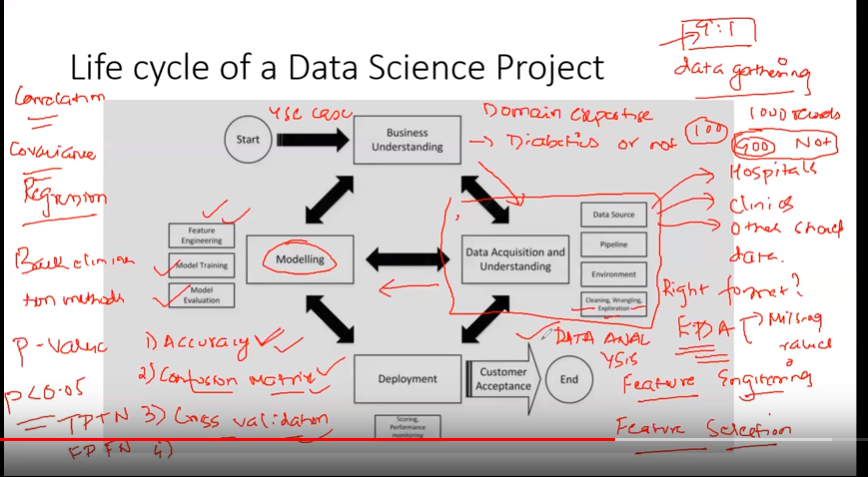
# **1)Complete Life Cycle of a Data Science Project**:

1. **Collect data**
2. **Data preprocessing** (to get data in the right format)
   1. : *Feature engineering (EDA, Data cleaning, Data Wrangling)*
   2. *: Feature selection (avoid features which are not needed and won’t help in model accuracy)*
3. **Check balance vs imbalance Data**?
4. **Data Modeling** (Use below techniques to choose the right algorithm)
   1. *Accuracy*
   2. *Confusion Metrics*
   3. *Cross-Validation (best)*
5. **Model Deployment:** We deploy the model in the form of API(web services) as our model is like a black box (take input and gives output) using Flask or AWSS.
6. If Everything goes fine then our model is good, otherwise, we will start it from starts(we can collect more data and all). So it is a continuous process - to follow the same cycle again and again.

****

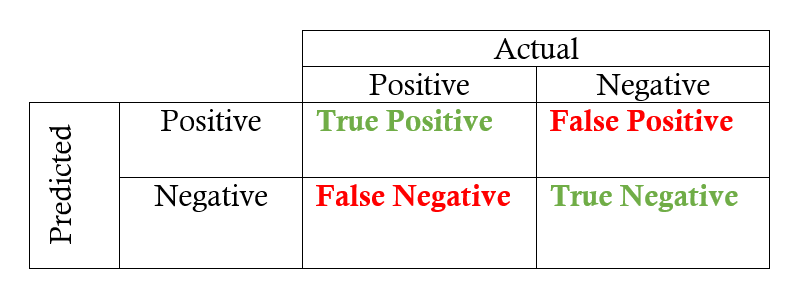
**----------------------------------------------------------------------------------------------------------------**

# **2)Confusion matrix, Precision, Recall**

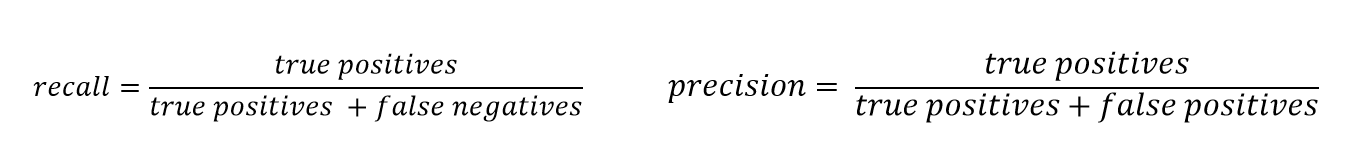
* **TP, TN, FP, FN**
* **TPR, TNR, FPR, FNR**

**Confusion matrix:** will help you to see the accuracy

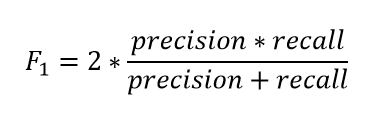
* **Confusion Matrix**: a table showing correct predictions and types of incorrect predictions.



* **Precision:** the number of true positives divided by all positive predictions. Precision is also called Positive Predictive Value. It is a measure of a classifier’s exactness. Low precision indicates a high number of false positives.
* **Recall:** the number of true positives divided by the number of positive values in the test data. The recall is also called Sensitivity or the True Positive Rate. It is a measure of a classifier’s completeness. Low recall indicates a high number of false negatives.



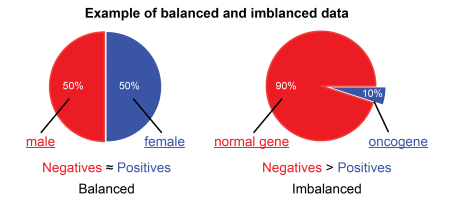
* **F1-Score**: the weighted average of precision and recall.



**----------------------------------------------------------------------------------------------------------------**

# **3)Balanced vs Imbalanced Dataset and how to handle Imbalanced Dataset**

Ref: ([**https://medium.com/analytics-vidhya/what-is-balance-and-imbalance-dataset-89e8d7f46bc5**](https://medium.com/analytics-vidhya/what-is-balance-and-imbalance-dataset-89e8d7f46bc5)**)**

****

## **Problem with an Imbalanced Datasets**

Let’s say you are working in a leading tech company and company is giving you a task to train a model on detecting fraud detection. But here’s the catch. The fraud transaction is relatively rare;

So you start to train your model and get over 95% accuracy.

You feel good and present your model in front of company CEO and Share Holders.

When they give inputs to your model so your model is predicting “Not a Fraud Transaction” every time.

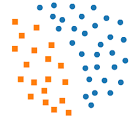
This is clearly a problem because many machine learning algorithms are designed to maximize overall accuracy.

Now, what happens?? You get 95% accuracy but your model in predicting wrong every time??

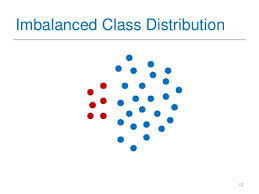
Let’s find out why?

## **What are Balanced and Imbalanced Datasets?**

**Balanced Dataset:** — Let’s take a simple example if in our data set we have positive values which are approximately same as negative values. Then we can say our dataset in balance



**Imbalanced Dataset:** — If there is a very high difference between positive values and negative values. Then we can say our dataset in Imbalance Dataset.



## **Techniques to Convert Imbalanced Dataset into Balanced Dataset**

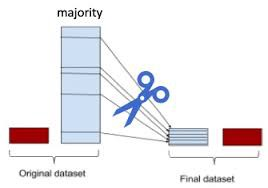
Imbalanced data is not always a bad thing, and in real data sets, there is always some degree of imbalance.

That said, there should not be any big impact on your model performance if the level of imbalance is relatively low.

Now, let’s cover a few techniques to solve the class imbalance problem.

1. **Under-sampling (Down Sampling):**

Unlike oversampling, this technique balances the imbalance dataset by reducing the size of the class which is in abundance. But Since it is removing observations from the original data set, it might discard useful information.

****

**Advantages**

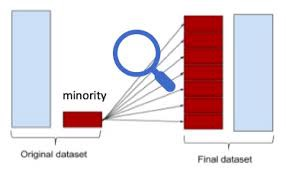
* Run-time can be improved by decreasing the amount of training dataset.
* Helps in solving the memory problems

**Disadvantages**

* Losing some critical information

**2)Over-sampling (Up Sampling):**

* This technique is used to modify the unequal data classes to create balanced datasets. When the quantity of data is insufficient, the oversampling method tries to balance by incrementing the size of rare samples.
* Over-sampling increases the number of minority class members in the training set. The advantage of over-sampling is that no information from the original training set is lost, as all observations from the minority and majority classes are kept. On the other hand, it is prone to overfitting.



**Advantages**

* No loss of information
* Mitigate overfitting caused by oversampling.

**Disadvantages**

* Overfitting

**3) Use Tree-Based Algorithms:**

The final tactic we'll consider is using **tree-based algorithms**. Decision trees often perform well on imbalanced datasets because their hierarchical structure allows them to learn signals from both classes.

In modern applied machine learning, tree **ensembles** (Random Forests, Gradient Boosted Trees, etc.) almost always outperform singular decision trees.

**Note: Well, tree ensembles have become very popular because they perform extremely well on many real-world problems**

**----------------------------------------------------------------------------------------------------------------**

**4)Select Best Model using cross-validation in Python**

**Jupyter Notebook:**(**model\_selections\_K\_Fold\_cross-validation(ModelBoost).ipynb)**

**----------------------------------------------------------------------------------------------------------------**

**5)Principal Component Analysis:**

**Another google doc for this**: (Create Pipeline, PCA, MinMaxScaler, StandardScaler using Sklearn: <https://docs.google.com/document/d/149h8ByEAMhLzGV56KcegJv23MJj1D9Gl-65x0Y_SOoc/edit>)

**Note: As the number of dimensions increases, it is a curse for our model accuracy because:**

**Your accuracy will be impacted based upon your dimensions (more dimension, less accuracy)**

**Jupyter Notebook: (Principal Component Analysis(code).ipynb)**

* Here in the **Jupyter** notebook, the total number of dimensions we have is 30.
* We need to reduce this to 2 dimensions.
* Here in 2 dimensions, we will be having after PCA:

1) WDBC-Malignant

2) WDBC-Benign

**----------------------------------------------------------------------------------------------------------------**

**6) TPR, FPR, FNR, TNR, Confusion Metric**

**----------------------------------------------------------------------------------------------------------------**

**7) Precision, Recall and F1-Score:** These we usually use in information retrieval.

**----------------------------------------------------------------------------------------------------------------**

**8) GridSearchCV-** Select the best hyperparameter for any classification Model**(in order to increase the accuracy of the model)**

**Jupyter Notebook: (GridSearchCV(Code).ipynb)**

**----------------------------------------------------------------------------------------------------------------**

**9) RandomizedSearchCV -** Select the best hyperparameter for any classification:

* **GridsearchCV is usually very slow in comparison to this one.**
* With small data sets and lots of resources, Grid Search will produce accurate results. However, with large data sets, the high dimensions will greatly slow down computation time and be very costly.
* In this instance, it is advised to use Randomized Search since the number of iterations is explicitly defined by the data scientist.

**Juputer:** **(RandomizedSearchCV(Code).ipynb)**

**----------------------------------------------------------------------------------------------------------------**

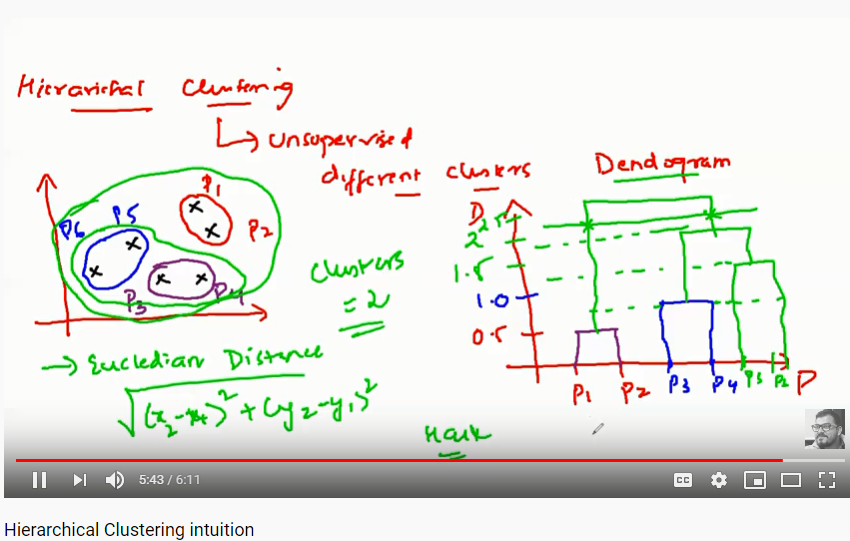
**10) KNN:** Check jupyter for it once and mention that here.

**----------------------------------------------------------------------------------------------------------------**

**11) Hierarchical Clustering Intuition:**

(Unsupervised machine learning algo)

**Jupyter Notebook:** **(Hierarchical\_Clustering.ipynb)**

****

**There are two types of hierarchical clustering:**

1. **Agglomerative** and 2) **Divisive**. In the former, data points are clustered using a bottom-up approach starting with individual data points, while in the latter top-down approach is followed where all the data points are treated as one big cluster and the clustering process involves dividing the one big cluster into several small clusters.

**Agglomerative Clustering:**

#### **Steps to Perform Hierarchical Clustering**

Following are the steps involved in agglomerative clustering:

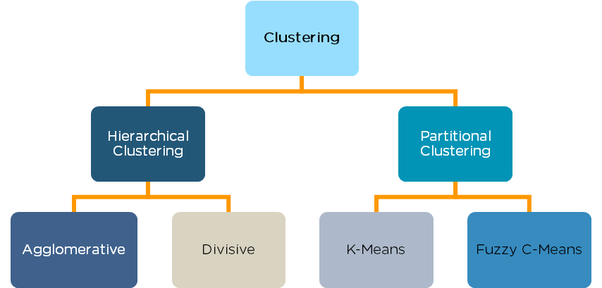
1. At the start, treat each data point as one cluster. Therefore, the number of clusters at the start will be K, while K is an integer representing the number of data points.
2. Form a cluster by joining the two closest data points resulting in K-1 clusters.
3. Form more clusters by joining the two closest clusters resulting in K-2 clusters.
4. Repeat the above three steps until one big cluster is formed.
5. Once single cluster is formed, [dendrograms](https://en.wikipedia.org/wiki/Dendrogram) are used to divide into multiple clusters depending upon the problem. We will study the concept of dendrogram in detail in an upcoming section.

There are different ways to find distance between the clusters. The distance itself can be Euclidean or Manhattan distance. Following are some of the options to measure distance between two clusters:

1. Measure the distance between the closes points of two clusters.
2. Measure the distance between the farthest points of two clusters.
3. Measure the distance between the centroids of two clusters.
4. Measure the distance between all possible combinations of points between the two clusters and take the mean.

#### **Role of Dendrograms for Hierarchical Clustering**

**Very good reference: (**[**https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/**](https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/)**)**

****

**Difference between K-Means and Hierarchical Clustering:**

1. In K means clustering we have to define the number of clusters to be created beforehand, Which is sometimes difficult to say. Whereas in Hierarchical clustering data is automatically formed into a tree shape form (dendrogram) and we can chose which trees are significant.

### **When should I go for K-Means Clustering and when for Hierarchical Clustering ?**

Well, Answer is pretty simple, if your data is small then go for Hierarchical Clustering and if it is large then go for K-Means Clustering.

**----------------------------------------------------------------------------------------------------------------**

**12) Visualize the Decision tree graph:**

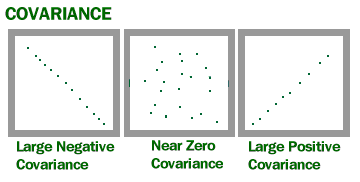
**Jupyter Notebook:** **(Visualize Decision Tree.ipynb)**

**----------------------------------------------------------------------------------------------------------------**

**13) Covariance in statistics:**

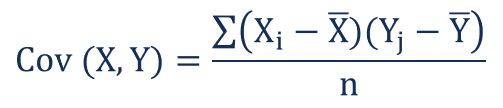
**Covariance is just helping us to quantify the relationship between features(between random variables in the dataset)**

Important topic in case of data preprocessing, **Covariance** is a measure of how much two [random variables](https://www.statisticshowto.datasciencecentral.com/random-variable/) vary together. It’s similar to [variance](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/variance/), but where variance tells you how a *single* variable varies, **co** variance tells you how **two** variables vary together.



* **Positive covariance**: Indicates that two variables tend to move in the same direction.
* **Negative covariance**: Reveals that two variables tend to move in inverse directions.

For example, the covariance between two random variables X and Y can be calculated using the following formula (for population):



Where:

* Xi – the values of the X-variable
* Yj – the values of the Y-variable
* X̄ – the mean (average) of the X-variable
* Ȳ – the mean (average) of the Y-variable
* n – the number of data points

Example:

Calculate covariance for the following data set:

x: 2.1, 2.5, 3.6, 4.0 (mean = 3.1)

y: 8, 10, 12, 14 (mean = 11)

Substitute the values into the formula and solve:

Cov(X,Y) = ΣE((X-μ)(Y-ν)) / n-1

= (2.1-3.1)(8-11)+(2.5-3.1)(10-11)+(3.6-3.1)(12-11)+(4.0-3.1)(14-11) /(4-1)

= (-1)(-3) + (-0.6)(-1)+(.5)(1)+(0.9)(3) / 3

= 3 + 0.6 + .5 + 2.7 / 3

= 6.8/3

= 2.267

The result is positive, meaning that the variables are positively related.

**Note-Impo.:**

* View Covariance as an indicator of how the variables move in correspondence with each other, like whether they move together in the same direction or in the opposite direction or they exhibit random patterns indicating that the variables are independent.
* Correlation coefficient on the other hand is some sort of a standard form of covariance that measures the strength of linear relationship among the two variables, which isn't captured by covariance alone.
* so if covariance is positive we can say that the variables are positively related but a correlation coefficient would further let us investigate into the degree of linear relationship among them.
* If you are familiar with the formula of Correlation coefficient you would know the sign of the coefficient is always determined by Covariance as standard deviations are always non negative quantities.

### **Example of Covariance**

John is an investor. His portfolio primarily tracks the performance of the [S&P 500](https://www.marketwatch.com/investing/index/spx) and John wants to add the stock of ABC Corp.

Before adding the stock to his portfolio, he wants to assess the directional relationship between the stock and the S&P 500.

John does not want to increase the unsystematic risk of his portfolio. Thus, he is not interested in owning securities in the portfolio that tend to move in the same direction.

**----------------------------------------------------------------------------------------------------------------**

**14) Gaussian distribution or Normal Distribution in Statistics:**

1. **Random Variable:**

A *random variable*, usually written *X*, is a variable whose possible values are numerical outcomes of a random phenomenon. There are two types of random variables, *discrete* and *continuous*.

* Discrete Random Variable
* Continuous Random Variable

## **Discrete vs. Continuous Variables**

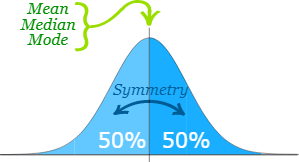
If a [variable](https://stattrek.com/Help/Glossary.aspx?Target=Variable) can take on any value between two specified values, it is called a continuous variable; otherwise, it is called a discrete variable.

**Some examples will clarify the difference between discrete and continuous variables.**

* Suppose the fire department mandates that all firefighters must weigh between 150 and 250 pounds. The weight of a firefighter would be an example of a continuous variable; since a fire fighter's weight could take on any value between 150 and 250 pounds.
* Suppose we flip a coin and count the number of heads. The number of heads could be any integer value between 0 and plus infinity. However, it could not be any number between 0 and plus infinity. We could not, for example, get 2.5 heads. Therefore, the number of heads must be a discrete variable.

1. **Gaussian Distribution(Normal Distribution):**

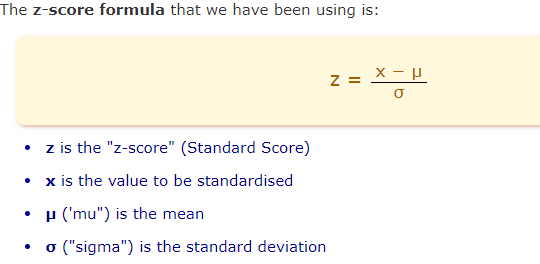
**Impo. :** Gaussian distribution (also known as normal distribution) is a bell-shaped curve, and it is assumed that during any measurement values will follow a normal distribution with an equal number of measurements above and below the mean value.



The **Normal Distribution** has:

* [**mean**](https://www.mathsisfun.com/mean.html) **=** [**median**](https://www.mathsisfun.com/median.html) **=** [**mode**](https://www.mathsisfun.com/mode.html)
* symmetry about the centre
* 50% of values less than the mean  
  and 50% greater than the mean

Note:



**Note: Use above formula to transform Normal/Gaussian distribution to the Standard Normal Distribution**

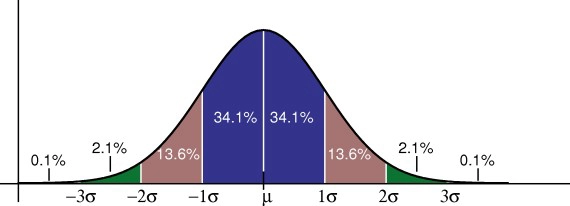
A [**normal distribution**](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/normal-distributions/), sometimes called the bell curve, is a distribution that occurs naturally in many situations. For example, the bell curve is seen in tests like the SAT and GRE. The bulk of students will score the average (C), while smaller numbers of students will score a B or D. An even smaller percentage of students score an F or an A. This creates a distribution that resembles a bell (hence the nickname). The bell curve is symmetrical. Half of the data will fall to the left of the [mean](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/statistics-definitions/mean-median-mode/#mean); half will fall to the right.

he [empirical rule](https://www.statisticshowto.datasciencecentral.com/empirical-rule-2/) tells you what percentage of your data falls within a certain number of [standard deviations](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/standard-deviation/) from the [mean](https://www.statisticshowto.datasciencecentral.com/mean):

• **68%** of the data falls within one [standard deviation](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/standard-deviation/) of the [mean](https://www.statisticshowto.datasciencecentral.com/mean).

• **95%** of the data falls within two [standard deviations](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/standard-deviation/) of the [mean](https://www.statisticshowto.datasciencecentral.com/mean).

• **99.7%** of the data falls within three [standard deviations](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/standard-deviation/) of the [mean](https://www.statisticshowto.datasciencecentral.com/mean).



The standard deviation controls the spread of the distribution. A smaller standard deviation indicates that the data is tightly clustered around the [mean](https://www.statisticshowto.datasciencecentral.com/mean); the normal distribution will be taller. A larger standard deviation indicates that the data is spread out around the [mean](https://www.statisticshowto.datasciencecentral.com/mean); the normal distribution will be flatter and wider.

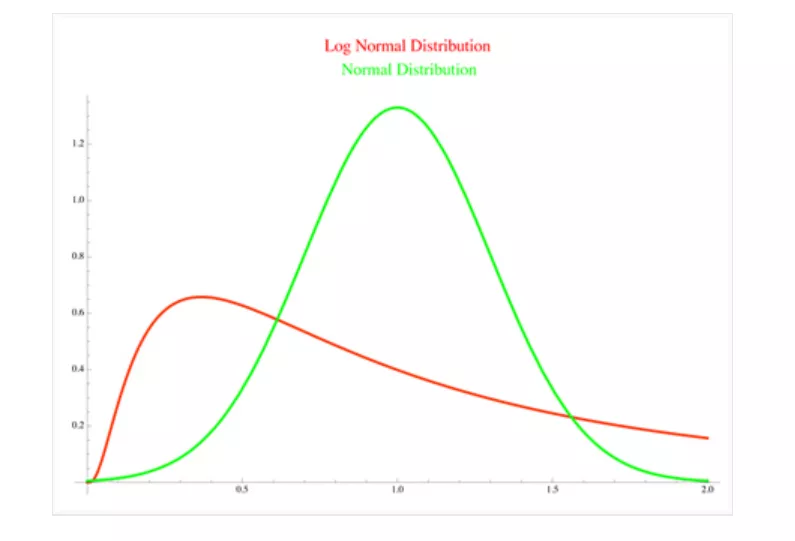
## **Properties of a normal distribution**

* The [mean, mode and median](https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/statistics-definitions/mean-median-mode/) are all equal.
* The curve is symmetric at the center (i.e. around the mean, μ).
* Exactly half of the values are to the left of center and exactly half the values are to the right.
* The total area under the curve is 1.

**----------------------------------------------------------------------------------------------------------------**

**15) Log Normal Distribution in statistics:**

A log-normal distribution is a statistical distribution of logarithmic values from a related normal distribution. A log-normal distribution can be translated to a [normal distribution](https://www.investopedia.com/articles/investing/102014/lognormal-and-normal-distribution.asp) and vice versa using associated logarithmic calculations.



## **Understanding Normal and Lognormal**

A normal distribution is a probability distribution of outcomes that is symmetrical or forms a bell curve. In a normal distribution 68% of the results fall within one standard deviation and 95% fall within two standard deviations.

While most people are familiar with a normal distribution, they may not be as familiar with log-normal distribution. A normal distribution can be converted to a log-normal distribution using logarithmic mathematics. That is primarily the basis as log-normal distributions can only come from a normally distributed set of random variables.

There can be a few reasons for using log-normal distributions in conjunction with normal distributions. In general most log-normal distributions are the result of taking the natural log where the base is equal to e=2.718. However, the log-normal distribution can be scaled using a different base which affects the shape of the lognormal distribution.

Overall the log-normal distribution plots the log of random variables from a normal distribution curve. In general, the log is known as the exponent to which a base number must be raised in order to produce the random variable (x) that is found along a normally distributed curve.

**Impo Notes:**

* **Log-normal distributions are most often used in finance to model stock prices, index values, asset returns, as well as exchange rates, derivatives, etc.**
* The log-normal distribution is the probability distribution of a [random variable](https://brilliant.org/wiki/continuous-random-variables-definition/) whose [logarithm](https://brilliant.org/wiki/logarithms-3/) follows a [normal distribution](https://brilliant.org/wiki/normal-distribution/). It models phenomena whose relative growth rate is independent of size, which is true of most natural phenomena including the size of tissue and blood pressure, income distribution, and even the length of [chess games](https://brilliant.org/wiki/chess-puzzles/).

**----------------------------------------------------------------------------------------------------------------**

**16) Binomial Distribution:**

Binomial distribution is a [probability distribution](https://www.investopedia.com/terms/p/probabilitydistribution.asp) that summarizes the likelihood that a value will take one of two independent values under a given set of parameters or assumptions.

* The underlying assumptions of the binomial distribution are that there is only one outcome for each trial, that each trial has the same probability of success, and that each trial is [mutually exclusive](https://www.investopedia.com/terms/m/mutuallyexclusive.asp), or independent of each other.
* Binomial distribution is a common [discrete](https://www.investopedia.com/terms/d/discrete-distribution.asp) distribution used in statistics, as opposed to a continuous distribution, such as the normal distribution. This is because the binomial distribution only counts two states, typically represented as 1 (for a success) or 0 (for a failure) given a number of trials in the data. The binomial distribution, therefore, represents the probability for x successes in n trials, given a success probability p for each trial

## **Criteria for Using Binomial Distributions**

The binomial distribution is used to model the probabilities of occurrences when specific rules are met.

* Rule #1: There are only two **mutually exclusive** outcomes for a discrete random variable (i.e., success or failure).
* Rule #2: There is a fixed number of **repeated trials** (i.e., successive tests with no outcome excluded).
* Rule #3: Each trial is an **independent event** (meaning the result of one trial doesn't affect the results of subsequent trials).
* Rule #4: The probability of success for each trial is fixed (i.e., the probability of obtaining a successful outcome is the same for all trials).

**Example:**

*Given a couple has 5 children, what is the probability that exactly 3 will be boys?*

* Possible outcomes: boy or girl
* Fixed number of repeated independent trials: 5
* Out of 5 trials, exactly 3 children are boys = success
* Probability of success (0.5) + probability of failure (0.5) = 1

**Note:** The binomial distribution model is an important probability model that is used when there are two possible outcomes (hence "binomial"). In a situation in which there were more than two distinct outcomes, a multinomial probability model might be appropriate, but here we focus on the situation in which the outcome is dichotomous.

**----------------------------------------------------------------------------------------------------------------**

**17) Population vs Sample in Statistics:**

## Population vs Sample

**The main difference between a population and sample has to do with how observations are assigned to the data set.**

* A population includes all of the [elements](https://stattrek.com/Help/Glossary.aspx?Target=element) from a set of data.
* A sample consists of one or more observations drawn from the population.

**----------------------------------------------------------------------------------------------------------------**

**18) Feature Engineering in Python:**

**Github:** [**https://github.com/krishnaik06/Feature-Engineering/blob/master/02.1\_Numerical\_variables.ipynb**](https://github.com/krishnaik06/Feature-Engineering/blob/master/02.1_Numerical_variables.ipynb)

**19) Python lambda (Anonymous Functions) | filter, map, reduce:**

In Python, anonymous function means that a function is without a name. As we already know that def keyword is used to define the normal functions and the lambda keyword is used to create anonymous functions. It has the following syntax:

lambda arguments: expression

* This function can have any number of arguments but only one expression, which is evaluated and returned.
* One is free to use lambda functions wherever function objects are required.
* You need to keep in your knowledge that lambda functions are syntactically restricted to a single expression.
* It has various uses in particular fields of programming besides other types of expressions in functions.

**Filter:** # Python code to illustrate

# filter() with lambda()

**li = [5, 7, 22, 97, 54, 62, 77, 23, 73, 61]**

**final\_list = list(filter(lambda x: (x%2 != 0) , li))**

**print(final\_list)**

**Map:**

**li = [5, 7, 22, 97, 54, 62, 77, 23, 73, 61]**

**final\_list = list(map(lambda x: x\*2 , li))**

**print(final\_list)**

**Reduce:**

**from functools import reduce**

**li = [5, 8, 10, 20, 50, 100]**

**sum = reduce((lambda x, y: x + y), li)**

**print (sum)**

**20)**

# **Python Iterables vs Iterators:**

Iterable is an object, which one can iterate over. It generates an Iterator when passed to iter() method. Iterator is an object, which is used to iterate over an iterable object using \_\_next\_\_() method. Iterators have \_\_next\_\_() method, which returns the next item of the object.

Iteration is a general term for taking each item of something, one after another. Any time you use a loop, explicit or implicit, to go over a group of items, that is iteration.

In Python, iterable and iterator have specific meanings.

An iterable is an object that has an \_\_iter\_\_ method which returns an iterator, or which defines a \_\_getitem\_\_ method that can take sequential indexes starting from zero (and raises an IndexError when the indexes are no longer valid). So an iterable is an object that you can get an iterator from.

An iterator is an object with a next (Python 2) or \_\_next\_\_ (Python 3) method.

Whenever you use a for loop, or map, or a list comprehension, etc. in Python, the next method is called automatically to get each item from the iterator, thus going through the process of iteration.

**Generator-Function :** A generator-function is defined like a normal function, but whenever it needs to generate a value, it does so with the [yield keyword](https://www.geeksforgeeks.org/use-yield-keyword-instead-return-keyword-python/) rather than return. If the body of a def contains yield, the function automatically becomes a generator function.

Note: **Import All Important Python Data Science Libraries Using Pyforest**

* Pip install pyforest

**21)**