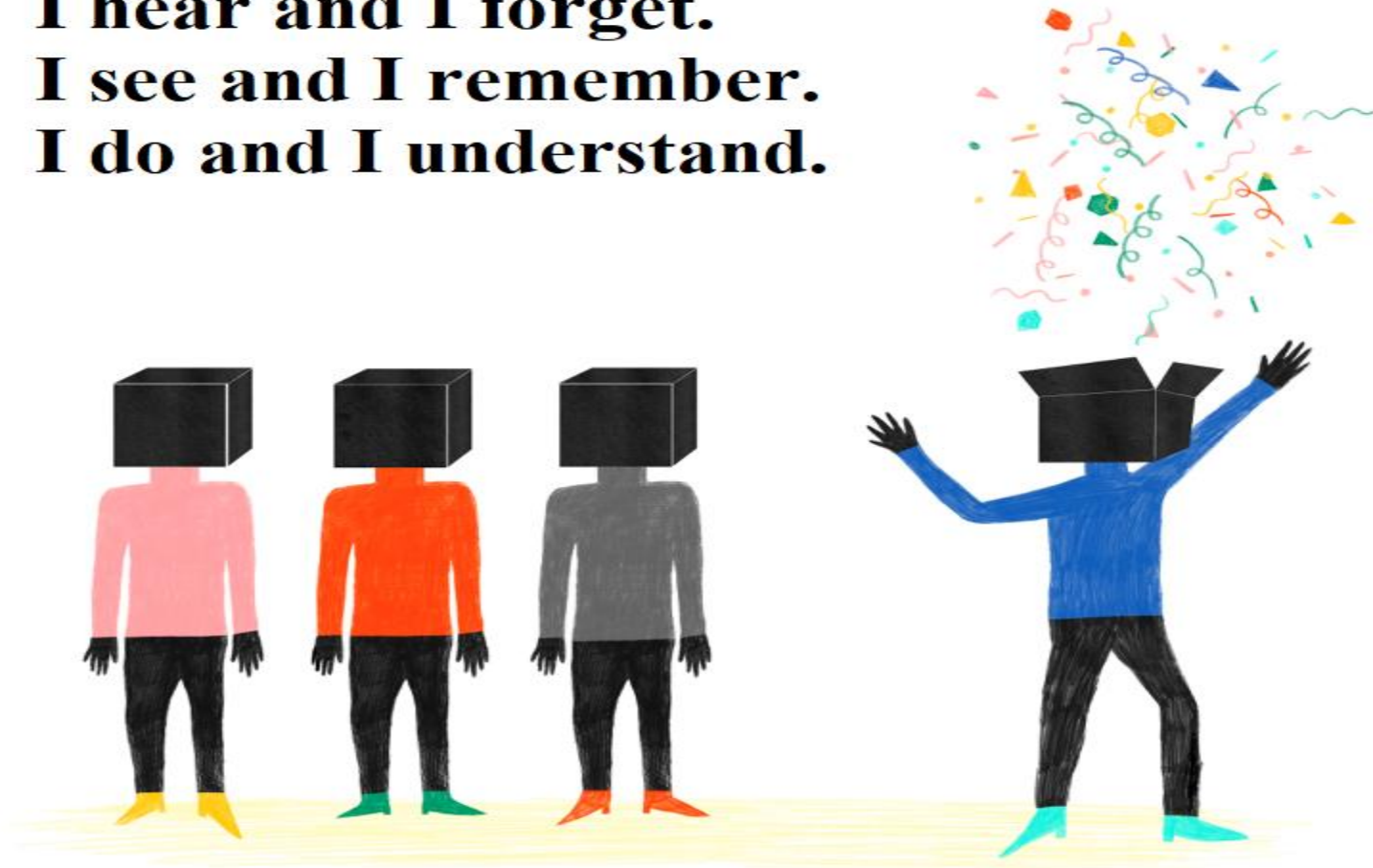


**I hear and I forget.
I see and I remember.
I do and I understand.**



Chandan Verma

Corporate Trainer(Machine Learning,AI,Cloud Computing,IOT)

www.facebook.com/verma.chandan.070

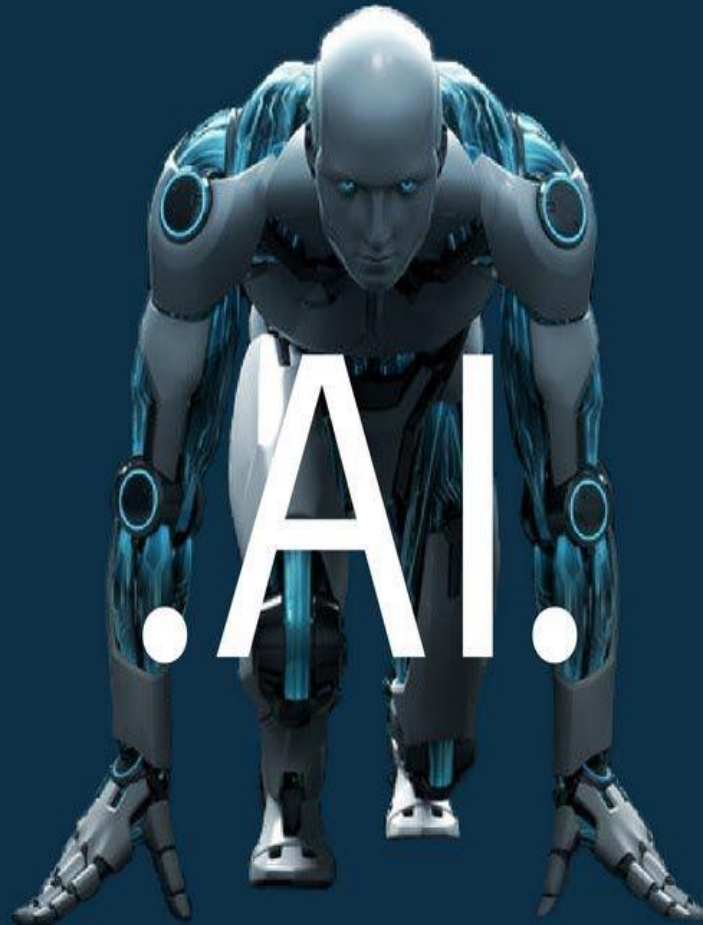
www.chandanverma.com

LINEAR
ALGEBRA

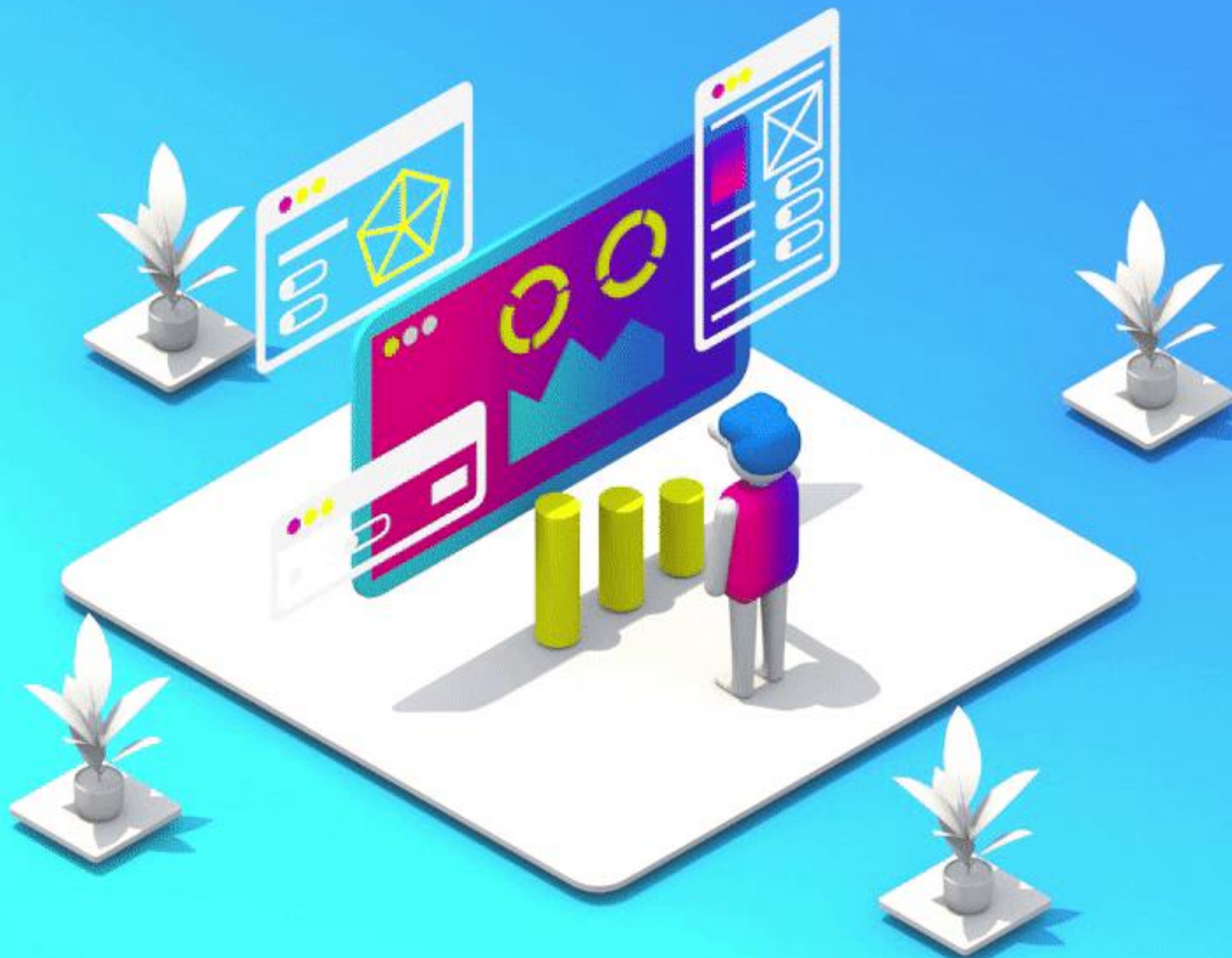
PROBABILITY

STATISTICS

CALCULUS



Math of AI.





Statistics

Statistics is the study of collecting, analyzing and studying data and come up with inferences and prediction about future.

Statistics is the base of all Data Mining and Machine learning algorithms

Statistics is the study of the collection, analysis, interpretation, presentation, and organization of data

Task of statistics

- Designing surveys and experiments
- Summarizing and understanding data
- Estimating population behavior
- Prediction or estimation of future

Statistics

- **Descriptive Statistics**
- **Inferential Statistics**
- **Predictive Modeling**

Statistics is used to summarize numbers for example finding out descriptive statistics like Mean, Median, Mode, Standard Deviation, Variance, Percentiles, Testing hypotheses etc

Descriptive Statistics

Descriptive statistics are very important because if we simply presented our raw data it would be hard to visualize what the data was showing, especially if there was a lot of it. Descriptive statistics therefore enables us to present the data in a more meaningful way, which allows simpler interpretation of the data.

Measures of central tendency: these are ways of describing the central position of a frequency distribution for a group of data.

Measures of spread: these are ways of summarizing a group of data by describing how spread out the scores are

Descriptive statistics.

- Descriptive statistics summarizes or describes characteristics of a data set.
- Descriptive statistics consists of two basic categories of measures: measures of central tendency and measures of variability or spread.
- Measures of central tendency describe the center of a data set.
- Measures of variability or spread describe the dispersion of data within the set.

Descriptive Statistics

1. Mean
2. Median
3. Mode
4. Variance and Standard Deviation
5. Percentiles
6. Testing hypotheses

Descriptive statistics.

Type of Variable	Best Measure of Central Tendency
Categorical	Mode
Ordinal	Median
Continuous (not skewed)	Mean
Continuous (skewed)	Median

Mean Median Mode

Mean, median and mode are together called the measures of central tendency

Mean is given by the total of the values of the samples divided by the number of samples.

The **median** is the number in an ordered set of data that is in the middle.

Mode represents the most common value in a data set. Mode is most useful when you need to understand clustering or number of 'hits'.

The Mean

$$\bar{x} = \frac{\sum x}{N}$$

N is odd

$$\text{Median} = \left(\frac{n + 1}{2} \right)^{th} \text{term}$$

N is even

$$\text{Median} = \frac{\left(\frac{n}{2} \right)^{th} \text{term} + \left(\frac{n}{2} + 1 \right)^{th} \text{term}}{2}$$

Expectation and Moments of the Distribution

Expected value — The expected value of a random variable, also known as the mean value or the first moment, is often noted $E[X]$ or μ and is the value that we would obtain by averaging the results of the experiment infinitely many times. It is computed as follows

$$E[X] = \sum_{i=1}^n x_i f(x_i)$$

$$E[X] = \int_{-\infty}^{+\infty} x f(x) dx$$

Expected Value

The expected value of f is the probability-weighted “average” value of $f(x_i)$.

$$E(f) = \sum_i p(x_i) \cdot f(x_i)$$

Expected value in continuous space

$$E(f) = \int_{x=a \rightarrow b} p(x) \cdot f(x)$$

Variance

Variance — The variance of a random variable, often noted $\text{Var}(X)$ or σ^2 is a measure of the spread of its distribution function. It is determined as follows:

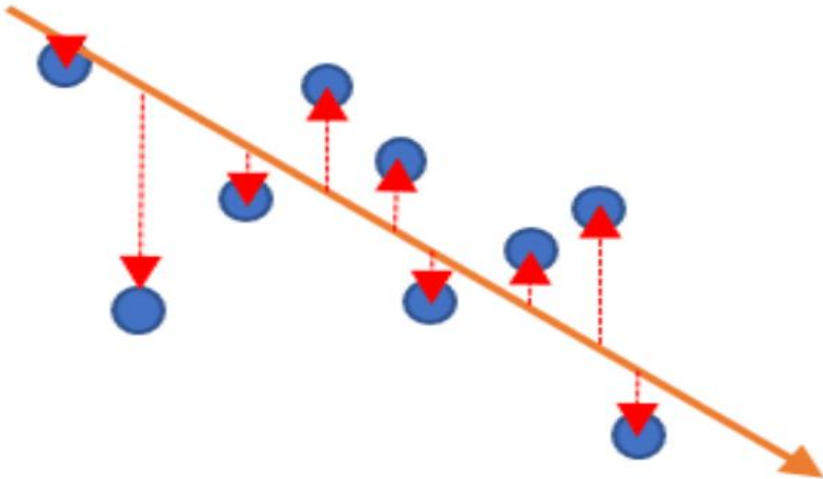
$$\text{Var}(X) = E[(X - E[X])^2] = E[X^2] - E[X]^2$$

Standard deviation — The standard deviation of a random variable, often noted σ , is a measure of the spread of its distribution function which is compatible with the units of the actual random variable. It is determined as follows:

$$\sigma = \sqrt{\text{Var}(X)}$$

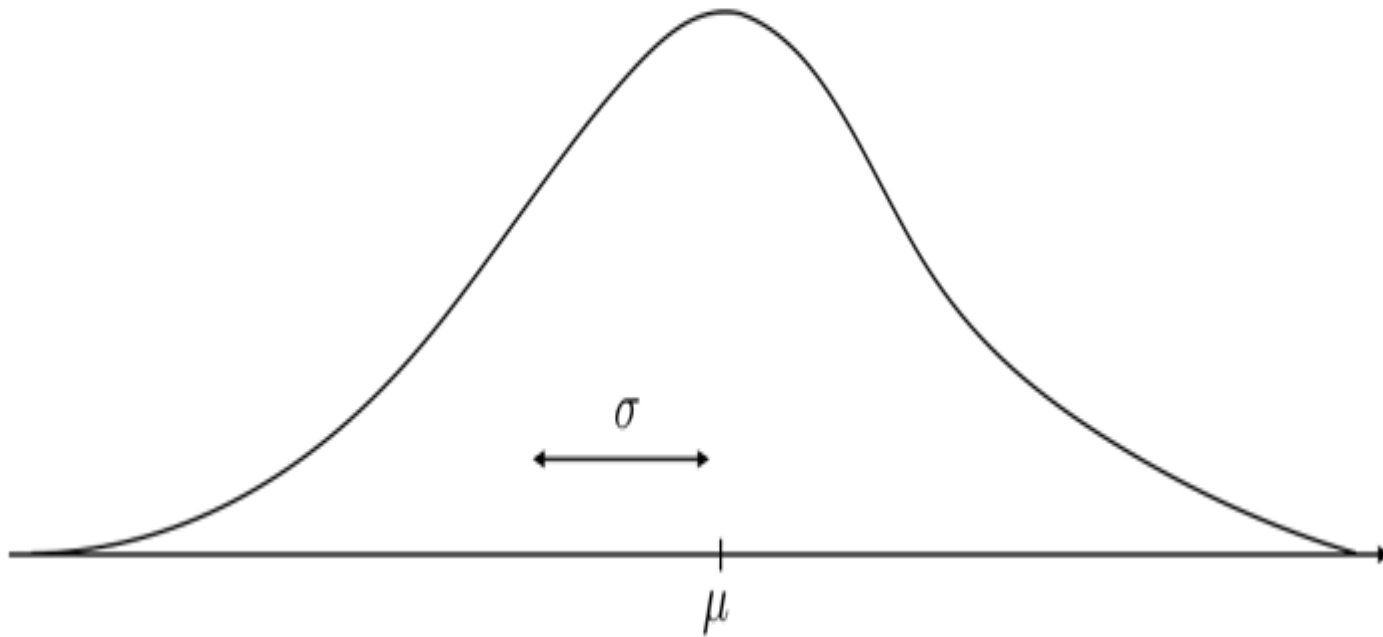
Variance

Variance is a measure of the variability or spread in a set of data. The variance measures how far each number in the set is from the mean



$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$$

Mean and Standard deviation



Normal Distribution

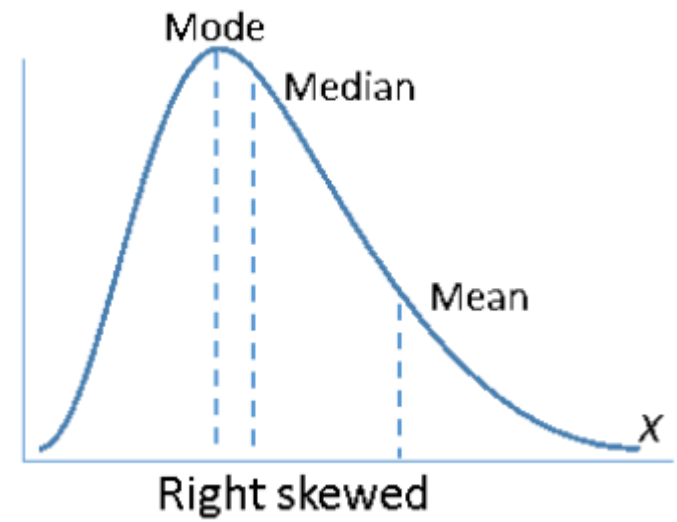
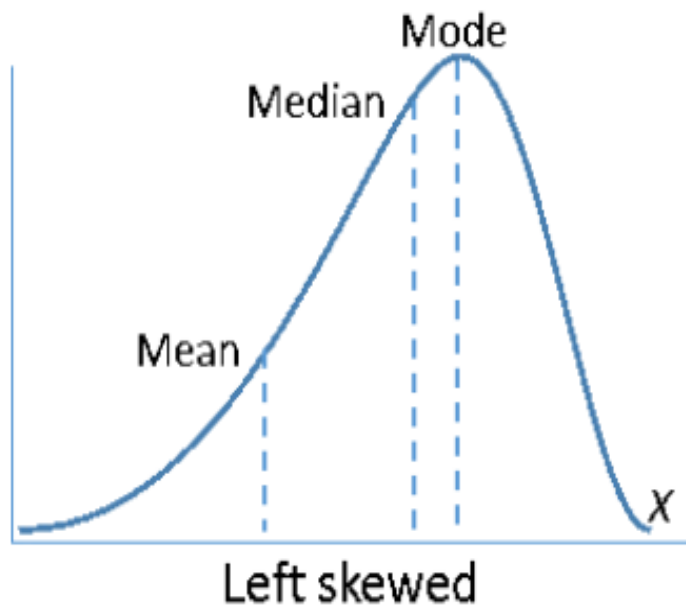
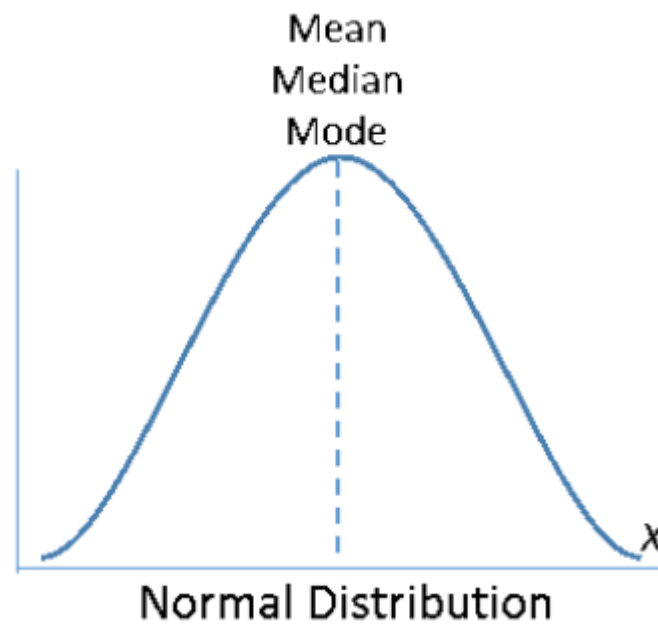
A variable is said to be normally distributed or have a normal distribution if its distribution has the shape of a normal curve — a special bell-shaped curve. ... The graph of a normal distribution is called the normal curve, which has all of the following properties: **mean, median, and mode are equal.**

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$


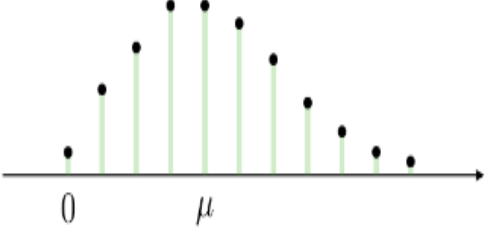
Standardised Normal Distribution

A standard normal distribution is a normal distribution with mean 0 and standard deviation 1

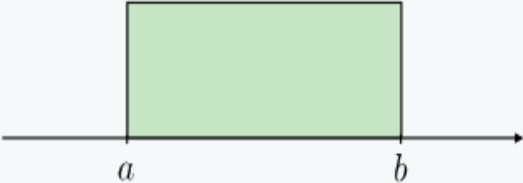
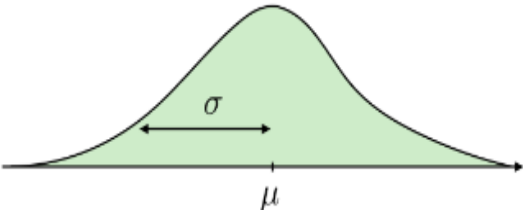

$$x_{new} = \frac{x - \mu}{\sigma}$$



Discrete distributions

Distribution	$P(X = x)$	$\psi(\omega)$	$E[X]$	$\text{Var}(X)$	Illustration
$X \sim \mathcal{B}(n, p)$	$\binom{n}{x} p^x q^{n-x}$	$(pe^{i\omega} + q)^n$	np	npq	
$X \sim \text{Po}(\mu)$	$\frac{\mu^x}{x!} e^{-\mu}$	$e^{\mu(e^{i\omega}-1)}$	μ	μ	

Continuous distributions

Distribution	$f(x)$	$\psi(\omega)$	$E[X]$	$\text{Var}(X)$	Illustration
$X \sim \mathcal{U}(a, b)$	$\frac{1}{b-a}$	$\frac{e^{i\omega b} - e^{i\omega a}}{(b-a)i\omega}$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$	
$X \sim \mathcal{N}(\mu, \sigma)$	$\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$	$e^{i\omega\mu - \frac{1}{2}\omega^2\sigma^2}$	μ	σ^2	
$X \sim \text{Exp}(\lambda)$	$\lambda e^{-\lambda x}$	$\frac{1}{1 - \frac{i\omega}{\lambda}}$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$	

Correlation



Understanding the Relationship between two Variables

What determines house prices?

Size : Typically, larger houses are more expensive



Covariance and Correlation

Covariance — We define the covariance of two random variables X and Y , that we note σ_{XY}^2 or more commonly $\text{Cov}(X, Y)$.

$$\text{Cov}(X, Y) \triangleq \sigma_{XY}^2 = E[(X - \mu_X)(Y - \mu_Y)] = E[XY] - \mu_X\mu_Y$$

Correlation — By noting σ_X, σ_Y the standard deviations of X and Y , we define the correlation between the random variables X and Y , noted ρ_{XY} .

$$\rho_{XY} = \frac{\sigma_{XY}^2}{\sigma_X\sigma_Y}$$

Correlation

Correlation is a statistical technique that can show whether and how strongly pairs of variables are related.

The sample correlation coefficient, r , is used to quantify the strength of the linear association between two variables

It can be useful in data analysis and modeling to better understand the relationships between variables.

Correlation Vs Regression

Correlation is described as the analysis which lets us know the association or the absence of the relationship between two variables 'x' and 'y'.

Regression analysis, predicts the value of the dependent variable based on the known value of the independent variable

How correlation is calculated

The main statistic to measure this **correlation** is called **covariance**.

1. It is the relationship between a pair of random variables where change in one variable causes change in another variable.
2. It can take any value between $-\infty$ to $+\infty$, where the negative value represents the negative relationship whereas a positive value represents the positive relationship.
3. It is used for the linear relationship between variables.
4. It gives the direction of relationship between variables

For Population

$$\text{Cov}(X, Y) = \frac{\sum (X_i - \bar{X})(Y_j - \bar{Y})}{n}$$

For a sample covariance

$$\text{Cov}(X, Y) = \frac{\sum (X_i - \bar{X})(Y_j - \bar{Y})}{n - 1}$$

Where:

- X_i – the values of the X-variable
- Y_j – the values of the Y-variable
- \bar{X} – the mean (average) of the X-variable
- \bar{Y} – the mean (average) of the Y-variable
- n – the number of the data points

Calculating correlation

$$\rho(X, Y) = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}$$

Where:

- $\rho(X, Y)$ – the correlation between the variables X and Y
- $\text{Cov}(X, Y)$ – the covariance between the variables X and Y
- σ_X – the standard deviation of the X-variable
- σ_Y – the standard deviation of the Y-variable

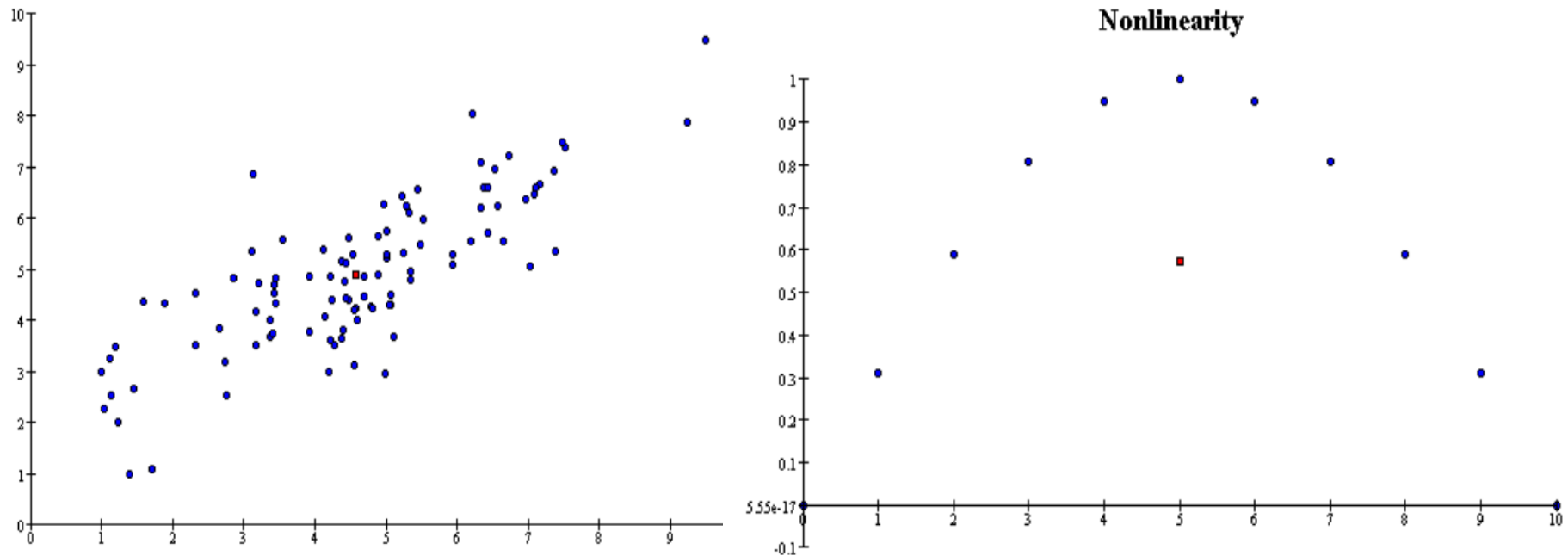
Pearson Correlation Coefficient

Pearson correlation measures the linear association between continuous variables.

A relationship is linear when a change in one variable is associated with a proportional change in the other variable

$$\rho_{X, Y} = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum (X_i - \bar{X})^2 \sum (Y_i - \bar{Y})^2}}$$

Correlation does not measure nonlinear association, only linear association. The correlation coefficient is appropriate only for quantitative variables, not ordinal or categorical variables, even if their values are numerical.



Assumptions for Pearson(r)

- For the Pearson r correlation, both variables should be normally distributed .
- include linearity and homoscedasticity.

Homoscedasticity

Homoscedasticity also referred to as homogeneity of variance or uniformity of variance.

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

Heteroscedasticity (the violation of homoscedasticity) is present when the size of the error term differs across values of an independent variable. The impact of violating the assumption of homoscedasticity is a matter of degree, increasing as heteroscedasticity increases.

Spearman's Correlation

Spearman's correlation coefficient, (ρ , also signified by r_s) measures the strength and direction of association between two ranked variables.

Spearman's correlation measures the strength and direction of monotonic association between two variables. Monotonicity is "less restrictive" than that of a linear relationship

$$\rho_{rank_X, rank_Y} = \frac{cov(rank_X, rank_Y)}{\sigma_{rank_X} \sigma_{rank_Y}}$$

If all ranks are unique (i.e. there are no ties in ranks),

$$\rho_s = 1 - \frac{6 \sum d_i^2}{N(N^2 - 1)}$$

where $d_i = rank(X_i) - rank(Y_i)$ is the difference between the two ranks of each observation and N is the number of observations.

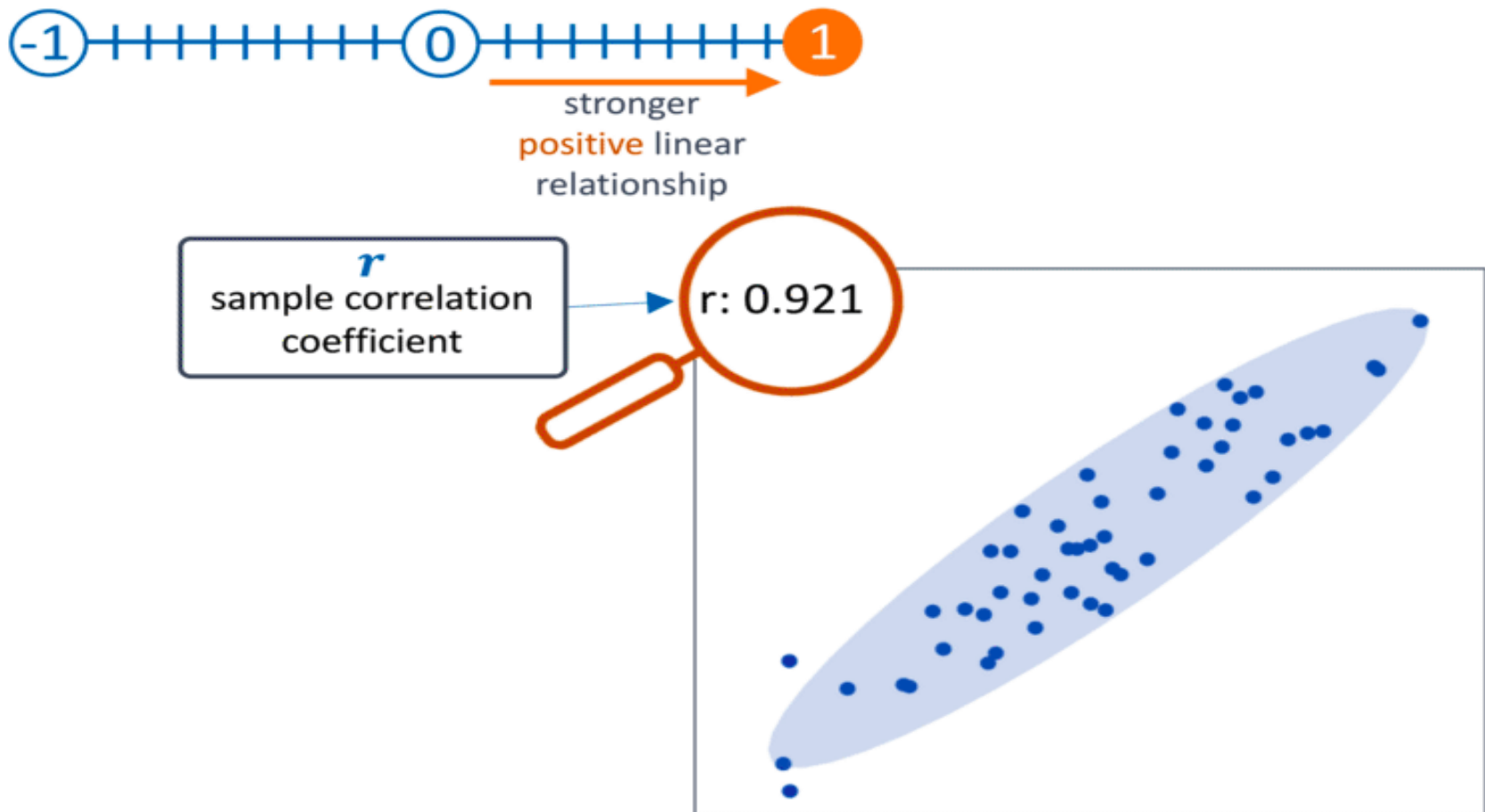
Spearman's vs Pearson's coefficient

Pearson's Measures the strength of the linear relationship between normally distributed variables.

When the variables are not normally distributed or the relationship between the variables is not linear, it may be more appropriate to use the Spearman rank correlation method

Pearson is most appropriate for measurements taken from an **interval scale** (temperature, dates, lengths, etc), while the Spearman is best for measurements taken from **ordinal scales** (rank orders, spectrum of values (agree, neutral, disagree), or healthy vs non-healthy).

- **Positive Correlation:** both variables change in the same direction.
- **Neutral Correlation:** No relationship in the change of the variables.
- **Negative Correlation:** variables change in opposite directions.



Multicollinearity

Multicollinearity is linear dependence between independent variables.

Multicollinearity is a state of very high intercorrelations or inter-associations among the independent variables. It is therefore a type of disturbance in the data, and if present in the data the statistical inferences made about the data may not be reliable.

Generally occurs when the variables are highly correlated to each other.

In almost any business, it is useful to express one quantity in terms of its relationship with others. For example, sales might increase when the marketing department spends more on TV advertisements, or a customer's average purchase amount on an e-commerce website might depend on a number of factors related to that customer.

when we talk of ‘correlation’ between two variables, we are referring to their ‘relatedness’ in some sense

The performance of some algorithms can deteriorate if two or more variables are tightly related, called multicollinearity

If your dataset has perfectly positive or negative attributes then there is a high chance that the performance of the model will be impacted by a problem called—**“Multicollinearity”**.

Multicollinearity happens when one predictor variable in a multiple regression model can be linearly predicted from the others with a high degree of accuracy.

This can lead to skewed or misleading results. Luckily, decision trees and boosted trees algorithms are immune to multicollinearity by nature. When they decide to split, the tree will choose only one of the perfectly correlated features.

However, other algorithms like Logistic Regression or Linear Regression are not immune to that problem and you should fix it before training the model.

Variance-Covariance

Variance measures the variation of a single random variable (like the height of a person in a population), whereas covariance is a measure of how much two random variables vary together (like the height of a person and the weight of a person in a population).

$$\sigma_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

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

Variance-Covariance Matrix

$$C = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})^T$$

$$C = \begin{pmatrix} \sigma(x, x) & \sigma(x, y) \\ \sigma(y, x) & \sigma(y, y) \end{pmatrix}$$

σ^2 COV COV COV COV
 COV σ^2 COV COV COV
 COV COV σ^2 COV COV
 COV COV COV σ^2 COV
 COV COV COV COV σ^2

The diagonal entries of the covariance matrix are the variances and the other entries are the covariances. For this reason, the covariance matrix is sometimes called the _variance-covariance matrix_

Correlation Matrix

A correlation matrix is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables. A correlation matrix is used as a way to summarize data, as an input into a more advanced analysis, and as a diagnostic for advanced analyses.

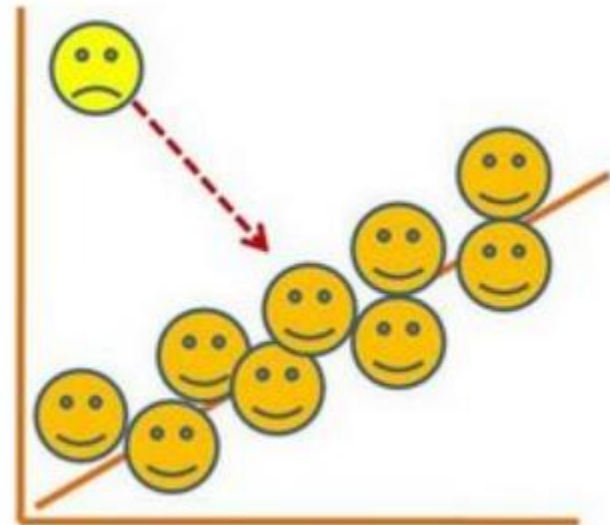
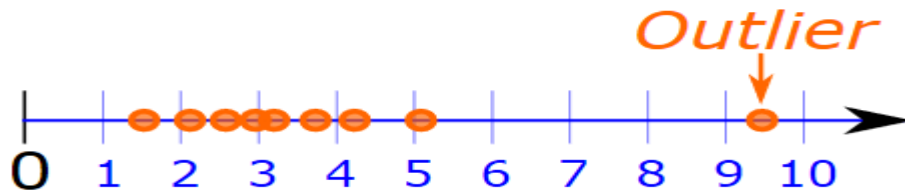
1.00	-0.34	-0.08	-0.04	0.08	0.26
-0.34	1.00	-0.37	0.08	0.02	-0.55
-0.08	-0.37	1.00	-0.31	-0.19	0.10
-0.04	0.08	-0.31	1.00	0.41	0.16
0.08	0.02	-0.19	0.41	1.00	0.22
0.26	-0.55	0.10	0.16	0.22	1.00

Applications of a correlation matrix

- To summarize a large amount of data where the goal is to see patterns.
- To input into other analyses. For example, people commonly use correlation matrixes as inputs for exploratory factor analysis, confirmatory factor analysis, structural equation models, and linear regression when excluding missing values pairwise
- As a diagnostic when checking other analyses. For example, with linear regression a high amount of correlations suggests that the linear regression's estimates will be unreliable.

Outlier

An **outlier** is any value that is numerically distant from most of the other data points in a set of data.



Name	Performance
Ms Dhoni	0.15
Virat	0.11
Rohit	0.06
Dinesh	0.06
Pant	0.12
Rahul	-0.56

The mean is:

$$(0.15+0.11+0.06+0.06+0.12-0.56) / 6 = -0.06 / 6 = -0.01m$$

So, on average the performance went DOWN.

Rahul's result is an "Outlier" ... what if we remove Rahul's result?

Let us try the results **WITHOUT** Rahul:

$$\text{Mean} = (0.15+0.11+0.06+0.06+0.12)/5 = 0.1 \text{ m}$$

When we remove outliers we are **changing the data**, it is no longer "pure", so we shouldn't just get rid of the outliers without a good reason

The **median** ("middle" value):

including Rahul is: **0.085**

without Rahul is: **0.11** (went up a little)

The mode (the most common value):

including Rahul is: **0.06**

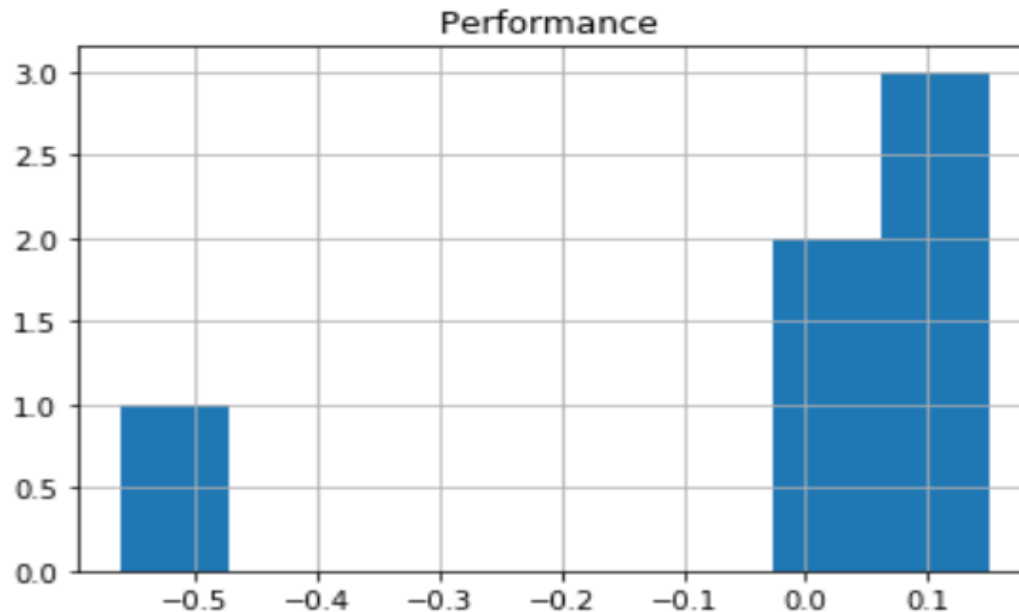
without Rahul is: **0.06** (stayed the same)

The mode and median didn't change very much.

They also stayed around where most of the data is.

Detecting Outliers

The easiest way to detect an outlier is by creating a graph. We can spot outliers by using histograms, scatterplots, number line



Types of outliers

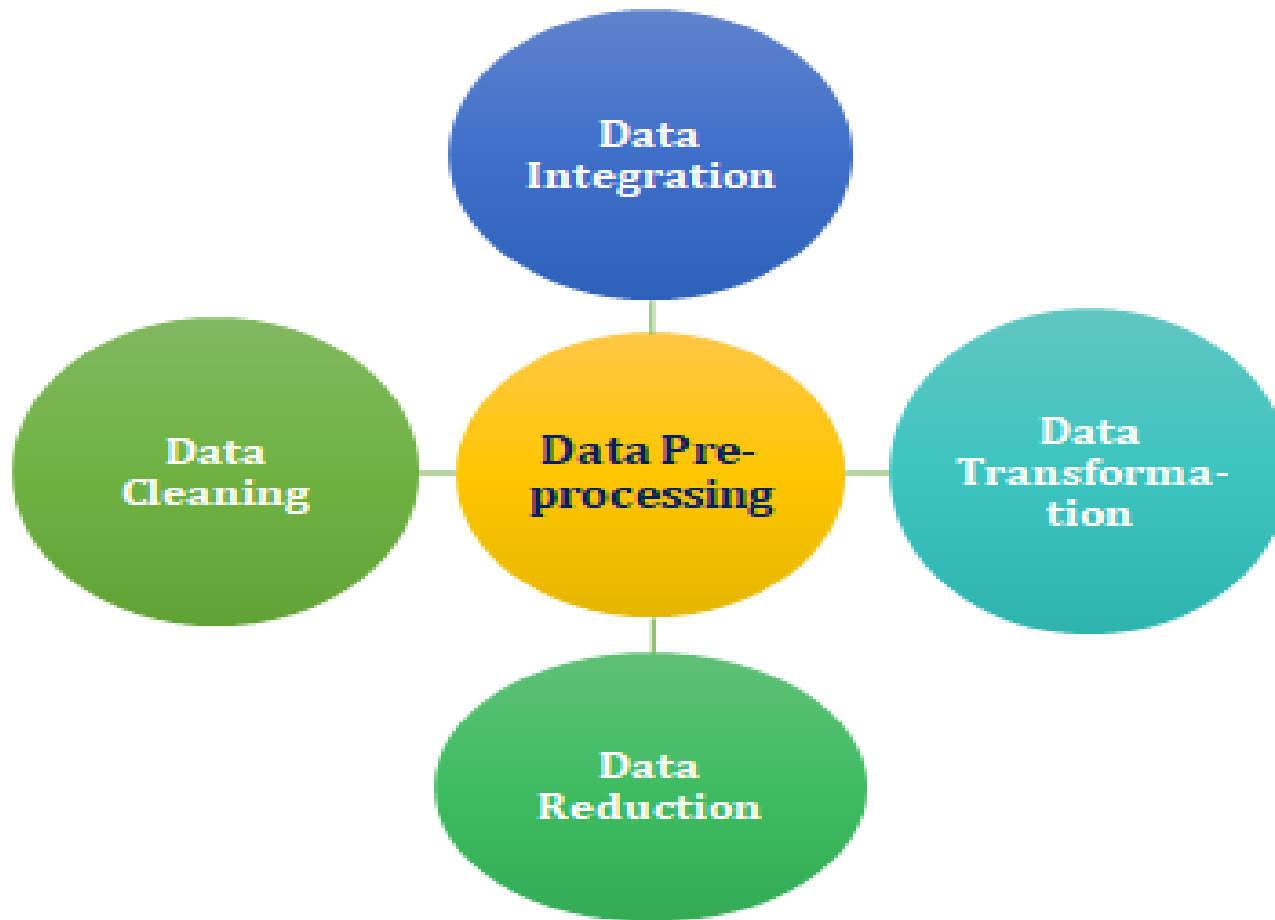
- Univariate
- Multivariate

Univariate outliers can be found when looking at a distribution of values in a single feature space. Multivariate outliers can be found in a n -dimensional space (of n -features).

Most popular methods for outlier detection

- Z-Score or Extreme Value Analysis (parametric)
- Probabilistic and Statistical Modeling (parametric)
- Linear Regression Models (PCA, LMS)
- Proximity Based Models (non-parametric)
- Information Theory Models
- High Dimensional Outlier Detection Methods (high dimensional sparse data)

Data Preprocessing



Country	Age	Salary	Purchased
France	44.000	72000.000	No
Spain	27.000	48000.000	Yes
Germany	30.000	54000.000	No
Spain	38.000	61000.000	No
Germany	40.000	nan	Yes
France	35.000	58000.000	Yes
Spain	nan	52000.000	No
France	48.000	79000.000	Yes
Germany	50.000	83000.000	No
France	37.000	67000.000	Yes

Preprocessing

Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

Pre-processing refers to the transformations applied to your data before feeding it to the algorithm.

Data Preprocessing-Steps

- ✓ Data Cleaning
- ✓ Data Integration
- ✓ Data Transformation
- ✓ Data Reduction

- Separate data into Feature and Response variables
- Taking care of missing data - i by replacing the missing data with average/Mode of the rest of the column .
- Encode categorical data – vocabulary
- Feature scaling - this is important because many machine learning algorithms use Euclidian distance and it can happen that if the features are not within the same scale, one feature will dominate the other .

Feature-Scaling

Bringing The Features onto Same Scale

F1:0-100

F2:1000-100000

Approaches to bringing different features onto the same scale:

1.Normalization

2.Standardization

Normalization

Data normalization is the process of re scaling one or more attributes to the range of 0 to 1. This means that the largest value for each attribute is 1 and the smallest value is 0.

$$x' = \frac{x - \bar{x}}{x_{max} - x_{min}}$$

Standardisation

It is a very effective technique which re-scales a feature value so that it has distribution with 0 mean value and variance equals to 1.

$$z = \frac{x_i - \mu}{\sigma}$$

sklearn.preprocessing

The sklearn.preprocessing package provides several common utility functions and transformer classes to change raw feature vectors into a representation that is more suitable for the downstream estimators

StandardScaler
MinMaxScaler
MaxAbsScaler
binarization

```
import pandas as pd
mydata=pd.read_csv(r"data_preprocessing.csv")
from sklearn import preprocessing
from sklearn import preprocessing
minmaxscaler=preprocessing.MinMaxScaler(feature_range=(
0,1))
stand=preprocessing.StandardScaler()
X_normalize=minmaxscaler.fit_transform(X)
X_standardisation=stand.fit_transform(X)
```

Handling Missing Data

NaN (acronym for Not a Number), it is a special floating-point value recognized by all systems that use the standard IEEE floating-point representation:

isnull(): Generate a boolean mask indicating missing values

notnull(): Opposite of isnull()

dropna(): Return a filtered version of the data

fillna(): Return a copy of the data with missing values filled or imputed

Missing Value

#NaN shows missing values

- **mydata.isnull()**
- **mydata.notnull()**

count number of missing values in each column

- **mydata.isnull().sum()**

#see the all rows of missing values in the column

- **mydata[mydata.City.isnull()]**

Remove Rows With Missing Values

Drop missing observations

df.dropna()

Drop rows where all cells in that row is NA

df.dropna(how='all')

Create a new column full of missing values

df['location'] = np.nan

Drop column if they only contain missing values

df.dropna(axis=1, how='all')

Drop rows that contain less than five observations

df.dropna(thresh=5)

drop missing values

```
mydata.dropna(how='any')
```

```
# drop row if all of the columns are missing
```

```
mydata.dropna(how='all')
```

```
# drop row if either C1 or c2 are missing
```

```
mydata.dropna(subset=['City','State'],how='any')
```

```
drop row if both C1 and Shape c2 are missing
```

```
mydata.dropna(subset=['City','State'],how='all')
```

Fillna()

Pandas provides the fillna() function for replacing missing values with a specific value.

Fill in missing data with zeros

df.fillna(0)

Fill in missing data with zeros

fillna(dataset.mean(), inplace=True)

Pima Indians Diabetes Database

Pregnancies	Number of times pregnant
Glucose	Plasma glucose concentration a 2 hours in an oral glucose tolerance test
BloodPressure	Diastolic blood pressure (mm Hg)
SkinThickness	Triceps skin fold thickness (mm)
Insulin	2-Hour serum insulin (mu U/ml)
BMI	Body mass index (weight in kg/(height in m)^2)
DiabetesPedigree	Diabetes pedigree function
Age	Age (years)
Outcome	Class variable (0 or 1)

columns have an invalid zero minimum value

- 1: Plasma glucose concentration
- 2: Diastolic blood pressure
- 3: Triceps skinfold thickness
- 4: 2-Hour serum insulin
- 5: Body mass index

Count the invalidZeros

```
zero_colum=mydata.iloc[:,1:6]  
(zero_colum==0).sum()
```

Glucose 5
BloodPressure 35
SkinThickness 227
Insulin 374
BMI 11

Handle Missing Data

isnull(): Check If Any Value is NaN

isnull().sum(): Return number of missing value per column.

describe()

Replac the 0 with NaN

```
mydata.iloc[:,1:6]=mydata.iloc[:,1:6].replace(0,np.NaN)
```

count the number of NaN values in each column

```
mydata.isnull().sum()
```

Impute Missing Values

Imputing refers to using a model to replace missing values. There are many options we could consider when replacing a missing value, for example:

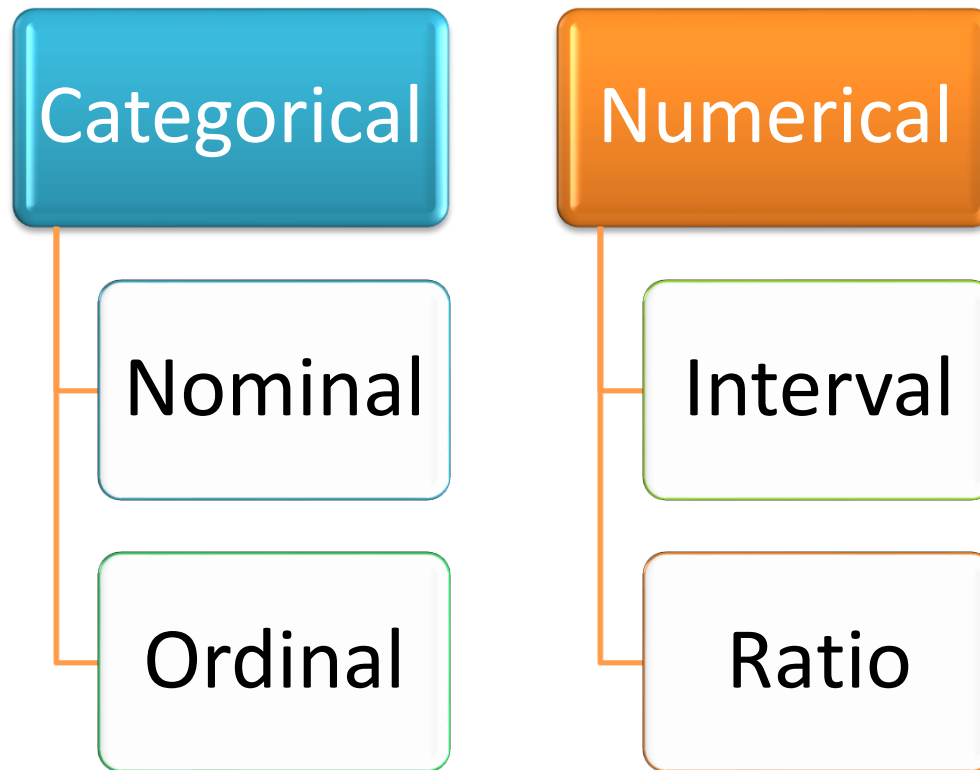
- A constant value that has meaning within the domain, such as 0, distinct from all other values.
- A value from another randomly selected record.
- A **mean, median or mode** value for the column.
- A value estimated by another predictive model.

sklearn.preprocessing.Imputer

```
Imputer(missing_values='NaN',strategy='mean',axis=0)
```

```
from sklearn.preprocessing import Imputer  
imp = Imputer(missing_values='NaN', strategy='mean',  
axis=0)  
imp.fit_transform(X)
```

Types of Data



Categorical

Qualitative data are often termed categorical data.

Nominal (Unordered list)

A variable that has two or more categories, without any implied ordering.

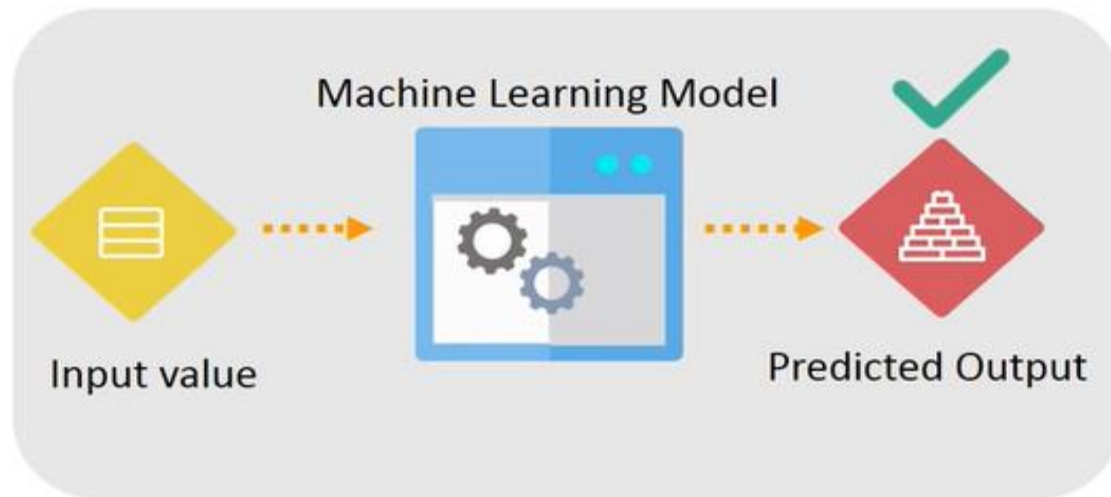
Ordinal Variable (Ordered list)

A variable that has two or more categories, with clear ordering.

- Gender - Male, Female
- Marital Status - Unmarried, Married, Divorcee
- State - New Delhi, Haryana, U.P

- Scale - Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree
- Rating - Very low, Low, Medium, Great, Very great

ENCODING



machine learning algorithms cannot operate on label data directly. They require all input variables and output variables to be numeric.

Converting Categorical Data to Numerical Data

- Label encoding
- One hot encoding
- Target mean encoding
- Binary encoding
- Hash encoding

Label encoding

Label Encoder: It is used to transform non-numerical labels to numerical labels (or nominal categorical variables). Numerical labels are always between 0 and $n_classes-1$.

original dataset

x_1	x_2	y
5	8	calabar
9	3	uyo
8	6	owerri
0	5	uyo
2	3	calabar
0	8	calabar
1	8	owerri

dataset with encoded labels

x_1	x_2	y
5	8	0
9	3	2
8	6	1
0	5	2
2	3	0
0	8	0
1	8	1

```
from sklearn.preprocessing import LabelEncoder  
le=LabelEncoder()  
cardata.iloc[:,0]=le.fit_transform(cardata.iloc[:,0])
```

Label encoding has the advantage that it is straightforward but it has the disadvantage that the numeric values can be “misinterpreted” by the algorithms. For example, the value of 0 is obviously less than the value of 4 but does that really correspond to the data set in real life

One Hot Encoding

One hot encoding is the most widespread approach, and it works very well unless your categorical variable takes on a large number of values

One hot encoding creates new (binary) columns, indicating the presence of each possible value from the original data. Let's work through an example.

original dataset

x_1	x_2	y
5	8	calabar
9	3	uyo
8	6	owerri
0	5	uyo
2	3	calabar
0	8	calabar
1	8	owerri

one-hot encoding of feature x_2

x_1	x_3	$x_{2,0}$	$x_{2,1}$	$x_{2,2}$	y
5	8	1	0	0	calabar
9	3	0	1	0	uyo
8	6	0	0	1	owerri
0	5	0	1	0	uyo
2	3	1	0	0	calabar
0	8	1	0	0	calabar
1	8	0	0	1	owerri

split_train_test

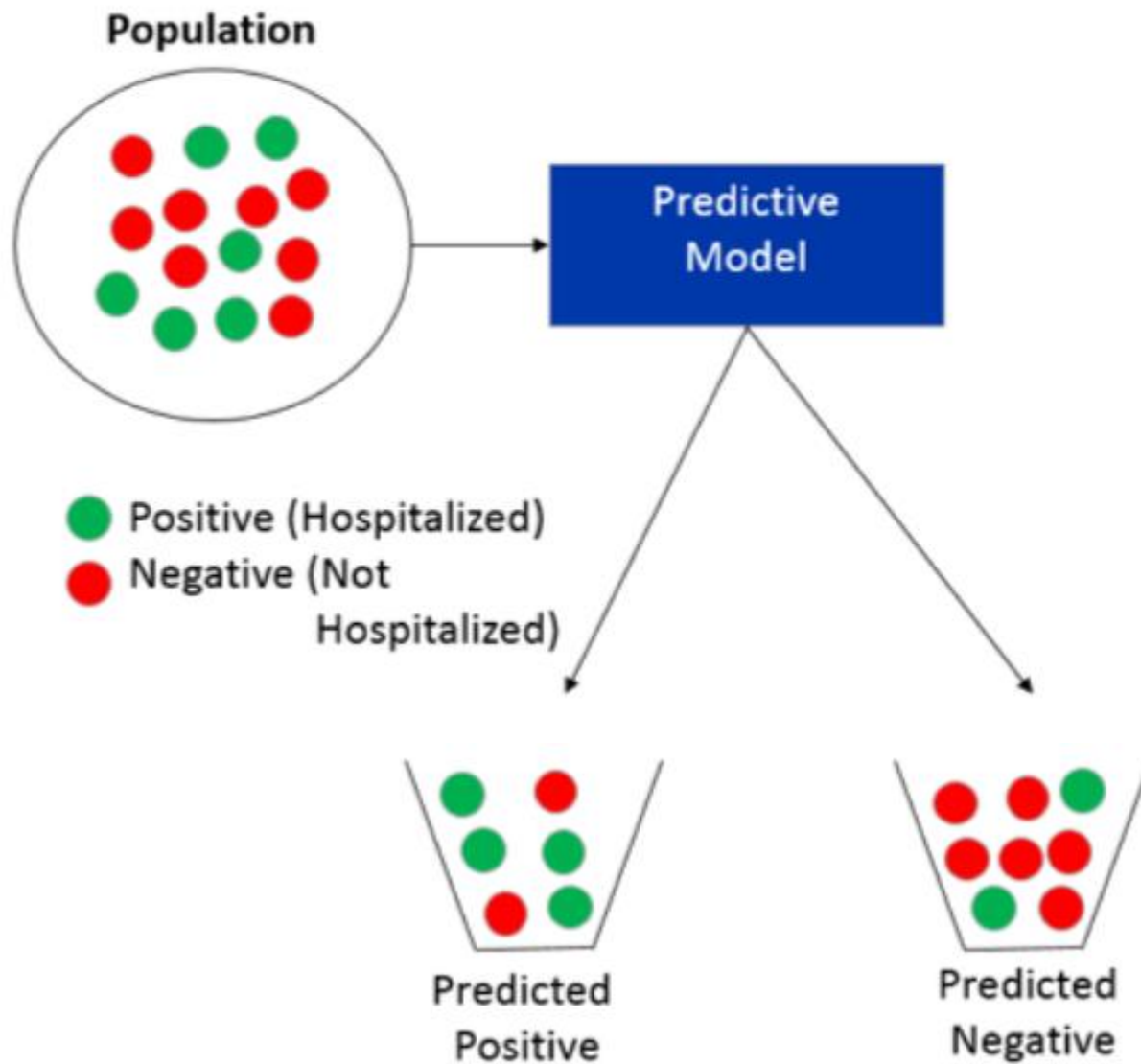
```
def split_train_test(data,test_ratio):  
    shuffled_indices=np.random.permutation(len(data))  
    test_set_size=int(len(data)*test_ratio)  
    test_indices=shuffled_indices[:test_set_size]  
    train_indices=shuffled_indices[test_set_size:]  
    return data.iloc[train_indices],data.iloc[test_indices]
```

Confusion Matrix

The confusion matrix is simply a square matrix that reports the counts of the **true positive**, **true negative**, **false positive** and false **negative**.

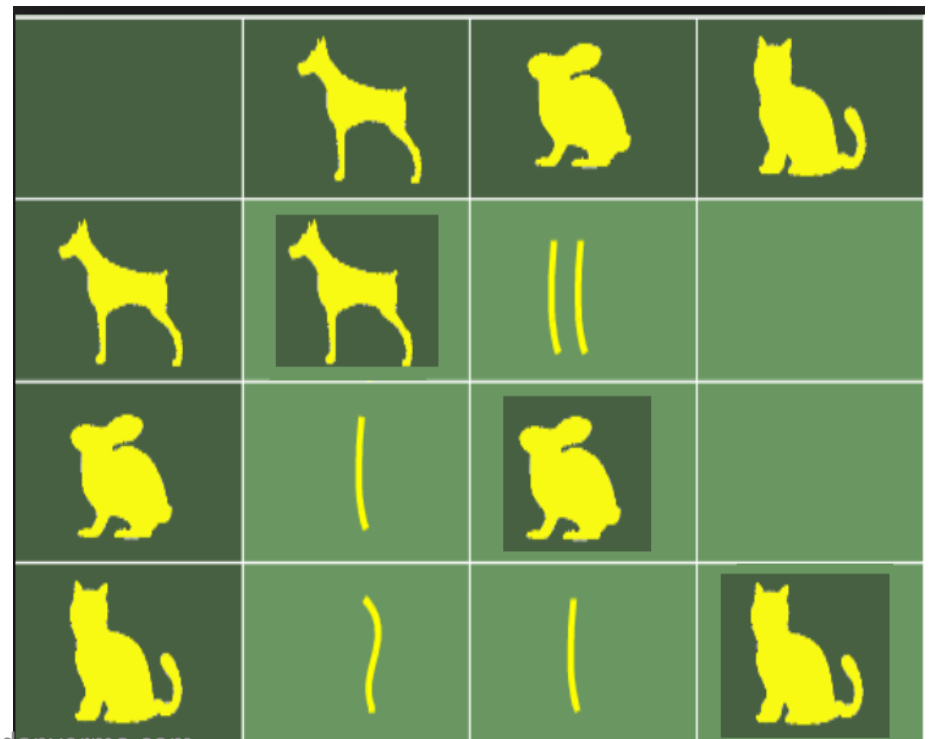
Confusion matrix is a table used to investigate the performance of a classification model where the actual test values are known.

It has two rows and two columns describing the true positives, false positives, false negatives and true negatives.



The confusion matrix will be very helpful in this situation to assess the performance of our model.

		Predicted class	
		P	N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

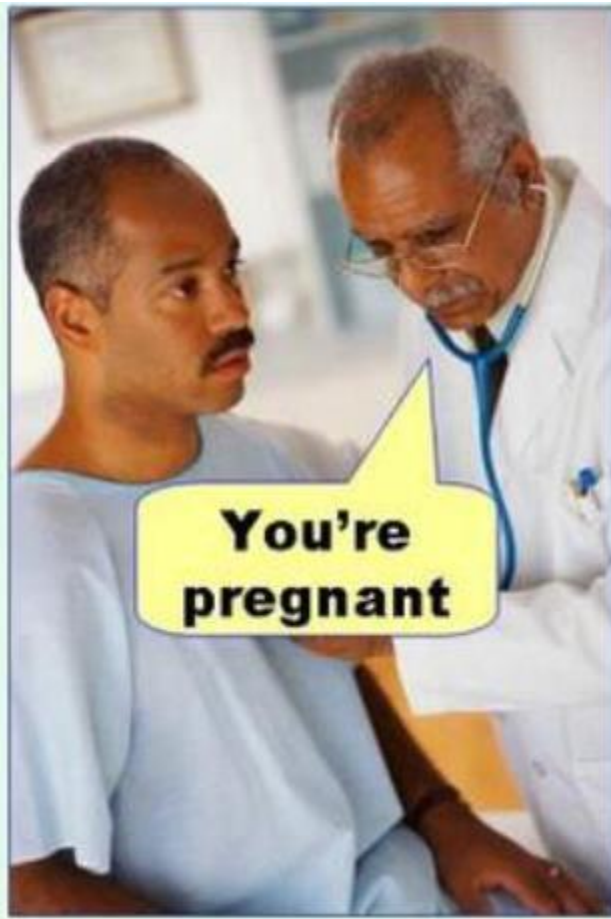


True Positives(TP): These are cases in which we predicted yes (they have the disease), and they do have the disease.

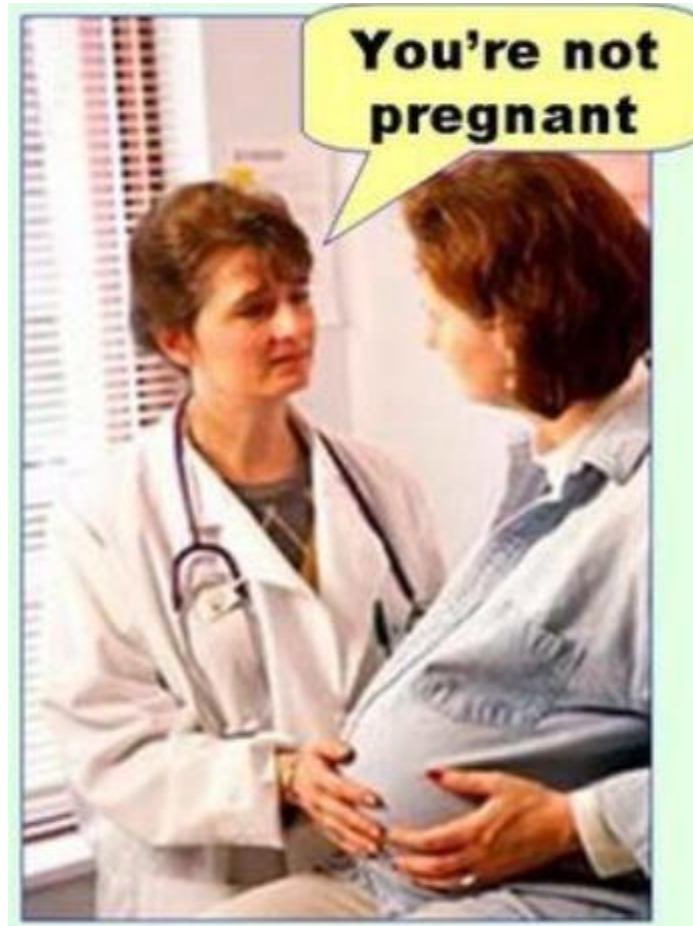
False Positives(FP): We predicted yes, but they don't actually have the disease. (Also known as a "Type I error.")

False Negatives(FN): We predicted no, but they actually do have the disease. (Also known as a "Type II error.")

True Negatives(TN): We predicted no, and they don't have the disease..



False Positives(FP)



False Negatives(FN)

Optimizing a classification model

Both the prediction error and accuracy provide general information about how many samples are misclassified.

Accuracy(ACC): Percentage for correct predictions

$$\text{Accuracy} = \frac{\text{all correct}}{\text{all}}$$

$$\frac{TP + TN}{TP + FN + FP + TN}$$

Prediction error(ERR) or Misclassification Rate: Percentage for incorrect predictions.

$$\text{Misclassification Rate} = \frac{\text{all incorrect}}{\text{all}}$$

$$\frac{FN + FP}{TP + FN + FP + TN} = 1 - \text{Acc}$$

Precision and Recall

Sensitivity(Recall): Percentage of correct predictions for the actual positives(True Positive Rate).

$$\text{Sensitivity} = \frac{\text{true positives}}{\text{all positives}}$$

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall: ability of a classification model to identify all relevant instances

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

High recall, low precision: This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives

Low recall, high precision: This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP)

F1 Score

single metric that combines recall and precision using the harmonic mean

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

F1 Score might be a better measure to use if we need to seek a balance between Precision and Recall and there is an uneven class distribution (large number of Actual Negatives).

sklearn.metrics.confusion_matrix

```
sklearn.metrics.confusion_matrix(y_true,  
y_pred, labels=None, sample_weight=None)
```

`y_true` :Ground truth (correct) target values.

`y_pred` :Estimated targets as returned by a classifier.

`labels` :List of labels to index the matrix.