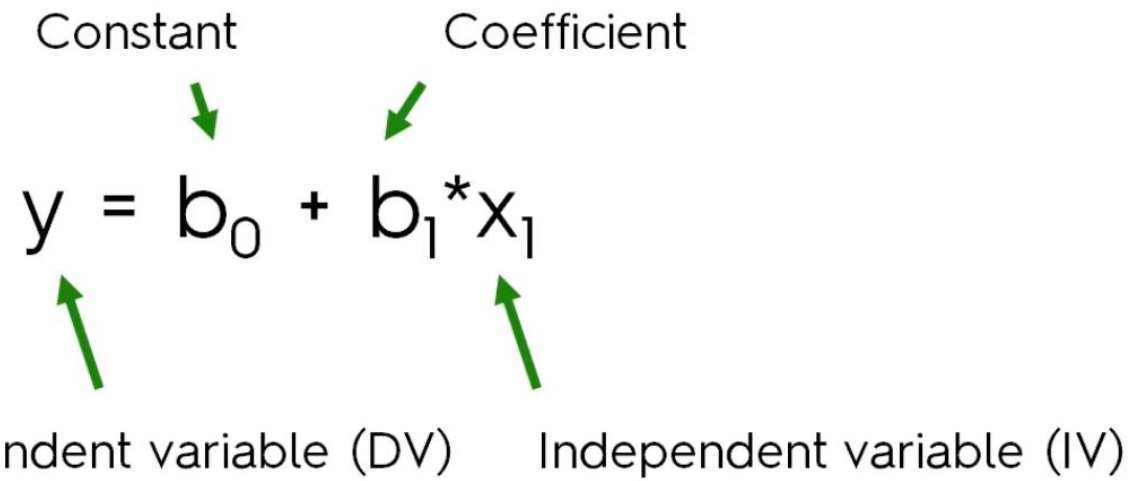
student_score.csv

Hours	Scores
2.5	21
5.1	47
3.2	27
8.5	75
3.5	30
1.5	20
9.2	88
5.5	60
8.3	81
2.7	25
7.7	85
5.9	62
4.5	41
3.3	42
1.1	17
8.9	95
2.5	30
1.9	24
6.1	67
7.4	69
2.7	30
4.8	54
3.8	35
6.9	76
7.8	86

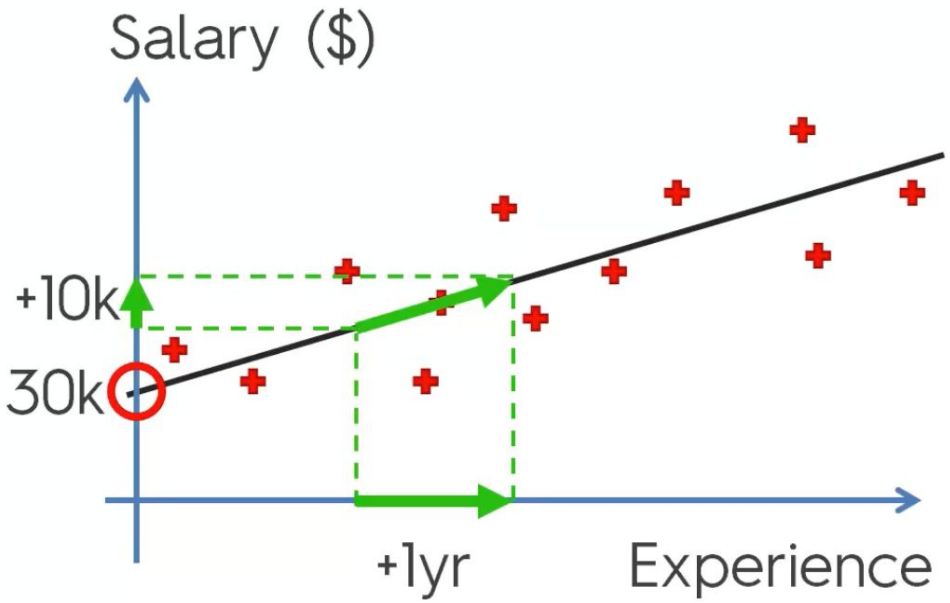
Linear Regression : Hands On



Simple Linear Regression



Simple Linear Regression:

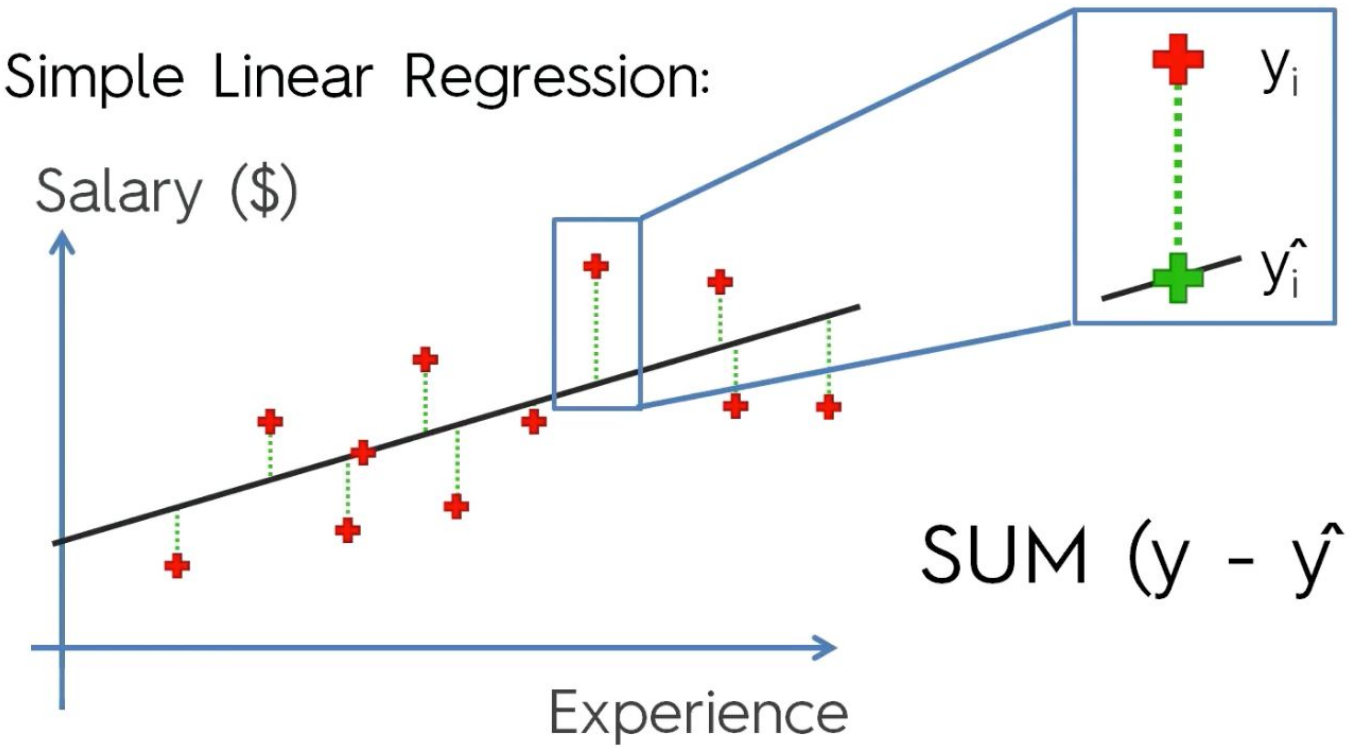


$$y = b_0 + b_1 * x$$

Salary = b_0 + b_1 *Experience

Ordinary Least Squares

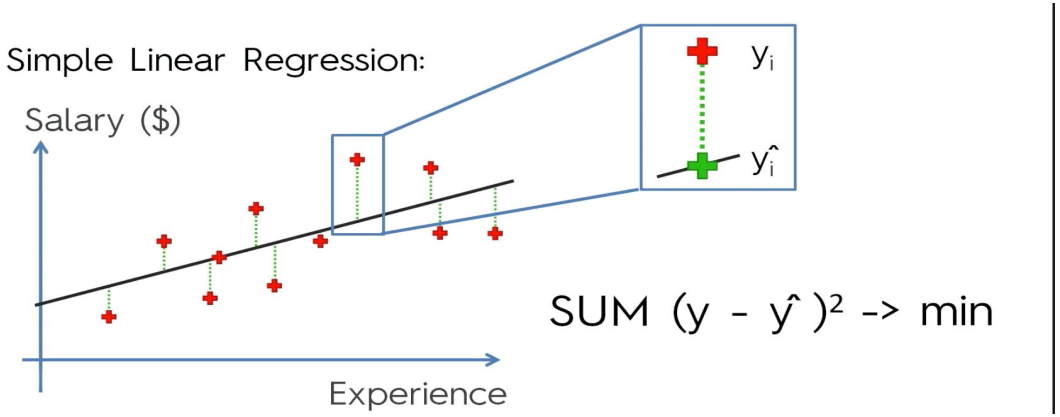
Simple Linear Regression:



$$SUM\ (y - \hat{y})^2 \rightarrow min$$

Evaluation Metrics

- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)



MAE= [∑ abs(actual_value - prdicted_value)] / n

mi=abs(actual_value-predicted_value)

m1=abs(37731.0 - 40748.961841)=3017.961841

m2=abs(122391.0 - 122699.622956)=308.6229559

m3=abs(57081.0 - 64961.657170)=7880.65717

m4=abs(63218.0 - 63099.142145)=118.8578551

m5=abs(116969.0 - 115249.562855)=1719.437145

m6=abs(109431.0 - 107799.502753)=1631.497247

MAE=(m1 + m2 + m3 + m4 + m5 + m6)/ 6

MAE=(3017.961841+308.6229559+7880.65717+118.8578551+1719.437145+1631.497247) / 6

MAE=(14677.03421) / 6

MAE=2446.172369

MSE= [∑ (actual_value - prdicted_value)²] / n

mi= (actual_value - prdicted_value)²

m1= (37731.0 - 40748.961841)² = 9108093.672

m2= (122391.0 - 122699.622956)² = 95248.12893

m3= (57081.0 - 64961.657170)² = 62104757.43

m4= (63218.0 - 63099.142145)² = 14127.18973

m5= (116969.0 - 115249.562855)² = 2956464.097

m6= (109431.0 - 107799.502753)² = 2661783.266

MSE=(m1 + m2 + m3 + m4 + m5 + m6) / 6

MSE=(9108093.672+95248.12893+62104757.43+14127.18973+2956464.097+2661783.266) / 6

MSE=(76940473.79) / 6

MSE=12823412.3

Root Mean Squared Error (RMSE)

RMSE= SQRT(MSE) RMSE= SQRT(12823412.3) RMSE= 3580.979237

R-squared?

What is R-squared?

R squared or Coefficient of determination, or R^2 is a measure that provides information about the goodness of fit of the regression model. In simple terms, it is a statistical measure that tells how well the plotted regression line fits the actual data. R squared measures how much the variation is there in predicted and actual values in the regression model.

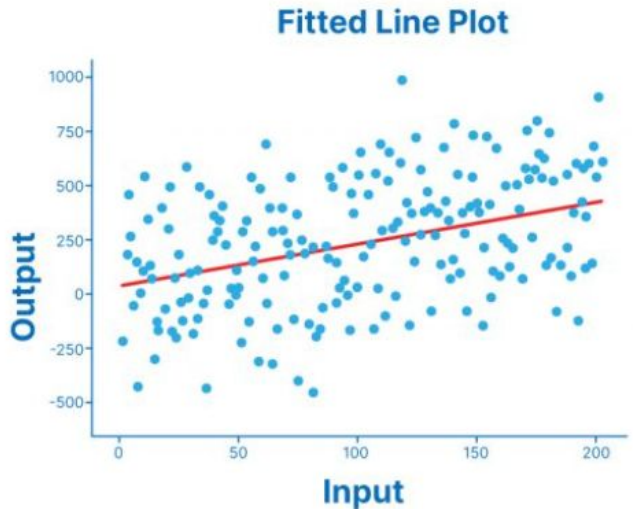
•R-squared values range from 0 to 1, usually expressed as a percentage from 0% to 100%.

$$\text{R-Squared} = 1-\left(\frac{SSR}{SST}\right)=1-\frac{\sum_{i=1}^{n}(yi-\widehat{yi})^2}{\sum_{i=1}^{n}(yi-\overline{yi})^2}$$

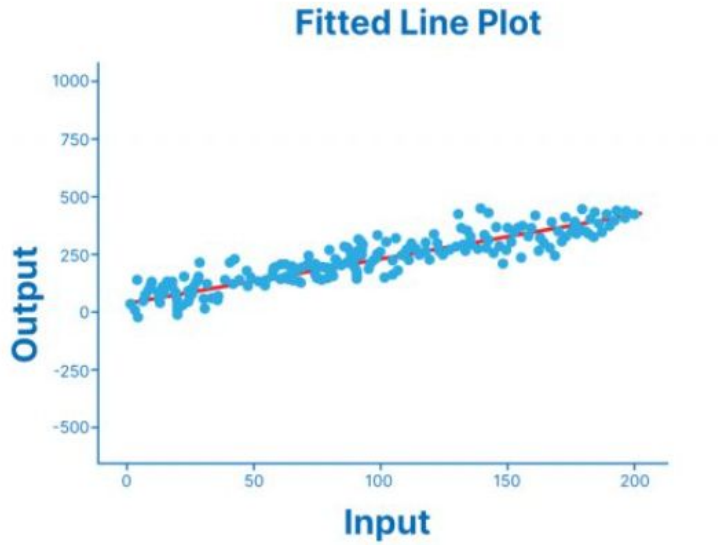
Predicated

Actual Mean

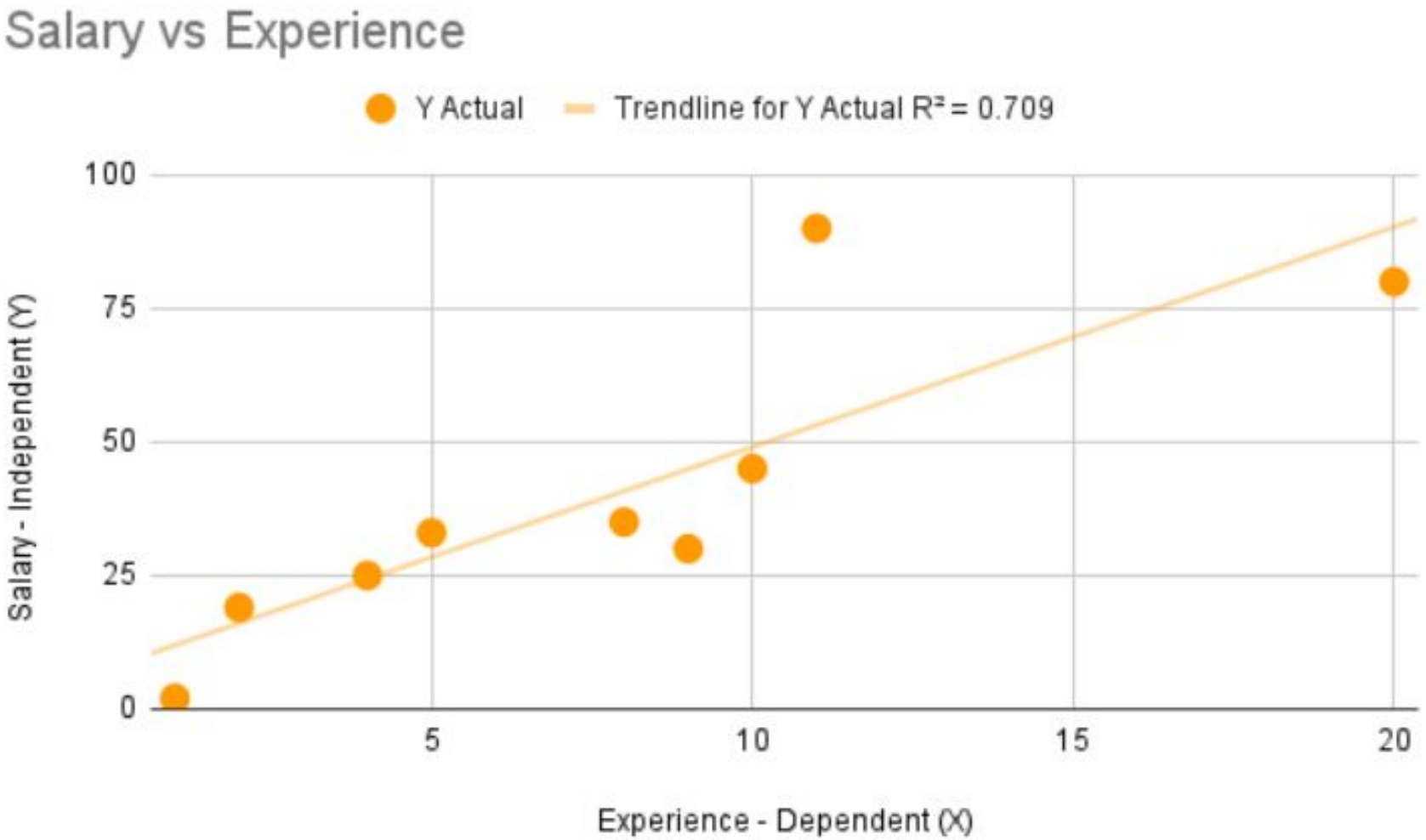
where:
SSR is the sum of squared residuals (i.e., the sum of squared errors)
SST is the total sum of squares (i.e., the sum of squared deviations from the mean)



R squared value will be less than 0.5 or near 0.



R squared will be close to 1

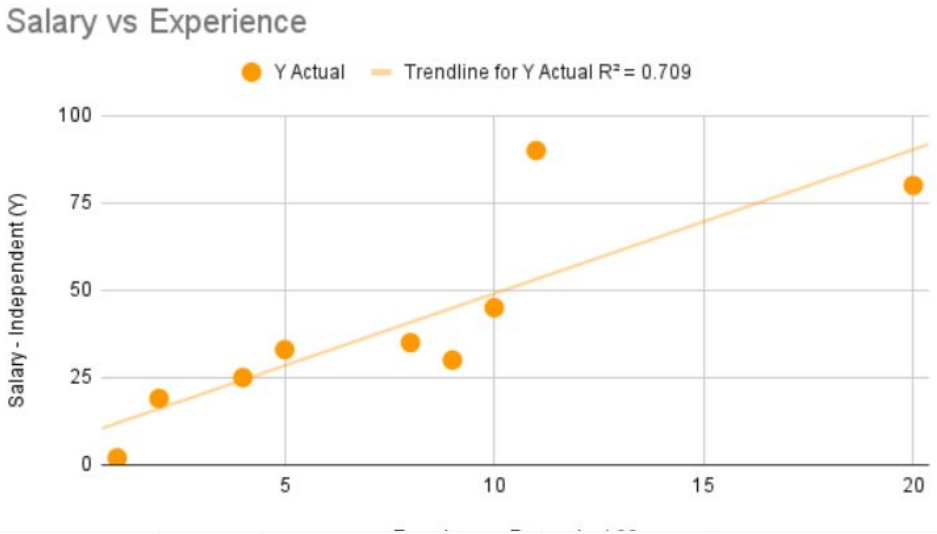


SSR

$$\sum_{i=1}^{n} (yi - \hat{yi})^2$$

SST

$$\sum_{i=1}^{n} (yi - \bar{yi})^2$$



R-Squared = 1-(SSR/SST)

Suppose we have a data set having values of X and Y.

1. We have to find Xi(mean) and Yi(mean).

2. Calculate Xi-Xi and Yi-Yi and then do (Xi-Xi)2

3. Now calculate **(Xi-Xi)(Yi-Yi)**

Now we have to calculate R squared so let's try to calculate it

x_i	y_i	$xi- \bar{x_i}$	$y_i-\bar{y_i}$	$(xi- \bar{x_i})^2$	$(xi- \bar{x_i})(y_i-\bar{y_i})$
11	90	3.22	50.11	10.38	161.47
10	45	2.22	5.11	4.94	11.36
2	19	-5.78	-20.89	33.38	120.69
8	35	0.22	-4.89	0.05	-1.09
4	25	-3.78	-14.89	14.27	56.25
20	80	12.22	40.11	149.38	490.25
1	2	-6.78	-37.89	45.94	256.80
9	30	1.22	-9.89	1.49	-12.09
5	33	-2.78	-6.89	7.72	19.14

SSR

$$\sum_{i=1}^{n} (yi - \hat{yi})^2$$

SST

$$\sum_{i=1}^{n} (yi - \bar{yi})^2$$

$\hat{y_i}$	$y_i - \hat{y_i}$	SSR	$y_i-\bar{y_i}$	SST
53.17	36.83	1356.46	50.11	2511.12
49.05	-4.05	16.39	5.11	26.12
16.07	2.93	8.56	-20.89	436.35
37.10	-2.10	4.39	-4.89	23.90
20.61	4.39	19.29	-14.89	221.68
86.56	-6.56	42.97	40.11	1608.90
8.24	-6.24	38.98	-37.89	1435.57
41.22	-11.22	125.82	-9.89	97.79
24.73	8.27	68.39	-6.89	47.46

$$\text{R-Squared} = 1-\left(\frac{SSR}{SST}\right)=1-\frac{\sum_{i=1}^{n}(yi-\widehat{yi})^2}{\sum_{i=1}^{n}(yi-\overline{yi})^2}$$

where:
SSR is the sum of squared residuals (i.e., the sum of squared errors)
SST is the total sum of squares (i.e., the sum of squared deviations from the mean)

```
actual_minus_predicted = sum((y_test - y_pred)**2)
actual_minus_actual_mean = sum((y_test - y_test.mean())**2)
r2 = 1 - actual_minus_predicted/actual_minus_actual_mean
print('R²:', r2)
```

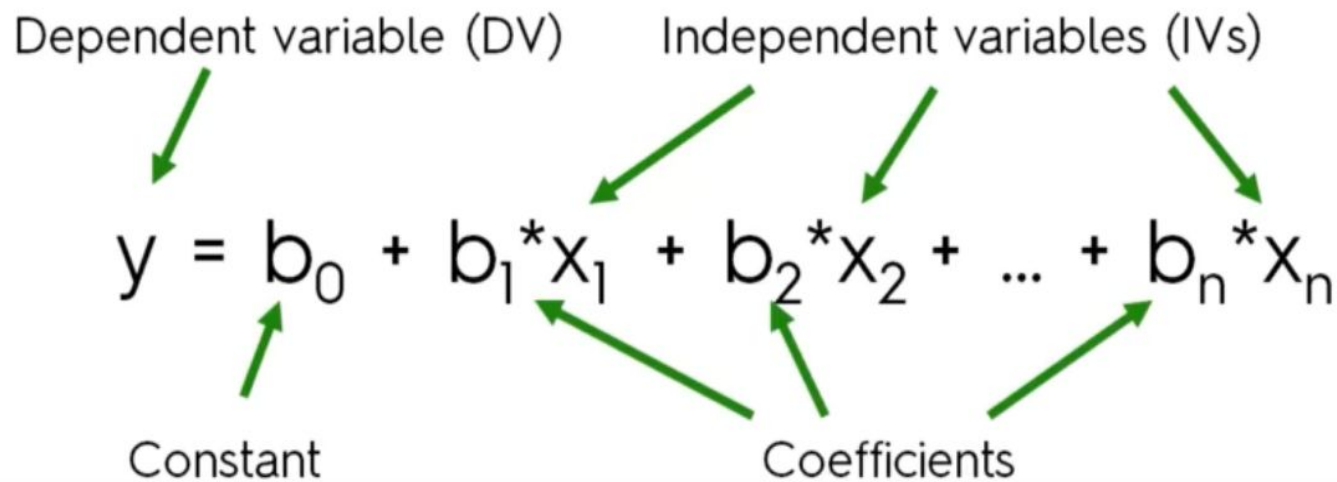
```
R²: 0.988169515729126
```

Salary Dataset write a program for simple linear regression in python

Simple Linear Regression

$$y = b_0 + b_1 * x_1$$

Multiple Linear Regression



Multiple Linear Regression

Advertising_sales.csv

Independent Variables:
- TV
- Radio
- Newspaper

Dependent Variable:
- Sales

	TV	Radio	Newspaper	Sales
1	230.1	37.8	69.2	22.1
2	44.5	39.3	45.1	10.4
3	17.2	45.9	69.3	9.3
4	151.5	41.3	58.5	18.5
5	180.8	10.8	58.4	12.9
6	8.7	48.9	75	7.2
7	57.5	32.8	23.5	11.8
8	120.2	19.6	11.6	13.2
9	8.6	2.1	1	4.8
10	199.8	2.6	21.2	10.6
11	66.1	5.8	24.2	8.6
12	214.7	24	4	17.4
13	23.8	35.1	65.9	9.2
14	97.5	7.6	7.2	9.7
15	204.1	32.9	46	19
16	195.4	47.7	52.9	22.4
17	67.8	36.6	114	12.5
18	281.4	39.6	55.8	24.4
19	69.2	20.5	18.3	11.3
20	147.3	23.9	19.1	14.6
21	218.4	27.7	53.4	18
22	237.4	5.1	23.5	12.5
23	13.2	15.9	49.6	5.6
24	228.3	16.9	26.2	15.5
25	62.3	12.6	18.3	9.7
26	262.9	3.5	19.5	12
27	142.9	29.3	12.6	15
28	240.1	16.7	22.9	15.9
29	248.8	27.1	22.9	18.9

Multiple Linear Regression

Y= b1*X1 + b2*X2 + b3*X3 + b0

X1= TV

X2= Radio

X3= Newspaper

and

Y=Sales

Sales= b1*(TV) + b2*(Radio) + b3*(Newspaper) + b0

Multiple Linear Regression

Problem Statement :-

Data of 50 companies.
R&D Spends/Administration/Marketing Spend/State Profit
(Independent Variables) (Dependent Variable)

A VC Fund is interested in investing in these companies. But has questions like :-
Where companies perform better? Are these companies are those who spend more money on R&D Spend or on Marketing Spend ?
Help VC Fund to build a model

Dummy Variable/Categorical Variable/One hot encoding

Profit	R&D Spend	Admin	Marketing	State
192,261.83	165,349.20	136,897.80	471,784.10	New York
191,792.06	162,597.70	151,377.59	443,898.53	California
191,050.39	153,441.51	101,145.55	407,934.54	California
182,901.99	144,372.41	118,671.85	383,199.62	New York
166,187.94	142,107.34	91,391.77	366,168.42	California

Dummy Variables

New York	California
1	0
0	1
0	1
1	0
0	1

$$y = b_0 + b_1*x_1 + b_2*x_2 + b_3*x_3 + b_4*D_1$$

Dummy Variable Trap

Not Truly Independent Variables
Multicollinearity = High correlation between 2 or more independent variables

Profit	R&D Spend	Admin	Marketing	State	Dummy Variables: New York	Dummy Variables: California
192,261.83	165,349.20	136,897.80	471,784.10	New York	1	0
191,792.06	162,597.70			ia	0	1
191,050.39	153,441.51			ia	0	1
182,901.99	144,372.41			rk	1	0
166,187.94	142,107.34			ia	0	1

$D_2 = 1 - D_1$

$$y = b_0 + b_1*x_1 + b_2*x_2 + b_3*x_3 + b_4*D_1 + b_5*D_2$$

Dummy Variable Trap

Profit	R&D Spend	Admin	Marketing	State
192,261.83	165,349.20	136,897.80	471,784.10	New York
191,792.06	162,597.70	151,377.59	443,898.53	California
191,050.39	153,441.51	101,145.55	407,934.54	California
182,901.99	144,372.41	118,671.85	383,199.62	New York
166,187.94	142,107.34	91,391.77	366,168.42	California

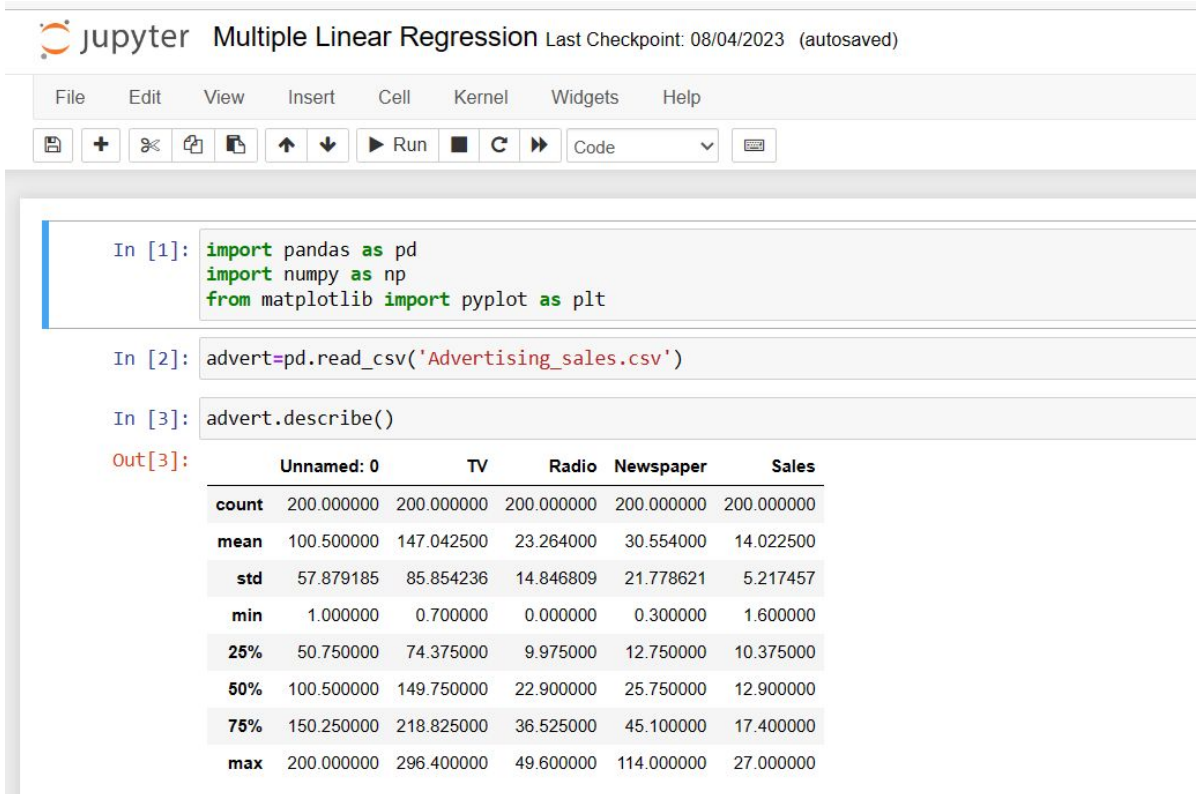
Dummy Variables

New York	California
1	0
0	1
0	1
1	0
0	1

$$y = b_0 + b_1*x_1 + b_2*x_2 + b_3*x_3 + b_4*D_1 + b_5*D_2$$

Always omit one dummy variable

Multiple Linear Regression: Hands-On



	Unnamed: 0	TV	Radio	Newspaper	Sales
count	200.000000	200.000000	200.000000	200.000000	200.000000
mean	100.500000	147.042500	23.264000	30.554000	14.022500
std	57.879185	85.854236	14.846809	21.778621	5.217457
min	1.000000	0.700000	0.000000	0.300000	1.600000
25%	50.750000	74.375000	9.975000	12.750000	10.375000
50%	100.500000	149.750000	22.900000	25.750000	12.900000
75%	150.250000	218.825000	36.525000	45.100000	17.400000
max	200.000000	296.400000	49.600000	114.000000	27.000000

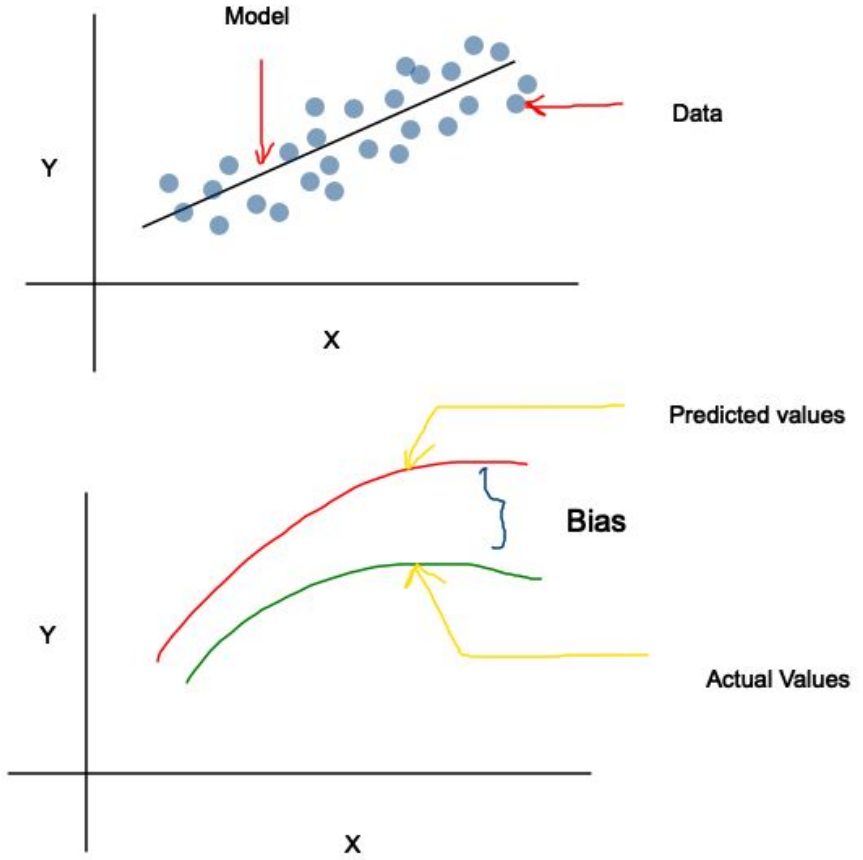
Bias and Variance

Bias-

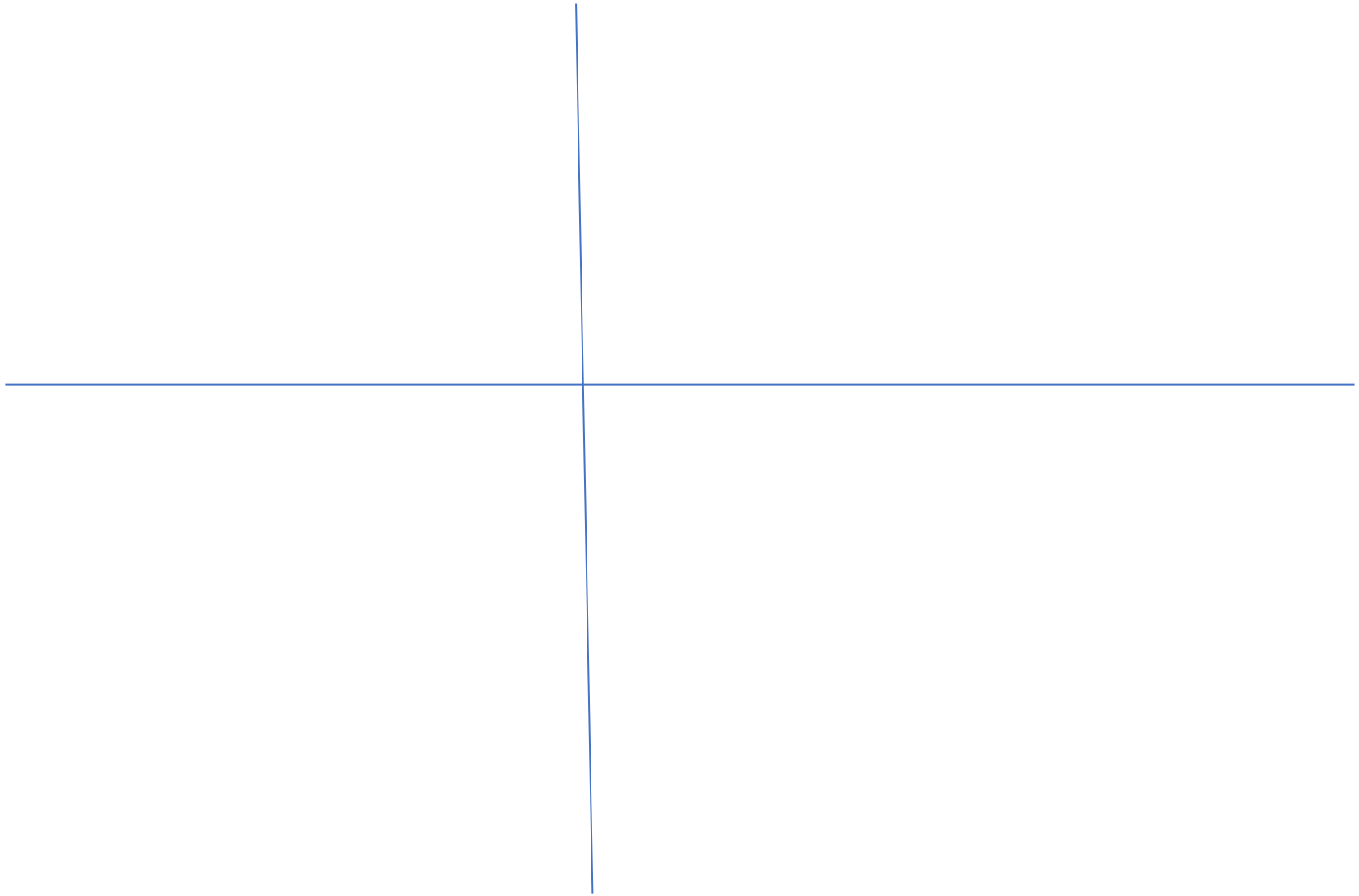
- Prediction Error, introduced in the model
- It is the difference between predicted value and actual value.
- Taring Data.

Variance-

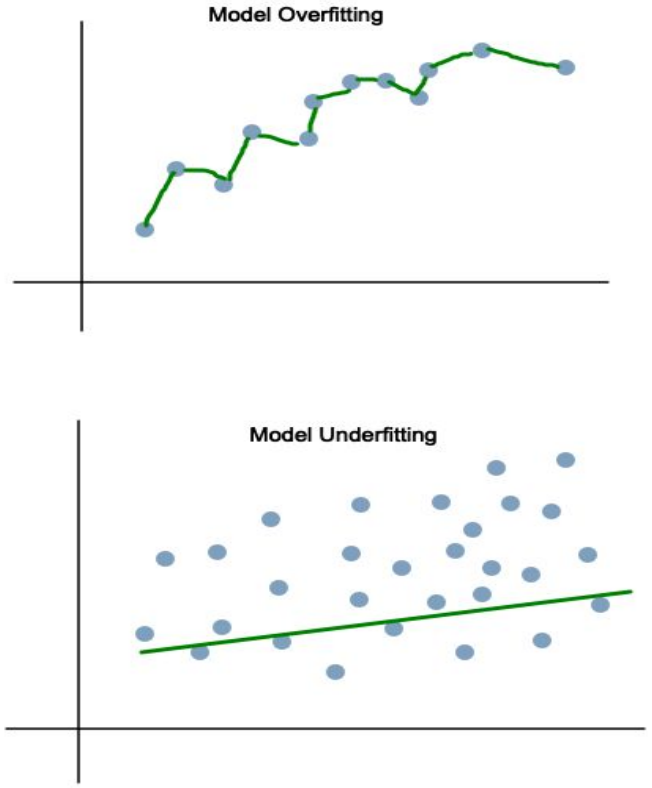
- Test Data.
- Error differences



Model Overfitting and Underfitting



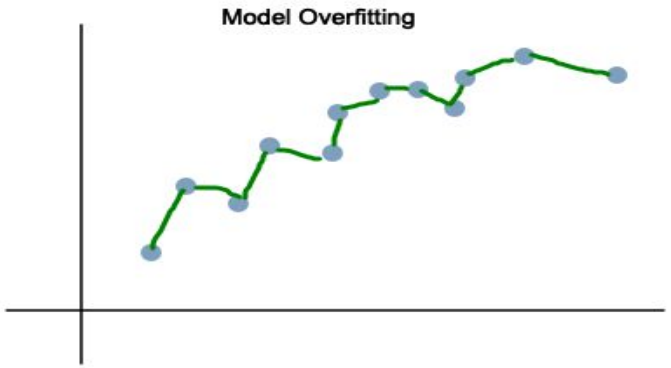
Model Overfitting and Underfitting

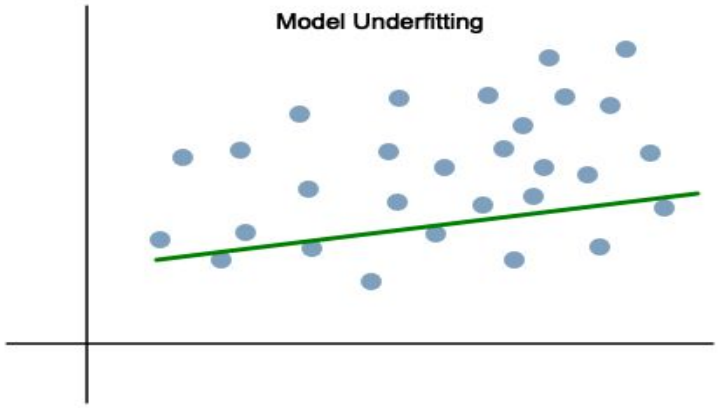


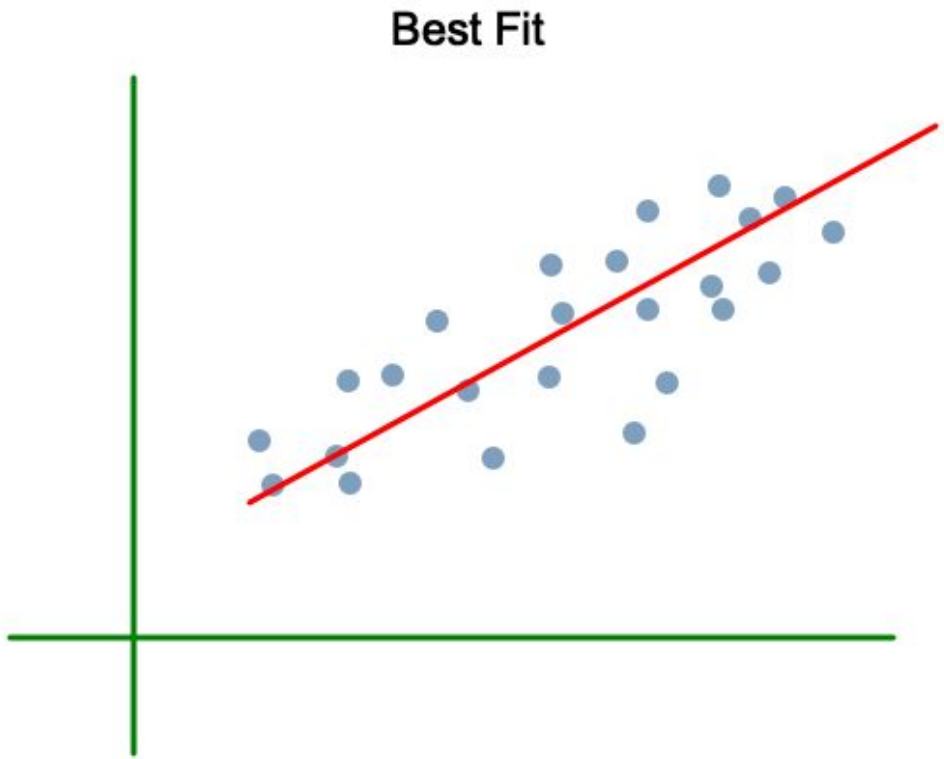
How to avoid Underfitting in Model

Increase number of features

Increase training time of model



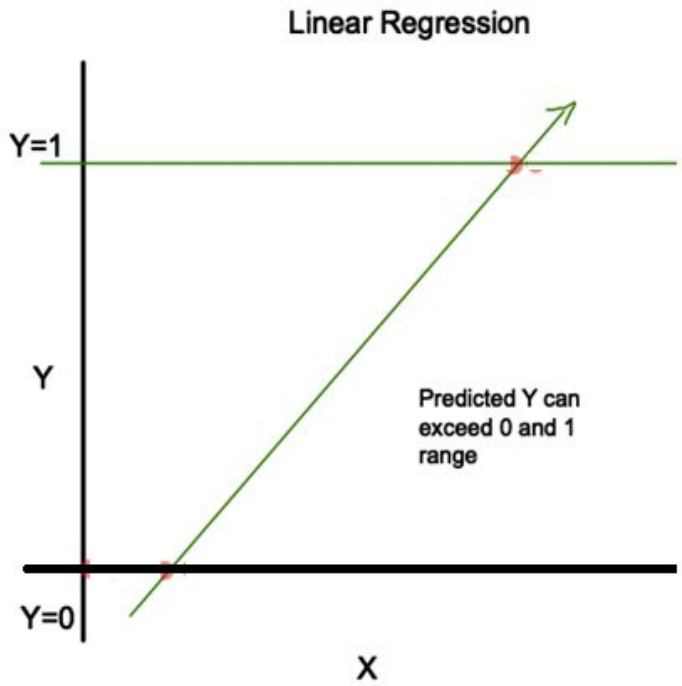


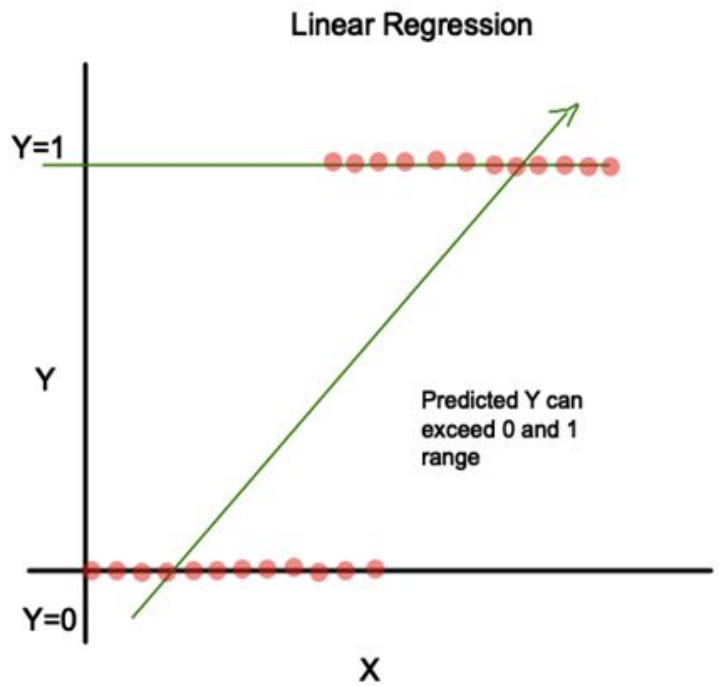


Logistic Regression

Limitations: Linear Regression

- Linear regression is not capable of predicting probability
- However, in many situations, the response variable is qualitative or, in other words, categorical.
- If you use linear regression to model a binary response variable, for example, the resulting model may not restrict the predicted Y values within 0 and 1.
- For example, gender is qualitative, taking on values male or female
- Logistic Regression comes in the picture


Linear Regression
Y=1
Y
Predicted Y can exceed 0 and 1 range
Y=0
X


Linear Regression
Y=1
Y
Predicted Y can exceed 0 and 1 range
Y=0
X

Logistic Regression

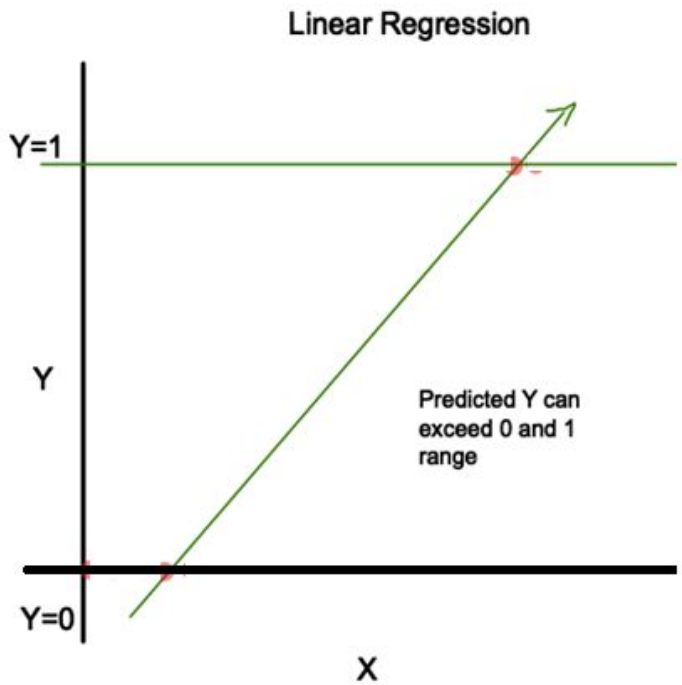
- Don’t go by its name!
- It is a classification, and not a regression algorithm.
- It is used to estimate or predict discrete values (Binary values like 0/1, yes/no, true/false)
- Set of independent variable(s) is given
- Based on Probability
- Also known as logit regression

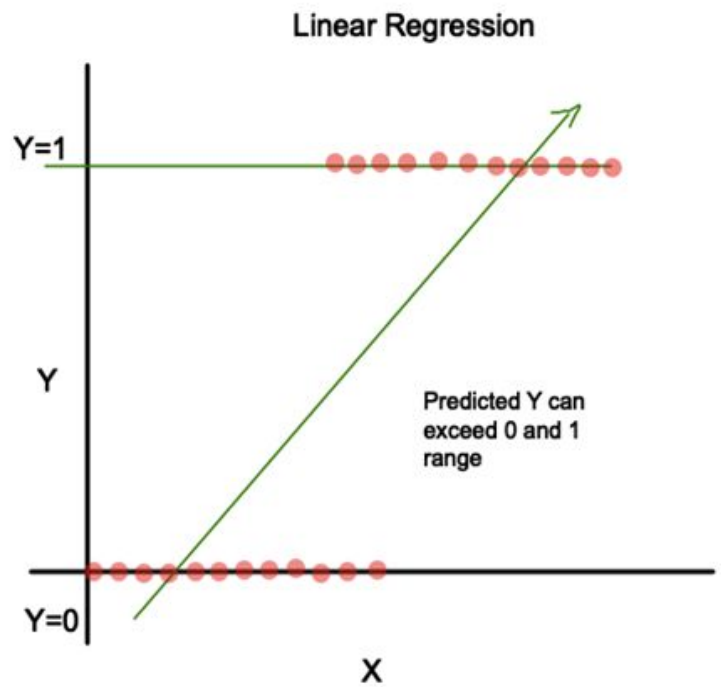
Logistic Regression

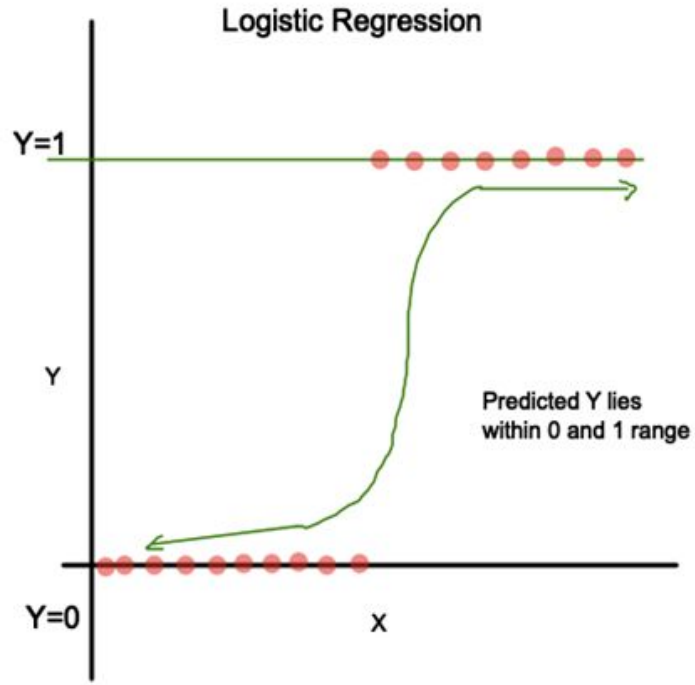
- It measures the probability of a binary response as the value of response variable based on the mathematical equation relating it with the predictor variables.
- The general mathematical equation for logistic regression is −
- y = 1/(1+e^-(a+b1x1+b2x2+b3x3+...))

Following is the description of the parameters used −

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


Linear Regression
Y=1
Y
Predicted Y can
exceed 0 and 1
range
Y=0
X


Linear Regression
Y=1
Y
Predicted Y can
exceed 0 and 1
range
Y=0
X


Logistic Regression
Y=1
Y
Predicted Y lies
within 0 and 1 range
Y=0
X

Machine Learning Key Terms

Country, Age, Salary

Independent Variable □ X□ Predictor

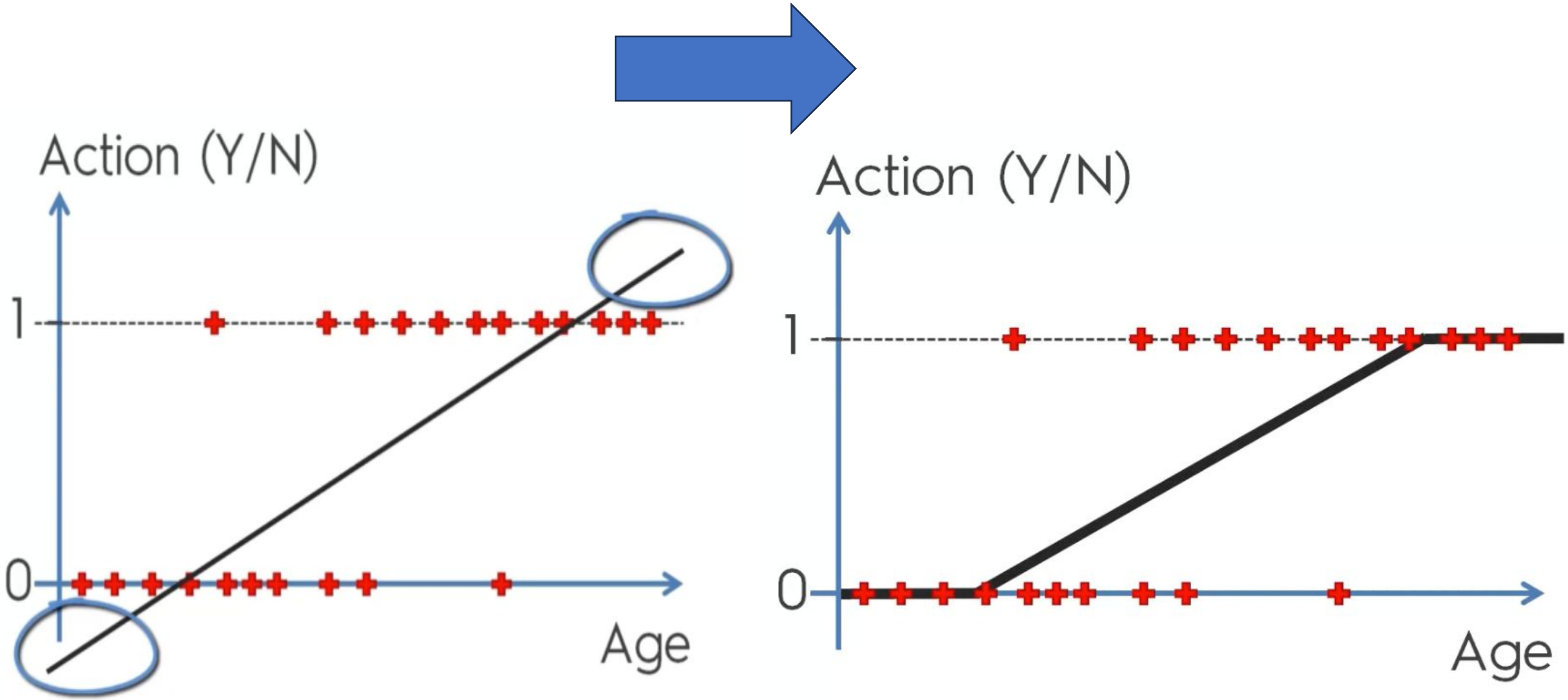
Purchased

Dependent Variable□ Y□ Response

Country	Age	Salary	Purchased
France	44	72000	No
Spain	27	48000	Yes
Germany	30	54000	No
Spain	38	61000	No
Germany	40		Yes
France	35	58000	Yes
Spain		52000	No
France	48	79000	Yes
Germany	50	83000	No
France	37	67000	Yes

Logistic Regression

Unlike regression where you predict a continuous number, you use classification to predict a category. There is a wide variety of classification applications from medicine to marketing

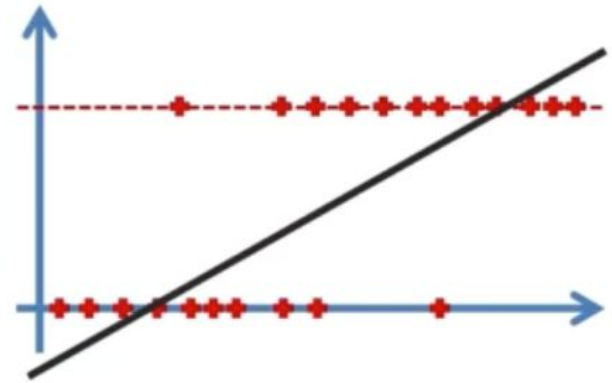

Action (Y/N)
1
0
Age
Action (Y/N)
1
0
Age

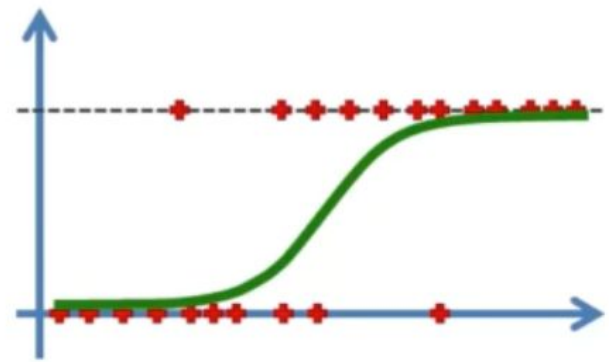
$$y = b_0 + b_1 * x$$

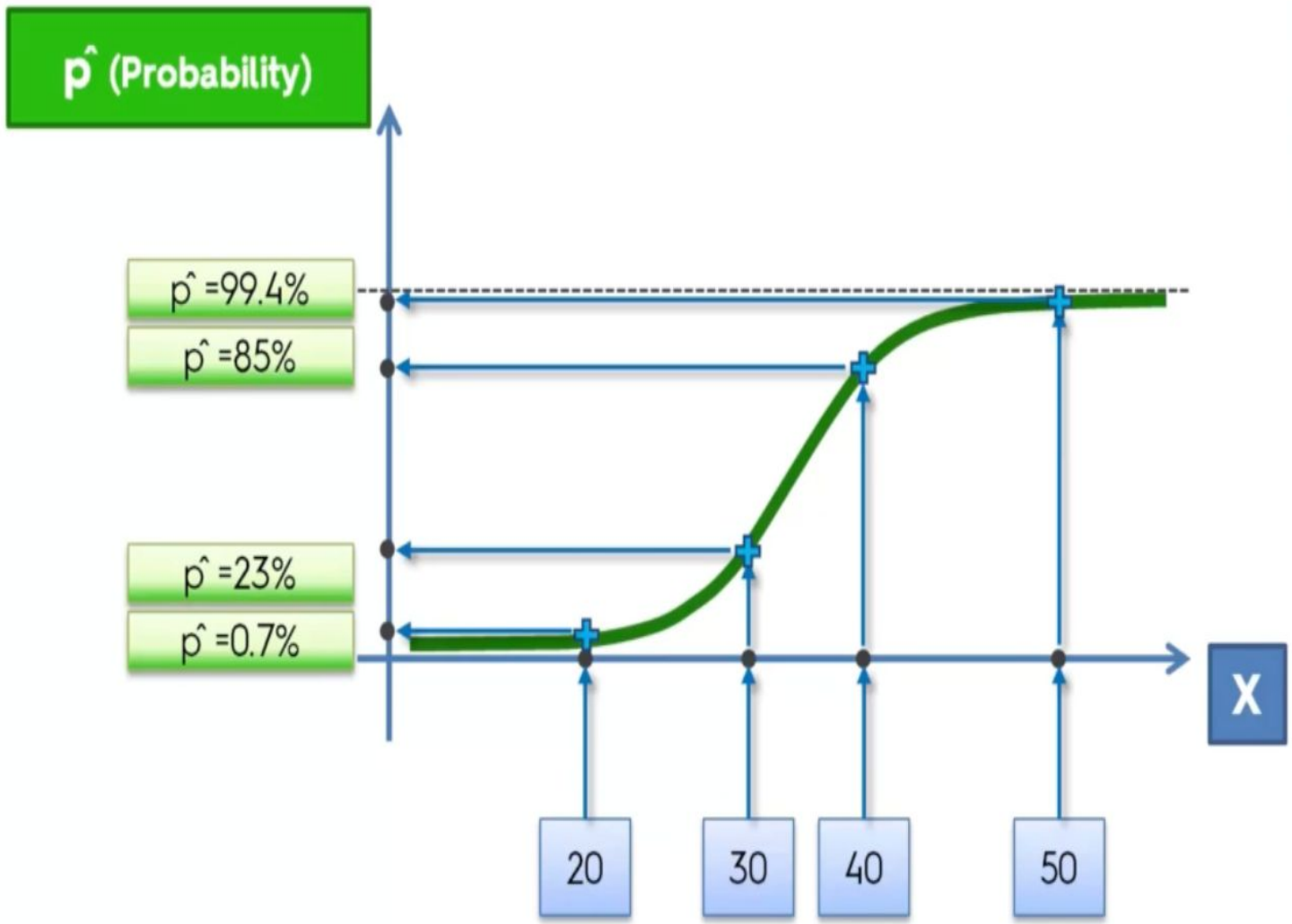
Sigmoid Function

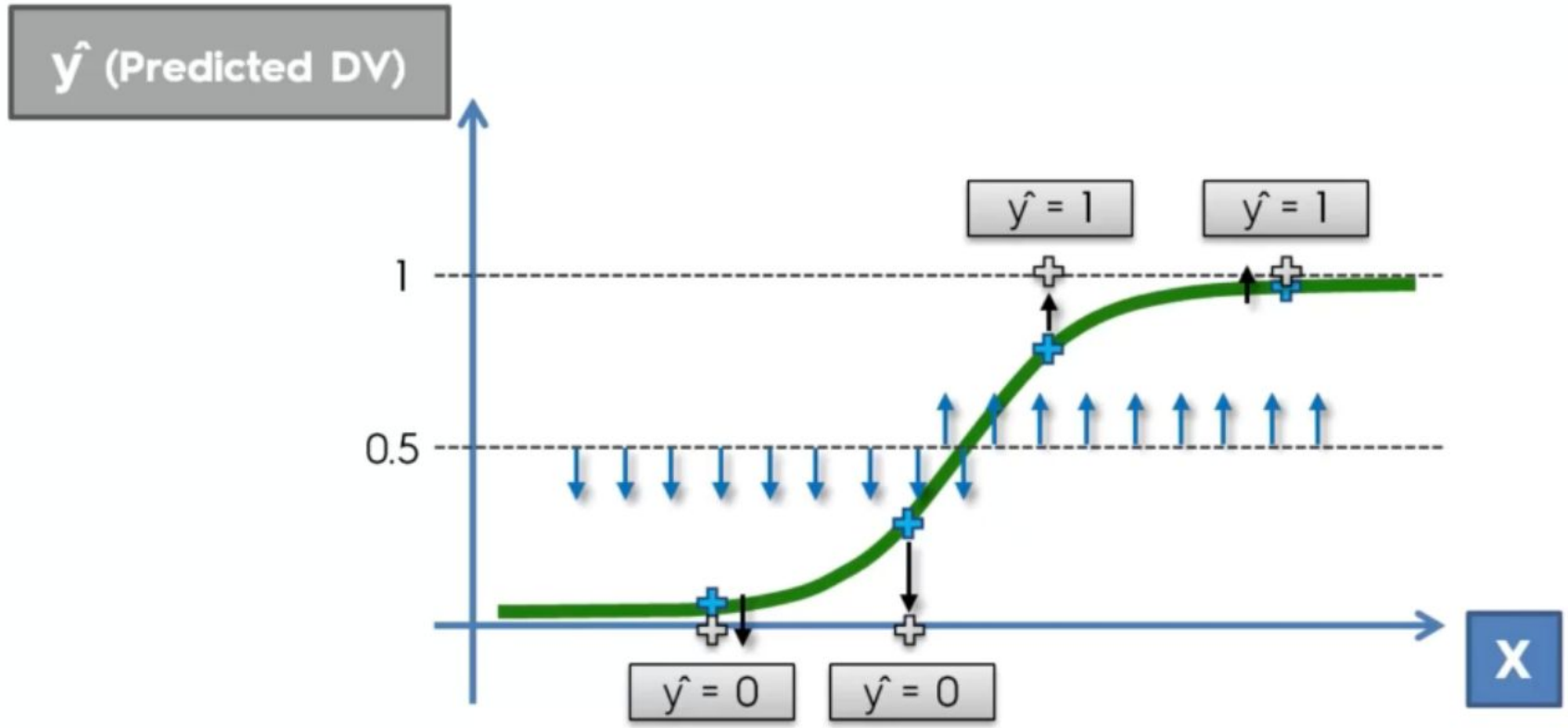
$$p = \frac{1}{1 + e^{-y}}$$

$$\ln\left(\frac{p}{1-p}\right) = b_0 + b_1 * x$$



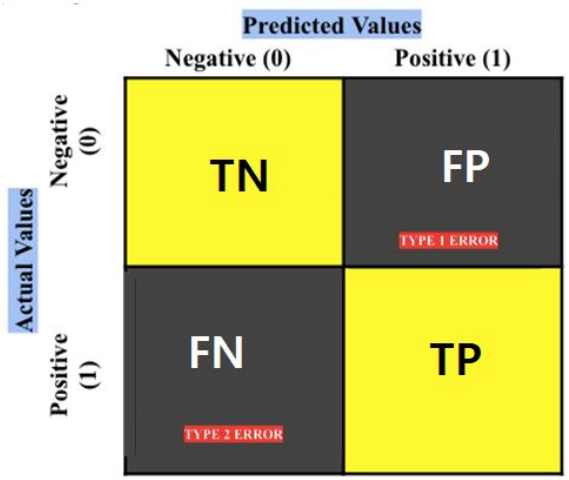



p̂ (Probability)
p̂ =99.4%
p̂ =85%
p̂ =23%
p̂ =0.7%
X
20
30
40
50


ŷ (Predicted DV)
1
0.5
ŷ = 1
ŷ = 1
ŷ = 0
ŷ = 0
X

Try Building a Model

Confusion Matrix

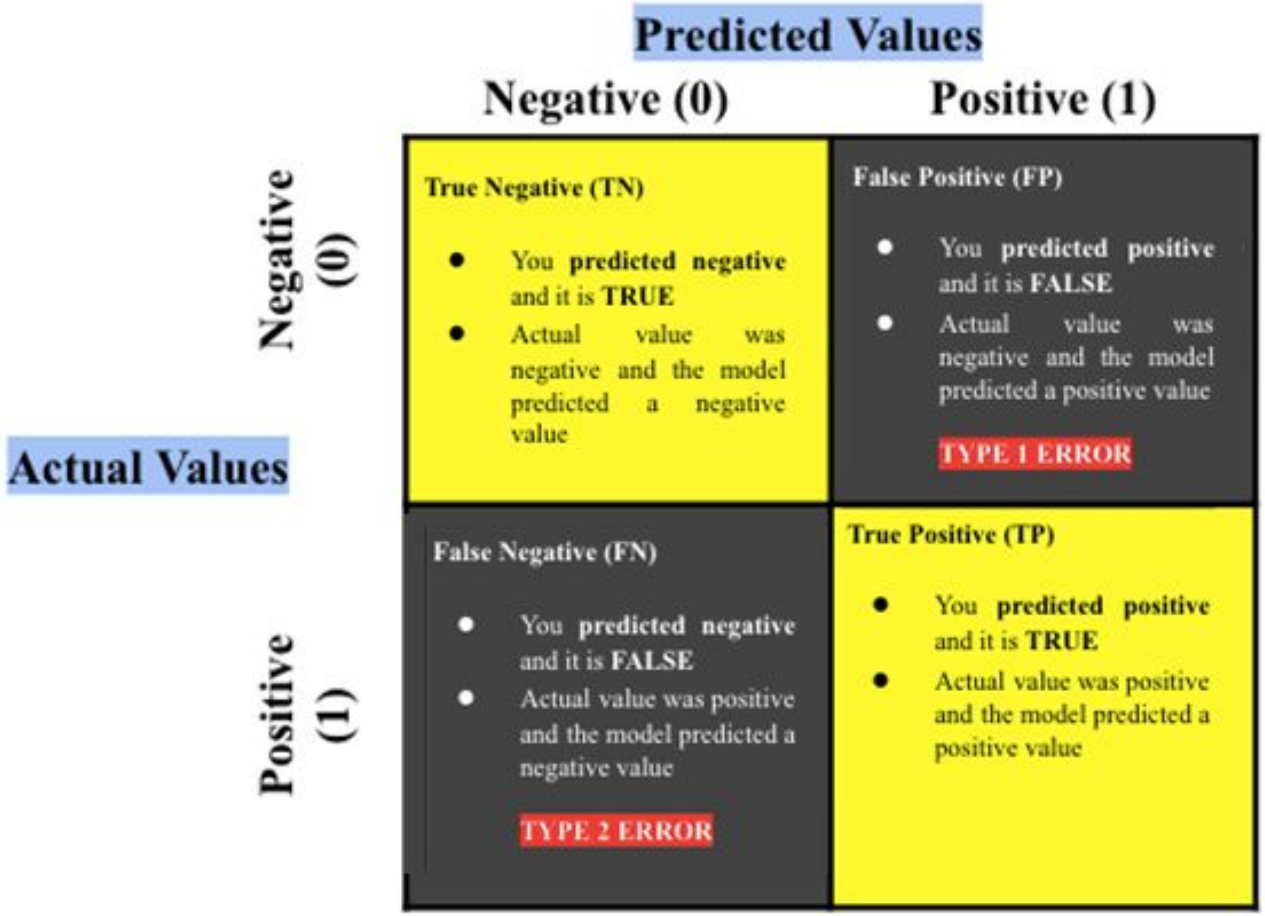


2 X 2 Confusion Matrix for Binary Classification

TP - True Positive
TN - True Negative
FP - False Positive
FN - False Negative

- Since this is binary classification, target variable has two values or two classes: **positive and negative**
- The **columns** of confusion matrix represents **actual values** of target variable.
- The **rows** of confusion matrix represent **predicted value** of target variable.

TP, TN, FP and FN of Confusion Matrix



Let's understand Confusion Matrix:

Diabetes Test Data

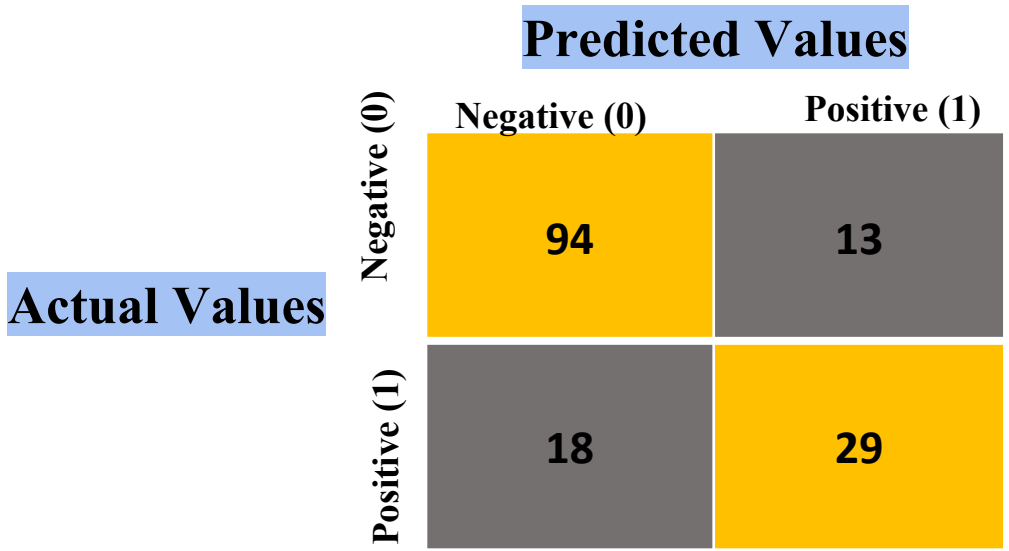
ID	Actual Diabetic	Predicted Diabetic	Outcome
1	1	1	TP
2	0	1	FP
3	1	0	FN
4	0	0	TN
5	1	1	TP
6	0	0	TN
7	1	0	FN
8	0	1	FP
	. . .	. . .	
154	1	1	TP

Example: Diabetes Prediction

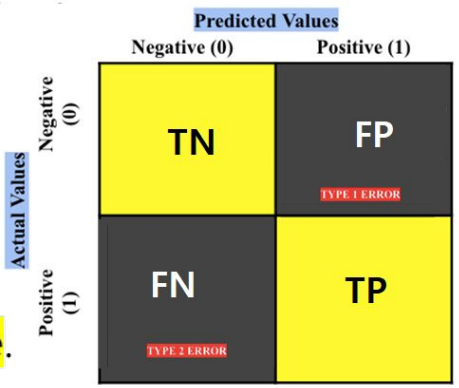
1 ---> Diabetic

0 ---> Not Diabetic

cf=confusion_matrix(y_test, y_pred)



Accuracy|Precision|Recall|F1 Score

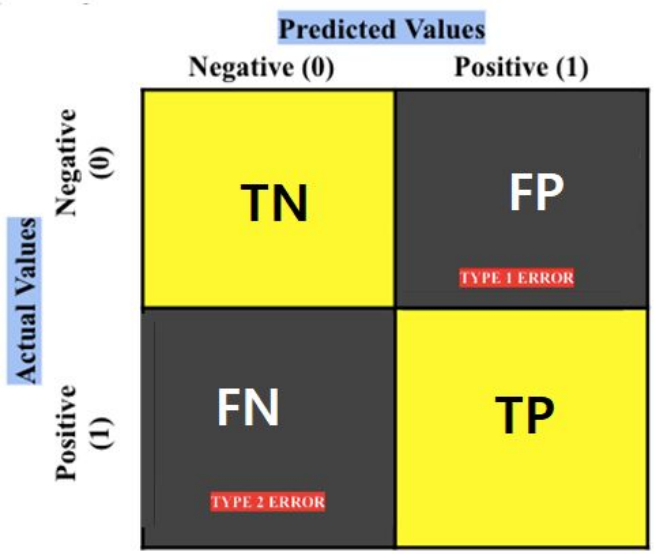


- **Accuracy** = $\frac{(TP+TN)}{(TP+FP+TN+FN)}$ = $\frac{(TP+TN)}{(Total)}$; ERROR =1- Accuracy
- **Precision** tells us how many of the correctly predicted cases turned out to be positive.

  Precision= TP / (TP+FP) = TP/ Predicated Yes
- **Recall** tells us how many of the actual positive cases we were able to predict correctly with our model. Recall = TP / (TP+FN) = TP/ Actual Yes
- **F1 Score** = 2 / ((1/Recall) + (1/Precision))
- Support is the number of actual occurrences of the class in the specified dataset.

Accuracy

- **Accuracy** = $\frac{(TP+TN)}{(TP+FP+TN+FN)}$ = $\frac{(TP+TN)}{(Total)}$; ERROR =1- Accuracy
- Accuracy= Number of correct Predictions/ Size of dataset



When does Accuracy fail?

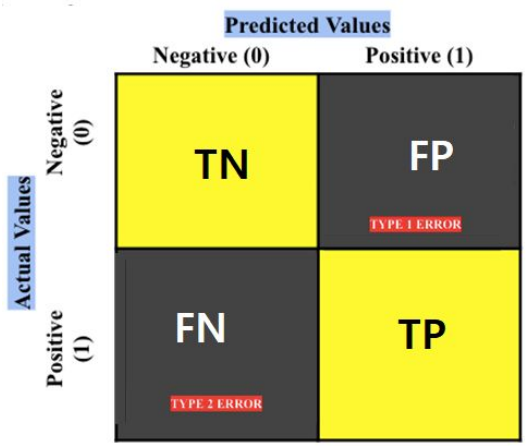
Precision

- **Precision** tells us how many of the correctly predicted cases turned out to be positive.

  Precision= TP / (TP+FP) = TP/ Predicated Yes

- Precision= Number of correctly predicated positive cases/ Number of total positive predictions made by a model
- Example:
- Precision = Number of correctly predicated people with diabetes /

  Number of people model predicated that they have diabetes.



When to check Precision?

What proportion of predicated positive is truly positive?

	Sent to Spam	Not sent to spam
Spam	100	170
Not Spam	30	700

	Sent to Spam	Not sent to spam
Spam	100	190
Not Spam	10	700

Recall

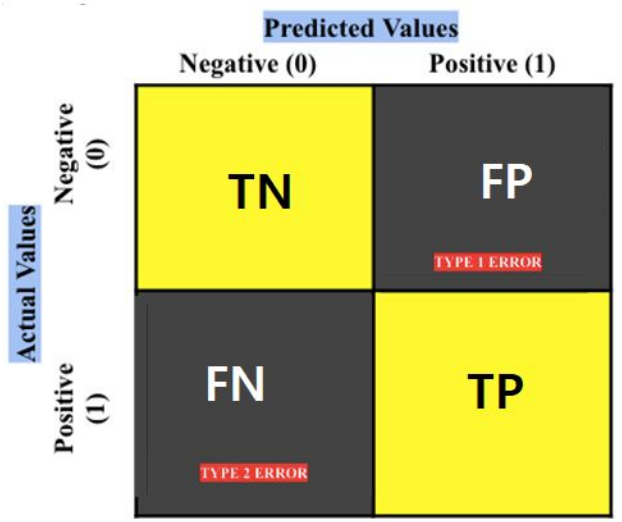
- **Recall** tells us how many of the actual positive cases we were able to predict correctly with our model.
- Recall = TP / (TP+FN) = TP/ Actual Yes
- Recall= Number of correctly predicted positive cases /

  Number of total positive cases in that dataset

  Example:

  Recall= Number of correctly predicated people with diabetes/

  Number of people who have diabetes in dataset



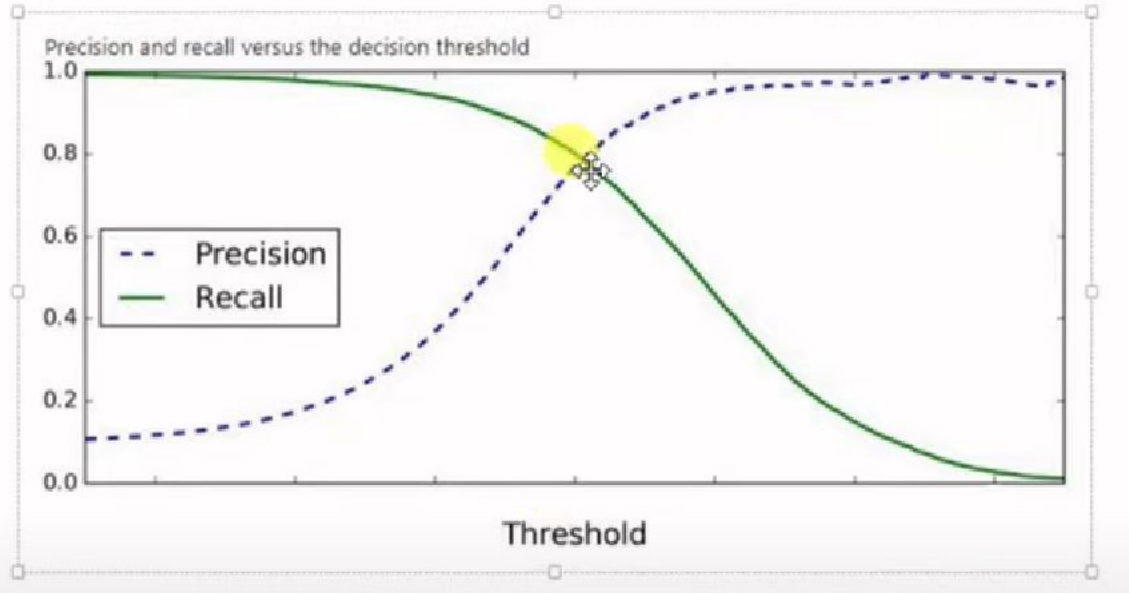
When to check Recall?

What proportion of actual positives is correctly classified?

	Detected Cancer	Not Detected
Has Cancer	1000	200
No Cancer	800	8000

	Detected Cancer	Not Detected
Has Cancer	1000	500
No Cancer	500	8000

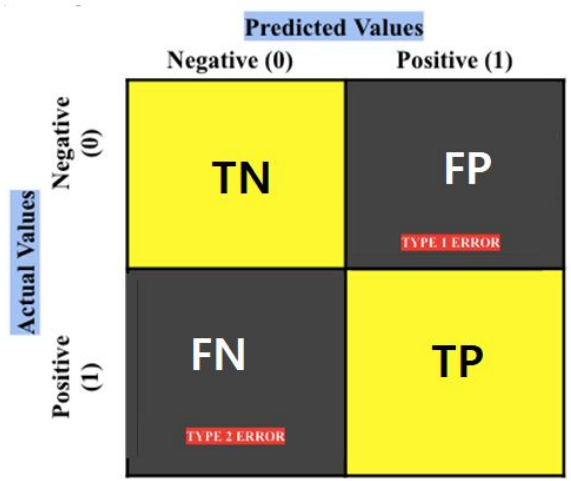
Precision and Recall Trade off



Not able to identify which type of error (Type1/ Type2) is more dangerous or problematic?

F1 Score and Support

- **F1 Score** = 2 / ((1/Recall) + (1/Precision))
- F1 Score is a measure combining both precision and recall.



- **Support** is the number of actual occurrences of the class in the specified dataset.

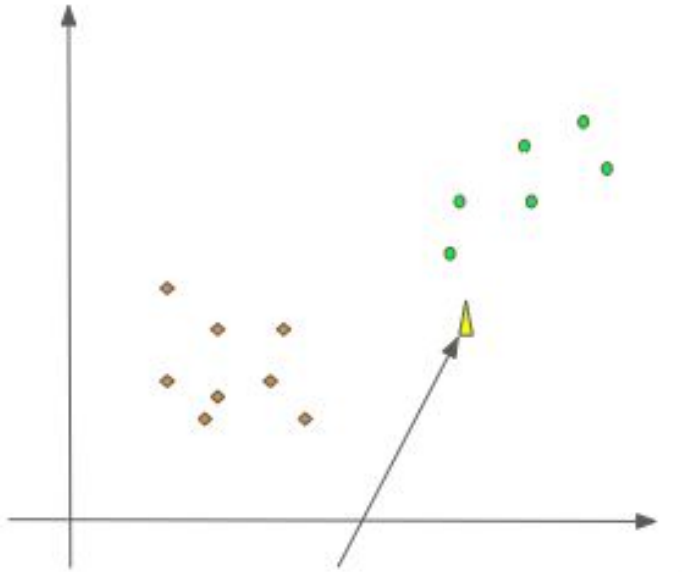
Limitations of logistic regression

- Non-linear problems can’t be solved with logistic regression because it has a linear decision surface
- Linearly separable data is rarely found in real-world scenario
- It is tough to obtain complex relationships using logistic regression
- To handle nonlinearity Decision Tree comes in picture

K Nearest Neighbor

K Nearest Neighbour

- Supervised learning algorithm
- Used for classification as well as regression problems
- It assumes the similarity between new case and available cases



K Nearest Neighbour- Metrics

- Euclidean Distance- (p=2)

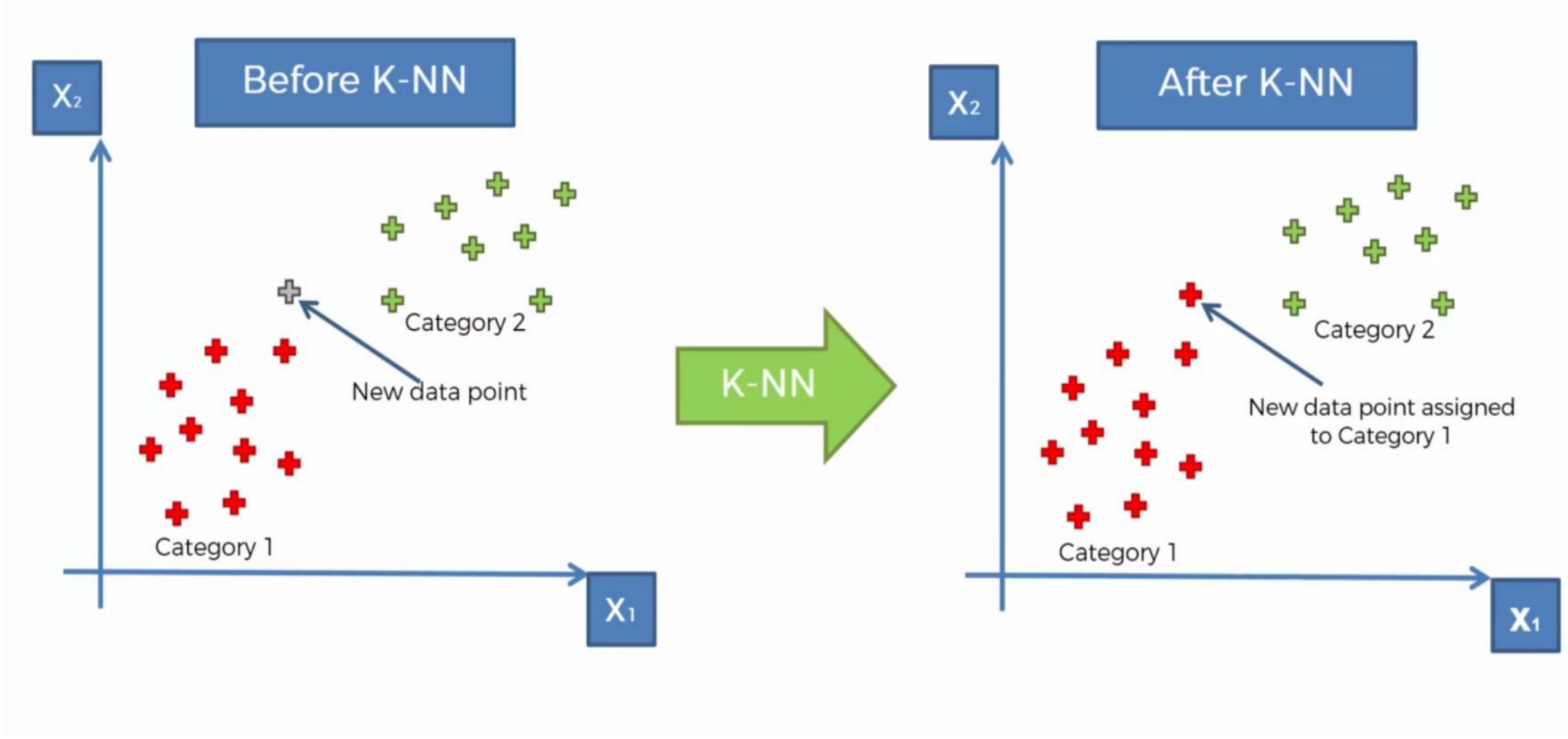
  $D= ((x1-x2)^2 + (y1-y2)^2)^{½}$

- Manhattan Distance- (p=1)

  $D= (x1-x2)+(y1-y2)$

- Minkowski-

  $D= ((x1-x2)^p + (y1-y2)^p)^{1/p}$


Before K-NN
X2
X1
Category 2
New data point
Category 1
K-NN
After K-NN
X2
X1
Category 2
New data point assigned
to Category 1
Category 1

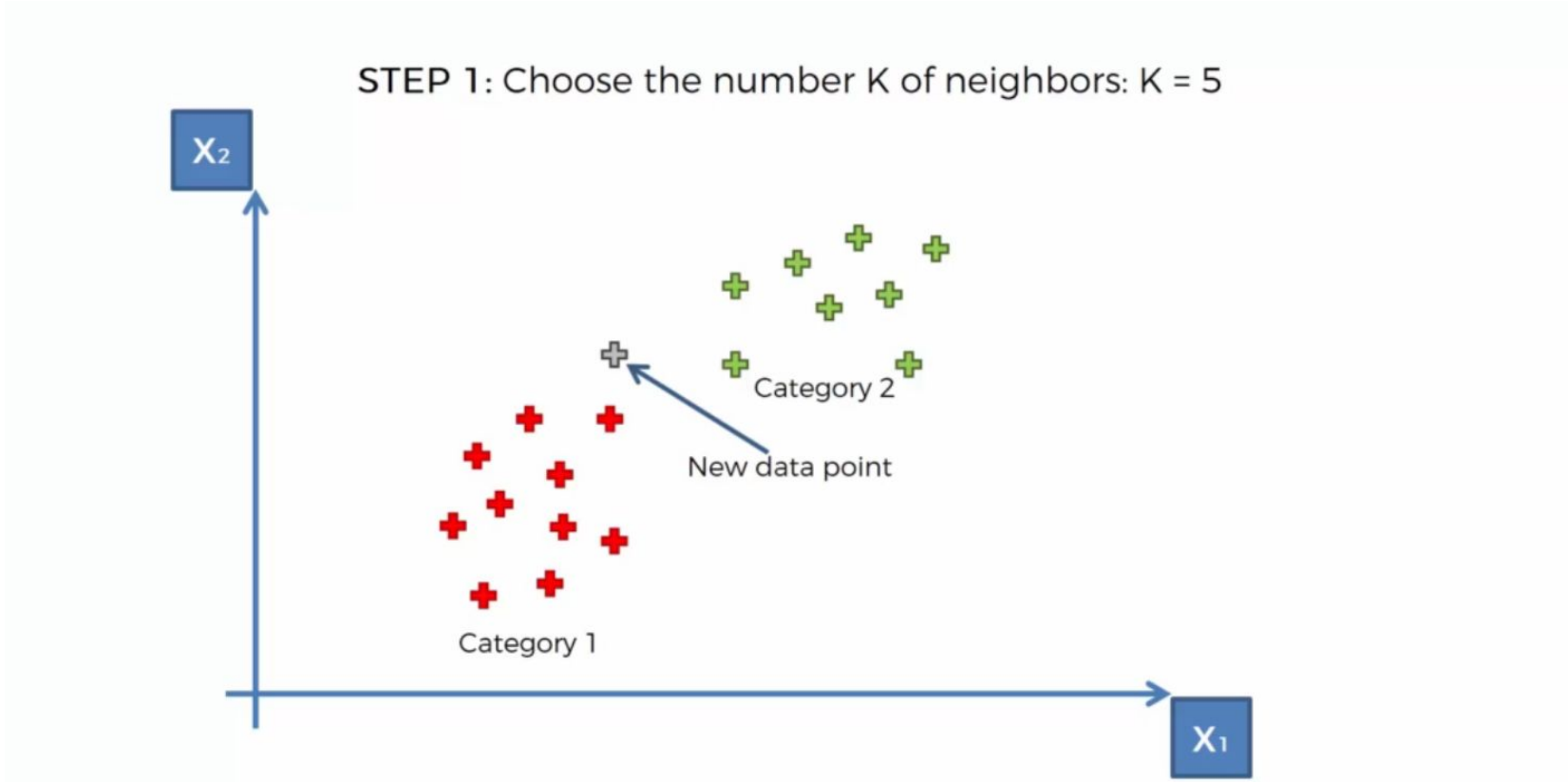
STEP 1: Choose the number K of neighbors

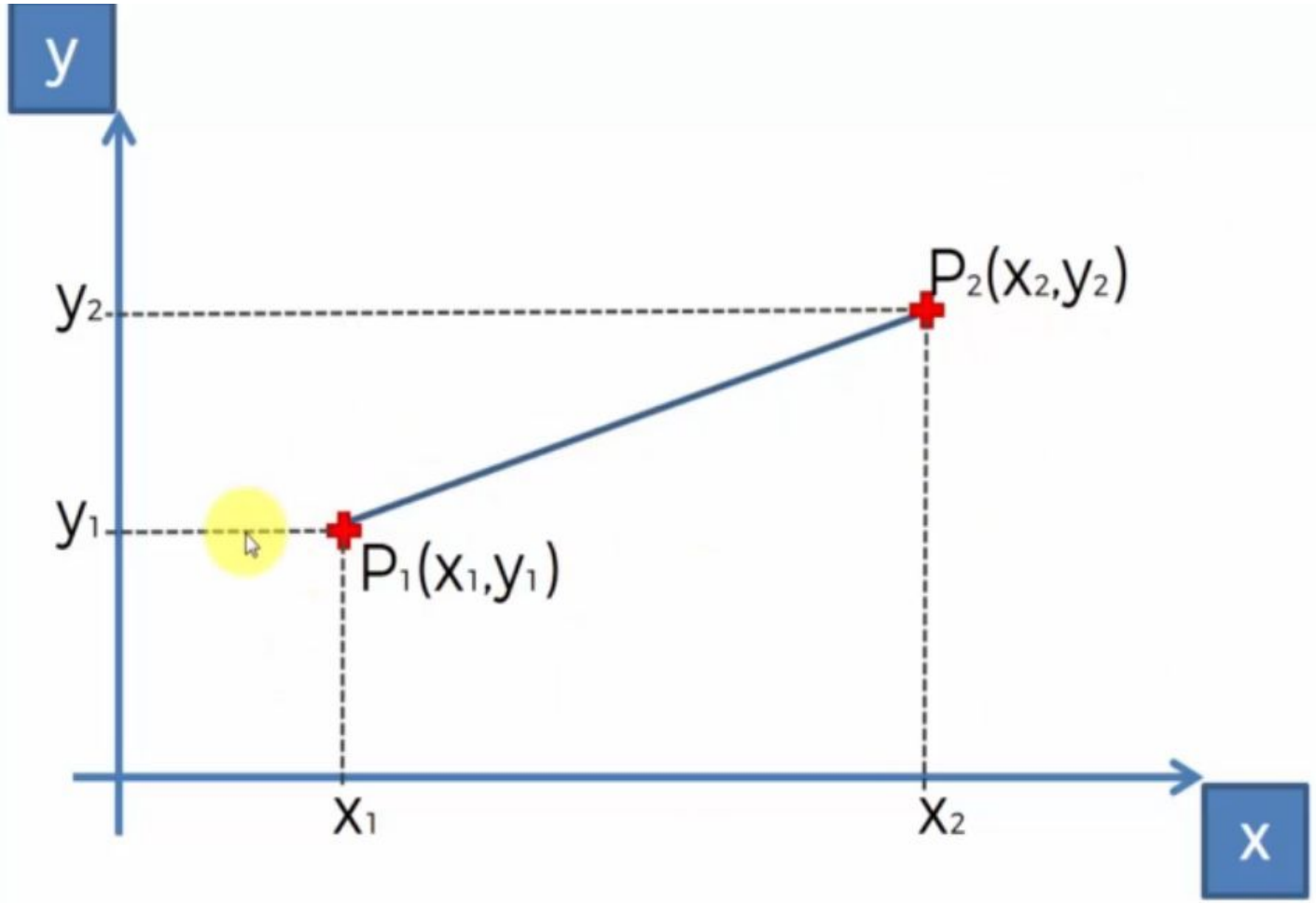
STEP 2: Take the K nearest neighbors of the new data point, according to the Euclidean distance

STEP 3: Among these K neighbors, count the number of data points in each category

STEP 4: Assign the new data point to the category where you counted the most neighbors

Your Model is Ready

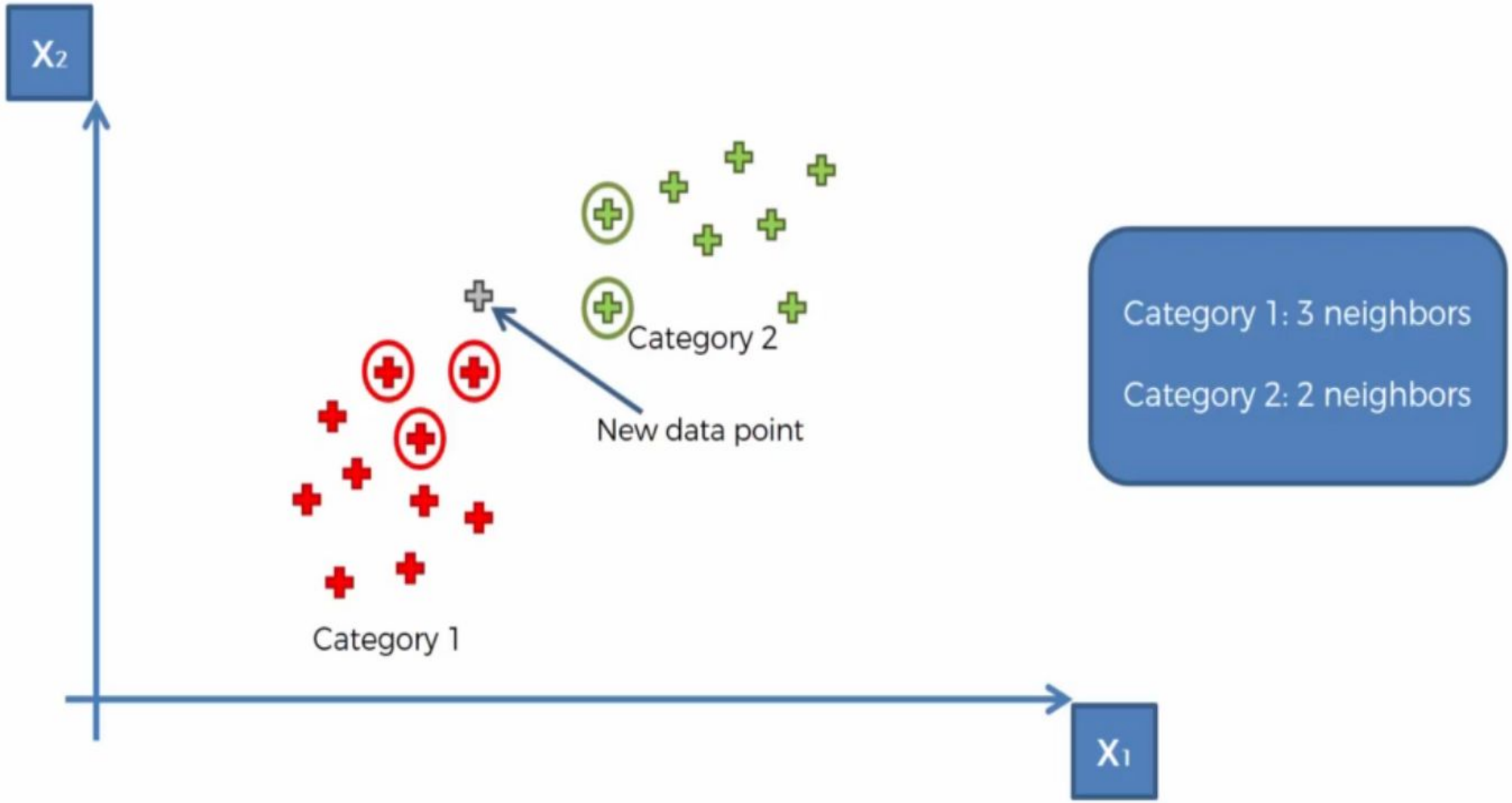

STEP 1: Choose the number K of neighbors: K = 5
X2
Category 2
New data point
Category 1
X1



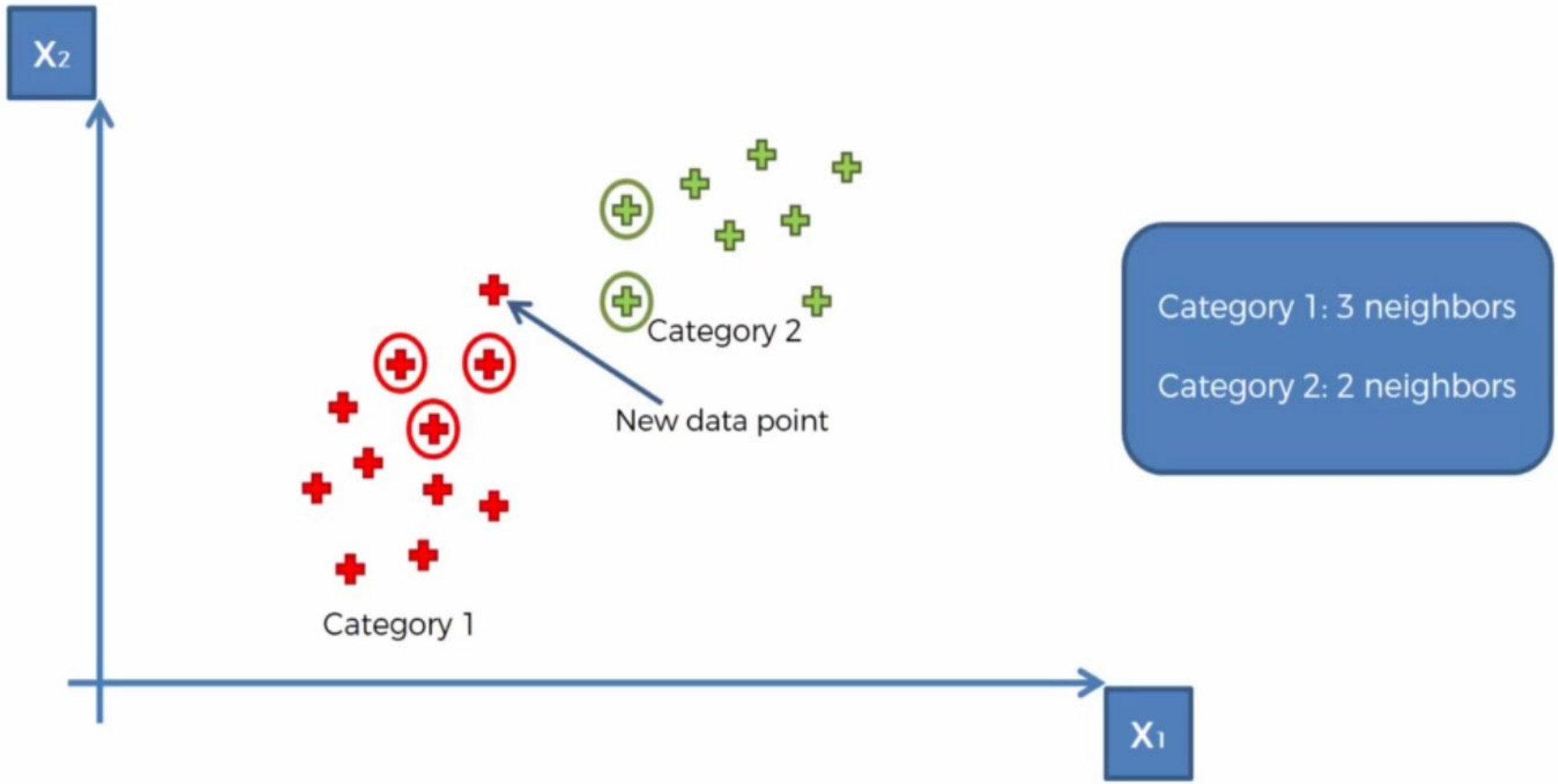
Euclidean Distance between P_1 and $P_2 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

STEP 2: Take the K = 5 nearest neighbors of the new data point, according to the Euclidean distance

STEP 3: Among these K neighbors, count the number of data points in each category



STEP 4: Assign the new data point to the category where you counted the most neighbors

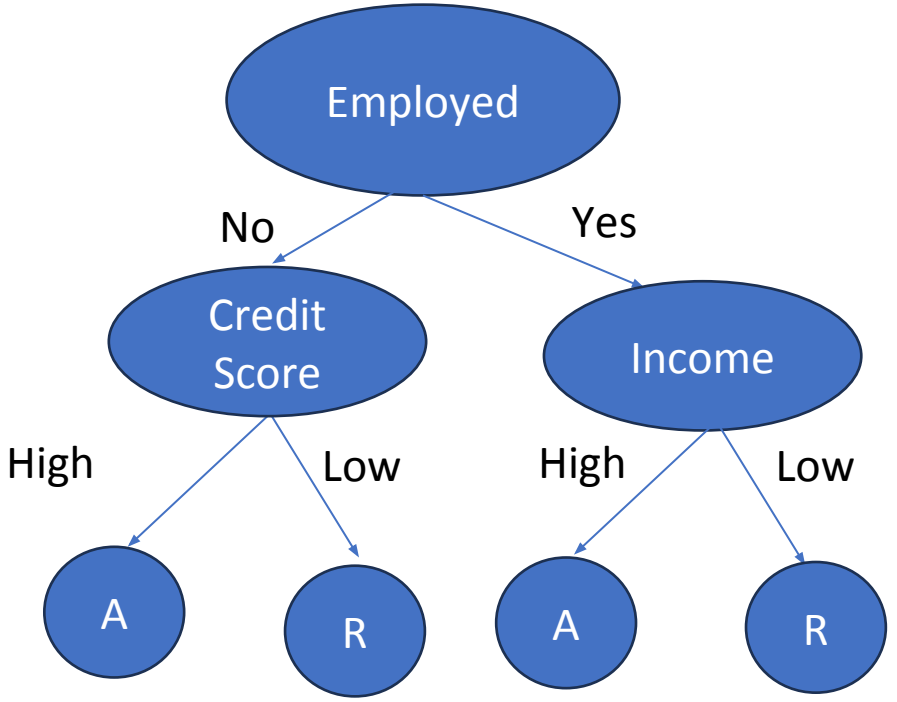


Decision Tree

Decision tree

Example. 1

Employed	Credit Score	Income	Loan A/R
Y			
Y			
Y			
N			
N			



Decision tree

Example. 2

Gender	Occupation	Suggestion
F	Student	PUBG
F	Programmer	GitHub
M	Programmer	Whatsapp
F	Programmer	GitHub
M	Student	PUBG
M	Student	PUBG

Decision tree

Example. 2

Gender	Occupation	Suggestion
F	Student	PUBG
F	Programmer	GitHub
M	Programmer	Whatsapp
F	Programmer	GitHub
M	Student	PUBG
M	Student	PUBG

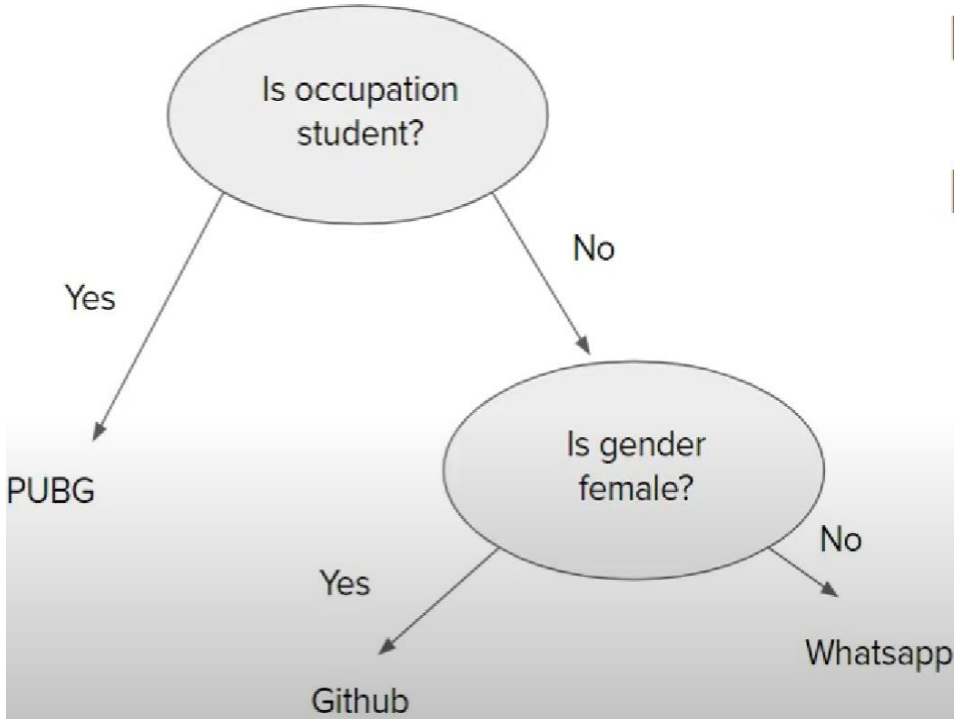
```
If occupation==student
    print(PUBG)
Else
    If gender==female
        print(Github)
    Else
        print(Whatsapp)
```

What is the tree

Example. 2

Gender	Occupation	Suggestion
F	Student	PUBG
F	Programmer	GitHub
M	Programmer	Whatsapp
F	Programmer	GitHub
M	Student	PUBG
M	Student	PUBG

```
If occupation==student
    print(PUBG)
Else
    If gender==female
        print(Github)
    Else
        print(Whatsapp)
```



Decision tree

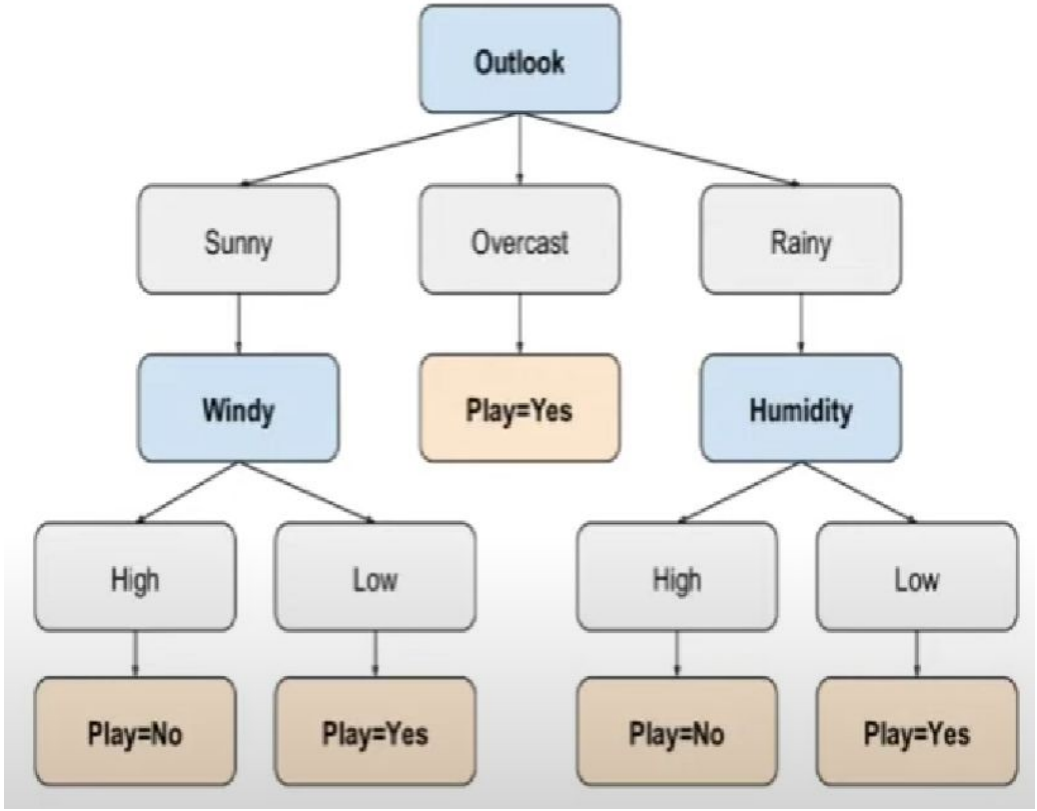
Example. 3

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Decision tree

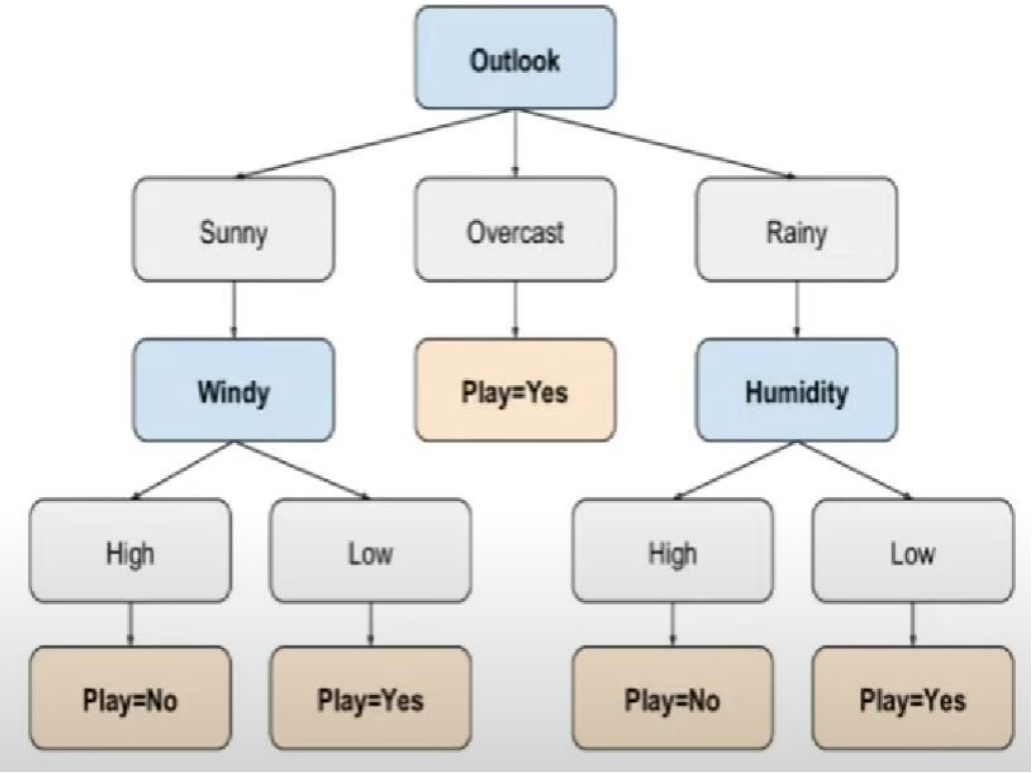
Example. 3

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no



Decision tree

Example. 3



Input Query Point:
[Outlook, temp, humidity, windy]
[Rainy, Mild, High, Strong]

Decision tree : What if we have numerical data?

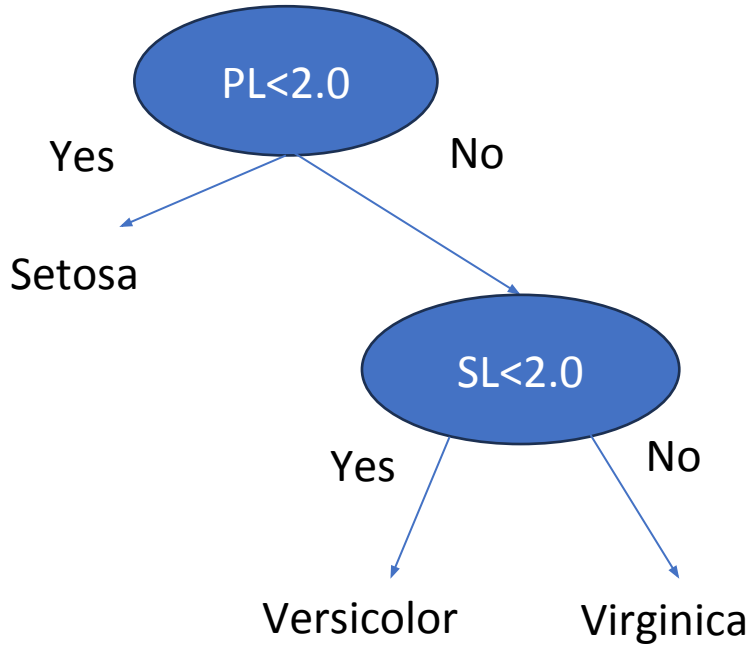
Example. 4

Petal Length	Sepal Length	Type
1.34	0.34	Setosa
3.45	1.45	Versicolor
1.69	0.98	Setosa
2.56	1.79	Virginica
3.00	1.13	Versicolor
1.3	0.88	Setosa

Decision tree : What if we have numerical data?

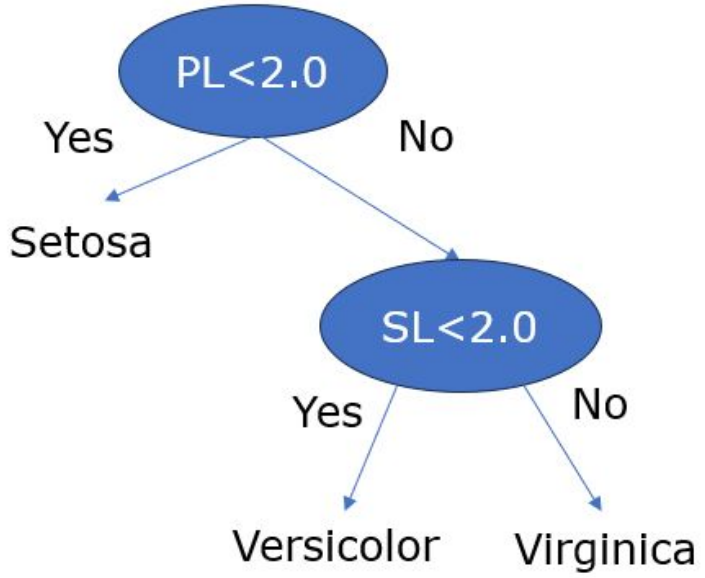
Example. 4

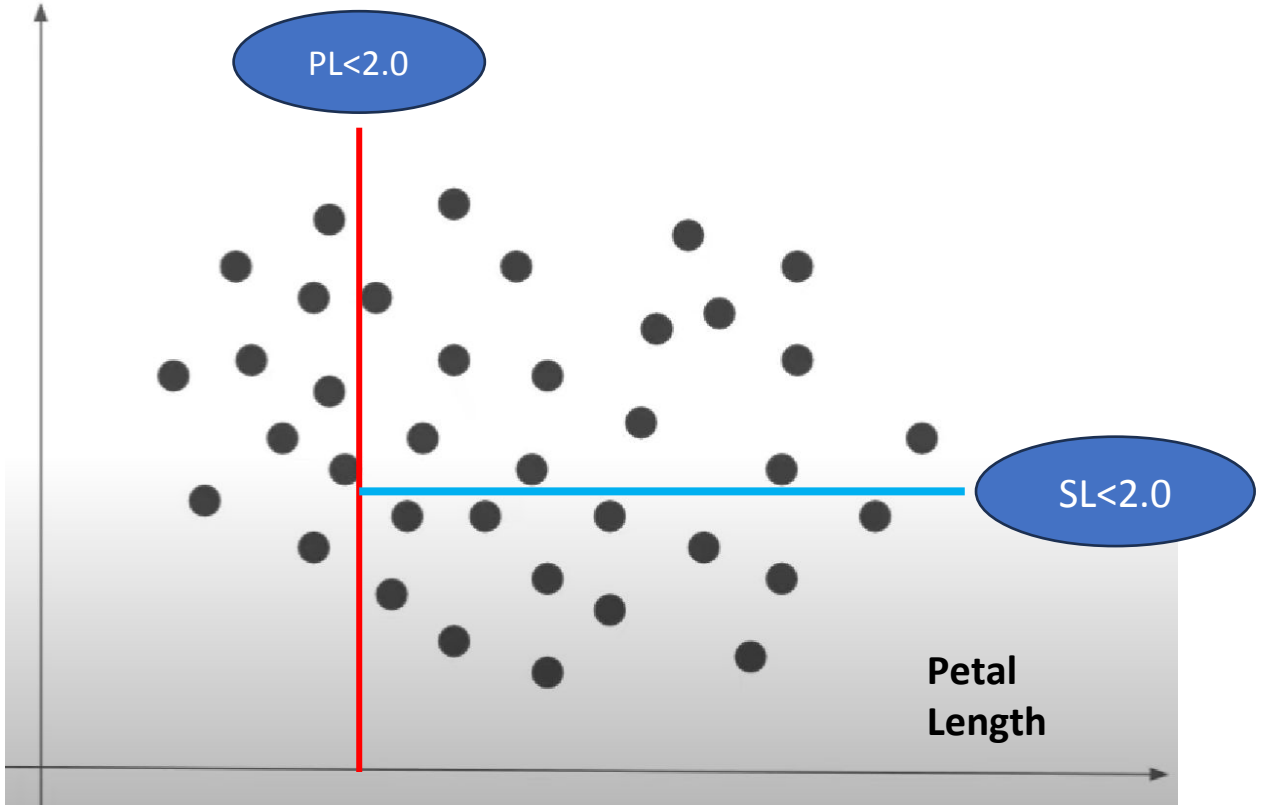
Petal Length	Sepal Length	Type
1.34	0.34	Setosa
3.45	1.45	Versicolor
1.69	0.98	Setosa
2.56	1.79	Virginica
3.00	1.13	Versicolor
1.3	0.88	Setosa



Decision tree : Geometric Intuition

Example. 4

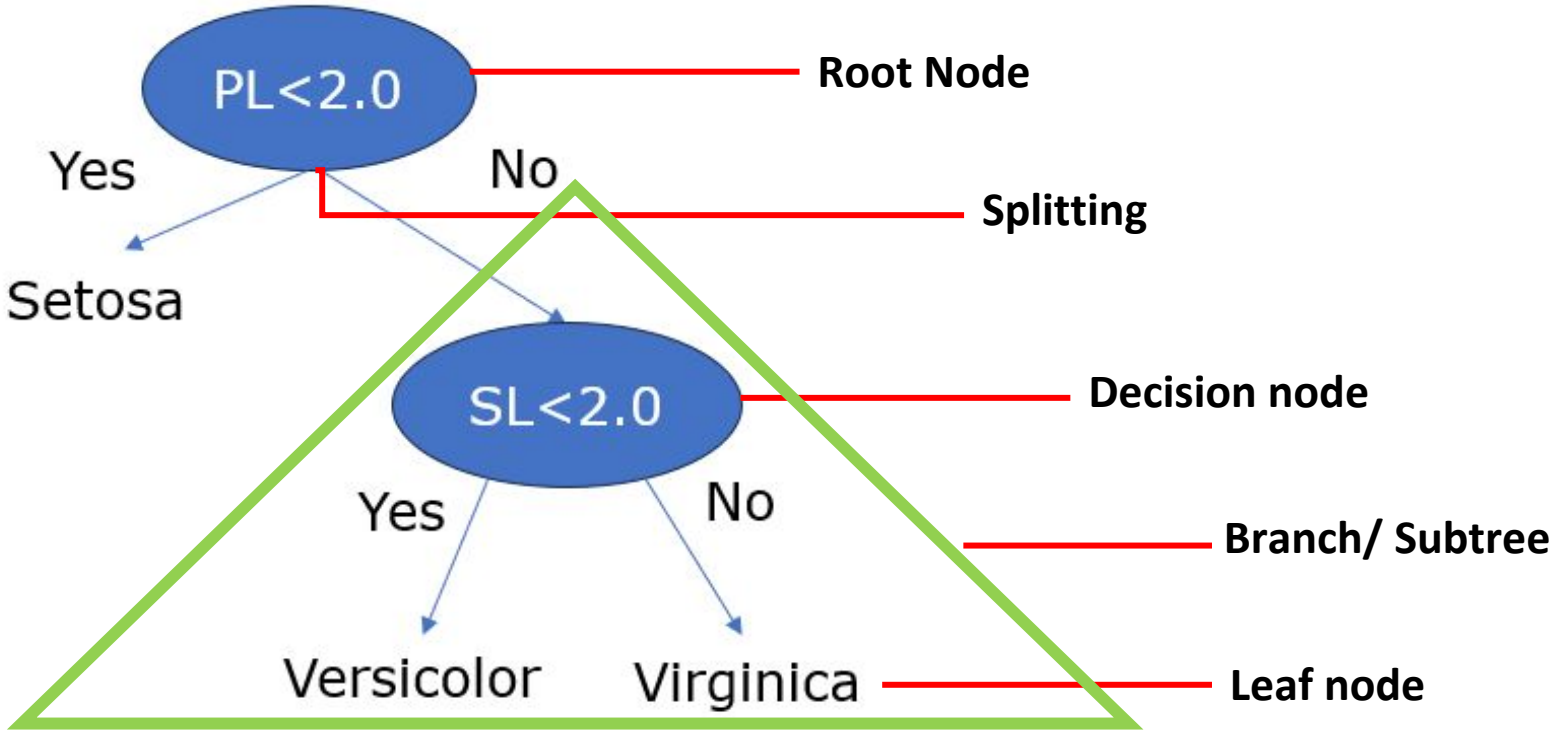




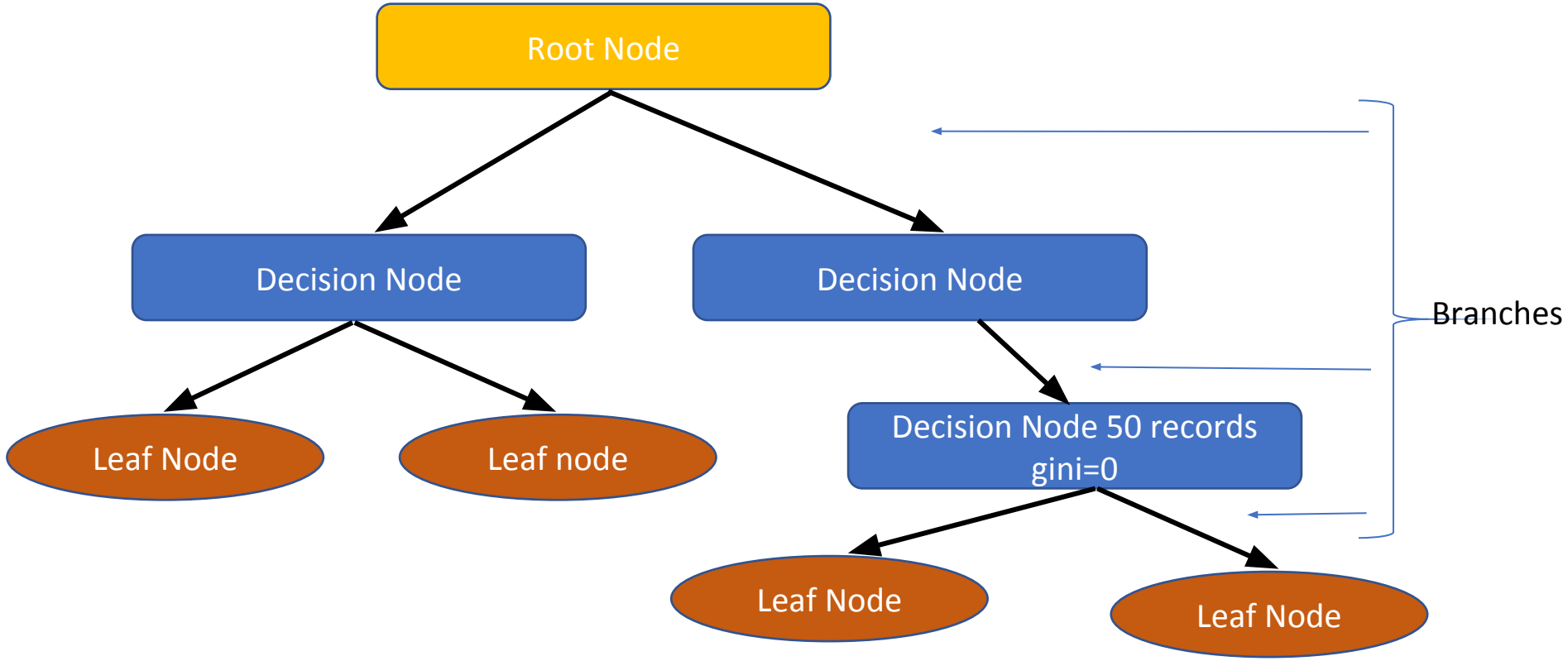
Pseudo Code

- Begin with your training dataset, which should have some feature variables and classification or regression output.
- Determine the “ best feature” in the dataset to split the data on; more on how we define “ best feature” later.
- Split the data into subset that contain the correct values fir this best feature. This splitting basically define a node on the tree i.e each node is a splitting point based on a certain feature from our data.
- Recursively generate new tree nodes by using the subset of data created from step 3.

Terminology



Structure of decision tree

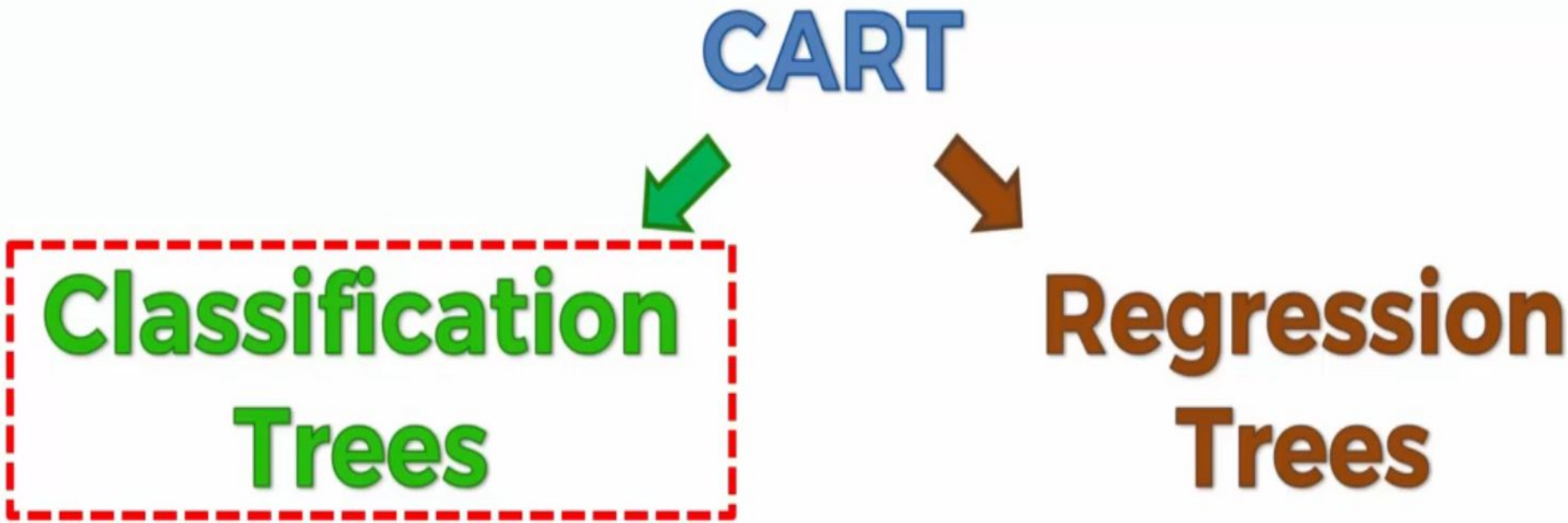


Why decision tree algorithm?

- It is considered to be the most understandable Machine Learning algorithm and it can be easily interpreted.
- It can be used for **classification and regression** problems.
- Unlike most Machine Learning algorithms, it **works effectively with non-linear data.**
- Constructing a Decision Tree is a very quick process since it uses **only one feature per node to split the data.**

Advantages- Decision Tree

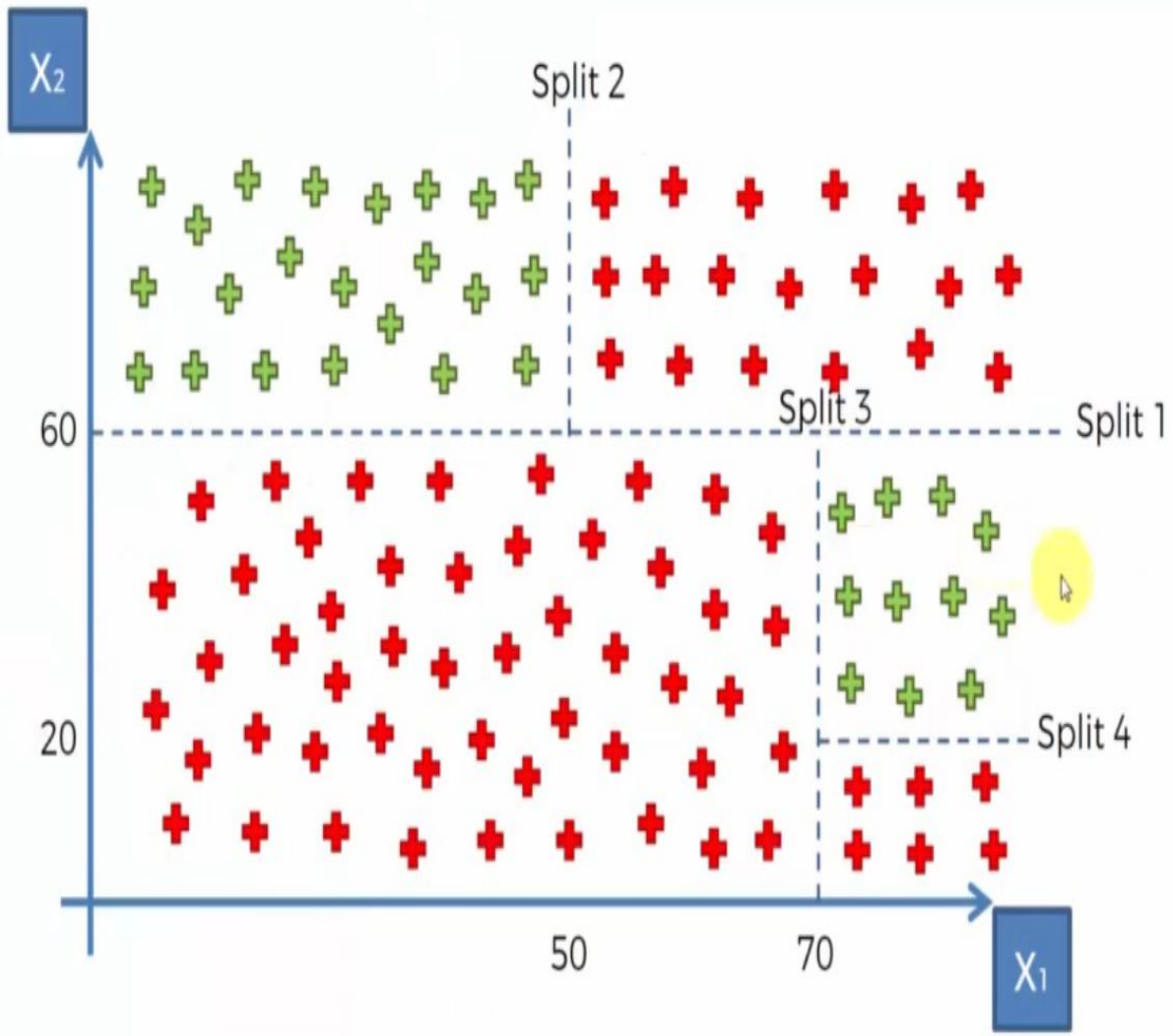
- Intuitive and easy to understand.
- Minimal data preparation is required.
- Decision trees perform classification without requiring much computation.
- Decision trees are capable of handling both continuous and categorical variables.
- CART – Classification and Regression Trees
- Decision trees provide a clear indication of which fields are most important for prediction or classification.
- The cost of using the tree for inference is logarithmic in the number of data points used to train the tree.

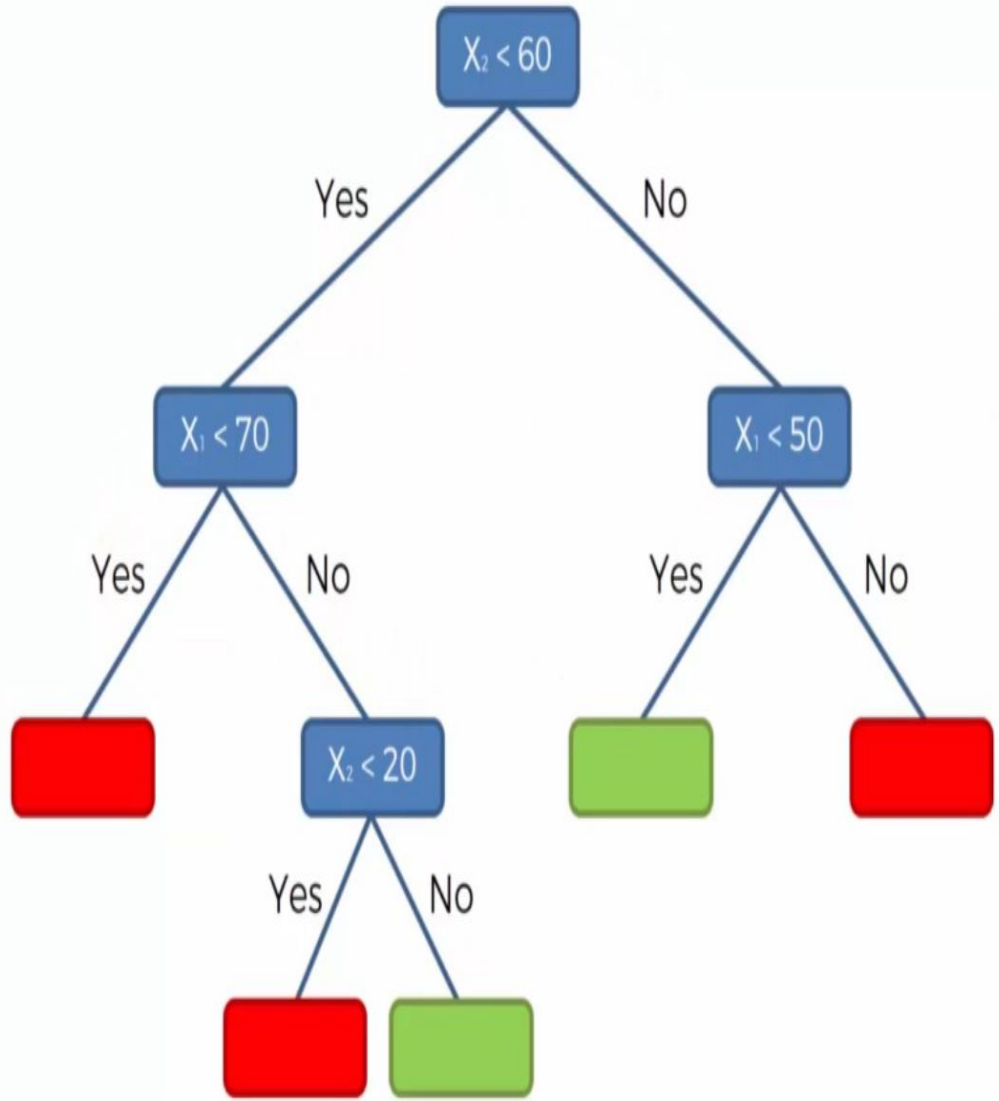

CART
Classification Trees
Regression Trees

[Akinator, the mind reading genie]

https://en.akinator.com/

Nelson Mandela


X2
Split 2
Split 3
Split 1
60
20
Split 4
50
70
X1


X2 < 60
Yes
No
X1 < 70
X1 < 50
Yes
No
Yes
No
X2 < 20
Yes
No

Questions

- **How to decide which columns should be considered as root node?**
- **How to select subsequent decision node?**
- **How to decide splitting criteria in case of numerical columns?**

Attribute Selection Measure

Two popular methods for attribute selection:

1. Gini Index
2. Information Gain -> Entropy

Gini Index

- Gini Index measures the degree or probability of a particular element being wrongly classified when it is randomly chosen.
- 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 there exists only one class (pure), and '1' denotes that the elements are randomly distributed across various classes (impure).

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

Attribute Selection Measure (ASM)

- Entropy- Entropy is the measurement of impurities or randomness in the data points.
- $\text{Entropy} = \sum_{i}^{c} -p_i \log_2 (p_i)$
- **Gini Index**$= 1 - \sum_{I}^{n} (p_i)^2$

- Information Gain=Entropy(Parent)-[(Weighted Avg) *Entropy(Children)

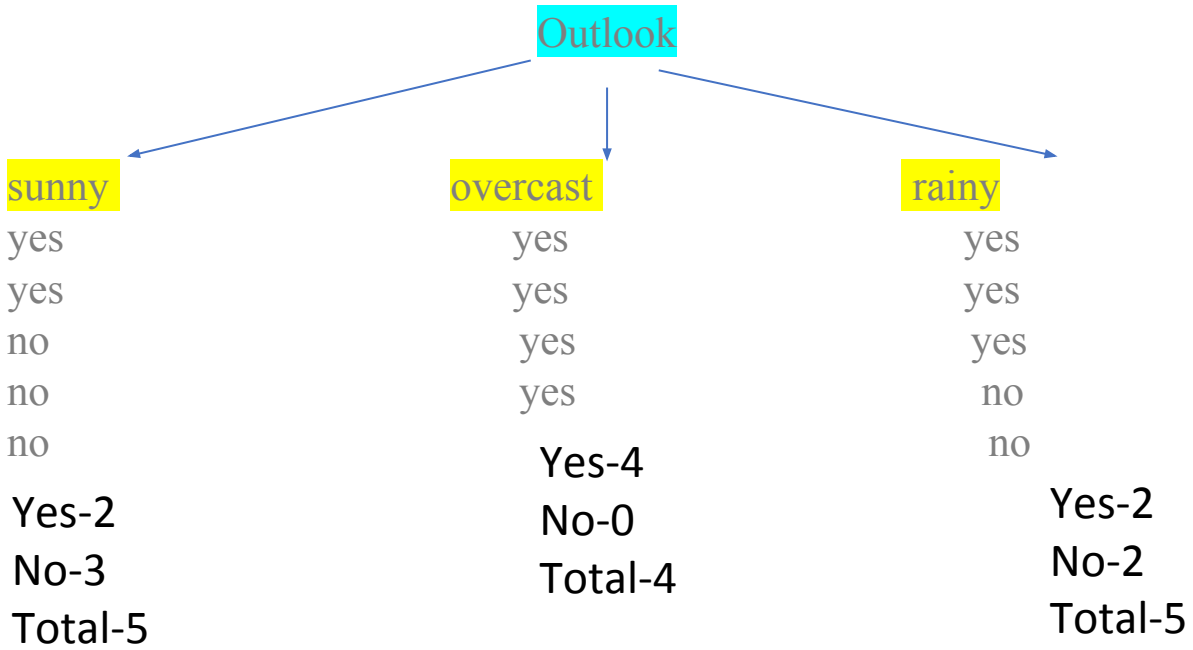
- Entropy(Parent)= $-P(yes)*\log_2 P(yes) - P(no)*\log_2 P(no)$

where,
P(yes)= probability of yes
P(no)= probability of no

Decision Tree- Gini Index Calculation (outlook)

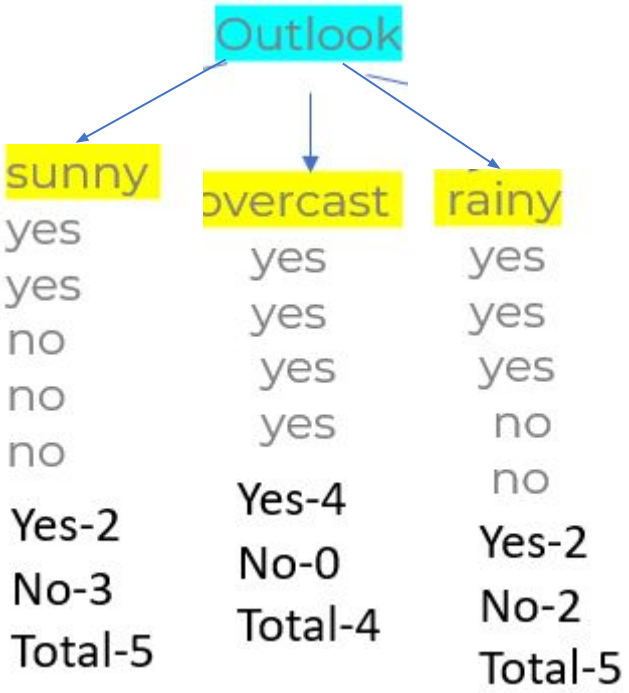
- **Gini Index** $= 1 - \sum_{i=1}^{n} (p_i)^2$

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no



Decision Tree- Gini Index Calculation (outlook)

- **Gini Index** $= 1 - \sum_{i}^{n} (p_i)^2$



P(outlook=sunny): 5/14
P(outlook=overcast): 4/14
P(outlook=rainy): 5/14

- If (outlook = sunny & play = yes), probability = 2/5
- If (outlook=sunny & play=no), probability = 3/5

Gini index = 1 - ((2/5)^2 + (3/5)^2) = 0.48

- If (outlook = overcast & play = yes), probability = 4/4
- If (outlook=overcast & play=no), probability = 0/4

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

- If (outlook = rainy & play = yes), probability = 3/5
- If (outlook=rainy & play=no), probability = 2/5

Gini index = 1 - ((3/5)^2 + (2/5)^2) = 0.48

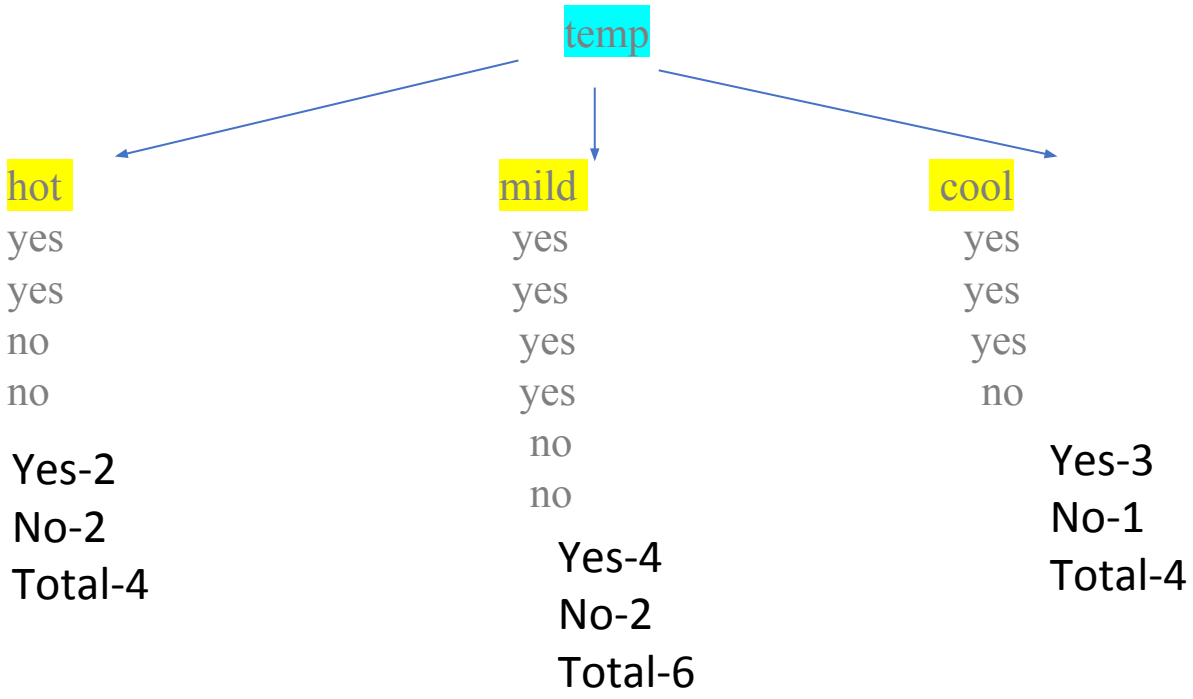
Weighted sum of the Gini Indices can be calculated as follows:

Gini Index for outlook = (5/14)0.48* + *(4/14)**0 +(5/14)*0.48 = 0.343**

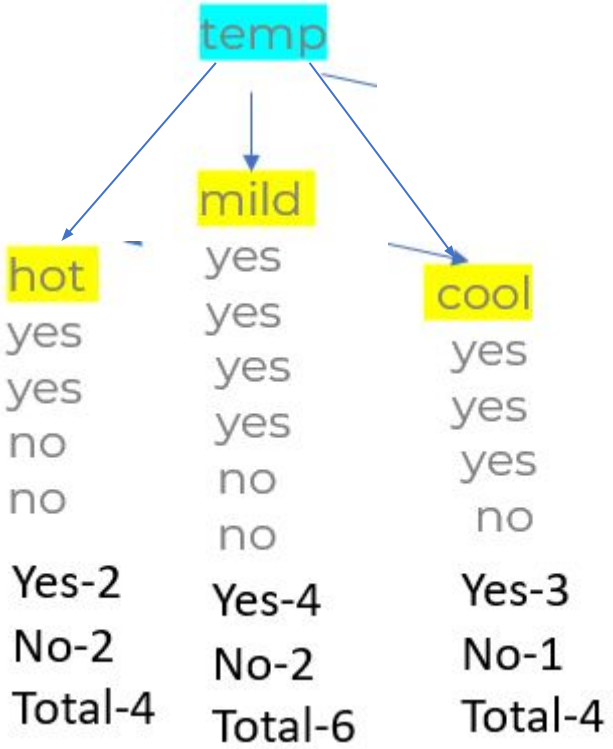
Decision Tree- Gini Index Calculation (temp)

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no



Decision Tree- Gini Index Calculation (temp)



P(temp=hot): 4/14
P(temp=mild): 6/14
P(temp=cool): 4/14

- If (temp=hot & play = yes), probability = 2/4
- If (temp=hot & play=no), probability = 2/4

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

- If (temp=mild & play = yes), probability = 4/6
- If (temp=mild & play=no), probability = 2/6

- Gini index = 1 - ((4/6)^2 + (2/6)^2) = 0.44

- If (temp=cool & play = yes), probability = 3/4
- If (temp=cool & play=no), probability =1 /4

Gini index = 1 - ((3/4)^2 + (1/4)^2) = 0.375

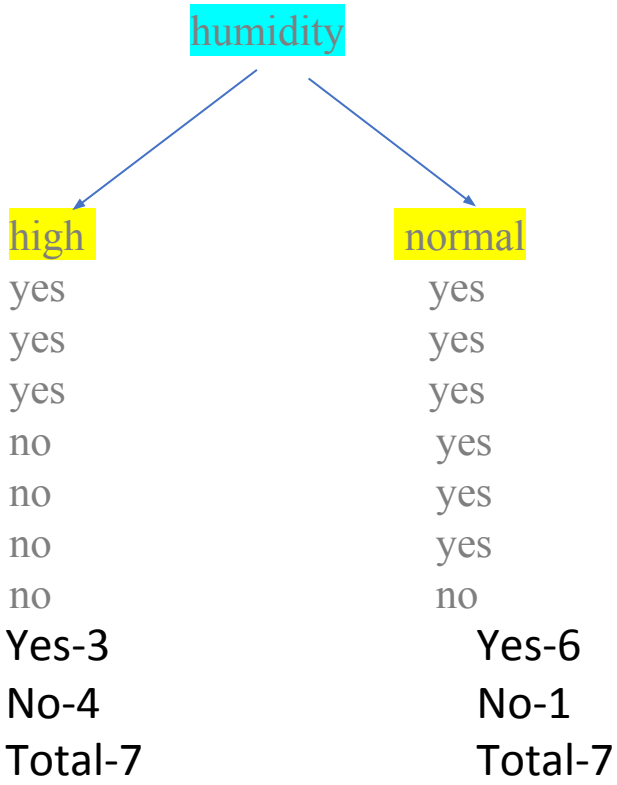
Weighted sum of the Gini Indices can be calculated as follows:

Gini Index for temp = (4/14)*0.5 + (6/14)*0.44+(4/14)*0.375= 0.438

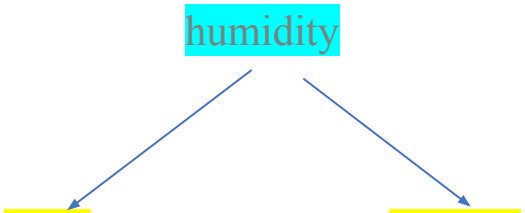
Decision Tree- Gini Index Calculation (humidity)

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no



Decision Tree- Gini Index Calculation (humidity)



high	normal
yes	yes
yes	yes
yes	yes
no	yes
no	yes
no	yes
no	no
Yes-3	Yes-6
No-4	No-1
Total-7	Total-7

P(humidity=high): 7/14
P(humidity=normal): 7/14

- If (humidity=high & play = yes), probability = 3/7
- If (humidity=high & play=no), probability = 4/7

Gini index = 1 - ((3/7)^2 + (4/7)^2) = 0.489

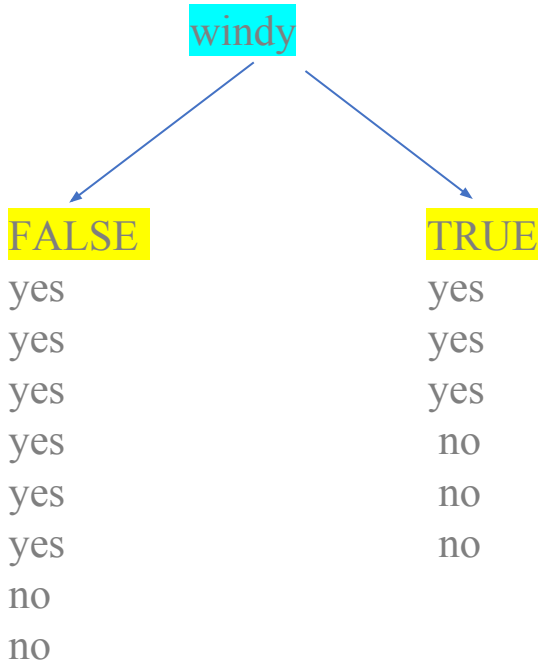
- If (humidity=normal & play = yes), probability = 6/7
- If (humidity=normal & play=no), probability = 1/7

- Gini index = 1 - ((6/7)^2 + (1/7)^2) = 0.244

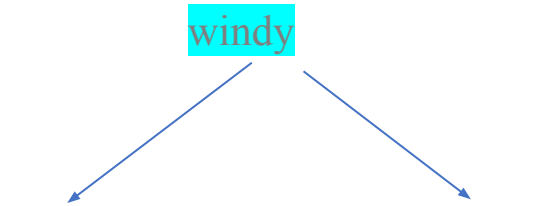
Weighted sum of the Gini Indices can be calculated as follows:
Gini Index for humidity = (7/14)0.489+ (7/14)**0.244= 0.366**

Decision Tree- Gini Index Calculation (windy)

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no



Decision Tree- Gini Index Calculation (windy)



FALSE	TRUE
yes	yes
yes	yes
yes	yes
yes	no
yes	no
yes	no
no	
no	

P(windy=FALSE): 8/14
P(windy=TRUE): 6/14

- If (humidity=high & play = yes), probability = 6/8
- If (humidity=high & play=no), probability = 2/8

Gini index = 1 - ((6/8)^2 + (2/8)^2) = 0.375

- If (humidity=normal & play = yes), probability = 3/6
- If (humidity=normal & play=no), probability = 3/6

- Gini index = 1 - ((3/6)^2 + (3/6)^2) = 0.5

Weighted sum of the Gini Indices can be calculated as follows:
Gini Index for windy = (8/14)0.375*+ (6/14)*0.5= 0.428**

Decision Tree- Gini Index Calculation

Attributes/Features	Gini Index
outlook	0.343
temp	0.438
humidity	0.366
windy	0.428

- ‘outlook’ has the lowest Gini Index and hence it will be chosen as the root node.
- Same procedure will be repeated to determine the sub-nodes or branches of the decision tree.

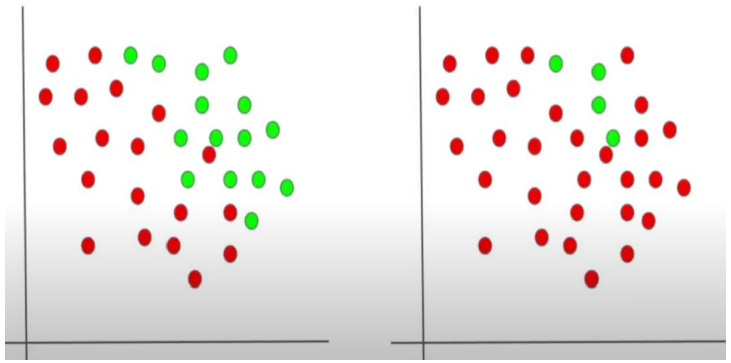
What is Entropy?

- Entropy is nothing but a measure of disorder.
- Measure of purity/Impurity

Example1

Ice Water Vapor

Example2



Example3

Salary	Age	Purchase
20000	21	Yes
10000	45	No
60000	27	Yes
15000	31	No
12000	18	No

Salary	Age	Purchase
34000	31	No
15000	25	No
69000	57	Yes
25000	21	No
32000	28	No

How to calculate Entropy?

- Entropy is nothing but a measure of disorder.
- Entropy= $\sum_i^c -p_i \log_2(p_i)$
- Where pi is simply the probability of a class 'I' in our data.

For e.g if our data has only 2 class labels Yes and No.

$$E(D)= -p_{yes}\log_2(p_{yes}) - p_{no}\log_2(p_{no})$$

Example3

Salary	Age	Purchase
20000	21	Yes
10000	45	No
60000	27	Yes
15000	31	No
12000	18	No

Salary	Age	Purchase
34000	31	No
15000	25	No
69000	57	Yes
25000	21	No
32000	28	No

$H(d)= -P_y\log_2(P_y) -P_n\log_2(P_n)$

$H(d)= -2/5\log_2(2/5) -3/5\log_2(3/5)$

$H(d)= 0.97$

$H(d)= -P_y\log_2(P_y) -P_n\log_2(P_n)$

$H(d)= -1/5\log_2(1/5) -4/5\log_2(4/5)$

$H(d)= 0.72$

Entropy Observation

- More the uncertainty more is entropy
- For a 2 class problem the min entropy is 0 and max is 1

Salary	Age	Purchase
20000	21	No
10000	45	No
60000	27	No
15000	31	No
12000	18	No

Example4

$$H(d)= -P_y \log_2(P_y) -P_n \log_2(P_n)$$

$$H(d)= -0/5\log_2(0/5) -5/5\log_2(5/5)$$

$$H(d)= 0$$

- For more than 2 class the min entropy is 0 and max can be greater than 1

Example5

Salary	Age	Purchase
20000	21	Yes
10000	45	No
60000	27	Yes
15000	31	No
30000	30	Maybe
12000	18	No
40000	40	Maybe
20000	20	Maybe

$$H(d)= - P_y \log_2(P_y) - P_n \log_2(P_n) - P_m \log_2(P_m)$$

$$H(d)= -2/8\log_2(2/8) -3/8\log_2(3/8) - 3/8\log_2(3/8)$$

$$H(d)= 1.56$$

Information Gain

- Information gain is used to determine which feature/attribute gives us the maximum information about a class.
- Information gain is based on the concept of entropy, which is the degree of uncertainty, impurity or disorder.
- Information gain aims to reduce the level of entropy starting from the root node to the leaf nodes.

Decision Tree- Information Gain

- Information Gain=Entropy(Parent)-[(Weighted Avg) *Entropy(Children)
- Entropy(Parent)= $-P(yes)*\log_2 P(yes) - P(no)*\log_2 P(no)$

where,
P(yes)= probability of yes
P(no)= probability of no

Out of 14 records, Count of yes=9 and Count of no=5

Entropy(Parent)=$-P(yes)\log_2 P(yes)-P(no)\log_2 P(no)$
Entropy Parent)=$-(9/14) * \log_2 9/14 - (5/14) * \log_2 5/14$
Entropy(Parent)=0.41+0.53= **0.94**

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Decision Tree- Information Gain

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

STEP1: Entropy of Parent

Out of 14 records, Count of yes=9 and Count of no=5

Entropy(Parent)=-P(yes) $\log_2$ P(yes)-P(no) $\log_2$ P(no)

Entropy Parent)=-(9/14) * $\log_2$ 9/14 - (5/14) * $\log_2$ 5/14

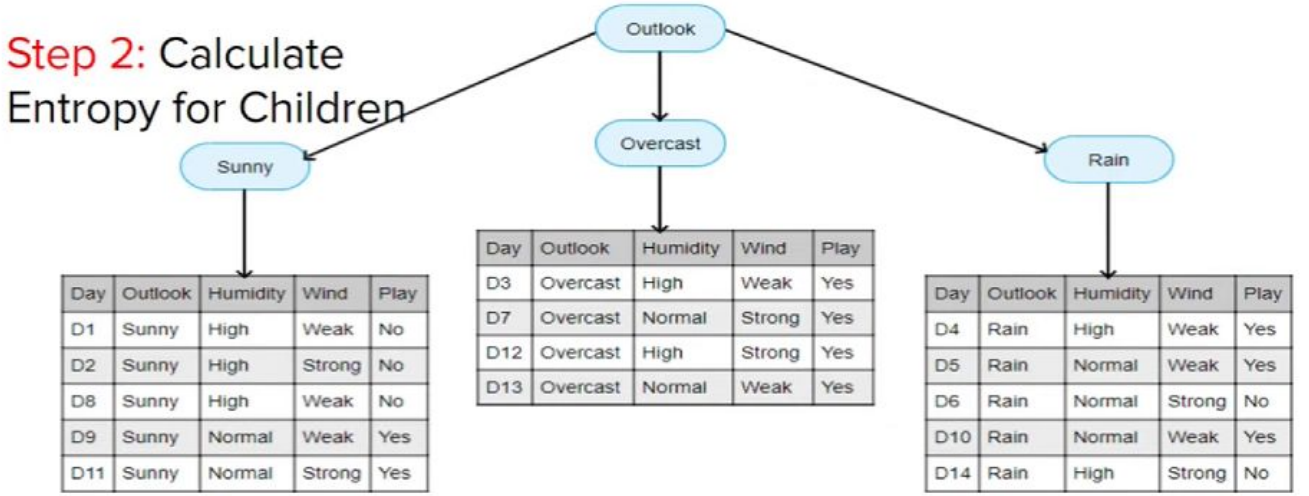
Entropy(Parent)=0.41+0.53= **0.94**

Decision Tree- Information Gain

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Step 2: Calculate Entropy for Children



Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D8	Sunny	High	Weak	No
D9	Sunny	Normal	Weak	Yes
D11	Sunny	Normal	Strong	Yes

Day	Outlook	Humidity	Wind	Play
D3	Overcast	High	Weak	Yes
D7	Overcast	Normal	Strong	Yes
D12	Overcast	High	Strong	Yes
D13	Overcast	Normal	Weak	Yes

Day	Outlook	Humidity	Wind	Play
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No
D10	Rain	Normal	Weak	Yes
D14	Rain	High	Strong	No

E(S) =	E(O) =	E(R) =
-2/5log(2/5) - 3/5log(3/5)	-5/5log(5/5) - 0/5log(0/5)	-3/5log(3/5) - 2/5log(2/5)
E(S)= 0.97	E(O)= 0	E(S)= 0.97

Decision Tree- Information Gain

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

STEP3: Calculate weighted Entropy of Children

Weighted Entropy = 5/14 * 0.97 + 4/14 * 0 + 5/14 * 0.97

W.E(Children) = **0.69**

Decision Tree- Information Gain

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

STEP4: Calculate Information Gain

Information Gain = E(Parent) - {Weighted Average} * E(Children)

IG= 0.97-069=0.25

So the information gain(or the decrease in entropy/impurity) when you split this data on the basis of **Outlook** condition/column is 0.25

Decision Tree- Information Gain

Tennis.csv

outlook	temp	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Step 5 : Calculate Information Gain for all the columns

Whichever column has the highest Information Gain(maximum decrease in entropy) the algorithm will select that column to split the data.

Step 6 : Find Information Gain recursively

Decision tree then applies a recursive greedy search algorithm in top bottom fashion to find Information Gain at every level of the tree.

Once a leaf node is reached (Entropy = 0), no more splitting is done.

Decision Tree- Information Gain

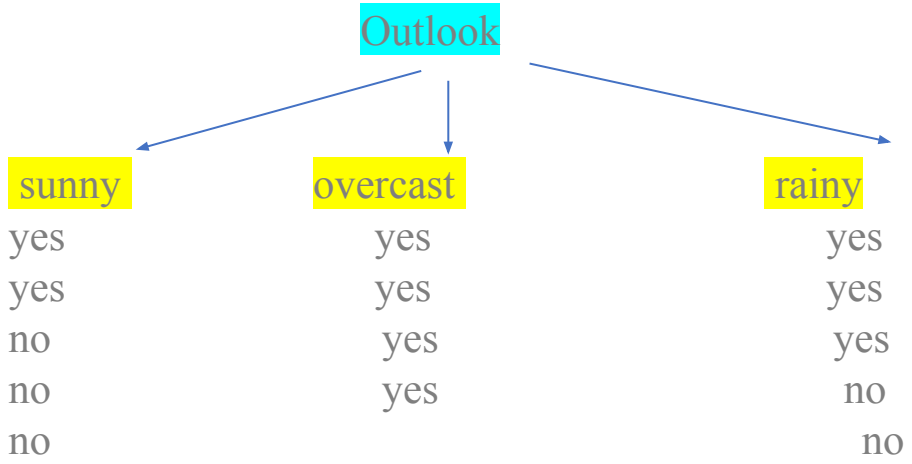
E(outlook=sunny)= -(2/5) *log2 2/5 - (3/5) *log2 3/5= 0.971

E(outlook=overcast)=-(4/4) *log2 4/4 = 0

E(outlook=rainy)=-(3/5) *log2 3/5 - (2/5) *log2 2/5= 0.971

Information from outlook:
I(outlook)=(5/14)*0.971+(4/14)*0+(5/14)*0.971
I(outlook)=0.693

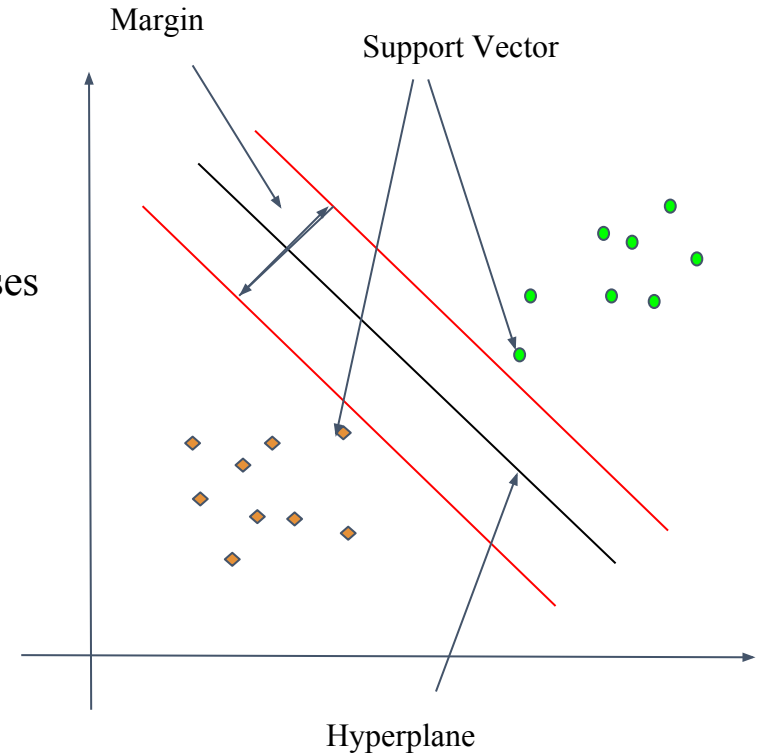
Information gained from outlook=E(S)-I(outlook)
Information gained from outlook=0.94-0.693
Information gained from outlook=0.247



Support Vector Machine

Support Vector Machine

- Supervised learning algorithm
- Used for classification as well as regression problems
- Creates best line or decision boundary to segregate n-dimensional space into classes
- The best decision boundary (Hyperplane)
- SVM selects extreme points or vectors to create hyperplane
- Extreme points are called as support vectors
- Maximum marginal hyperplane



Support Vector Machine

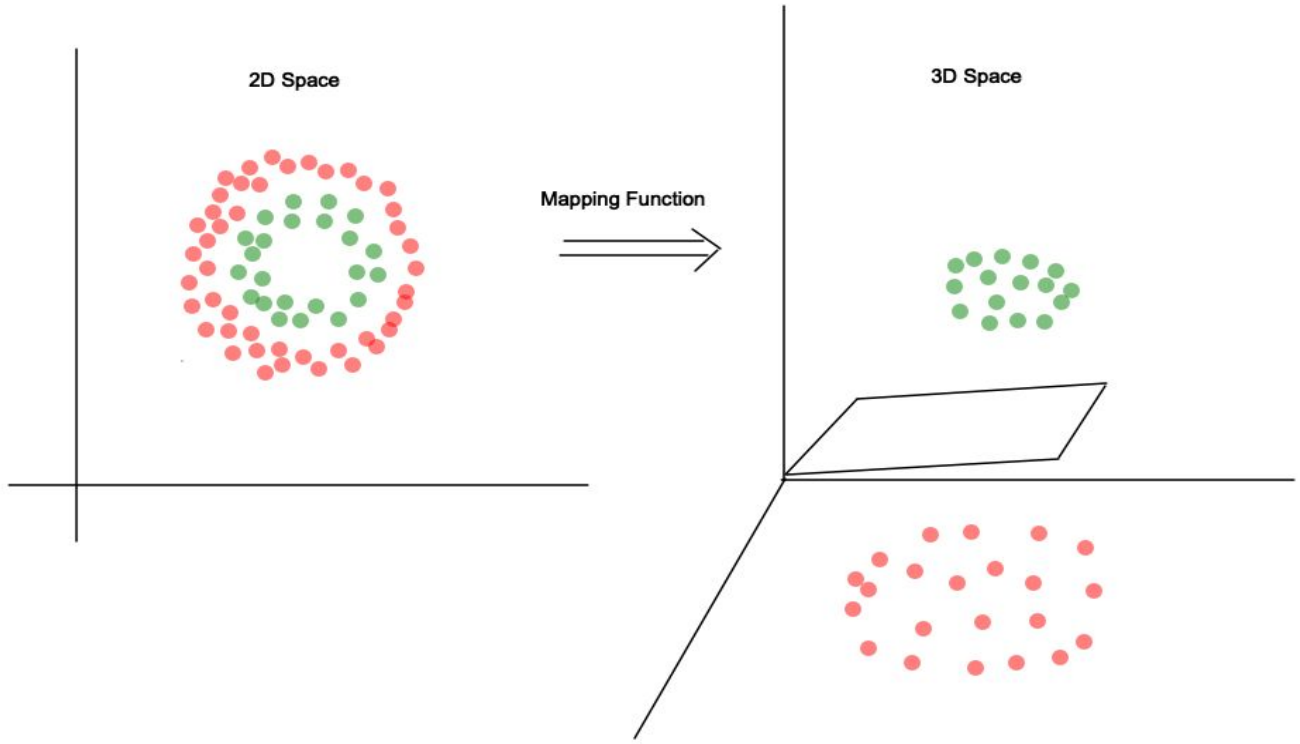
- **Supervised learning algorithm** mostly used for **classification** but it can be used also for **regression**.
- Based on the labeled data (training data) the algorithm tries to find the **optimal hyperplane** which can be used to classify new data points.
- **Other classification algorithms** learns the **differences** while **SVM** learns **similarities**.
- Example: Apple and Lemon
- Support Vector- 1) Apple that is similar to lemon and 2) Lemon that is similar to apple.
- Margin is **orthogonal** to the boundary and **equidistant** to the support vectors.
- **Support vectors** are data points that **defines the position and the margin of the hyperplane**.

Support Vector Machine

The basic steps of the SVM are:

1. Select two hyperplanes (in 2D) which separates the data with no points between them (red lines)
2. Maximize their distance (the margin)
3. The average line (here the line half-way between the two red lines) will be the decision boundary

Support Vector Machine



Support Vector Machine