For My Parents,

Who continue to inspire me

Manish Mawatwal

101 Questions

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, stored in a database and / or published in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior written permission of the publisher.

Table of Contents

Statistics

Probability

Python

Machine Learning

Deep Learning

**Preface**

Welcome to the world of machine learning, statistics, and deep learning! This book aims to provide you with a concentrated list of questions and ideas to help you understand the fundamental concepts underlying these fascinating fields.

Learning complex subjects can often be a daunting task, especially when faced with a vast sea of information and resources. When I began my own exploration of machine learning, statistics, and deep learning, I found myself overwhelmed by the sheer volume of material available. It was challenging to navigate through the multitude of theories, algorithms, and techniques, which left me feeling uncertain about where to begin and how to progress.

Reflecting on my experiences, I realized that if I could go back and start anew, I would have appreciated a concise and structured approach to learning these subjects. I yearned for a resource that presented a carefully curated selection of questions and ideas, guiding me through the fundamentals and paving the way for deeper understanding.

This book is my attempt to provide aspiring learners like yourself with such a resource. By condensing and organizing the essential concepts, theories, and methodologies, I aim to streamline your learning journey and offer a clear roadmap to grasp the core principles of machine learning, statistics, and deep learning.

Throughout the pages of this book, you will find a collection of questions that delve into various aspects of these fields. These questions are designed to encourage critical thinking and prompt exploration, helping you to develop a solid foundation of knowledge. Additionally, accompanying each question are concise explanations and insights that aim to demystify complex concepts and foster intuitive understanding.

It is important to note that this book does not attempt to replace comprehensive textbooks or in-depth courses. Rather, its purpose is to serve as a supplement—a companion that complements your learning process. By providing a concentrated list of questions and ideas, I hope to spark curiosity, prompt self-reflection, and inspire further exploration.

Remember, the journey of learning is both personal and iterative. Each individual absorbs knowledge in their own unique way, and it is crucial to adapt and tailor the material to suit your specific needs. Embrace the process, embrace the challenges, and let this book be your guide as you embark on a rewarding expedition through the world of machine learning, statistics, and deep learning.

I wish you the best of luck in your endeavours and hope that this book proves to be a valuable resource on your path to understanding the fundamentals of these fascinating fields.

Happy learning!

Manish Mawatwal

**Acknowledgments**

I am filled with deep gratitude as I reflect upon the completion of this book. The journey has been challenging yet rewarding, and I owe my success to the unwavering support and encouragement of numerous individuals who have touched my life along the way.

First and foremost, I extend my heartfelt appreciation to my parents. Their unconditional love, constant belief in me, and countless sacrifices have been the foundation upon which I have built my aspirations. Their unwavering support and guidance have been invaluable, and I am forever grateful for their unwavering faith in my abilities.

To Marya, a dear friend and confidante, thank you for being a constant source of inspiration and motivation. Your unwavering belief in my potential and your encouragement during the toughest of times have been instrumental in shaping my journey. Your insightful feedback and endless discussions have enriched my understanding and contributed immensely to the refinement of my ideas.

To my “Broskies”, my group of friends who have stood by me through thick and thin, thank you for your unwavering camaraderie. Your support, laughter, and shared moments of celebration and solace have given me strength and rejuvenated my spirit. You have been the pillars of my support system, and I cherish the bonds we share.

I am also grateful to my extended family for their constant encouragement and belief in my abilities. Your presence in my life has provided a sense of belonging and a network of support that has been invaluable throughout my journey. Your words of encouragement and shared wisdom have inspired me to strive for excellence.

To everyone else who has played a part, whether through mentorship, guidance, or a kind word of encouragement, I extend my heartfelt gratitude. Each interaction, no matter how small, has left an indelible mark on my journey. Your presence in my life has shaped my perspective and contributed to my growth as an individual.

I would also like to express my deepest appreciation to the countless individuals who have dedicated their time, expertise, and resources to the creation of this book. From editors and reviewers to researchers and professionals in the field, your invaluable contributions have enhanced the quality and depth of this work.

Finally, I am indebted to the readers who will embark on this intellectual journey with me. It is your curiosity, open-mindedness, and eagerness to learn that motivate me to share my knowledge and insights. It is my sincere hope that this book will contribute to your understanding and inspire further exploration in the realms of [subject matter].

Thank you, from the bottom of my heart, to each and every person who has played a part in shaping my path and supporting me on this remarkable journey. Your belief in me has fuelled my determination, and I am profoundly grateful for your presence in my life.

With utmost appreciation,

Manish Mawatwal

**Chapter 1: Statistics**

# What is Central Limit Theorem?

# Hypothesis Testing and Statistical significance of an insight

Used to see if a certain experiment yields meaningful results.

It essentially helps to assess the statistical significance of insight by determining the odds of the results occurring by chance.

Hypothesis testing is used to find out the statistical significance of the insight.

Left-tailed test: p-value = cdf(x)

Right-tailed test: p-value = 1- cdf(x)

Two-tailed test: p-value = 2\*min{cdf(x), 1-cdf(x)}

Null hypothesis is denoted by H0.

**Alternative hypothesis(H1)** - A statement used to contradict the null hypothesis, is the statement that must be true if the null hypothesis is false. It is the opposing point of view that gets proven right when the null hypothesis is proven wrong.

**Statistical significance of an insight** - After calculating the p-value, the null hypothesis is assumed true, and the values are determined. To fine-tune the result, the alpha value, which denotes the significance, is tweaked.

If the p-value turns out to be less than the alpha, then the null hypothesis is rejected, but if it is greater than alpha, the null hypothesis is accepted. This ensures that the result obtained is statistically significant.

# Probability, Likelihood, Prior Probability

# Assumption of Normality

# One sample t-test

A statistical hypothesis test in which we check if the mean of the sample data is statistically or significantly different from the population’s mean.

E.g., We measure the grams of protein for a sample of energy bars. The label claims that the bars have 20 grams of protein. We want to know if the labels are correct or not.

# PAC Learning?

# Long-tailed Distribution

# Confidence interval, p-values, point estimates?

**Confidence interval** - The probability that a population parameter will fall between a set of values for a certain proportion of times. They are used in hypothesis testing and regression analysis.

A range of values likely containing the population parameter is given by the confidence interval. The Confidence Coefficient/Confidence level is denoted by , which gives the probability or likeness. The **level of significance** is given by *α*.

# Standard Deviation, Expectation

# Normal Distribution, empirical rule, z-score

Quantitative data is known as numeric data. Qualitative data is known as categorical data.

# Symmetric Distribution

# Handling Missing Data

* Prediction of the missing values
* Assignment of individual (unique) values
* Deletion of rows, which have the missing data
* **Mean imputation or median imputation** - Null values in a dataset are replaced directly with the corresponding mean of the data. It is a bad practice as it removes the accountability for feature correlation. The data will have low variance and increased bias, adding to the dip in accuracy of the model.
* Using random forests, it supports the missing values

# Exploratory Data Analysis, Six sigma, Sensitivity

**EDA** - The process of performing investigations on data to understand the data better. Initial investigations are done to determine patterns, spot abnormalities, test hypotheses, and also check if the assumptions are right.

# Outliers, Inliers

# Quartile, 5-number summary

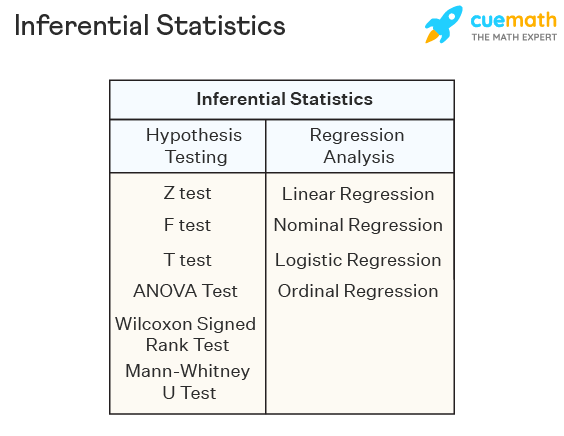
# Kurtosis

# Difference between population and sample?

Population is the entire group that you want to draw conclusion about. A population is a large volume of observations. The sample is a small portion of that population. A sample is the specific group that you will collect data from.

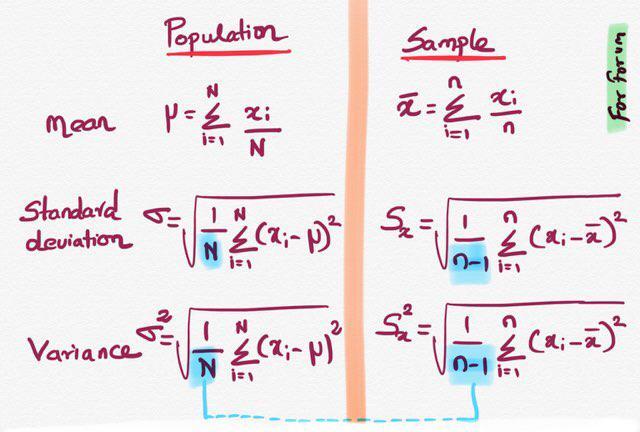
|  |  |
| --- | --- |
| Population parameters | Sample parameters |
| * Mean = * Standard deviation = | * Mean = * Standard deviation = s |

# Descriptive and Inferential Statistics

Descriptive statistics summarizes a sample set of data like the standard deviation or the mean. Inferential statistics is used to draw conclusions from the test data that are subjected to random variations. Inferential statistics is used in research, government operations, quality control and quality assurance teams in MNCs.

# Bessel’s Correction

It is a factor that is used to estimate a populations’ standard deviation from its sample. It causes the standard deviation to be less biased, thereby, providing more accurate results. Bessel’s correction advocates the use of n-1 instead of n in the formula of standard deviation.



# DOE

# Pareto Principle

80/20 rule, which means that 80% of the results are obtained from 20% of the causes in an experiment.

# Skewness

# Degree of Freedom

DF is the number of options at hand when performing an analysis. It is used with t-distribution, chi-square test.

Degree of freedom refers to the maximum number of logically independent values, which are values that have the freedom to vary, in the data sample.

Example, consider a data sample consisting of five positive integers. The values of the five integers must have an average of six. If four of the items within the data set are {3, 8, 5, and 4}, the fifth number must be 10. Because the first four number can be chosen at random, the degree of freedom is four.

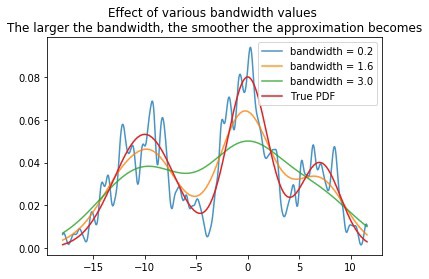
Consider a data sample consisting of one integer. That integer must be odd. Because there are constraints on the single item within the data set, the degrees of freedom are zero.

Degrees of freedom tells you how many units within a set can be selected without constraints to still abide by a given rule overseeing the set. For example, consider a set of five items that add to an average value of 20. Degrees of freedom tell you how many of the items (4) can be randomly selected before constraints must be put in place. In this example, once the first four items are picked, you no longer have liberty to randomly select a data point because you must “force balance” to the given average.

# What does kernel density estimation do?

The Kernel Density Estimation is a mathematic process of finding an estimate probability density function of a random variable. The estimation attempts to infer characteristics of a population, based on a finite data set. The data smoothing problem often is used in signal processing and data science, as it is a powerful way to estimate probability density. In short, the technique allows one to create a smooth curve given a set of random data. However, the estimation can also be used to generate points that only appear to have come from a specific sample set. This feature is particularly useful in project simulations and in object modelling.

The Kernel Density Estimation works by plotting out the data and beginning to create a curve of the distribution. The curve is calculated by weighing the distance of all the points in each specific location along the distribution. If there are more points grouped locally, the estimation is higher as the probability of seeing a point at that location increases. The kernel function is the specific mechanism used to weigh the points across the data set. The bandwidth of the kernel changes its shape. A lower bandwidth limits the scope of the function and leads to the estimate curve looking rough and jagged. By tweaking the parameters of the kernel function (bandwidth and amplitude), one changes the size and shape of the estimate.



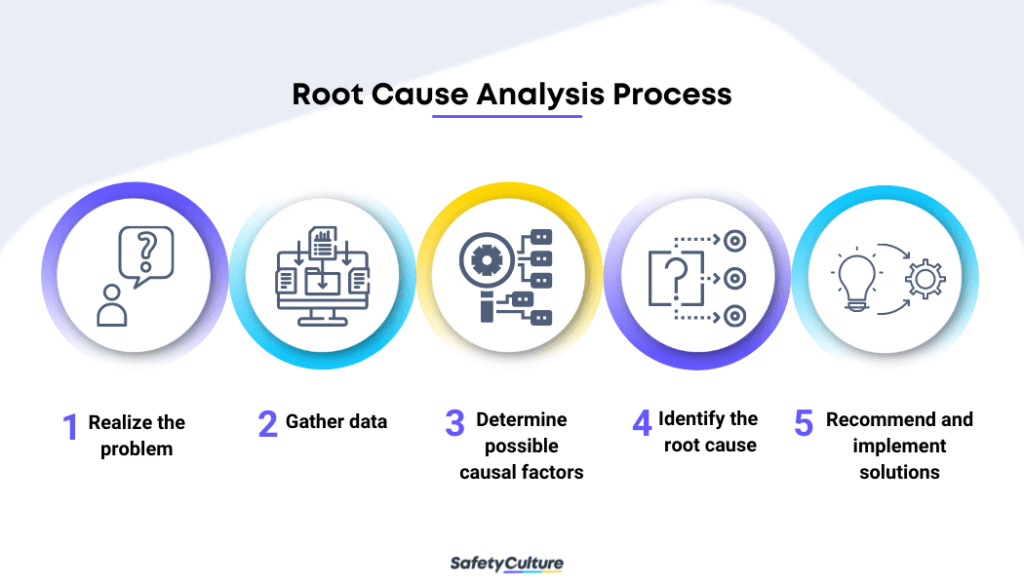
# Heteroscedastic Model

# Autocorrelation

# Correlation and Covariance

# Confounding Variable, Statistical Interaction

# Cherry-picking, p-hacking, Significance Chasing, Root cause analysis?

**Root Cause Analysis** - It is a problem-solving technique used for isolating the root causes of faults or problems. A factor is called root cause if its deduction from the problem-fault-sequence averts the final undesirable event from recurring. Initially developed to analyse industrial accidents. Example: If the higher crime rate in a city is directly associated with the higher sales in a red-coloured shirt, it means that they are having a positive correlation. However, this does not mean that one causes the other. Causation can be tested using A/B testing or hypothesis testing.

# Confidence level and Significance level

# Difference between Confidence test and Hypothesis test

# Power of Test

# Bayesian and Frequentist

# Reason of using Median instead of Mean

There are cases in statistical analysis in which the median may provide a better understanding of the data than the mean. E.g., when dealing with datasets that have many outliers, the median can provide a better idea of a data trend without the distortion that outliers might cause in the average. It is best to use the mean to describe the centre of a dataset when the distribution is mostly symmetrical and there are no outliers. When a distribution is skewed, the median does a better job of describing the centre of the distribution than the mean.

# Normalisation and Standardization

|  |  |
| --- | --- |
| Standardization | Normalization |
| Technique of converting data in such a way that it is normally distributed and has a standard deviation of 1 and a mean of 0. | Technique of converting all data values to lie between 0 and 1. This is also known as min-max scaling. |
|  |  |

# When should you use a t-test vs a z-test?

z-test and t-test are statistical tests used to determine whether the two-population means are different when the variances are known, and the sample is large.

z-test is used for hypothesis testing with a normal distribution. It is used to determine population variance in the case where a sample is large or n > 30. Population standard deviation is required for z-test.

t-test is used with a t-distribution and used to determine population variance when you have a sample size n < 30. T-test requires degree of freedom. If there is an increase in DF, the t-distribution will reach closer to the normal distribution. If DF > 30, this means that the t-distribution at hand is having all of the characteristics of a normal distribution.

Where, is degrees of freedom, N is sample size.

# What is ANOVA test?

Analysis of Variance is a statistical formula used to compare variances across the means or average of different groups. A range of scenarios uses it to determine if there is any difference between the means of different groups.

ANOVA test is the way to find out if survey or experiment results are significant. It helps us to figure out if we need to reject the null hypothesis or accept the alternate hypothesis. Types of ANOVA:

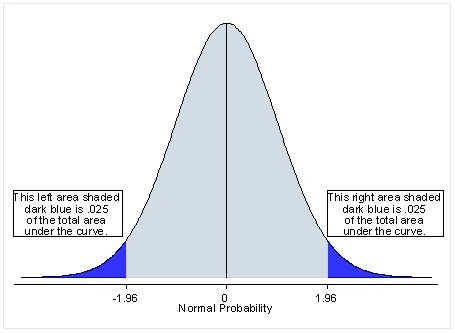
* One way ANOVA
* Two-way ANOVA

One-way ANOVA is the hypothesis test in which only one categorical variable or the single factor is taken into consideration.

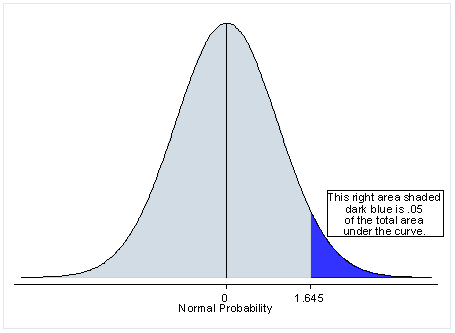
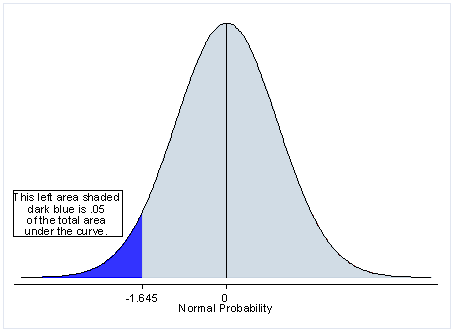
Two-ways ANOVA examines the effect of two independent factors on a dependent variable. It also studies the inter-relationship between independent variables influencing the values of the dependent variable, if any.

# One-tailed and two-tailed tests?

If you are using a significance level of 0.05, a two-tailed test allots half of your alpha to testing the statistical significance in one direction and half of your alpha to testing statistical significance in the other direction.  This means that .025 is in each tail of the distribution of your test statistic. When using a two-tailed test, regardless of the direction of the relationship you hypothesize, you are testing for the possibility of the relationship in both directions.  For example, we may wish to compare the mean of a sample to a given value *x* using a t-test.  Our null hypothesis is that the mean is equal to *x*. A two-tailed test will test both if the mean is significantly greater than *x* and if the mean significantly less than *x*. The mean is considered significantly different from *x* if the test statistic is in the top 2.5% or bottom 2.5% of its probability distribution, resulting in a p-value less than 0.05.



If you are using a significance level of .05, a one-tailed test allots all of your alpha to testing the statistical significance in the one direction of interest.  This means that .05 is in one tail of the distribution of your test statistic. When using a one-tailed test, you are testing for the possibility of the relationship in one direction and completely disregarding the possibility of a relationship in the other direction.  Example- Our null hypothesis is that the mean is equal to *x*. A one-tailed test will test either if the mean is significantly greater than *x* or if the mean is significantly less than *x*, but not both. Then, depending on the chosen tail, the mean is significantly greater than or less than *x* if the test statistic is in the top 5% of its probability distribution or bottom 5% of its probability distribution, resulting in a p-value less than 0.05.  The one-tailed test provides more power to detect an effect in one direction by not testing the effect in the other direction.



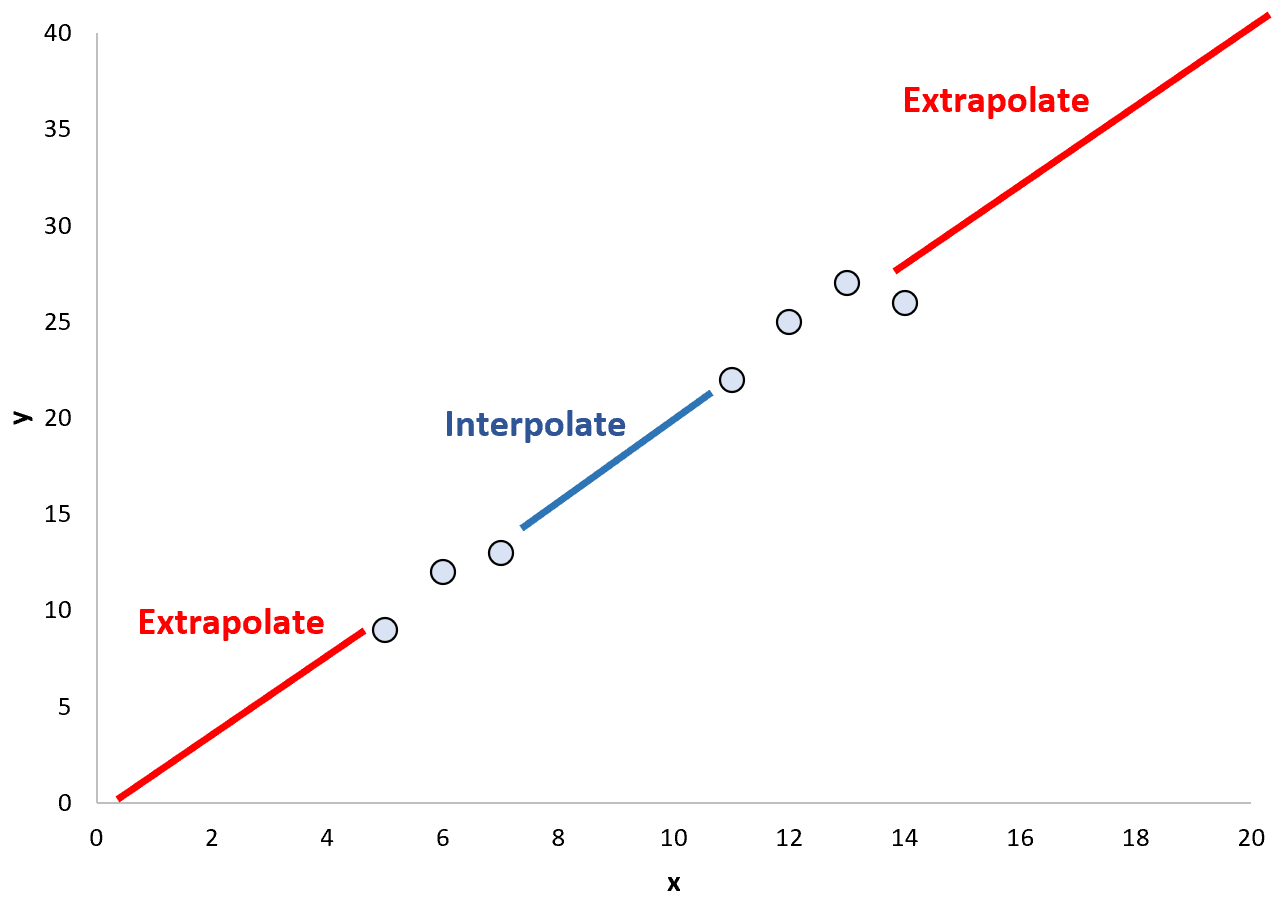
Because the one-tailed test provides more power to detect an effect, you may be tempted to use a one-tailed test whenever you have a hypothesis about the direction of an effect. Before doing so, consider the consequences of missing an effect in the other direction.  Imagine you have developed a new drug that you believe is an improvement over an existing drug.  You wish to maximize your ability to detect the improvement, so you opt for a one-tailed test. In doing so, you fail to test for the possibility that the new drug is less effective than the existing drug.  The consequences in this example are extreme, but they illustrate a danger of inappropriate use of a one-tailed test.

So, when is a one-tailed test appropriate? If you consider the consequences of missing an effect in the untested direction and conclude that they are negligible and in no way irresponsible or unethical, then you can proceed with a one-tailed test. For example, imagine again that you have developed a new drug. It is cheaper than the existing drug and, you believe, no less effective.  In testing this drug, you are only interested in testing if it less effective than the existing drug.  You do not care if it is significantly more effective.  You only wish to show that it is not less effective. In this scenario, a one-tailed test would be appropriate.

# Interpolation and Extrapolation

Interpolation is a prediction made using inputs that lie within the set of observed values. Generally, interpolations are more accurate.

Extrapolation is when a prediction is made using an input that’s outside the set of observed values.



# What is Chi-Square Statistic? Chi-square Test?

Test that measures how a model compares to actual observed data. The data used in calculating a chi-square statistic must be random, raw, mutually exclusive, drawn from independent variables, and drawn from a large enough sample. It is also called Z squared. It is also called the "goodness of fit" statistic because it measures how well the observed distribution of the data fits with the distribution that is expected if the variables are independent.

Where c = Degrees of freedom, O = Observed value, E = Expected value

There are two main kinds of chi-square tests: the test of independence, which asks a question of relationship, such as, "Is there a relationship between student gender and course choice?"; and the goodness-of-fit test, which asks something like "How well does the coin in my hand match a theoretically fair coin?" There are two kinds of chi-square tests: the **test of independence**, which asks a question of relationship, such as, “Is there a relationship between gender and SAT scores?”; and the **goodness-of-fit test**, which asks something like “If a coin is tossed 100 times, will it come up heads 50 times and tails 50 times?”

Chi-squared test, a statistical method, is used to check the "goodness of fit test".

It is used to test if a sample of data came from a population with a specific distribution. Chi-Square test include-

1. **Goodness of fit test** - Used to determine whether or not a categorical variable follows a hypothesized distribution.
2. **Test of independence** - Used to determine whether or not there is a significant association between two categorical variables.  
   Let’s suppose you have two random categorical variables X and Y. Now if you want to check if there's any association between X and Y. For example,  
   Researchers want to know if gender(X) is associated with political party(Y) preference in a certain town so they survey 500 voters and record their gender and political party preference. They can perform a Chi-Square Test of Independence to determine if there is a statistically significant association between voting preference and gender.

We can use the Chi-Square test in one of the following situations:  
- When we want to estimate observed distribution matches with the expected distribution, this is also referred to as the goodness of fit test.  
- When we want to estimate whether two random variables are associated or not.

The chi squared test is one type of hypothesis testing, so to solve any problem using the Chi-Square Test we follow the same steps as hypothesis testing.  
  
Hypothesis test steps:  
Define the Null hypothesis (H0)  
Define the alternative hypothesis (H1)  
Design the Test statistic(T)  
Take the T & H0 & H1 find the p-value  
If the p-value is > 5%(5% rule is arbitrary) then we fail to reject h0   
Else reject the H0 and accept the H1

# Law of Large Numbers

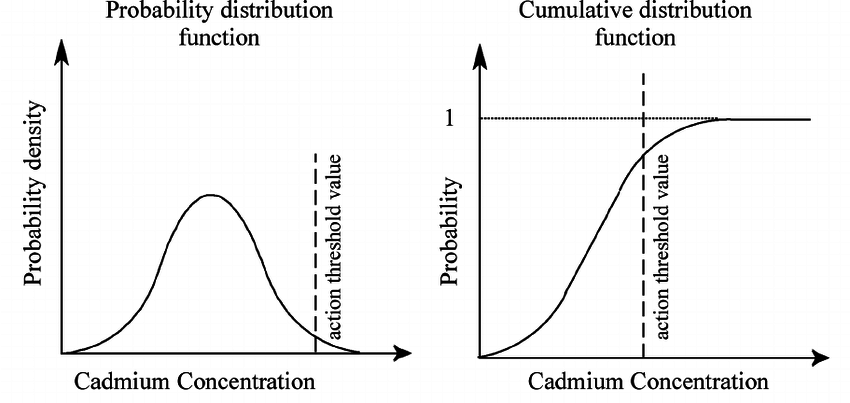
# Difference between binomial distribution and geometric distribution?

# Difference between disjoint events and independent events?

# Cumulative Distribution Functions and Probability Density Functions?

**Probability Density Function** can be defined only for **continuous random variables** and it tells us the probability of the random variable X falling within a range of values (a, b) by computing the integral of this variable's PDF over that range:

PDF is used when we want to know the probability that an observation relies on a certain range, with the restriction that the observation comes from a continuous random variable.



# Power law and Power Law Distribution?

# Probability Mass Function and Probability Density Function?

# Bernoulli Trial

# Shifted geometric distribution

# Identify if a coin is biased

We perform a hypothesis test.

According to the null hypothesis, the coin is unbiased if the probability of head flipping is 50%. Perform the below steps:

* Flip the coin 500 times
* Calculate the p-value
* Compare the p-value against the alpha

If p-value > alpha: Null hypothesis holds good and the coin is unbiased

If p-value < alpha: Null hypothesis is rejected and the coin is biased

# References

1. Intellipaat
2. Mygreatlearning
3. Analyticsindiamag
4. Indeed.com
5. Investopedia.com
6. Mlstack.cafe

HR

Tell me about yourself?

Are you willing to relocate to various parts of India?

Since the company is spread across the globe, you might have to relocate for a project or a job role. Do consider this fact before you answer this question.

“I am open to relocation if my role compulsorily demands it.”

### Q3. Where do you see yourself in 5 years?

**Ans.** This question is meant to assess if you plan to stick with the company in the long run. You must include the company in your future plans. A sample answer would be:

“I expect to excel in my current role and get promoted to a senior or a managerial position in Deloitte within five years. I look forward to contributing to the growth of the company”

### Q4. What are your expectations from this role?

**Ans.** Talk about how this role will help you with newer challenges and chances to improve your skills to use. Also, talk about how you will be able to upgrade your skill sets through this job role. A sample answer would be:

“I expect more challenging tasks and projects where I can showcase my skills. I also look forward to opportunities where I can test myself and explore myself professionally.”

### Q5. What made you want to find a new job?

**Ans.** This is a behavioural question to test if you have a critical bone. The interviewer would like to assess whether you criticise former employees or undermine the current ones. Through your answer, they can understand if you are a good fit for the organisation or not. A sample answer would be:

“I am looking for a switch to explore opportunities for professional growth. I would to learn and upgrade my skill sets through new challenges.”

### Q6. Why should we hire you Deloitte?

**Ans.** I am a hardworking person who likes to take on challenges. I am able to meet deadlines since I have good time management skills. I can handle stress most of the time which keeps my performance positively unaffected.

# What made you apply for the position at Deloitte/E&Y?

I am committed to deliver my clients with the best possible service, and if I talk about your company, your company’s market share, plus the exposure, would give me great opportunity at personal and professional level, and I could uplift my skills after joining your company. I want to upskill my leadership skills, organisational skills, I want to work as a team leader or team player. And I want to make a big impact on community in large. It is one of the big 4 auditing and consultancy firm in the world.

# What type of work environment do you prefer?

I enjoy working in a highly collaborative and energetic environment, the people in the team have good healthy competition which improves my work quality and my work efficiency, my team is helping and they have got my back. When I am in a fast-paced environment I feel highly motivated, it gets me excited to work in the next day because I am with such a good and efficient team, and hence I could achieve my goals easily.

# Where do you see yourself in 5 years? Describe your ideal job? What are your long-term goals? What do you want to achieve in your career?

I am a long run kind of guy. I believe in the future of the company. I would like to learn more data analytics, machine learning pipeline and software development cycle in the next five years. I want to become the very best at what I do, gain **hands-on practical experience** in **managing people** and I always wanted to become a technical expert in the field.

I am quite excited about this opportunity because I believe that it is a step in the right direction and would allow me to achieve my goals. The main driver in my career is professional growth and self-improvement.

# How do you feel about working under pressure?

I am someone how thrive in working under pressure. When I was in my under-grad college, I had to give 6 exams, followed my 6 quizzes in a period of 3 days, followed my externals. I could manage my studies, the tension didn’t overpower me, and I was able to motivate myself and yes, I can achieve any given thing in a given period of time. I managed my studies and co-curricular activities every day. When I am under pressure, I try to be positive and take control of the situation and try to be motivated as much as I can.

# What is ‘goodwill’?

It is an intangible asset associated with the purchase of one company by another. The value of a company’s name, brand reputation, loyal customer base, solid customer service, good employee relations, and proprietary technology represent aspects of goodwill. This value is why one company may pay a premium for another.

The value of goodwill typically arises in an acquisition of a company. The amount that the acquiring company pays for the target company that is over and above the target’s net assets at fair value usually accounts for the value of the target’s goodwill.

# What are the common problems that an analyst encounters during the analysis?

* Handling duplicates.
* Collecting the meaningful data at right time
* Handling data storage problems.
* Making data secure and dealing with compliance issue.

# Difference between P&L statement and Balance Sheet?

|  |  |
| --- | --- |
| P&L statement | Balance sheet |
| Statement of income and expenses. | Statement of assets and liabilities and capital. |
| It shows the results of operations for a period. | It shows the financial position of the company as at the data it is drawn up. |

# Geek is going to Hyderabad and wants to know if it’s raining. Geek calls 3 random friends staying at Hyderabad to ask this question. Each friend has a 2/3 chance of telling the truth and 1/3 chance of lying. All the three friends gave the response as “Yes, it is Raining”. What is the probability that it’s actually raining in Hyderabad?

8/9

As all three friends said Yes so either all of them are telling truth or all are lying.

Case 1: All telling truth 2/3\*2/3\*2/3 = 8/27

Case 2: All lying 1/3 \* 1/3 \* 1/3 = 1/27

case 1 is favourable to raining and case 2 is against it

So, probability that it is raining in Hyderabad is = 8/9

So, it is approximately 89% chance that it is raining in Hyderabad.

# What is RAROC?

Risk Adjusted Return On Capital is a risk-adjusted measure of the Return On Investment. It does this by accounting for any expected losses and income generated by capital, with the assumption that riskier projects should be accompanied by higher expected returns.

# What are your strengths and weaknesses?

Strengths:

* Goal-oriented and versatile
* Ability to handle pressure
* Strong communication skills

Weaknesses which you can address:

* Self-critical
* Multi-tasker

# How many times are the hands of a clock coincide in a day?

The hands coincide 22 times in a day.

The hands of a clock coincide 11 times in every 12 hours (Since between 11 and 1, they coincide only once, i.e., at 12 o'clock).

AM 12:00,1:05,2:11,3:16,4:22,5:27,6:33,7:38,8:44,9:49,10:55

PM 12:00,1:05,2:11,3:16,4:22,5:27,6:33,7:38,8:44,9:49,10:55

The hands overlap about every 65 minutes, not every 60 minutes.

# How many times does 1 appear between 1 and 100?

21 times

1-10 (2 times), 11-20 (10 times), 21-30 (1 time), 31-40 (1 time), 41-50 (1 time), 51-60 (1 time), 61-70 (1 time), 71-80 (1 time), 81-90 (1 time), 91-100 (2 time).

Most difficult time during your studies?

What do you look for in the people you work with?

What is the biggest challenge you have faced by far?

What’s your biggest weakness?

What are your skills?

Why this company?

What was your teamwork experience?

What kind of career do you want?

What is your dream job?

Tell me about a time you carried out a training program, what were the successes/failure?

Most difficult situation in life?

Puzzles

# **Flipping a coin**

You are in a room blindfolded. There are 10 coins placed in front of you where, 5 of them are placed heads up and 5 are placed heads down. It is not possible to determine which side is up by touching them. The task is to separate these coins into two piles of 5 such that both the piles have an equal number of heads up. You are allowed to flip the coins any number of times?

**Solution**

Step 1: Take the coins and arrange them into two piles with 5 coins each.

Step 2: Fixing one pile, flip all the coins in the other pile.

Conclusion: The number of heads in both the piles will become equal. It happens because the coins have only two probabilities, they can either have heads or a tail.

We know that are the 10 coins: H H H H H T T T T T

Now, let us consider the following cases:

Case 1: Let us consider the coins are divided in two piles in the following order

P1: H T T T T and P2: T H H H H. Now if we flip P1, then P1: T H H H H. Therefore, both the piles will have equal no. of heads.

Case 2: Let us consider the coins are divided in the following order

P1: T H T H H and P2: H T H T T. Now if we Flip P1, it becomes P1: H T H T T. Therefore, both the piles will have equal no. of heads.

What is happening here is that we are fixing the number of heads in one pile whereas in the other pile we are flipping them. The logic here is that the number of heads up and heads down coins is fixed in the beginning.

When we are dividing them into two piles, if one pile gets “x” heads up then the other pile will have “x” tails up and then we flip the other pile. It becomes “x” tails and the heads become tails.

# **Bag of coins**

You are given 10 bags, with each bag containing 1000 coins. However, one of the bags contains forgeries. All the coins are identical in every aspect except that the forged coins weigh 1.1 gram while the real coins weigh 1 gram. What is the minimum number of times you need to weigh the bags to figure out the bag with the forged coins?

**Solution**

We are allowed to weigh the bags only once to figure out the forged bags.

Let us number the bags, from bag 1 to bag 10.

Now from bag 1, we take 1 coin, and from bag 2 we take 2 coins, from bag 3 we take 3 coins, and so on. Now we take all these coins and weigh them together.

Between all the 10 bags we will have 55 coins.

1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 55

Now we know that the real coins all weigh 1 gram while the forged coins weigh 1.1 gram or have an extra 0.1 gram of weight.

Therefore, by observing the extra weight in our picked coins, we can figure out the forged bag.

Say the weight of the collective coins comes as 55.5 grams. Then we can say that there is an additional 0.5-gram weight, which is 0.1×5, therefore bag 5 from where we took 5 coins is the forged bag. Similarly, if the weight was 55.7, then it would be bag 7.

To sum it up, if the weight is 55, then the bag numbered n is the bag with the forgeries.

# **Three Glasses and 10 Coins**

You have 3 glasses and 10 coins. The objective is to distribute the ten coins amongst the three glasses such that each glass has an odd number of coins?

**Solution**

We can see that there is no way we can divide 10 into three odd parts, as there will always be one even number. Therefore, we cannot conventionally divide the coins into three odd parts.  
However, interviewers ask questions like this to check a candidate’s capacity to think out of the box. This problem for instance although we cannot solve it conventionally, we can do the following:

Let us distribute the coins in the following way 5: 3: 2, now we have odd numbers in the 1st and 2nd glass while the third glass has even. Pick up the third glass and place it inside the 2nd glass. In this way, the 2nd and third glasses essentially become one and contain 5 coins which is an odd number.

# **Burning Rope Puzzle**

There are two ropes. Each of them burns for one hour. They burn unevenly in different parts though – it means that one half can burn for 20 minutes and another half for 40 minutes. In addition, you have a box of matches. Measure 15 minutes?

**Solution**

Step1:  Ignite both the ends of the first rope and only 1 end of the second rope at the same time. As soon as your first rope is burnt completely in 30 mins; you are left with 30 min of the other rope with one end unburnt.

Step2: Now ignite both the ends of the leftover 30 min length of rope 2, it will burn in 15 mins.

# **Shopkeeper Puzzle: Profit or Loss**

A lady buys goods worth Rs 200 from a shop, whose shopkeeper is selling the goods with zero profit. The lady gives him a Rs 1000 note. The shopkeeper gets the change from the next shop, keeps Rs 200 for himself, and returns Rs 800 to the lady. Later the shopkeeper of the next shop comes with the Rs 1000 note saying “fake note” and takes his money back. How much of a loss or profit did the shopkeeper take?

**Solution**

The lady changed her Rs.1000 fake currency into goods and currency of total worth Rs.1000. Therefore, the shopkeeper changed his original Rs.1000 currency into a fake currency. Hence, the shopkeeper faces a loss which is Rs.1000.

# **Marbles Probability Puzzle**

You are a prisoner sentenced to death. The emperor offers you a last chance to live by playing a simple game. He says, “Divide these 100 marbles into these 2 bowls. You can divide them any way you like as long as you use all the marbles. Then I will blindfold you and mix the bowls around. You then can choose one bowl and remove ONE marble. If the marble is WHITE you will live, but if the marble is BLACK you will die.” How do you divide the marbles up so that you have the greatest probability of saving your life that is choosing a WHITE marble?

**Solution**

Say we put all the White marbles into JAR A and all the Black ones into JAR B then the chances for picking a white marble are as follows:

1/2 chance we pick BOWL A \* 50/50 chance we pick a White marble

1/2 chance we pick BOWL B \* 0/50 chance we pick a White marble

You would try different combinations, such as 25 of each coloured marble in a bowl or putting all white marbles in one jar and all the black in the other. You would still end up with a chance of 50%.

What if you put a single white marble in one bowl and the rest of the marble in the other bowl? This way, you are guaranteed at least a 50% chance of getting a white marble (since one marble picked at random, doesn’t leave any room for choice). Now that you have 49 white marbles left in the other bowl, you have a nearly even chance of picking a white marble (49 out of 99).

So, the maximum probability will be:

Bowl A: (1/2) \*1 = 1/2 (selecting the bowl A = 1/2, White marble from bowl A = 1/1)

Bowl B: (1/2) \*(49/99) (selecting the Bowl B = 1/2, White marble from Bowl B = 49/99)

Total probability = 74/99 (~3/4)

# **Gold Bar puzzle**

You’ve someone working for seven days and a gold bar to pay him. The gold bar is segmented into seven connected pieces. You must give them a piece of gold at the end of every day. What and where are the fewest number of cuts to the bar of gold that will allow you to pay him 1/7th each day?

**Solution**

The smallest number in which we can cut the gold bar is 2, so it can be divided into 3 parts. Now the segment in which we will divide the bar will be 1,2, and 4 segments long.

Now the process follows as:

Day 1- After he completes the work give him 1 segment of the gold bar. Today you have given him 1/7th of the whole gold bar.

Day 2- At the end of day 2 give him the 2-segment gold bar and take the 1 segment of gold bar you have given him yesterday. Now you have given him 2/7th of the whole gold bar.

Day 3- At the end of day three give him the 1 segment of gold bar. Now at the end of 3 days of working he has 3 segments of the whole gold bar. And, as per condition, we have given him 3/7th of the whole gold bar till now.

Day 4- After completing the work on day 4 give him the 4-segment gold block and the 2 and 1-segment gold block from him. Now he has 4 gold blocks after working for 4 days and you have given him 4/7th of the total gold bar.

Day 5- At the end of day 5 give him 1 segment gold bar that you have. Now he has 5 segment gold bars after working for 5 days.

Day 6- At the end of day 6 give him 2 segment gold bars and take 1 segment gold bar from him. Now he has 6 segment gold bar and you have given him 6/7th of the whole gold bar after working for 6 consecutive days.

Day 7- At the end of day 7 give him the 1 segment gold bar. Now he has the whole gold bar after working consecutively for the whole week. and you have also fulfilled the condition of giving him 1/7th of the whole gold bar per day without splitting it into 7 pieces i.e., dividing the bar into minimum possible segments.

In case you have to divide the bar into 1/5th, make two cuts (1 unit and two 2 units).

# **Three Mislabelled jars of fruits**

There are three mislabelled jars, where one Jar contains apples, one contains oranges and the third one contains the mixture of both. In how many minimum numbers of withdrawals one can label the jars correctly?

**Solution**

Suppose you pick from jar labelled as Apple and Oranges and you got Apple from it. That means that jar should be Apple as it is incorrectly labelled. So, it has to be Apple jar. Now the jar labelled Oranges has to be Mixed as it cannot be the Oranges jar as they are wrongly labelled and the jar labelled Apple has to be Oranges.

Similar scenario applies if it’s an Oranges taken out from the jar labelled as Apple and Oranges. So, you need to pick just one fruit from the jar labelled as Apple and Oranges to correctly label the jars.

# **Ant and Triangle puzzle**

Three ants are sitting at the three corners of an equilateral triangle. Each ant starts randomly picks a direction and starts to move along the edge of the triangle. What is the probability that none of the ants collide?

**Solution**

The ants can only avoid a collision if they all decide to move in the same direction (either clockwise or anti-clockwise). If the ants do not pick the same direction, there will be a collision. Each ant has the option to either move clockwise or anti-clockwise. There is a one-in-two chance that an ant decides to pick a particular direction. Using simple probability calculations, we can determine the probability of no collision.

N (No collision) = N (All ants go in a clockwise direction) + N (All ants go in an anti-clockwise direction) = 0.5×0.5×0.5 + 0.5×0.5×0.5 = 0.25

# **Crossing the Bridge Puzzle**

A bridge will collapse in 17 minutes. 4 people want to cross it before it will collapse. It is a dark night and there is only one torch between them. Only two people can cross at a time. “A” takes a minute to cross. “B” takes 2 minutes, “C” takes 5 minutes and “D” takes 10 minutes. How do they all cross before the bridge cross?

**Solution**

A & B crosses the bridge in 2 mins and A returns with the torch in 1 min, so the total time taken by now is 3 min. Time left before the bridge collapsed = 14 mins

C & D crosses the bridge in 10 mins and B returns with the torch in 2 mins, so the total time taken by now is 15 min. Time left before the bridge collapsed = 2 mins

Now, A & B were only left out and the time left before the bridge collapsed in 2 min but the maximum time taken by A & B to cross the bridge is 2 mins.

Hence A, B, C, and D four will cross the bridge within 17 mins.

# **Ratio of Boys and Girls**

In a country where everyone wants a boy, each family continues having babies until they have a boy. After some time, the proportion of boys to girls in the country is X: 1. What is the value of X?

**Solution**

No. of girls = 0\*(Probability of 0 girl) + 1\*(Probability of 1 girl) + 2\*(Probability of 2 girls) + …

Number of girls = 0\*(X\*1/2) + 1\*(X\*1/2\*1/2) + 2\*(X\*1/2\*1/2\*1/2) + …

Number of girls = 0 + X/4 + 2\*X/8 + 3\*X/16 + …

Number of girls = Xb  
(Assume that the probability of having a boy or a girl is the same).

The proportion of boys to girls is 1: 1.

# **Blind Games**

You are in a dark room where a table is kept. There are 50 coins placed on the table, out of which 10 coins are showing tails and 40 coins are showing heads. The task is to divide this set of 50 coins into 2 groups (not necessarily the same size) such that both groups have the same number of coins showing the tails?

Divide the coins into a pile of 40 and a pile of 10 at random. The heads will be divided somehow between the two piles. N in one, and 10 – N in the other, where 0 <= N <= 10.

# **Chaos in the Bus**

There is a bus with 100 labelled seats (labelled from 1 to 100). 100 people are standing in a queue. Persons are also labelled from 1 to 100. People board the bus in sequence from 1 to n. The rule is, if a person ‘i’ boards the bus, he checks if seat ‘i’ is empty. If it is empty, he sits there, else he randomly picks an empty seat and sits there. Given that 1st person picks the seat randomly, find the probability that the 100th person sits in his place i.e., 100th seat in?

**Solution**

The probability that the last person ends up in his proper seat is exactly 1/2. First, observe that the fate of the last person is determined the moment either the first or the last seat is selected. This is because the last person will either get the first seat or the last seat. Since at each choice step, the first or last is equally probable to be taken, the last person will get either the first or last with an equal probability of 1/2.

# **Coins on Round Table**

Two players (Player A & Player B) are playing a game on a circular table of finite diameter. The game is to take turns and place a coin on the table. The coin should always be put down on the surface & can’t be stacked on top of one another. A limited amount of coins can be put on the table. The player who puts the last coin on the surface wins. Suggest a strategy for Player A, so that no matter what Player B does, he will always win. Player A will always make the first move?

**Solution**

Player 1 must, place the coin in the centre of the table. Wherever Player 2 keeps their coin, place the coin on the opposite side, at the same distance from boundary. This will keep on happening. If player 2 has space to keep the coin on the table. Then player 1 will also have space to keep the coin. As he has been placing his coin always diagonally opposite. This player 2 will run out of space first.

# **Helium Balloon Puzzle**

A helium balloon is tied with a string to the floor of a car. The windows of the car are closed. When I start the car and accelerate, in which direction will the balloon go? Backward? Forward? Or will it stay put?

**Solution**

The Helium balloon will also move forward as the car accelerates. When the car is stationary, the air inside the car is at rest with respect to its surrounding. The Helium inside the Helium Balloon is also at rest w.r.t air. When the car accelerates and moves forward, due to inertia it will push the air inside the car backward. Since Helium is lighter than air, it pushes forward and hence the balloon will also move forward.

# **Pair of Socks**

In a drawer, you have an equal number of pairs of socks that are identical and are of red and blue. You can only pull out one sock at a time. But you cannot look at the sock before pulling it out. What is the maximum number of tries it will take before you have a perfect pair of socks?

**Solution**

The maximum number of tries it will take before obtaining a perfect pair is 3.

1st Try: On the first try, we will pick either a red or a blue sock. Say we pick a red sock.

2nd Try: Now on the second try, we will again pick either a red or blue. If we pick a red, then this red and the earlier form a pair and we got the perfect pair. Else we pick a blue and go for the third try.

3rd Try: Now on the third try, we will pick red or blue. But now since we have a colour for each pair already, if we pick red, we have a pair of red socks, if we pick blue, we have a pair of blue socks.

# **Monty Hall Puzzle?**

You are playing a game where the host gives you 3 boxes. One of the boxes contains an expensive diamond whereas, the other 2 are empty. You are available with the option to pick any one of the three boxes. You picked one box of the three. Now among the two boxes, one of them is sure to be empty. The host opens one of the remaining boxes and reveals it to be empty. Now he gives you a choice, where you can either open the box, you had picked initially or pick the unopened box on the table. Whichever box you choose, you can open that box. Which box will you pick? Will you stay with your initial choice or switch? Why?

**Solution**

I will pick the box on the table to open as it increases the probability of getting the diamond from 1/3 to 2/3​. There are 3 boxes, and on picking one of the three boxes, the probability of winning is 1/3​.

While the probability of the diamond not being in the selected box and being in one of the other two boxes is 2/3​. Once the empty box is open – the 2/3​ probability of the diamond being in the unselected box shifts over to the remaining box. Thus, the unopened box that was previously not selected now has a probability of 1/3​ containing the diamond.

# **Rooms toggle?**

There are 100 doors in a row, all doors are initially closed. A person walks through all door’s multiple times and toggle (if open then close, if close then open) them in the following way: In the first walk, the person toggles every door. In the second walk, the person toggles every second door, i.e., 2nd, 4th, 6th, 8th, … In the third walk, the person toggles every third door, i.e., 3rd, 6th, 9th, … Likewise, in the 100th walk, the person toggles the 100th door. Which doors are open in the end?

**Solution**

A door is toggled in an ith walk if i divide door number.

For example, door number 45 is toggled in the 1st, 3rd, 5th, 9th,15th, and 45th walks.  
The door is switched back to an initial stage for every pair of divisors.  But there are door numbers that would open, for example, in 16, the divisors are (1,2,4,8,16). Similarly, all other perfect squares 4, 9, 16, 25, 36, 49, 64, 81 and 100 would become open.

# **Three light bulbs**

There is a room with a door (closed) and three light bulbs. Outside the room, there are three switches, connected to the bulbs. You may manipulate the switches as you wish, but once you open the door you can’t change them. Identify each switch with its bulb. All bulbs are in working condition.

**Solution**

Let the bulbs be X, Y, and Z. Turn on switch X for 5 to 10 minutes. Turn it off and turn on switch Y. Open the door and touch the light bulb. The light is on from the bulb, it is Y. Now we will check other two off bulbs, the bulb which is hot, it is X, the bulb which is cold, it is Z. In case the bulbs are LED, it is impossible to find the correct answer.

# **8 balls**

You have eight balls of the same size. Sev­en of them weigh the same, and one of them weighs slightly more. How can you find the ball that is heavier by using a balance and only two attempts at weighing?

**Solution**

You can put six of the balls on the balance (3 on each side). If one of the sides is heavier you will know that the heavier ball is on that side.

If not, the heavier ball is among the two that you did not measure and it will be real­ly easy to determine precisely which ball is heavier with your second weighing.

After you determine which side is heavier, you will be left with 3 balls to choose from. You have another attempt at weighing left. You can put two of the balls on the balance and see if one of them is heavier.

If it is, then you have found the heavier ball. If it is not, then the third ball is the one that is heavier.

# **Prisoner and Cigarette Problem**

A was serving in the prison and had a bad smoking problem. He would always be smoking in his cell. One day he figured out that using 3 cigarette butts, if he combines them together it forms 1 full cigarette, so using this technique he was able to smoke more cigarettes. On a particular day, A had the urge to smoke 5 cigarettes however he did not have any pack with him and only had 10 butts. He really wanted to smoke but since he did not have enough, he wondered what he could do. He looked over at his cellmate B’s pile of cigarette butts and considered stealing; however, B always kept his butts counted and if he finds one missing, he will beat A up. Help A to figure out a plan such that he is able to smoke 5 cigarettes.

**Solution**

To be able to smoke 5 cigarettes, A can perform the following steps: First with his 10 butts A uses 9 and form 3 cigarettes. He smokes those 3 and gets 3 butts. He turns these 3 butts and smokes them as 1 cigarette. Then he is left with 2 butts, 1 from his current cigarette and the other from his first 10 butts. He then “borrows” one butt from B’s pile and makes the 5th cigarette and when he gets the butt, he puts it back in B’s pile.

# **Russian Roulette**

Aman was caught stealing and brought in front of the King for justice. He pleaded to the King that he was a poor man and to show him some mercy. The King felt bad for him but couldn’t straight away forgive him as it would set a bad example. So instead, he puts forward an alternative solution. He tells the man that he won’t hang him, but instead use a six-chambered revolver to shoot him. He would put two bullets into the chamber in successive order and spin the chamber and then take a shot. If the man does not die on the first shot, he can then choose to take another shot immediately or ask them to spin the chamber again before shooting. If he survives the second shot, he is free to go. The man agrees to this. The executioner comes and loads the revolver and takes a shot. It is a blank. Now the king asks the man whether he wants to take the next one or should he spin the chamber a second time. What should the man choose to survive?

**Solution**

He should take the next shot without spinning the revolver. The first shot was among the 4 empty chambers. The bullets were placed one after the other, therefore one empty chamber is followed by a bullet and the other three empty chambers are followed by another bullet. The probability of the next shot to be loaded is 1/4. If the man opts for spinning the chamber the combination of the bullet’s changes.

Now the probability of the next shot being loaded increases to 1/2 of 50%. While without spinning the probability is 1/4 or 25%.

# **Mistake of Bank Cashier puzzle**

You went to a bank to cash out your cheque. By mistake the bank cashier gives you- The dollar amount in cents, and Cent amount in dollars. On the way home, you spend 5 cents, and then suddenly you notice that you have twice the amount of your cheque. How much have you written on the cheque?

**Solution**

Let’s assume cent = z and dollar = 100z, no. of dollars = x and no. of cents = y

So, initially he had 100zx + zy

Now after cashier’s mistake it becomes 100zy + zx

After spending 5 cents it becomes double

So now equation is 100zy + zx -5z = 200zx + 2zy

or 199x = 98y -5

Now, the value of y cannot be greater than 100 (otherwise it becomes $) so plotting values of y for 1 to 100, we get only 1 integer value for x i.e., 31 for y = 63

Hence the answer 31 dollars and 63 cents.

# **Heaven or Hell**

You are standing before two doors. One path leads to heaven and the other one leads to hell. There are two guardians, one by each door. You know one of them always tells the truth and the other always lies, but you don’t know who is the honest one and who is the liar. You can only ask one question to one of them to find the way to heaven. What is the question?

**Solution**

The question you should ask is “If I ask the other guard about which side leads to heaven, what would he answer?”. It should be fairly easy to see that irrespective of whom you ask this question, you will always get an answer which leads to hell. So, you can choose the other path to continue your journey to heaven.

If you ask the guard who speaks the truth about which path leads to heaven, as he speaks always the truth, he would say “left”. Now that the liar, when he is asked what “the other guard (truth-teller)” would answer, he would say “right”. Similarly, if you ask the liar about which path leads to heaven, he would say “right”. As the truth-teller speaks nothing but the truth, he would say “right” when he is asked what “the other guard (liar)” would answer. So, in any case, you would end up having the path to hell as an answer. So, you can choose the other path as a way to heaven. If you end up asking the question to the truthful one, he will speak the truth and he knows that other guard is going to lie so he will show the way to the heaven.  
If you end up asking the question to the liar, he will lie about the other and the answer will be the way to the heaven.

Important links

SQL in 1 video

<https://www.youtube.com/watch?v=D_wNQR3LeeM>

Docker with project

<https://www.youtube.com/watch?v=rr9cI4u1_88>

Guesstimate

# How many iPhones users are there in India at present?

Suppose, we are considering all models of the iPhone. There are 1.39 billion Indians. Out of which, 40% of the population includes children and senior citizens. This means that children and senior citizens will be excluded which leaves us with 834 million people that can own an iPhone.

Now, out of all these people, we will consider the upper class and upper-middle class who can own an iPhone. For this, we will exclude the lower middle class, which is around 14%.

This brings the probable number to 717 million. As per the statistics, the market share of the iPhone is 3.2%. This means there can be 22 million possible iPhone users in India.

# **What number of tennis balls can fit inside a room?**

First of all, you need to know the size of the tennis ball. You can do one of the two things: ask the interviewer or assume its size.

Now, calculate the volume of the room and divide this volume by the volume of tennis balls. You need to consider that the balls are round and a regular arrangement will leave empty space due to their shape.

Suppose, the room has only 4 seats. The room may fit 5 chairs in the vertical direction and 10 chairs in the horizontal direction. It seems as if this arrangement can be repeated 10 times to fill the room. This means that the room can roughly fill 500 seats.

The total space occupied by the seat should be considered here (sp).

Here sp= (4 x 2 x 1)ft = 8 ft. This means that the room’s volume is approximately sp x number of seats = 8 x 500 = 4000 cubic ft.

The tennis ball seems to occupy 4 cubic inches of area, the number of balls = volume of room/ area occupied by balls = 1000 balls.

Since tennis balls can be packed up to 70%, hence the total number of balls is 700.

# How many balls should you take out of a bag that has red and green balls in order to get two matching balls?

Suppose that you take out a red ball and then a green ball. After two times, you will automatically get either a red or green ball which means you will own a pair of matching balls the third time.

# How many cups of tea were consumed in Delhi in a month?

We will assume that fewer people will consume tea during the weekend since these are not working days. The next number to consider is the population. There are 20 million people in the city and let us assume that 20% of youngsters do not consume tea. Out of the rest, 30% consume tea on a daily basis, 20% consume tea occasionally and 10% do not consume tea. Let us say that daily drinkers could be having three cups of tea in a day and occasional drinkers consume tea twice a week.

Then, the total number of cups of tea consumed will be:

Daily drinkers – 3 x 0.2 x 7 = 4.2

Occasional drinkers – 1 x 0.2 x 1 = 0.2

Non-drinkers= 0

Total= Daily + Occasionally + Non-drinkers = 4.4 cups in a day

Per month = 4 x 4.4 x 1.4 crore = 24.64 crore cups.

# How much paint will be required for painting a 20 m x 20 m wall?

Let us estimate the amount of paint required for every square meter. Now, we will find the area to be painted. The wall to be painted will have the main area as 20 m x 20 m which is 400 square meters. Let us assume that the depth is 1 mm.

We will also consider that the oil in half of the paint has dried after a few hours of the paint application on the wall. Let us consider the width of the paint to be considered as 2 mm.

Thus, the volume to be painted is 400 square meters x 0.002 meters = 0.8 meters cube of paint is required.

# How many people live in your housing society?

Suppose that society has only apartment-style of flats. Recall the number of apartments is numbered from 1 to 40 which means that there are 40 apartments. Each apartment block has 9 floors with 3 apartments. So, 9x3x40 = 1080 flats.

On average, Indian households have 5 members. Some apartments may be unoccupied. If assuming that 10% of apartments are unoccupied. So, the number of people living in the society will be 1080\*0.9\*5 =4860.

# How many gallons of white house paint are sold in the US every year?

Assum­ing that there are 300 million people in the US and the average household contains 2.5 peo­ple then we can conclude that there are 120 million homes in the US.

* *Number of houses:* Many people live in apart­ments and other types of buildings different than houses. Let’s assume that the percentage of people living in houses is 50%. Hence, there are 60 million houses.
* *Houses that are painted in white:* Although white is the most popular colour, many people choose dif­ferent paint colours for their houses or do not need to paint them (using other types of techniques in order to cover the external surface of the house). Let’s hy­pothesize that 30% of all houses are painted in white, which makes 18 million houses that are painted in white.
* *Repainting:* People need to repaint their houses after a given amount of years. For the purposes of this exercise, let’s hypothesize that people repaint their houses once every 9 years, which means that every year 2 million houses are repainted in white.

I have never painted a house, but let’s assume that in order to repaint a house you need 30 gallons of white paint. This means the total US market for white house paint is 60 million gallons.

# References

1. Naukri.com

POWER BI

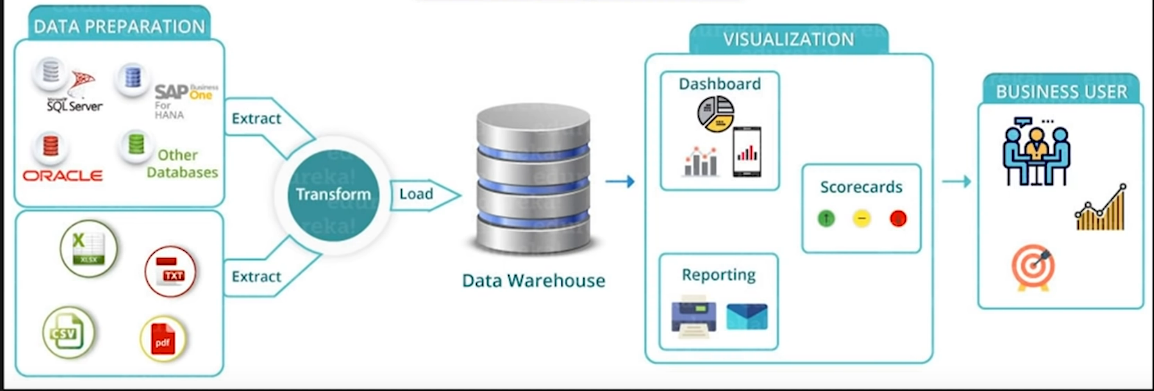
# Business Intelligence and Data Visualization

# What is Power BI?

Power BI is a business analytics service provided by Microsoft that provides interactive visualizations with self-service business intelligence capabilities. Benefits of Power BI

1. Pre-built dashboards and reports for popular SAAS solutions
2. Real-time dashboard updates
3. Secure, live connection to your data sources on-premises and in the cloud
4. Intuitive data exploration using natural language query
5. Integrated with familiar Microsoft products to utilize commitment for scale.
6. Fast deployment

Architecture of Power BI



Components of Power BI

# Building Blocks of Power BI

Machine Learning and Deep Learning

# Data Science

An interdisciplinary field that includes various scientific processes, algorithms, tools, and machine learning techniques to find common patterns and gather sensible insights from the given raw input data using statistical and mathematical analysis. Statisticians work a posteriori (revised or updated probability of an event occurring after taking into consideration new information), explaining the results and designing a plan, data scientists use historical data to make predictions.

Univariate data – Contains one variable.

Bivariate data – Contains two different variables.

Multivariate data – Contains three or more variables.

# Artificial Intelligence

Refers to the **simulation** of human intelligence in machines that are programmed to think like humans and **mimic** their actions. A subset of AI is machine learning (ML). Deep Learning techniques enable this automatic learning through the absorption of huge amounts of unstructured data such as text, images, or video.

# Machine Learning

ML is a field of computer science that gives computers the **ability** to learn without being **explicitly** programmed. ML can be categorized in the following categories

* Supervised ML – task oriented
* Semi-supervised ML
* Unsupervised ML
* Reinforcement learning (Method based on rewarding desired behaviours and/or punishing undesired ones)

Reason for popularity of DL in recent times

* Increase in the amount of **data** generated through various sources
* Growth in **hardware resources** required to run these models.
* **GPUs** are multiple times faster and they help us build bigger and deeper DL models in comparatively less time than required previously.

Father of ML is **Geoffrey Everest Hinton**.

# What is Semi-Supervised ML?

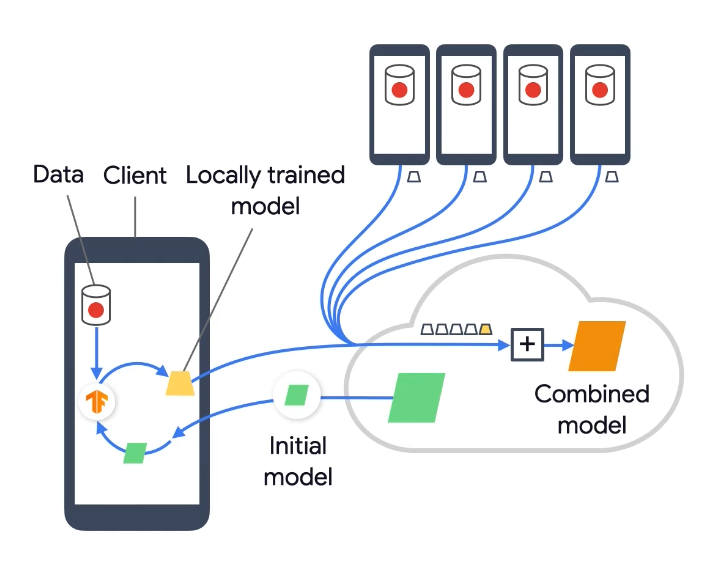
# What is Find-S algorithm?

A **basic concept learning algorithm** in ML. It finds the most specific hypothesis that fits all the positive examples. The find-S algorithm starts with the most specific hypothesis and generalizes this hypothesis each time it fails to classify an observed positive training data. Hence, the Find-S algorithm moves from the most specific hypothesis to the most general hypothesis.

# What is Genetic Algorithm?

# What is federated learning?

Federated learning is a way to train AI models **without anyone seeing or touching your data**, offering a way to unlock information to feed new AI applications. New AI models are being trained collaboratively on the edge, on data that never leave your mobile phone, laptop, or private server. This new form of AI training is called federated learning, and it’s becoming the standard for meeting a raft of **new regulations for handling and storing private data**. By processing data at their source, federated learning also offers a way to tap the **raw data streaming from sensors** on satellites, bridges, machines, and a growing number of smart devices at home and on our bodies.



# What is GraphML?

# What is backpropagation?

# Clustering

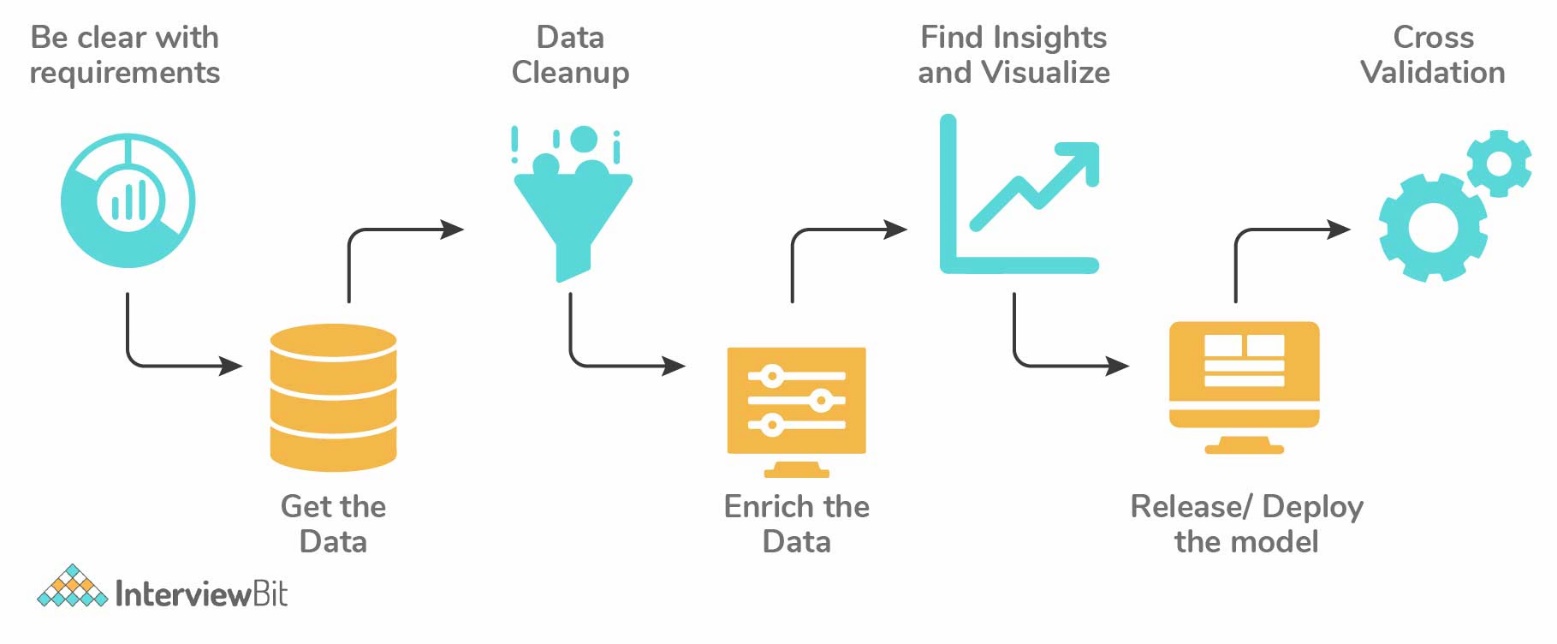
# Dimensionality reduction

# Supervised and Unsupervised Learning

|  |  |
| --- | --- |
| **Supervised** | **Unsupervised** |
| Requires labelled data. | Works on unlabelled data. |
| Used to create models that can be employed to predict or classify things. | Used to extract meaningful information out of large volumes of data. |
| E.g., Linear Regression, Decision Tree, logistic regression, SVM, Naïve Bayes, K-nearest neighbour, Neural networks, AdaBoost, Random Forest, Gradient Boosting, XgBoost. | E.g., K-means clustering, Apriori algorithm, hierarchical clustering, Anomaly Detection, Dimensionality Reduction, DBScan, PCA etc. |
| Has a feedback mechanism. | No feedback mechanism. |
| Used in Classification and Regression. | Dimension reduction. |

# Approach for Data Analytics Project

* Understand the business requirement/problem
* Explore the given data and analyse it carefully, in case of missing data, get the requirements clarified from the business
* Run the model, build meaningful visualization and analyse the results to get meaningful insights
* Release model implementation, and track the results and performance over a specified period to analyse the usefulness

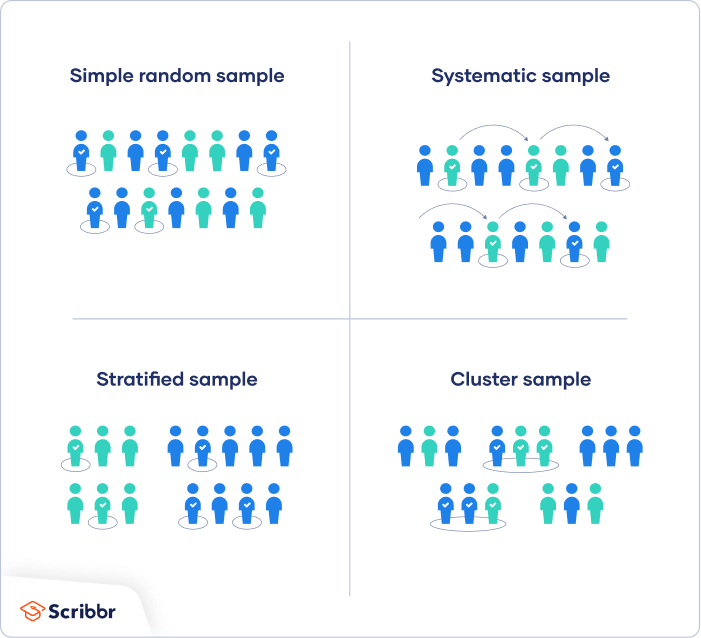


# Data Sampling

Data analysis cannot be done on the whole volume of data when it involves larger datasets. We take some data samples that can be used for representing the whole population and then perform analysis on it. We should be careful to take sample data out of the huge dataset so that it truly represents the entire dataset. It is one of the most important factors which decides the accuracy of a result.

Mainly there are two types of sampling techniques:

|  |  |
| --- | --- |
| **Probability Sampling** | **Non probability sampling** |
| It involves random selection which gives chance to every element to be selected. | It follows non-random selection which means the selection is done based on your ease or any other required criteria. |
| Simple Random Sampling, Stratified Sampling, Systematic Sampling, Cluster Sampling, Multi stage sampling | Convenience sampling, purposive sampling, Quota sampling, Snowball Sampling |



**Stratified Sampling**: Population is split into sub-populations. It allows you to conclude more precise results by ensuring that every sub-population is represented in the sample.

**Systematic Sampling**: Elements are chosen from an ordered sampling frame. The list is advanced in a circular fashion so once you reach the end of the list, it is progressed from top again.

**Cluster Sampling**: Each of the sampling units is a collection of clusters of elements. It involves dividing the sample population into separate groups, called clusters. Analysis is conducted on data from the sampled clusters.

**Convenience Sampling**: Method where data is collected from an easily accessible group.

**Purposive Sampling/Judgemental sampling** - is where the researchers use their expertise to select a sample that is useful or relevant to the purpose of the research.

**Snowball Sampling**: Used where the **population is difficult to access**. It can be used to recruit individuals via other individuals.

# Data resampling

Method to sample data for improving accuracy and quantify the uncertainty of population parameters.

* It ensures the model is good enough by training the model on different patterns of a dataset to ensure variations are handled
* It is done in cases where models need to be validated using random subsets or when substituting labels on data points while performing tests

# Parametric & non-parametric models

**Parametric models** have a **finite number of parameters**. To predict new data, you only need to know the parameters of the model. Examples include **linear regression**, **logistic regression**, and linear SVMs.

Non parametric models have an **unbounded number of parameters**, allowing for more flexibility. To predict new data, you need to know the parameters of the model and the state of the data that has been observed. Examples include **DT, KNN**.

# What is incremental learning? What is partial\_fit()?

For ML tasks where a **new batch of data** comes with time, **re-training** the model with the previous and new batch of data is a **computationally expensive** process. So instead of re-training the model with the entire set of data, use an incremental learning approach, where the **past learning** of the model will be restored and the same model will be trained with the new batch of data.

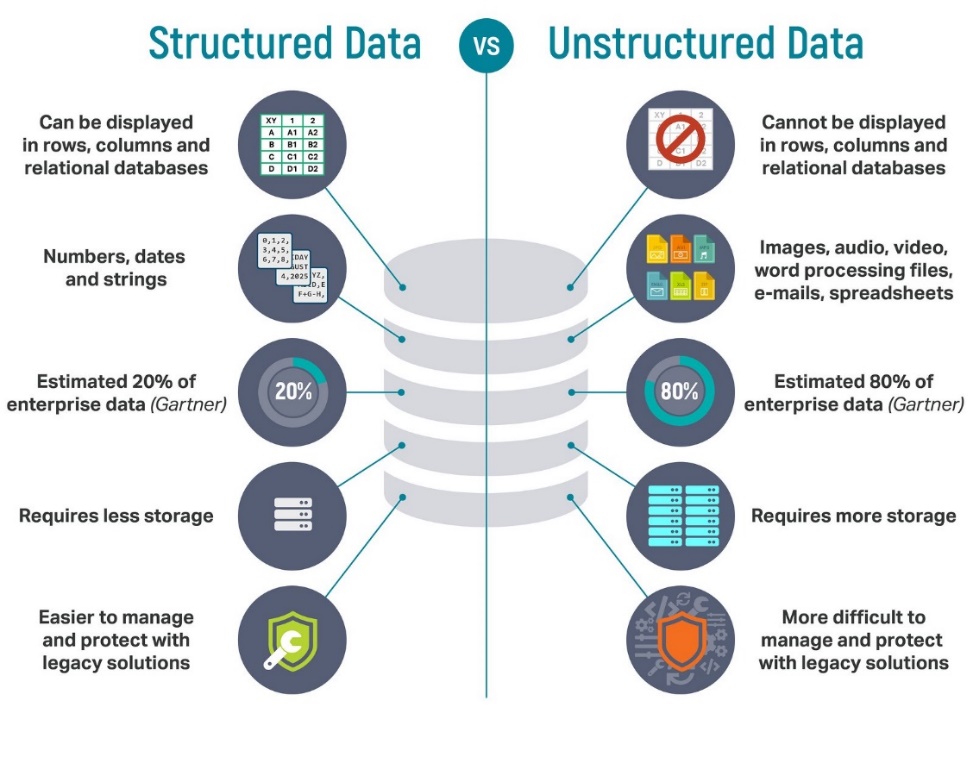
Also, for large datasets, where it’s not possible to load the entire data into the RAM at once, one can load the data in chunks and fit the training model for each chunk of data.

**partial\_fit**: For **incremental learning**, Scikit-learn has partial\_fit API, which has the ability to learn incrementally from the **batch of instances**. It is useful when the whole dataset is too big to fit in memory at once. This method is expected to be called several times consecutively on different chunks of a dataset so as to implement **out-of-core learning**. partial\_fit has some numerical stability and performance overhead, so it’s recommended to call partial\_fit function on a considerable large batch of data (that fits into the memory), to overcome the limitation of overhead.

# Structured and Unstructured Data

Structured data is data that has been **predefined** and **formatted** to a set structure before being placed in data storage, which is often referred to as **schema-on-write**. Example, **relational** **database**: the data has been formatted into precisely defined fields, such as credit card numbers or address, in order to be easily queried with SQL.

Unstructured data is data stored in its native format and not processed until it is used, which is known as **schema-on-read**. It comes in a myriad of file formats, including email, social media posts, presentations, chats, IoT sensor data, and satellite imagery.



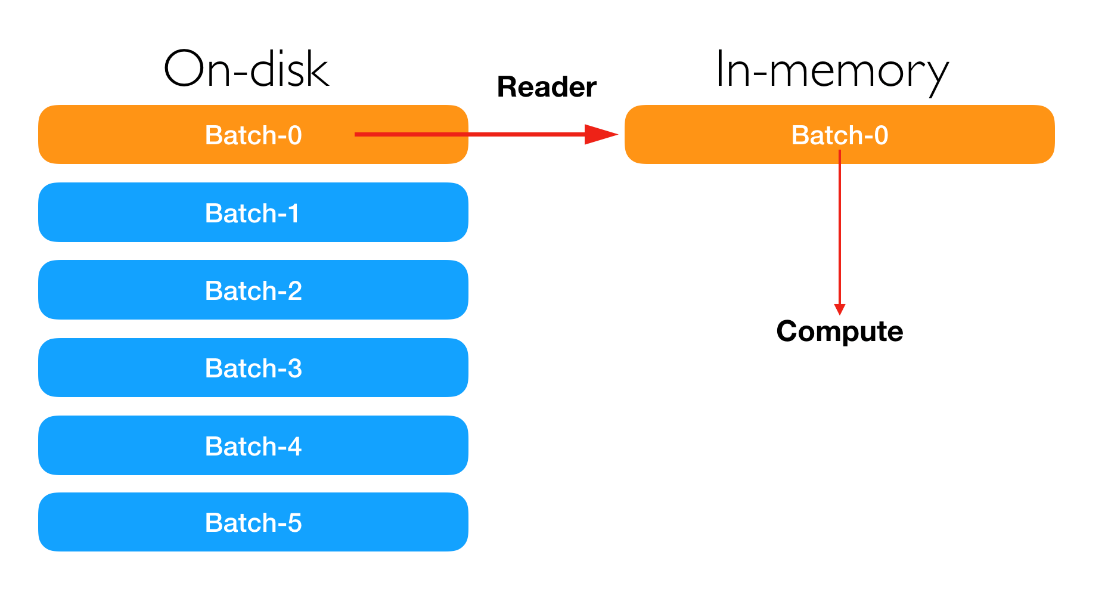
# Unreasonable Effectiveness of Data?

# What is No Free Lunch Theorem?

# Out of Core Learning

ML algorithms working with data cannot fit into the memory of a single machine, but that can easily fit into some data storage such as local hard disk or web repository. Your available RAM, the core memory on your single machine, may indeed range from a few gigabytes (sometimes 2 GB, more commonly 4 GB) up to 256 GB on large server machines. Large servers like cloud computing services such as Amazon Elastic Compute Cloud (EC2), whereas your storage capabilities can easily exceed terabytes of capacity using just an external drive.

Ability to **learn incrementally** from a mini-batch of instances is key to out-of-core learning as it guarantees that at any given time there will be only a small amount of data in the main memory. Choose a good size for the mini-batch that balances relevancy and memory footprint could involve some tuning.



# K-Nearest Neighbours (KNN)

# Naïve Bayes Classification and Gaussian Naïve Bayes

# Bias

Bias is an **error** introduced in the model because of **oversimplification** of a ML algorithm. Types of biases include

* Selection Bias
* Under coverage Bias - bias that occurs when some members of the population are inadequately represented in the sample
* Survivorship Bias - it is the logical error of focusing on aspects that support surviving a process and casually overlooking those that did not because of their lack of prominence. This can lead to wrong conclusions in numerous ways.
* Recall Bias
* Exclusion Bias
* Observer Bias

Some of the widely used low and high bias ML algorithms are

**Low bias** – **Decision tress**, **Support Vector Machines**, k-Nearest Neighbours, etc.

**High bias** – **Linear Regression**, **Logistic Regression**, Linear Discriminant Analysis, etc.

# Selection Bias

Also called as Selection effect, it denotes a situation when selected individuals or a group within a study differ in a manner from the population of interest that they give systematic error in the outcome. Selection Bias leads to skewing results, it can lead to false insights about a particular population group in a study.

* Sampling Bias: In a random population, some members of the population have fewer chances of getting included than others, resulting in a biased sample.
* Time interval bias: Trials may be stopped early if we reach any extreme value but if all variables are of similar invariance, the variables with the highest variance have a higher chance of achieving the extreme value.
* Data Bias: It is when specific data is selected arbitrarily and the generally agreed criteria is not followed.
* Attrition: The loss of the participants which did not complete the trial
* Observer selection: It is a kind of discrepancy or detection bias

When there is no randomization achieved while picking a part of the dataset for analysis. Selection Bias says that the sample analysed does not represent the whole population meant to be analysed.

# Variance

# Bias-Variance Trade-off

# Overfitting and Underfitting

# Long and Wide format data

|  |  |
| --- | --- |
| **Long format Data** | **Wide-Format Data** |
| Each row of the data represents the **one-time information** of a subject. Each subject would have its data in different/ multiple rows. | The **repeated responses** of a subject are part of separate columns. |
| Data is recognized by considering rows as groups. | Data is recognized by considering columns as groups. |
| Used in R analyses and to write into log files after each trial. | Used in stats packages for repeated measures ANOVAs. |



# Imbalanced Data

Data is unequally distributed across different categories leading to error in model performance and inaccuracy in results.Techniques to balance imbalanced data include, increasing the sample number of minority classes. The number of samples can be decreased for majority classes. Following are the approaches followed to balance data

* Perform **K-fold cross-validation** correctly while using over-sampling. The cross-validation should be done before over-sampling because if it is done later, then it would be like overfitting the model to get a specific result.
* Training set resampling: It is also possible to balance data by working on getting different datasets and this can be achieved by resampling.
  + 1. **Under-sampling**: This balances the data by reducing the size of the abundant class and is used when the data quantity is sufficient. By performing this, a new dataset that is balanced can be retrieved and this can be used for further modelling.
    2. **Over-sampling**: This is used when data quantity is not sufficient. This method balances the dataset by trying to increase the samples size. Instead of getting rid of extra samples, new samples are generated and introduced by employing the methods of repetition, bootstrapping, etc.
* Use the right **evaluation metrics**
  + 1. Specificity / Precision, Sensitivity, F1 score, AUC (Area Under the Curve)
    2. MCC (Matthews Correlation Coefficient): It represents the correlation coefficient between observed and predicted binary classifications.

# KPI, Lift, Robustness

**Key Performance Indicator**, measures how well the business achieves its **objectives**. An example of KPI in an organization is the **expense ratio**.

**Lift** indicates **how good** the model is at prediction versus if there was no model.

**Robustness** represents the **system’s capability** to handle differences and **variances** effectively.

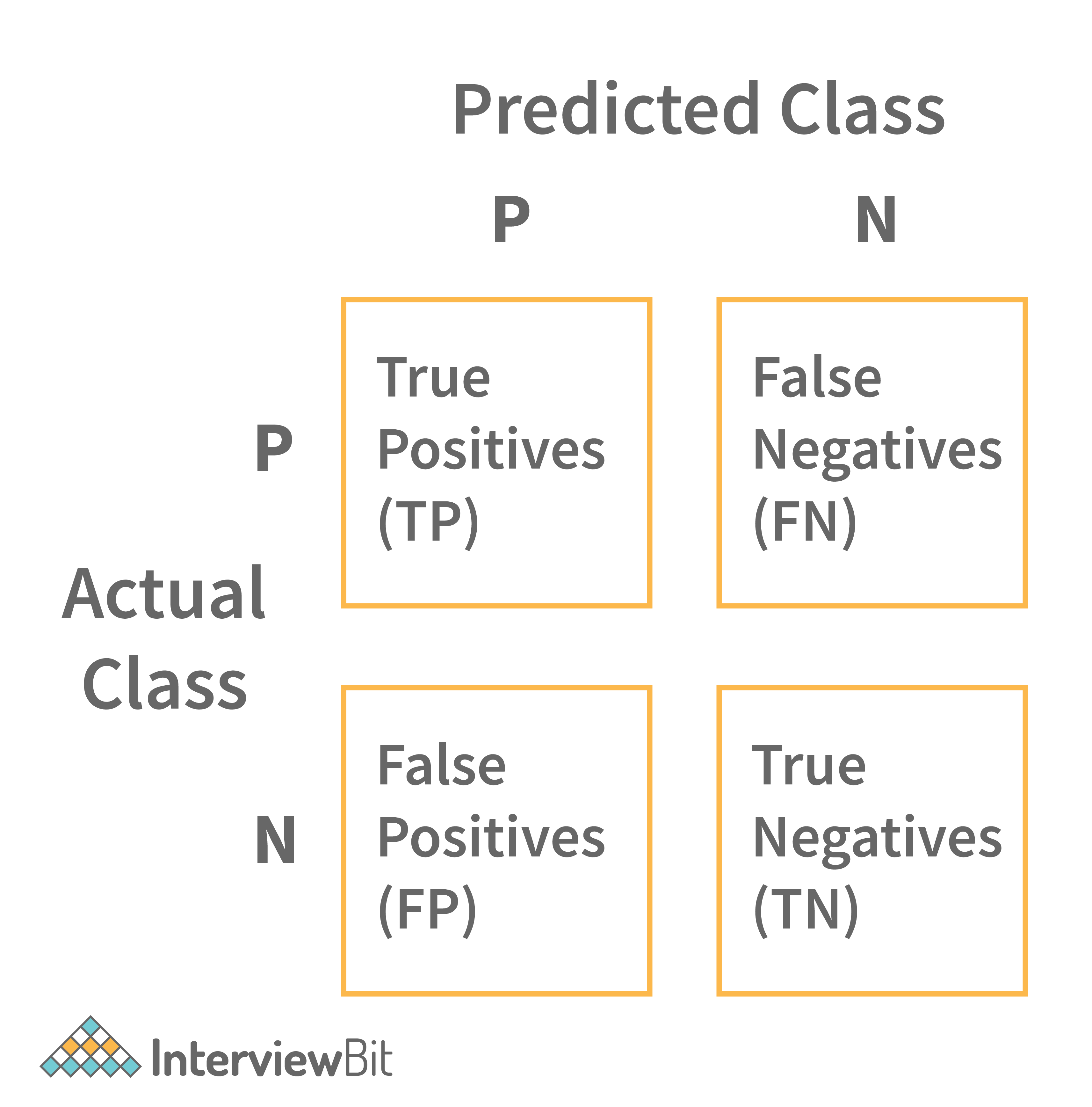
# Eigenvectors and Eigenvalues

# What is PCA?

# Regression tree?

A regression tree refers to an algorithm where the target variable and the algorithm is used to predict its value. As an example of a regression type problem, you may want to predict the selling prices of a residential house, which is a continuous dependent variable. Regression tress model nonlinear data.

# Confusion matrix

**Confusion matrix** -

# False Positive and False Negative Importance Case

* When both FP and FN are equally important

In **Banking** fields: Lending loans are the main sources of income to the banks. But if the repayment rate isn’t good, then there is a risk of huge losses instead of any profits. So, giving out loans to customers is a gamble as banks can’t risk losing good customers but at the same time, they can’t afford to acquire bad customers. This case is a classic example of **equal importance** in FP and FN scenarios.

* When a FP is important than a FN

**Medical** field, assume you have to give chemotherapy to patients. Assume a patient comes to that hospital and he is tested positive for cancer, based on the lab prediction but he actually doesn’t have cancer. This is a case of FP. Here it is of utmost danger to start chemotherapy on this patient when he actually does not have cancer. In the absence of cancerous cell, chemotherapy will do certain damage to his normal healthy cells and might lead to severe diseases, even cancer.

# R-Squared and Adjusted R-squared

# Deep Learning

Multiple layers of processing to extract high level features from the data.

* Image recognition
* Object detection
* Natural Language Processing

Deep Learning Frameworks

* Caffe
* Keras
* TensorFlow
* PyTorch
* Chainer
* Microsoft Cognitive Toolkit

**Neural Networks** - Networks that learn the patterns from the data and predicts the output for new data.

# What are Affine Transformations?

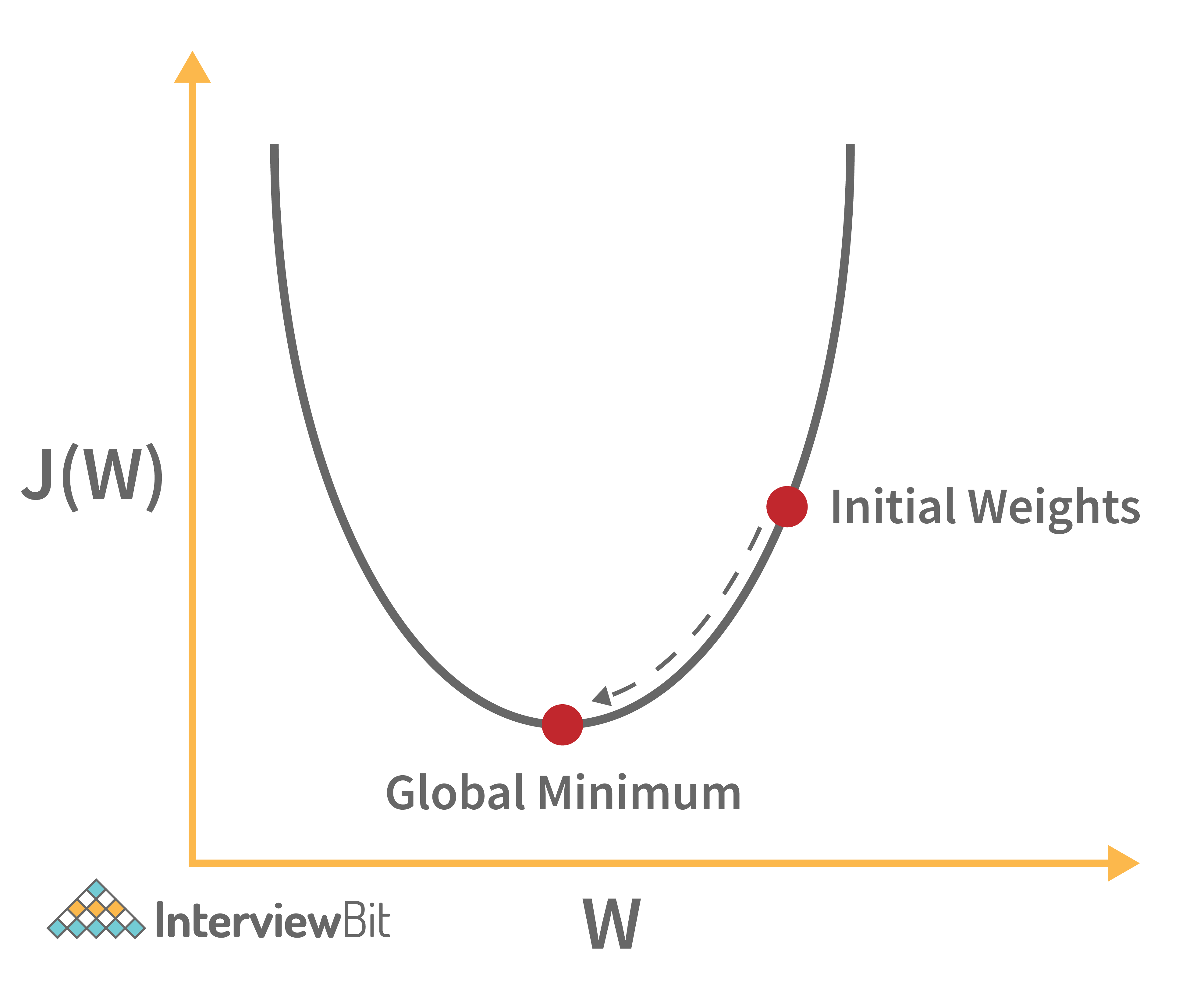
# Lazy Learning Algorithms

# Instance based learning?

# Gradient and Gradient Descent? Weights initialization?

**Gradient** - Measure of change in output with little change in input. Measure of change in weights with respect to change in error. Mathematically represented as the slope of a function.

**Gradient Descent** - Minimization function that minimizes the activation function. Gradient descent methods may not converge in some cases, as they reach a local minimum or a local optimum point in deep learning. Gradient Descent is governed by the data and the starting conditions.



Weight initialisation - We can either initialize the weights to **zero** or **assign them randomly**. Initializing the weights randomly gives better accuracy to the model since every neuron performs different computation. In zero initialisation the neural network stops learning as all the neurons have same weights. Zero weight initializes the weights with 0 integer and when used with any activation function such as sigmoid the derivative of the weights is 0.

ReLU – He normal or uniform

Sigmoid – Xavier or GLorot

Tanh - uniform

# Exploding Gradients and Vanishing Gradient

# Batch Learning?

# Data mining, Data Warehouse, Data Modelling, Database Design?

# Data Privacy in ML?

# When to use MAE and MSE?

MSE is continuous and differentiable at all points, and for gradient descent algorithm to work, the loss function must be differentiable at all points. Whereas in MAE, the mathematical expression is , which is not differentiable at all points. Hence, we choose MSE as our loss function.

When we have outliers, we use MAE, as MSE will give a very large amount of errors for outliers.

When we don’t have outliers, we use MSE. MSE gives an optimum amount of error due to which the loss gets reduced much faster as compared to MAE.

MSE is more sensitive to outliers in our predictions so it diminishes the effect of smaller errors and one large error could mislead us into thinking that the whole of our model is not working as it is supposed to be. Also, the error value it gives is not on the same scale as the data points.

# Decide number of hidden layers and nodes?

Choosing Hidden Layers

* If the data is linearly separable then you don’t need any hidden layers at all.
* If the data is less complex and is having fewer dimensions or features then neural networks with 1 to 2 hidden layers would work.
* If the data is having large dimensions of features, then to get an optimum solution, 3 to 5 hidden layers can be used.

Increasing hidden layers would also increase the complexity of the model and choosing hidden layers such as 8, 9 or more may sometimes lead to overfitting.

Choosing Nodes in Hidden Layers

* The number of hidden neurons should be between the size of the input layer and the output layer.
* Most appropriate number of hidden neurons is
* The number of hidden neurons should keep on decreasing in subsequent layers to get more and more close pattern and feature extraction and to identify the target class.

# Batch Gradient and Stochastic Gradient Descent

# SVM Classifier

# Computational Graph

# Cross-Validation

# Star Schema

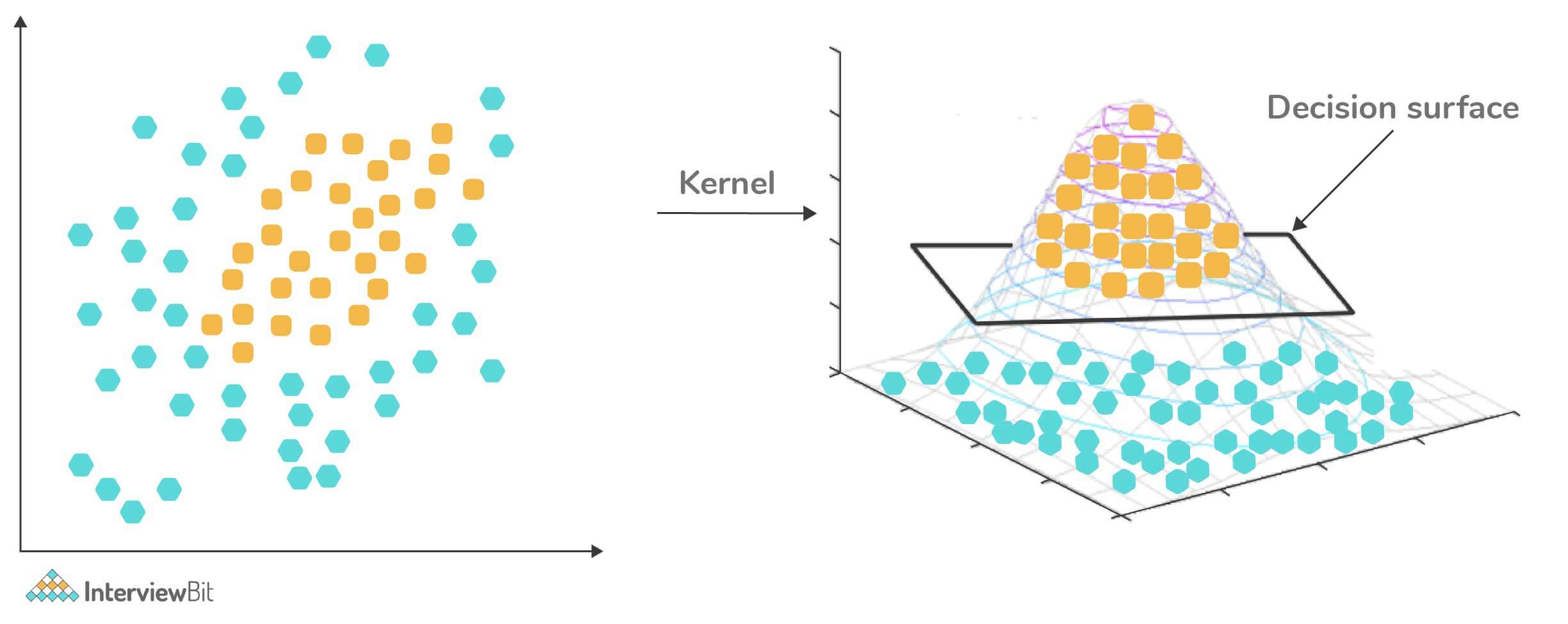
It is a traditional database schema with a central table. Satellite tables map IDs to physical names or descriptions and can be connected to the central fact table using the ID fields; these tables are known as lookup tables and are principally useful in real-time applications, as they save a lot of memory. Sometimes star schemas involve several layers of summarization to recover information faster.

# Handle Missing values

* First, we have to identify the variables with the missing values. If a pattern is identified, it could lead to interesting and meaningful insights. If there are no patterns, substitute the missing values with the median or mean values or we can simply ignore the missing values.
* If the missing value belong to categorical variables, then they are assigned default values such as mean, minimum, and maximum or median. The missing value is assigned the default value. If the data is normally distributed, we give the mean value.
* If 80% of the values are missing for a particular variable, then we would drop the variable instead of treating the missing values.

# Kernel Trick

Kernel functions are generalized dot product functions used for the computing dot product of vectors ‘xx’ and ‘yy’ in high dimensional feature space. Kernel trick is used for solving a non-linear problem by using a linear classifier by transforming linearly inseparable data into separable ones in higher dimensions.



# Treating Outlier in ML

* **Drop outlier** if it is a garbage value or in case of extreme values.
* Try a different model. Data detected as outlier by linear models can be fit by nonlinear models.
* **Normalizing** the data. Extreme data points are pulled to a similar range.
* Use algorithms that are less affected by outliers, e.g., **random forests**.
* Anomaly detection.

# Recommender system

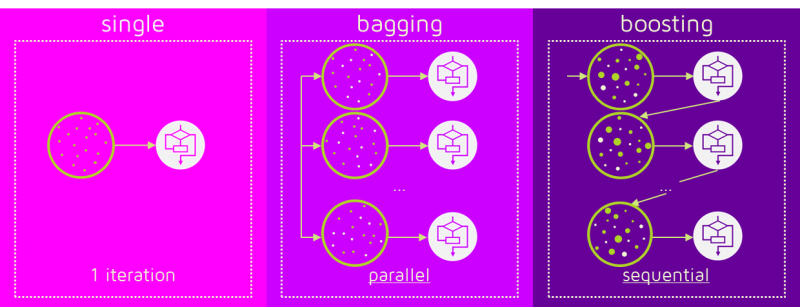
# Logistic Regression

# Decision Tree Algorithm

# Entropy in ML?

# Random Forest

# Ensemble Learning, Stacking, Bagging and Boosting



# Why is it Logistic Regression and not Logistic Classification?

Logistic Regression is closely related to linear regression, it is a generalized linear model. Logistic Regression just uses a sigmoid function and a threshold value to find the probability.

# How to handle multicollinearity?

# Feature Vectors

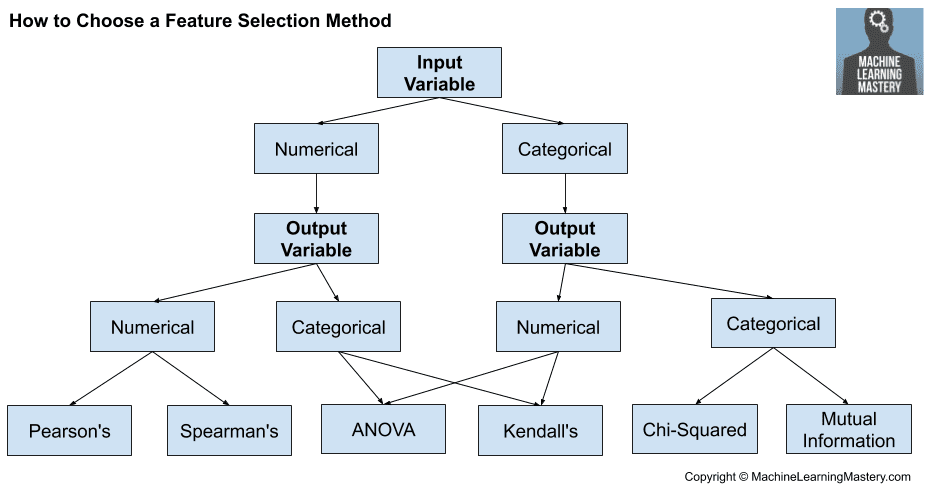
# Feature Selection methods

There are two main methods for feature selection, i.e., **filter and wrapper** methods

**Filter methods** - Linear Discrimination Analysis, ANOVA, Chi-Square, Backward Elimination – Lasso Regression, recursive Feature Elimination, Pearson’s Correlation – Variables (both dependent and independent) used for Pearson’s Correlation Coefficient must be quantitative. It will only test for the linear relationship between two variables.

**Wrapper methods**

* 1. Forward selection: We test one feature at a time and keep adding them until we get a good fit
  2. Backward selection: We test all the features and start removing them to see what works better
  3. Recursive feature selection: Recursively looks through all the different features and how they pair together

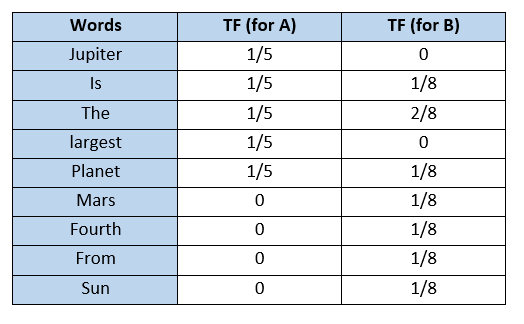
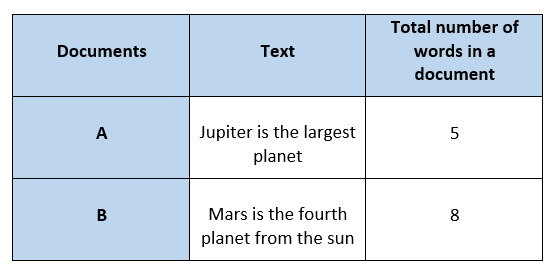


# A/B Testing

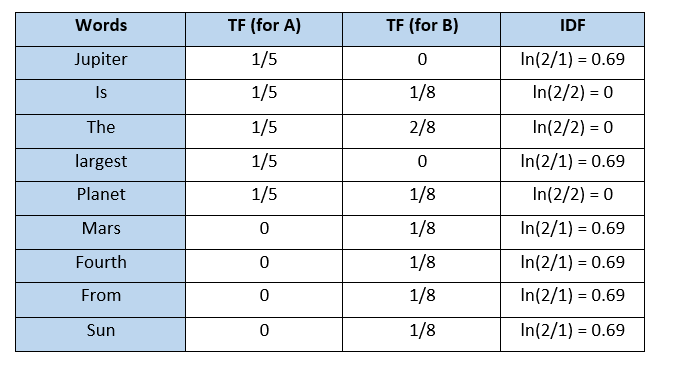
# Markov Chains

# TF/IDF Vectorization

Term Frequency-Inverse Document Frequency. It is a numerical measure that allows us to determine how important a word is to document in a collection of documents called a corpus. TF/IDF is often used in text mining and information retrieval.

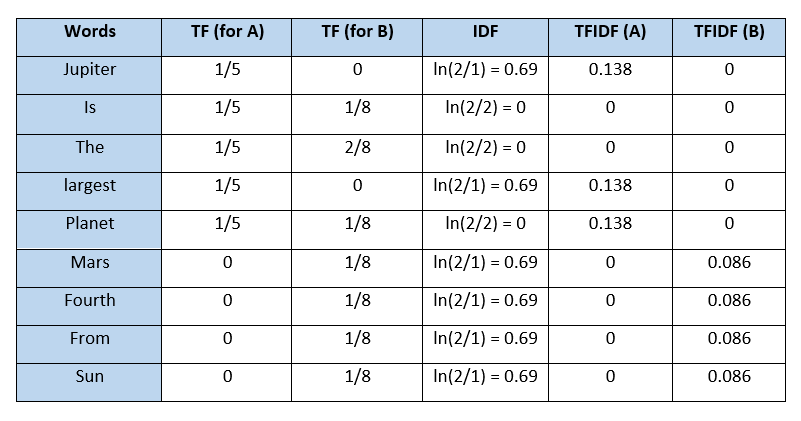


Inverse Document Frequency (IDF) is the measure of the importance of a word. IDF provides weightage to each word based on its frequency in the corpus D. IDF of a word (w) is defined as



The TF-IDF value increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general.

TFIDF gives more weightage to the word that is rare in the corpus (all the documents). TFIDF provides more importance to the word that is more frequent in the document.



One problem with TFIDF is that it is unable to capture the semantics. For example, funny and humorous are synonyms, but TFIDF does not capture that. Moreover, TFIDF can be computationally expensive if the vocabulary is vast.

# Cost function and Loss function

# ROC Curve

# Test set and Validation set

* Validation set is part of training set used to select parameters for avoiding model overfitting.
* Test set is used to evaluate the performance of the trained model.

# Regularization

# Hyperparameters and Parameters

* **Train-test split ratio**
* **Learning rate** - Learning rate **controls** how much we should **adjust the weights** concerning the loss gradient. Learning rates are randomly initialised. When the learning rate is too low, training of the model will progress very slowly as we are making minimal updates to the weights. It will take many updates before reaching the minimum point. If the learning rate is too high, this causes undesirable divergent behaviour to the loss function due to drastic updates in weights. It may fail to converge or even diverge.
* **Optimiser** - Deep learning is an iterative process. This is the key to **increase the speed and efficiency** of training a machine learning model. Example stochastic gradient descent, min-batch gradient descent, gradient descent with momentum and the Adam optimiser. Adam optimiser is the best one. Given an algorithm *f(x),* it helps in either minimisation or maximisation of the value of *f(x).* We use optimisation algorithms to train the neural network by optimising the cost function.
* **Activation function** - It introduces **non-linearity** in the neural network. This is done to help the leaning process. Decides whether a neuron should be activated or not. It will decide whether the neuron’s input to the network is important or not in the process of prediction.
* Choice of loss function
* **Dropout** - It is used for dropping out hidden and visible units of a network on a random basis. They prevent overfitting.
* **Number of iterations** (**Epochs**) - Epoch represents one iteration over the entire dataset.
* **Kernel or filter size**
* **Pooling size** - A method that is used to reduce the spatial dimensions of a CNN. It down samples operations for reducing dimensionality and creating pooled feature maps. Pooling in CNN helps in sliding the filter matrix over the input matrix.
* **Batch size** - **Batch** is a series of broken-down collections of the dataset, which help pass the information into the system. It is used when the developer cannot pass the entire dataset into the neural network at once.

Parameters are internal to the model. They are learned or estimated purely from the data during training as the algorithm used tries to learn the mapping between the input features and the labels or targets.

* Weights of linear and logistic regression models.
* Weights and biases of NN
* Cluster centroids in clustering

# Batch Normalization

# Online Machine Learning

Traditional machine learning techniques run in batch mode. For example, supervised learning tasks where the complete training data is fed in advance to train a model by applying certain algorithms. Such an approach requires entire training data available prior to the learning task and the process is also in offline mode due to expensive training costs.

Conventional techniques drawbacks include low efficiency in both time and space; poor scalability for large-scale applications because the model often has to retrain from scratch for new data.

Online learning is a combination of different techniques of ML where data arrives in sequential order and the learner (algorithm/model) aims to learn and update the best predictor for future data at every step. Models can be updated instantly for any change in data. Online learning is far more efficient and scalable for large-scale learning tasks in real-world data, analytics, and various applications where data is not only large in size but also arrives at high velocity.

Training a model can go wrong in lots of different ways: the algorithm itself might not be suitable, the model might fail to generalise well, the learning rate might be wrong, the regularisation might be too low or too high…the list goes on. Why on earth would we even attempt to learn immediately when there are no guarantees on what might happen?

The answer is simple: no matter how good a model is, or how much data you feed it, a model is still an imperfect representation of an environment. To make the best possible decisions right now, we can’t afford to have a model that only knows about things that happened yesterday.

Consider the following example. Let’s say we run a news website. We personalise our news by collecting data on what was clicked or not clicked, and by whom. Based on this information, we predict the types of news different visitors might like, and serve them relevant items.

One day, out of the blue, word gets out that the government is issuing a state of emergency, and will hold a press conference in an hour. Suddenly, everyone is interested in domestic affairs — even those who typically only read about sports or look at the funnies. When presented with a news piece about the conference, a huge percentage of the audience clicks it to learn more.

If you had gone the traditional route and batch trained your recommendation engine once a day, it would still be stuck offering the same type of content, even though the underlying world changed dramatically¹. You should be serving up domestic news right now, but aren’t because your system is too slow.

It gets worse: the following day, after the press conference and following a new training cycle, your engine would start actively recommending domestic news which, after 24 hours, isn’t necessarily interesting any more. It’s made two mistakes, both because it can’t react fast enough.

That’s the power of online learning: done properly, it can react in minutes or even seconds. With it, there is no such thing as “yesterday’s news”.

Online learning also requires an entirely different approach in terms of technical architecture. Since a model can, and will, change from second to second, you can’t just instantiate several instances like you can with traditional techniques. It’s not horizontally scalable. Instead, you are forced to have a single model instance that eats new data as fast as it can, spitting out sets of learned parameters behind an API. And the second that one set in one process gets replaced by a new one, all other processes must follow suit immediately. It’s an engineering challenge, because the most important part (the model) is only vertically scalable. It may not even be feasible to distribute between threads.

Learning immediately also requires fast access to new data. If you’re lucky enough to get all the data you need for a single training example as part of an API call, you’re good to go. But if something is not available client-side, you need to be able to grab that data from somewhere in milliseconds. Typically, that means using an in-memory store like Redis. “Big data” processing frameworks aren’t of much help. If you want to do both batch and online learning, Spark isn’t enough. If you do only online learning, Spark is useless.

# Evolution of CNN

There are four different layers of CNN.

* Convolutional layer
* ReLU layer
* Pooling layer
* Fully connected layer

It all started with LeNet in 1998 and eventually, after nearly 15 years, lead to ground-breaking models winning the ImageNet Large Scale Visual Recognition Challenge which includes AlexNet in 2012 to Google Net in 2014 to ResNet in 2015 to an ensemble of previous models in 2016. In the last two years, no significant progress has been made, and the new models are an ensemble of previous ground-breaking models.

* LeNet in 1998

LeNet is a 7-level convolutional network by LeCun in 1998 that classifies digits and used by several banks to recognise the hand-written numbers on cheques digitised in 32x32 pixel greyscale input images.

* AlexNet in 2012

AlexNet: It is considered to be the first paper/ model, which rose the interest in CNNs when it won the ImageNet challenge in the year 2012. It is a deep CNN trained on ImageNet and outperformed all the entries that year.

* VGG in 2014

VGG was submitted in the year 2013, and it became a runner up in the ImageNet contest in 2014. It is widely used as a simple architecture compared to AlexNet.

* GoogLeNet in 2014

In 2014, several great models were developed like VGG, but the winner of the ImageNet contest was GoogLeNet. GoogLeNet proposed a module called the inception modules that includes skipping connections in the network, forming a mini-module, and this module is repeated throughout the network.

* ResNet in 2015

There are 152 layers in the Microsoft ResNet. The authors showed empirically that if you keep on adding layers, the error rate should keep on decreasing in contrast to “plain nets” we're adding a few layers resulted in higher training and test errors.

# Power Analysis

Part of experimental design. It determines the sample size required to find the effect of a given size from a cause with a specific level of assurance.

# Maintenance of deployed model

1. Monitor: constant monitoring of all models is needed to determine their performance accuracy.
2. Evaluate: evaluation metrics of the current are calculated to determine if a new algorithm is needed
3. Compare: the new models are compared to each other to determine which model performs the best.
4. Rebuild: the best performing model is rebuilt on the current state of the data.

# Canary Deployment

# Blue Green Deployment

# Market Basket Analysis

# Time series problems

Time series is extrapolation whereas Regression is interpolation. Time series refers to an organized chain of data. Time-series forecasts what comes next in the sequence.

Regression can be applied to Time-series problems as well as to non-ordered sequences which are termed as features.

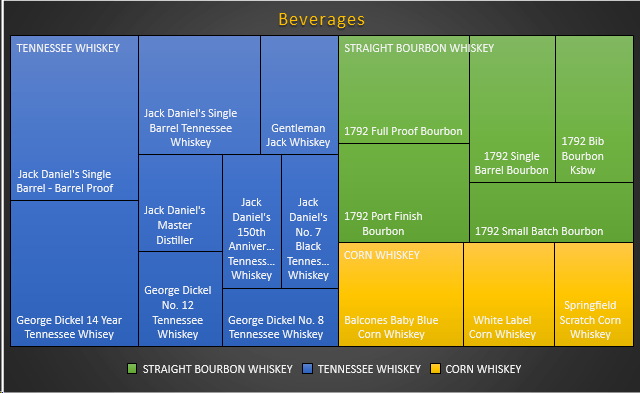
The observations close to one another in time are expected to be similar to the ones far away which provide accountability for seasonality. For instance, today’s weather would be similar to tomorrow’s weather but not similar to weather from 4 months from today. Hence, weather prediction based on past data becomes a time series problem.

# Intercluster and Intracluster

# Heatmap, Tree Map

**Heatmap** - 2-D graphical representation of data containing individual values in a matrix format. The values show the correlation values and are represented by various shades of the same colour.

**Tree Map** - It is a chart type that illustrates hierarchical data or part-to-whole relationships.



# Downsides of Visualisation

It gives estimation not accuracy, a different group of the audience may interpret it differently, improper design can cause confusion.

# In a time-interval of 15-minutes, the probability that you may see a shooting star or a bunch of them is 0.2. What is the percentage chance of you seeing at least one star shooting from the sky if you are under it for about an hour?

Let us say that probability that we may see a minimum of one shooting star in 15 minutes is ‘Prob’. So, Prob = 0.2

Now, the probability that we may not see any shooting star in the time duration of 15 minutes is

The probability that we may not see any shooting star for an hour is

So, the probability that we will see one shooting star in the time interval of an hour is

# ZCA Whitening

Zero Component Analysis Whitening.

ZCA transforms the data to zero mean and makes the features linearly independent of each other. Making the co-variance matrix as the identity matrix is called whitening.

# RandomizedSearchCV, GridSearchCV, BayesianSearchCV

**Randomized search CV** is used to perform a random search on hyperparameters. Randomized search CV uses a fit and score method, predict probability, transform etc. The parameters of the estimator used to apply these methods are optimized by cross-validated search over parameter settings. A fixed number of parameter settings is sampled from the specified distributions. The number of parameter settings that are tried is given by ‘n\_iter’.

**GridSearchCV** – Grid search is the process of performing hyperparameter tuning to determine the optimal values for a given model. It runs on all the possible range of hyperparameter values and outputs the best model.

**BayesianSearchCV** – Bayesian search keeps track of past evaluation results, which they use to form a probabilistic model mapping hyperparameters to a probability of a score on the objective function

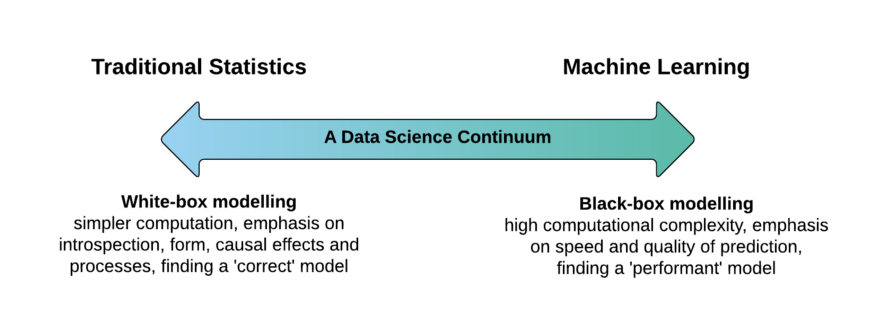
# Difference between black-box vs white-box models?

Black-box models such as NN, gradient boosting models or complicated ensembles often provide great accuracy. The inner workings of these models are harder to understand and they don’t provide an estimate of the importance of each feature on the model predictions, nor is it easy to understand how the different features interact.

Simpler models such as linear regression and DT on the other hand provide less predictive capacity and are not always capable of modelling the inherent complexity of the dataset (i.e., feature interactions). They are however significantly easier to explain and interpret.

Even in less complex scenarios, black-box models typically outperform white-box counterparts due to black-box models’ ability to capture high non-linearity and interactions between features: a multi-layer neural network applied to a churn detection use case. Despite their superior performance, black-box models have several drawbacks. The first disadvantage is a lack of explain ability both internally and externally to customers and regulators seeking explanations for why a decision was made.

The second disadvantage of black-box models is that there may be a slew of unseen issues affecting the output — such as overfit, spurious correlations, or “garbage in / garbage out” — that are impossible to detect due to a lack of understanding of the black-box model’s operations. Another drawback of not spending enough time understanding the reality beyond the black-box model is that it creates a “comprehension debt” that must be repaid over time through difficulty in maintaining performance, unexpected effects such as people gaming the system, or potential unfairness.



# What is explainable AI?

Explainable AI is a set of tools and frameworks to help you understand and interpret predictions made by your machine learning models, natively integrated with a number of Google's products and services. With it, you can debug and improve model performance, and help others understand your models' behaviour.

# t-SNE

t-Distributed Stochastic Neighbour Embedding is a non-linear dimensionality reduction algorithm used for exploring high-dimensional data. It maps multi-dimensional data to two or more dimensions suitable for human observation.

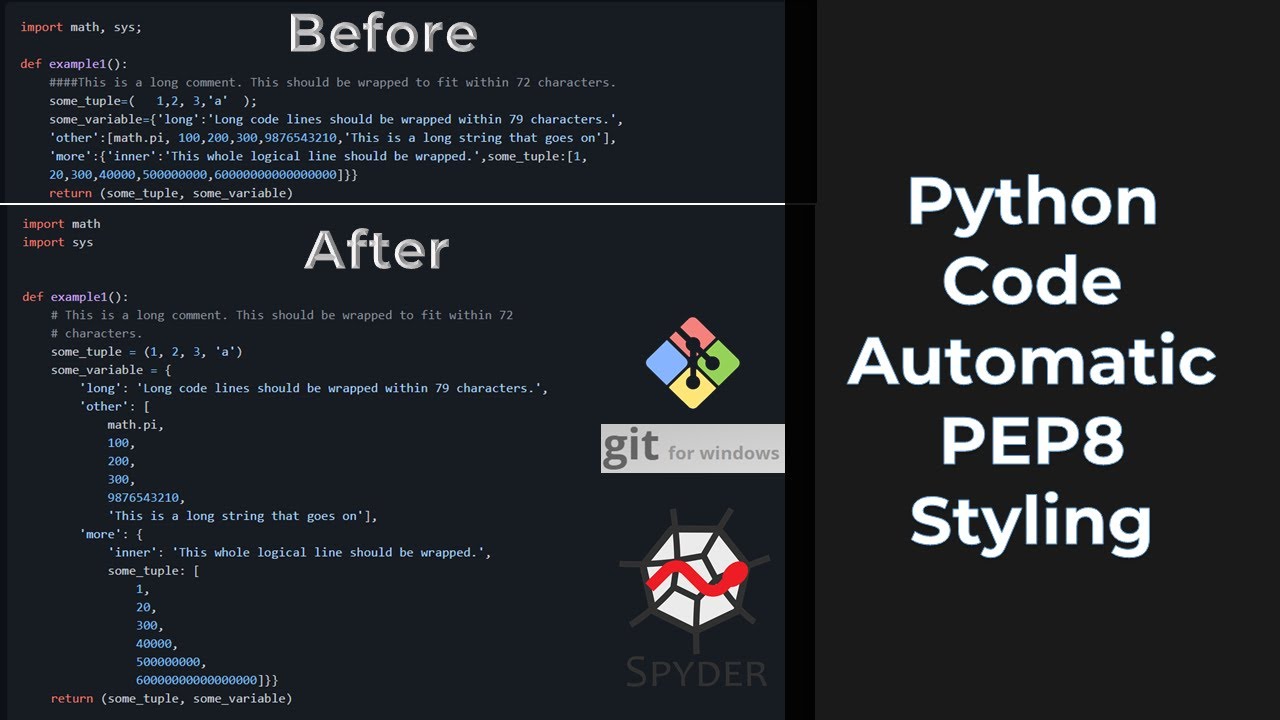
# References

1. Interviewbit
2. Intellipaat
3. Simplilearn
4. Hacker.io
5. Guru99
6. iNeuron
7. Springboard
8. mygreatlearning

Python

# PEP8

PEP 8 is a coding convection. It consists of coding guidelines that are a set of recommendations for Python language about making the Python more readable and usable for another person. PEP stands for Python Enhancement Proposal.



# What type of language is python

Python is a high level, interpreted, general-purpose programming language.

Python is capable of scripting, but it is considered as a general-purpose programming language.

Python is case-sensitive.

Whenever python exits, especially those Python modules which are having circular references to other objects or the objects that are referenced from the global namespaces are not always de-allocated or freed. It is impossible to de-allocate those portions of memory that are reserved by the C library. On exit, because of having its own efficient clean up mechanism, python would try to de-allocate/destroy every other object.

Python lays down the concept of prefixing the name of the variable, function or method with a single or double underscore to imitate the behaviour of protected and private access specifiers.

Indentation is required in python, and it is part of its syntax.

Python is an object-oriented programming language meaning it can enclose the codes within the objects. The property allows the storage of the data and the method in a single unit called the object.

# What is high level language?

A high-level language is any programming language that enables development of a program in a much more user-friendly programming context and is generally independent of the computer's hardware architecture.

# What is general purpose programming language?

A programming language designed to be used for building software in a wide variety of application domains, across a multitude of hardware configurations and operating systems.

# What is scripting language?

A scripting language is a programming language that is interpreted. It is translated into machine code when the code is run, rather than beforehand. Eg., JavaScript, Python, Ruby etc.

# What is an interpreted language?

An Interpreted language executes its statements line by line. E.g., Python, JavaScript, R, PHP, and Ruby. Programs written in an interpreted language runs directly from the source code, with no intermediary compilation step. Python code is fast to develop.

Python code is not as fast in execution compared to C++, since the code is not directly compiled and executed and an additional layer of python virtual machine is responsible for the execution. Hence it is little slow as compared to conventional languages like C, C++ etc.

# If a function doesn’t have a return statement, is it valid?

Yes, it is valid. By default, all the functions in Python return the particular value “None” unless an explicit return statement is present to return a value that’s not “None”.

# Difference between local and global namespaces?

Creation of global namespaces occurs when the program begins, while local namespaces are created within a function when that function is called.

# Exception handling in python?

Python uses the block <try-except> to handle exceptions.

The <try-except> block can help you check error details without program termination and may also offer suggestions for fixing the error. Python provides 3 words to handle exceptions, ‘try’, ‘except’ and ‘finally’.

Try:

#Try to do this

Except:

#If try block fails then do this

Finally:

#Always do this

# Describe compile-time and run-time code checking?

Compile-time and run-time code checking is done uniquely in Python and involves first checking a small portion during compile-time checking.

Then most other checks like name and type checks happen only during code execution. The instructions or **source code written using high-level language is required to get converted to machine code for a computer to understand.** During compile time, the source code is translated to a byte code like from .java to .class.  During compile time the compiler check for the syntax, semantic, and type of the code.

So, referencing a non-existent function in your Python code will still lead it to successful compilation.

But your code will fail with an exception only during code execution when the code execution path references the non-existent function.

A program’s life cycle is a runtime when the program is in execution. Following are the different types of runtime errors:

* Division by zero – when a number is divided by zero (0)
* Dereferencing a null pointer – when a program attempts to access memory with a NULL
* Running out of memory – when a computer has no memory to allocate to programs



# Differentiate between instance and class variable?

|  |  |
| --- | --- |
| **Instance variable** | **Class variable** |
| They are created locally within a class. Each object has its own copy of instance variables. | They are created globally in a class. Class variables are common to all objects of a class. |
| We can access them only within the particular object of the class. | We can access them within all the instances of that class. |
| They can be accessed only through an object reference. | They can be accessed using the class name or an object reference. |
| Changes made to instance variables through one object will not reflect in another. | Changes made to class variables through one object will reflect in another object. |

# Are ML models written in C++?

All AI frameworks are written in C++. Python is an interface to command such frameworks from a terminal.

# How do I access a module written in python from C?

# What are the data types in python?

Everything is an object in python; hence all the data types are actually classes. Variables are instances of the classes.

* Int, float, complex
* String
* List, tuple, range
* Set – It is an unordered collection datatype that is iterable, mutable, and has no duplicate items. Represented by {}. Since sets are unordered, we cannot access items using indexes as we do in lists.
* Dictionary - It is an unordered collection of items. Each item of a dictionary has a key/value pair. Dictionary elements are accessed via keys.

Syntax:

my\_dic = {<key1>:<value>, <key2>:<value>, ….}

* Bool
* Bytes, bytearray, memoryview

# Why tuples are faster than lists?

Tuple is stored in a **single block of memory**. Creating a tuple is faster than creating a list. Creating a list is slower because two memory blocks need to be accessed.

# What does break, continue, pass keyword do in python?

‘break’ and ‘continue’ help control loops in python.

“break” **breaks the current loop** from execution and transfers control to the next block. “continue” makes a **jump to the next iteration** of the loop without exhausting it.

The “pass” keyword is a no-operation statement in python. It signals that no action is required. It works as a **placeholder** in compound statements which are intentionally left blank.

def myEmptyFunc():

pass #do nothing

myEmptyFunc() #nothing happens

# What is pop()?

The pop() method removes the item at the given index from the list and **returns** the removed item.

Syntax: list.pop(index)

# What will be the output of the code below?

Output:

[]

It will not result in an ‘IndexError’. The code is trying to access the 10th object in the list which does not exist, so it will output: [].

# Explain generators vs iterators vs decorators?

**Iterator** has the whole sequence in memory before it returns the first result. An iterator uses the "return".

**Generator** calculates each result at the moment it is called for. The next result is unknown. A generator uses "yield".

You'd use a generator when processing a stream, or when memory consumption is important. Generators are iterators, but iterators are not generators.

**Constructor** is a special kind of method in python. Constructor will be called automatically.

class Employee:

    def \_\_init\_\_(self):

        print("Constructor is executed")

    def temp(self):

        print("Hello!")

emp = Employee()

A decorator allows adding functionality to an existing function by passing that existing function to a decorator, which executes the existing function as well as additional code.

They are represented by the @decorator\_name in python and are called in bottom-up fashion.

**Decorator function**: This takes , as an argument. It also defines a function, , which calls and executes some code, . Then it returns the function which is defined.

def ():

def logfunction\_called():

print(f’{ } called.’)

return logfunction\_called

We will write other functions that we’ll eventually add with decorator

def my\_name():

print(‘Manish’)

def friends\_name():

print(‘Momo’)

my\_name()

friends\_name()

Output:

Manish

Momo

Now add the decorator to both.

@

def my\_name():

print(‘Manish’)

@

def driends\_name():

print(‘Momo’)

my\_name()

friends\_name()

Output:

<function my\_name at 0x10f> called

Manish

<function friends\_name at 0x10e> called

Momo

# Is python call by value or call by reference? Does python allow arguments Pass by Value or Pass by Reference

Everything in Python is an object and all variables hold references to the objects. The reference values are according to the functions; as a result, you cannot change the value of the references. However, you can change the objects if it is mutable.

In case you pass arguments like whole numbers, strings or tuples to a function, the passing is like call-by-value because you cannot change the value of the immutable objects being passed to the function.

Whereas passing mutable objects can be considered as call by reference because their values are changed inside the function, then it will also be reflected outside the function.

#call by value

string = "Geeks"

def test(string):

    string = "Geeks for geeks"

    print("Inside function:", string)

test(string)

print("Outside function:", string)

Inside function: Geeks for geeks

Outside function: Geeks

#call by reference

my\_list = [10,20,30,40]

def test(list):

    list.append(50)

    print("Inside function:", list)

test(my\_list)

print("Outside function:", my\_list)

Inside function: [10, 20, 30, 40, 50]

Outside function: [10, 20, 30, 40, 50]

**Neither** the arguments are Pass by Value nor does python support Pass by Reference. Instead, they are **Pass by Assignment**. The parameter which you pass is originally a reference to the object not the reference to a fixed memory location. But the reference is passed by value.

# What is dynamically typed language?

**Typing** refers to type-checking in programming languages. In a strongly-typed language, such as Python, "1" + 2 will result in a type error since these languages don't allow for "type-coercion" (implicit conversion of data types).

On the other hand, a weakly-typed language, such as JS, will simply output "12" as result.

Type-checking can be done at two stages -

Static - Data Types are checked before execution.

Dynamic - Data Types are checked during execution.

Python is an interpreted language, executes each statement line by line and thus type-checking is done on the fly, during execution.

# Tabs vs Spaces?

Both are completely correct. The most popular way of indenting is with spaces only and also part of the PEP-8 standard.

# What is ‘id()’ function in python?

It is an inbuilt function in python. Function accepts a single parameter and is used to return the identity of an object. This identity has to be unique and constant for this object during the lifetime. Two objects with non-overlapping lifetimes may have the same id() value. If we relate this to C, then they are actually the memory address, here in Python it is the unique id. This function is generally used internally in Python.

# Difference between shallow copy and deep copy?

In Python, we use = operator to create a copy of an object. You may think that this creates a new object; it doesn't. It only creates a new variable that shares the reference of the original object.

old = [[1,2,3], [4,5,6], [7,8,'a']]

new = old

new[2][2]=9

print('Old List',old)

print('ID of old:',id(old))

print('New list',new)

print('ID of new',id(new))

Output:

Old List [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

ID of old: 2241843483968

New list [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

ID of new 2241843483968

Both variables old\_list and new\_list shares the same id i.e 140673303268168. So, if you want to modify any values in new\_list or old\_list, the change is visible in both.

In Python, there are two ways to create copies:

* Shallow Copy
* Deep Copy

A shallow copy creates a new object which stores the reference of the original elements. So, a shallow copy doesn't create a copy of nested objects, instead it just copies the reference of nested objects.

A deep copy creates a new object and recursively adds the copies of nested objects present in the original elements. Both the old list and the new list are independent.

# What are access specifiers?

Python control access modifications which are used to restrict access to the variables and methods of the class. Most programming languages has three forms of access modifiers, Public, Protected and Private in a class. Python uses ‘\_’ symbol to determine the access control for a specific data member or a member function of a class. They help in securing data from unauthorized access and in preventing it from being exploited. A class in Python has three types of access modifiers:

* **Public Access Modifier** - The members of a class that are declared public are easily accessible from any part of the program. All data members and member functions of a class are public by default.
* **Protected Access Modifier** - The members of a class that are declared protected are only accessible to a class derived from it. Data members of a class are declared protected by adding a single underscore ‘\_’ symbol before the data member of that class.
* **Private Access Modifier** - The members are accessible within the class only, most secure access modifier. Data members of a class are declared private by adding a double underscore ‘\_\_’ symbol before the data member of that class.

# What is python suite?

A group of individual statements, which make a single code block are called suites in Python. Compound or complex statements, such as if, while, def, and class require a header line and a suite.

Header lines begin the statement (with the keyword) and terminate with a colon (:) and are followed by one or more lines which make up the suite.

while expr==True:

  stmt1

  stmt2

# What does \*args, \*\*kwargs mean?

We use \*args when we aren’t sure how many arguments are going to be passed to a function, or if we want to pass a stored list or tuple of arguments to a function.

\*\*kwargs is used when we don’t know how many keywords argument will be passed to a function, or it can be used to pass the values of a dictionary as keyword arguments.

# Does python need to be compiled before it is run?

Compiled languages are based on code that can be executed directly on a computer’s processor. An interpreted language, isn’t in “machine code” before runtime. The translation happens at the same time as the program is executed. Python is an interpreted language so it does not need to be compiled.

# What are docstrings?

Docstrings are not comments, they are documentation strings. These docstrings are within triple quotes. They are not assigned to any variable and therefore, at times, serve the purpose of comments as well.



# What is Global Interpreter Lock (GIL) in python?

Python is not thread safe when it comes to memory management. So, if you are running multiple threads, the GIL is a bottleneck; it only allows one thread to access memory at a time. This means that only one thread can be in a state of execution at any point in time. Solutions are to use multi-processing, use extensions written in C, or use other python implementations like IronPython or Cython.

# Difference between concurrency vs parallelism?

Concurrency and parallelism are names for two different mechanisms for juggling tasks in programming. Concurrency involves allowing multiple jobs to take turns accessing the same shared resources, like disk, network, or a single CPU core. Parallelism is about allowing several tasks to run side by side on independently partitioned resources, like multiple CPU cores.

The goal of concurrency is to prevent tasks from blocking each other by switching among them when one is forced to wait on an external resource. A common example is completing multiple network requests. The crude way to do it is to launch one request, wait for it to finish, launch another, and so on. The concurrent way to do it is to launch all requests at once, then switch among them as they get responses back. Through concurrency, we can aggregate all the time spent waiting for responses.

Parallelism, by contrast, is about maximizing the use of hardware resources. If you have eight CPU cores, you don’t want to max out only one while the other seven lie idle. Rather, you want to launch processes or threads that make use of all those cores, if possible.

# What is lazy eval?

Lazy evaluation means that the object is evaluated when it is needed, not when it is created.

In python 2, range will return a list – this means that if you give it a large number, it will calculate range and return at the time of creation.

In python 3, however you get a special range object. Only when you consume it, will it actually be evaluated – in other words, it will only return the numbers in the range when you actually need them.

# What are Meta classes?

A meta class in python is a class of a class that defines how a class behaves. A class is itself an instance of a meta class. A class in python defines how the instance of the class will behave.

# Multithreading and Multiprocessing?

A **program** is an executable file which consists of a set of instructions to perform some tasks and is usually stored on the disk of your computer.

A **process** is what we call a program that has been loaded into memory along with all the resources it needs to operate. It has its own memory space.

A **thread** is the unit of execution within a process. A process can have multiple threads running as a part of it, where each thread uses the process’s memory space and shares it with other threads.

**Multithreading** is a technique where multiple threads are spawned by a process to do different tasks, at about the same time, just one after the other. This gives you the illusion that the threads are running in parallel, but they actually run in a concurrent manner. In python, the Global Interpreter Lock (GIL) prevents the threads from running simultaneously.

**Multiprocessing** is a technique where parallelism in its truest form is achieved. Multiple processes are run across multiple CPU cores, which do not share the resources among them. Each process can have many threads running in its own memory space.

# Memory management in python?

Memory allocation can be defined as allocating a block of space in the computer memory to a program. In python, memory allocation and deallocation method are automatic as the python developers created a garbage collector for python so that the user does not have to do manual garbage collection. [Garbage collection](https://www.geeksforgeeks.org/garbage-collection-python/) is a process in which the interpreter frees up the memory when not in use to make it available for other objects. Assume a case where no reference is pointing to an object in memory i.e., it is not in use so, the virtual machine has a garbage collector that automatically deletes that object from the heap memory.

Reference counting works by counting the number of times an object is referenced by other objects in the system. When references to an object are removed, the reference count for an object is decremented. When the reference count becomes zero, the object is deallocated.

# Built-in types available in python?

* Integer
* Complex number
* Floating point numbers
* Strings
* Built in functions

# What is a lambda function?

An anonymous function is known as a lambda function. This function can have any number of parameters but can have just one statement.

# What are generators?

Generators are a way of implementing iterators. A generator function is a normal function except that it contains yield expression in the function definition making it a generator function.

# What is the purpose of “end” in python? Print without a newline?

In Python, the default ‘print()’ function ends with a newline. The ‘print()’ function comes with a parameter called ‘end’ and the value of this parameter in ‘/n’ which means a new line. A print statement can be ended with any character/string using this parameter. Using this parameter, we can change the end character in the print statement according to our choice.

For example, in print a line instead of the new line in the end:

print(“Let us learn”, end = ‘ ’)

print(“python”)

print(“”knowledgehut”, end=’.’)

print(“com”, end=’ ‘)

Output:

Let us learn Python

Knowledgehut.com

# type() and identification number of an object in python?

We use the function type() and id().

type(<variableName>) – It gives us the type of the object that variable is pointing to.

id(<variablename>) – It gives us the unique identification number of the object that variable is pointing to.

# Additional data structures in python?

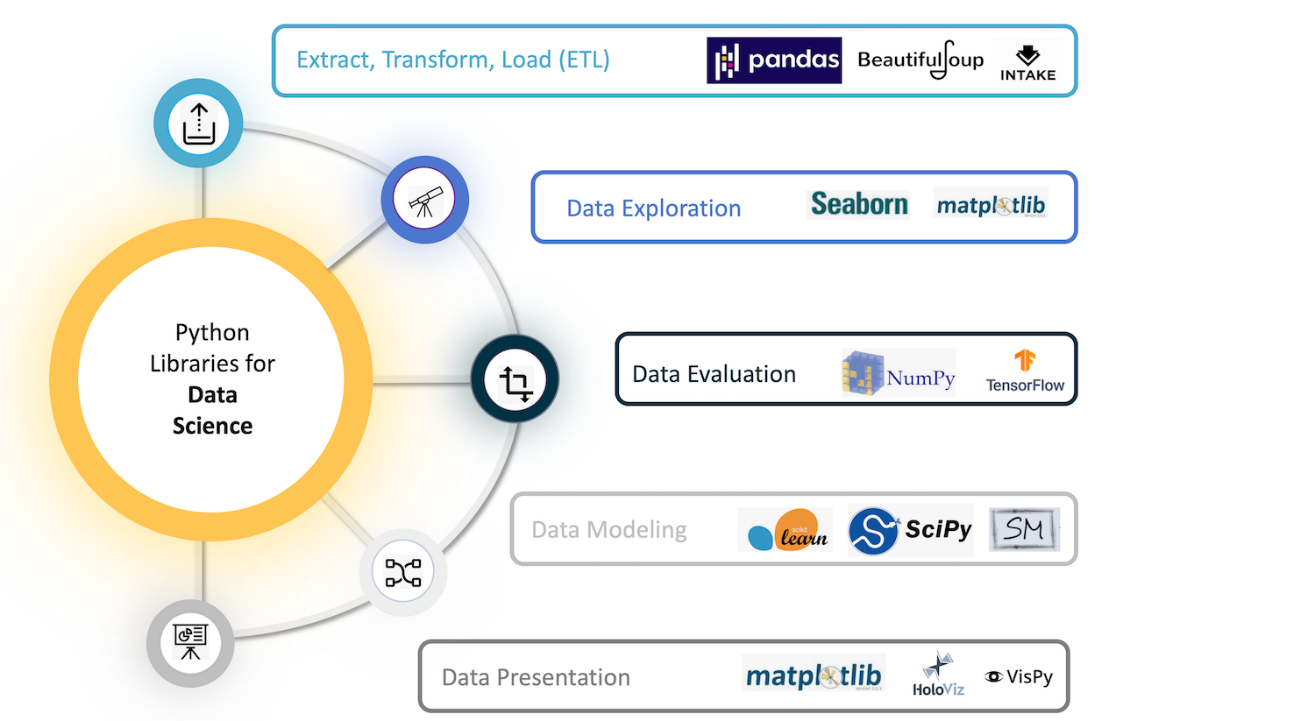
* Bisect
* Boolean
* Deque
* Float
* Heapq
* Integers

# What is Python module?

A module is a single file / files containing functions, definitions, and variables designed to do certain tasks. It is a extension file. It can be imported at any time during a session and needs to be imported only once. To import a python module, there are two ways:

# Library

A library is a collection of reusable functionalities of codes that allows us to perform a variety of tasks without having to write the code. It is a collection of modules. These codes can be used by importing and by calling that library’s method with a period (.).



# Name mutable and immutable objects

The mutability of a data structure is the ability to change the portion of the data structure without having to recreate it. Mutable objects are lists, sets, values in a dictionary.

Immutability is the state of the data structure that cannot be changed after its creation. Immutable objects are integers, strings, float, bool, tuples, keys of a dictionary. Immutable types are pointers to locations in memory so are subjected to change.

# Name some standard errors that occur in python?

* TypeError - It occurs when the expected type does not match with the given type of a variable.
* ValueError- It occurs when an expected value is not given, suppose you are expecting 6 elements in a list and you gave 2.
* NameError- It occurs when you are trying to access an undefined variable or a function.
* IOError- It occurs when you are trying to access a file that does not exist.
* IndexError- It occurs when you are trying to access an invalid index of a sequence.
* KeyError- It occurs when you use an invalid key to access a value in the dictionary.

# Compound Data Types

The data type that is constructed using simple, primitive, and basic data types.

# How will you convert a list into a tuple?

days = [‘sun’, ‘mon’, ‘tue’, ‘wed’, ‘thu’, ‘fri’, ‘sat’]

print(tuple(days))

Output:

(‘sun’,’mon’, ‘tue’, ‘wed’, ‘thu’, ‘fri’, ‘sat’)

# What is the difference between NumPy and SciPy?

NumPy is a python library which is written in C and therefore is one of the fastest and most compatible libraries among the rest.

NumPy would contain nothing but the array data type and the most basic operations: indexing, sorting, reshaping, basic element wise functions, etc. All numerical code would reside in SciPy. SciPy contains more fully-featured versions of the linear algebra modules, as well as many other numerical algorithms.

# Difference between list and tuple

List

* Enclosed within
* Mutable
* Slower than tuples
* E.g. [‘A’, 1, ‘h’]

Tuple

* Enclosed in
* Immutable
* Faster than lists
* Must be used when the order of the elements of a sequence matter.
* E.g. (‘Twenty’, 20, ‘XX’)

# Why is python numpy better than lists?

Python numpy arrays are fast, consume less memory and convenient with lots of functionality.

# Map function

Map function executes the function given as the first argument on all the elements of the iterable given as the second argument.

# String interpolation

* f strings
* % Operator

# Difference between “is” and “==”

‘is’ checks identity and ‘==’ checks equality.

Example: We will create some lists and assign them to names. ‘b’ points to the same object as ‘a’.

Output:

True

True

Output:

True

False

Output:

12345

12345

12456

‘c’ has a different ‘id’ than ‘a’ and ‘b’

# List and Dictionary comprehension

Python comprehensions are syntactic constructs providing a way to build a list, dictionary or set based on the existing list, dictionary or set whilst altering or filtering elements.

These are generally more compact and faster than normal functions and loops for creating lists. We must avoid writing very long comprehensions in one line to ensure that code is user-friendly and to maintain the readability.

Example of list comprehension

Output: 10, 20, 30, 40, 50

Example of dictionary comprehension

Output: {‘i’: ‘this’, ‘ii’: ‘month’, ‘iii’: ‘is’, ‘iv’: ‘March’}

# What is tuple unpacking

A tuple can be unpacked in the sense that its elements can be separated in the following manner:

We have a tuple x = (500, 352)

This tuple can be assigned to two new variables in this way: a, b = x

Now, printing a and b will result in: print(a) = 500 and print(b) = 352

Tuple unpacking helps to separate each value one at a time.

# Difference between %, / and //

returns a remainder

returns quotient

// is the floor division that rounds off the quotient to the bottom

Example:

11 % 2 -> 1

11 / 2 -> 5.5

11 // 2 -> 5

# Difference between indexing and slicing

Indexing is extracting or lookup one or particular values in a data structure.

Slicing retrieves a sequence of elements like strings, list, tuples etc.

# Difference between global and local variables

Global variables are the defined and declared outside a function.

Local variables are declared inside the function’s body.

# Difference between .py and .pyc files

*.py* files contain the source code of a program. Whereas, *.pyc* file contains the bytecode of your program. We get bytecode after compilation of *.py* file (source code).

*.pyc* files are not created for all the files that you run. It is only created for the files that you import.

# What is PYTHONPATH in python?

An environment variable to add additional directories where python will look for modules and packages. Useful in maintaining python libraries that you do not wish to install in the global default location.

# What is OOPS?

All objects are grouped into classes in object-oriented programming.

OOPs integrates real-world concepts into programming such as inheritance, polymorphism, and abstraction.

* OOPs reduces redundancy
* OOPs provides ability to bind both data and code together.
* Allows in keeping sensitive data confidential.
* OOPs improves code-readability.
* Polymorphism gives flexibility to the programs by allowing the entities to have multiple forms.

Java is not pure OOP language. The static keyword in Java allows us to use classes without the use of objects.

* **Object** – A real world entity having a particular state and behaviour. It is also called as an instance of a class.
* **Class** – A **user defined prototype** which basically is a **blueprint** from which an object can be created or instantiated. Classes can construct **instances** of objects. This is known as **instantiation**. An object exists physically in this world, but class does not exist.
* **Inheritance** – A concept that refers to an object gaining all the properties and behaviours of a parent object. It provides code reusability.

In the below example, , inherits from . And with that inheritance comes the instance methods of the parent class.

class Car():

def drive(self):

print(‘vroom’)

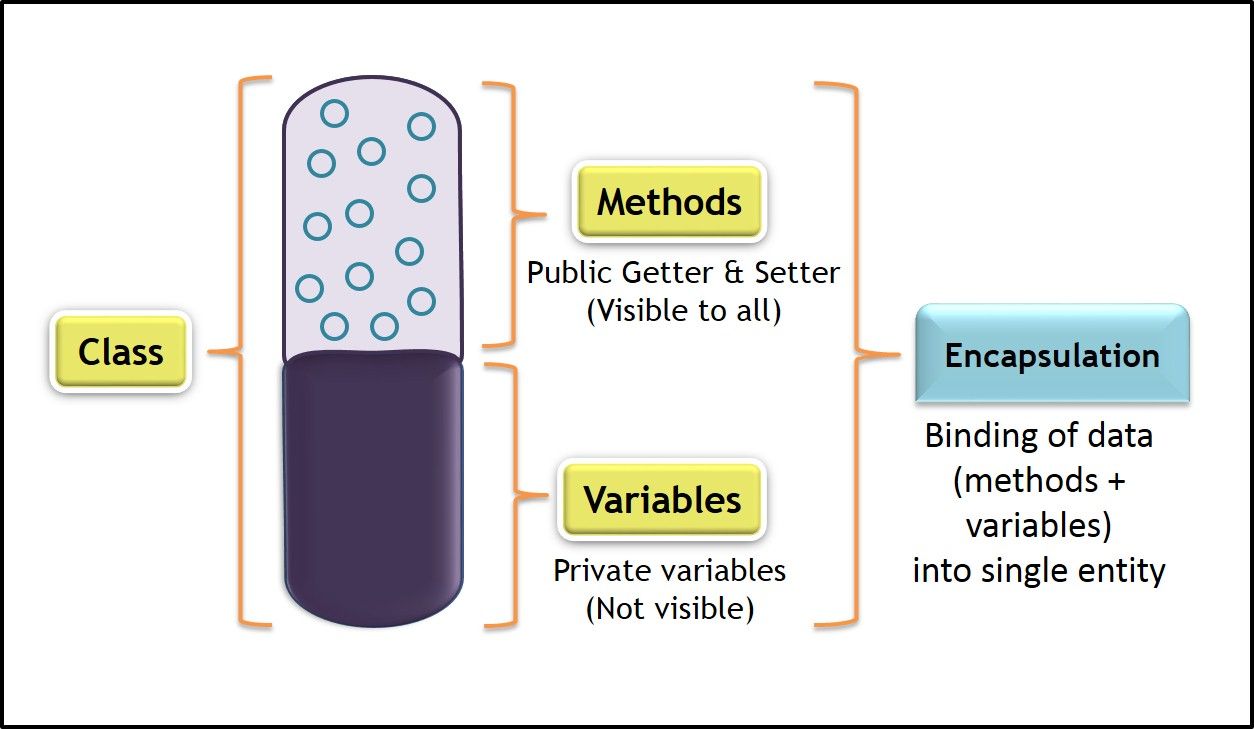
class Audi(Car)

pass

audi = Audi()

audi.drive()

* **Polymorphism** – A concept that allows a task to be performed in different ways.
* **Abstraction** – A concept that hides the internal details of an application and only shows the functionality.
* **Encapsulation** – A concept that refers to the wrapping of code and data together into a single unit. It means binding the code and the data together. Example, a python class.



Syntax:

class Employee:

    def temp(self):

        print("Hello!")

emp = Employee()

emp.temp()

Output:

Hello!

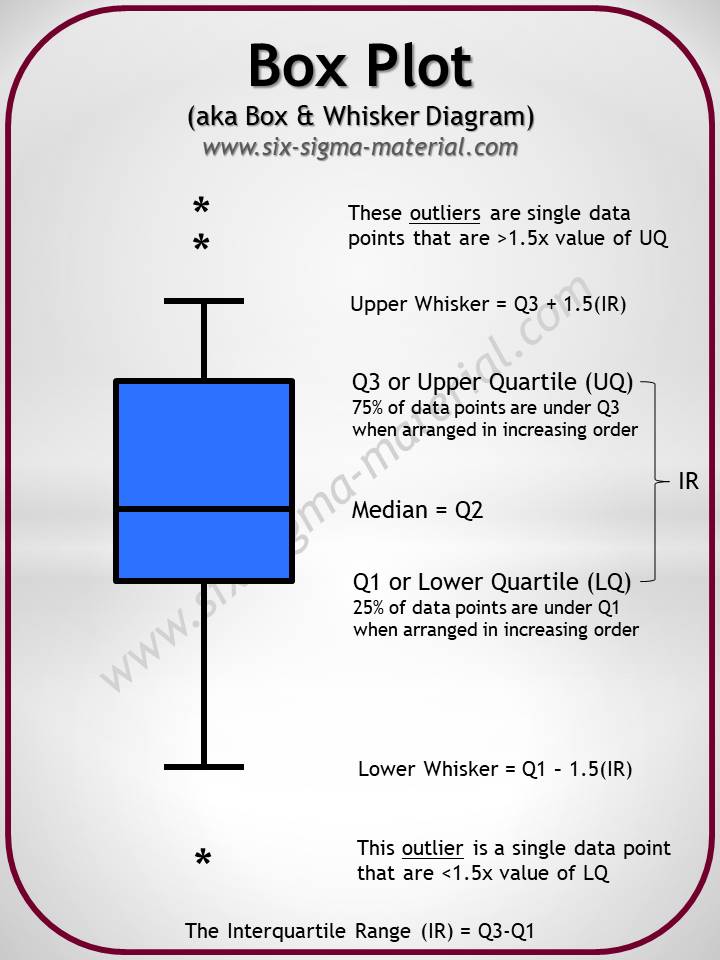
# Difference between data frame and matrices

|  |  |
| --- | --- |
| Data Frame | Matrices |
| A collection of series that share a common index. | A matrix in NumPy is constructed with multiple vectors. |
| It can hold multiple series, which are of different data types. | It can hold only one data type in the entire 2-D structure. |

# Scatter plot

It is 2-D data visualization plot that illustrates relationship between observations of two different variables, plotted against x and y axis respectively.

# Box plot



# Explain range and xrange?

Range generates a list of integers and there are 3 ways to use it. The function takes 1 to 3 arguments.

: generate integers from 0 to the “stop” integer.

Output: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

generate integers from the “start” to the “stop” integer.

Output: [2, 3, 4, 5, 6, 7, 8, 9]

generate integers from “start” to “stop” at intervals of “step”

Output: [2, 4, 6, 8]

**xrange** and **range** are exact same in terms of functionality.

The only difference is that range returns a python list object and xrange returns an xrange object.

# What does “self” refer to in a class

‘self’ refers to the instance of the class itself. It’s how we give methods access to and the ability to update the object they belong to.

E.g., Passing self to \_\_init\_\_() gives us the ability to set the colour of an instance on initialization.

class Shirt:

def \_\_init\_\_(self, colour):

= colour

s = Shirt(‘yellow’)

Output:

Yellow

# Difference between dictionaries and JSON

is python datatype, a collection of indexed but unordered keys and values.

JSON is just a string which follows a specified format and is intended for transferring data.

# Difference between and

is the object representing the function which can be assigned to a variable or passed to another . calls the function and returns it outputs.

def func():

(‘I m a function’)

#=> function \_\_main\_\_.func

#=> I m a function

# Check if a data set or time series is Random

We use the lag plot. If the lag plot for the given dataset does not show any structure, then its random.

# ORM

**Object Relational Mapping** map data models to database tables and simplifies database transactions.

SQLAlchemy is typically used in the context of Flask, and Django has its own ORM.

# What is namespace in python?

In python, every name introduced has a place where it lives and can be hooked for. This is known as namespace. It is like a box where the variable name is mapped to the object placed.

# What is pickling and unpickling?

Pickle module accepts any python object and converts it into a **string representation** and dumps it into a file using the **dump function**, this process is called pickling.

While the process of retrieving original python objects from the stored string representation is called unpickling.

# Memory status when python exists?

No, it is not freed, because the objects that are referenced from global namespaces of Python modules are not always de-allocated when Python exists.

# Difference between append and extend

adds a value to a list while adds values in another list to a list.

#=> [1, 2, 3, 6]

#=> [1, 2, 3, 4, 5]

# String functions

* To check if a string only contains numbers

–

#=> False

#=> True

* To check if a string contains letters

#=> False

#=> True

* To check if a string contains only numbers and letters

#=> False

#=> True

# References

1. Analytixlabs
2. https://towardsdatascience.com/53-python-interview-questions-and-answers-91fa311eec3f

SQL

# Requirements for Data

* Integrity
* Availability
* Security
* Independent of Application
* Concurrency

# SQL, PostgreSQL, MySQL

**Structured Query Language**, is the standard language for RDBMS. It is useful in handling organized data comprised of entities (variables) and relations between different entities of the data.

**PostgreSQL** is an enterprise-level, versatile, resilient, open-source, object-relational database management system that supports variable workloads and concurrent users.

**MySQL** is a RDBMS, like SQL Server, Oracle or IBM DB2, that is used to manage SQL databases. The default port for MySQL server is 3306

# SQL Sandbox

SQL Sandbox is a secure environment within SQL Server where untrusted programmes can be run. There are three different types of SQL sandboxes:

* Safe Access Sandbox: In this environment, a user may execute SQL activities like building stored procedures, triggers, and so on, but they can’t access the memory or create files.
* Sandbox for External Access: Users can access files without having the ability to alter memory allocation.
* Unsafe Access Sandbox: This contains untrustworthy code that allows a user to access memory.

# SQL Skills

SQL skills aid data analysts in the creation, maintenance, and retrieval of data from relational databases, which divide data into columns and rows. Fundamental abilities

* Database Management
* Structuring a Database
* Creating SQL clauses and statements
* SQL system skills like MySQL, PostgreSQL
* PHP expertise is useful
* Analyze SQL data
* OLAP skills

# NoSQL vs SQL

|  |  |
| --- | --- |
| NoSQL | SQL |
| Non-relational database | Relational database |
| Databases use dynamic schemas | Have a specified schema and employ structured query language |
| NoSQL database scale horizontally | SQL databases scale vertically |
| NoSQL databases are document, key-value, graph, or wide-column stores | SQL databases are table-based |
| NoSQL excels at unstructured data such as documents and JSON | SQL databases excel in multi-row transactions. |

# SQL Constraints

Constraints are used to specify the rules concerning data in the table. It can be applied for single or multiple fields in an SQL table during the creation of the table or after creating using the command. The constraints are:

* – Restricts NULL value from being inserted into a column
* – Verifies that all the values in a field satisfy a condition
* – Automatically assigns a default value if no value has been specified for the field
* – Ensures unique values to be inserted into the field. This provides uniqueness for the columns and helps identify each row uniquely. Unlike primary key, there can be multiple unique constraints defined per table.
* – Indexes a field providing faster retrieval of records
* – Uniquely identifies each record in a table. It must contain values and has an implicit constraint. A table in SQL is strictly restricted to have one and only one primary key, which is comprised of single or multiple fields.
* – Ensures referential integrity for a record in another table. A comprises of single or collection of fields in a table that essentially refers to the in another table.

# Limitations of Flat Files

* Dependency of program on physical structure of data
* Complex process to retrieve (accessing) data
* Loss of data on concurrent access
* Inability to give access based on record (security)
* Data redundancy and inconsistency

# Use-cases for Files

* On device access (offline access)
* Storing frequent incoming data
* Storing not so important data
* Low-cost data storage

# Database

A database is a shared collection of logically related data and description of these data, designed to meet the information needs of an organization. A database is an organized collection of data, stored and retrieved digitally from a remote or local computer system.

# Collation

Set of rules that determine how data is sorted and compared.

Rules defining the correct character sequence are used to sort the character data. It incorporates options for specifying case sensitivity, accent marks, kana character types, and character width. Below are the different types of collation sensitivity:

* Case sensitivity: A and a are treated differently
* Accent sensitivity: a and á are treated differently
* Kana sensitivity: Japanese kana characters Hiragana and Katakana are treated differently
* Width sensitivity: Same character represented in single-byte (half width) and double-byte (full-width) are treated differently

# Tables and Fields

**Table** is an organized collection of data stored in the form of rows and columns.

Columns in a table are called **fields**.

Rows are called **records**.

# ACID

* Atomicity: Ensures all-or-none rule for database modifications
* Consistency: Data values are consistent across the database
* Isolation: Two transactions are said to be independent of one another
* Durability: Data is not lost even at the time of server failure.

# DBMS

Database Management System is a software system that enables users to define, create, retrieve, maintain, update and control access to the database.

Database systems have high cost and require high end hardware configurations. DBMS is an interface between database application and database.

An Application Program interacts with a database by issuing an appropriate request (typically a SQL statement).

It ensures that our data is consistent, organized, and is easily accessible by serving as an interface between the database and its end-users or application software.

Functions of DBMS

* Data Management – Store, retrieve and modify data
* Integrity – Maintain accuracy of data
* Security – Access to authorized users only
* Concurrency – Simultaneous data access for multiple users
* Transaction – Modification to database must either be successful or must not happen at all
* Utilities – Data import/export, user management, backup, logging, performance analysis, recovery mechanism for data

There are two types of DBMS:

* Relational Database Management System, stores data in the form of a collection of tables, and relations can be defined between the common fields of these tables. Most modern database management systems like MySQL, Microsoft SQL Server, Oracle, IBM DB2, and Amazon Redshift are based on RDBMS.
* Non-Relational Database Management System, there is no concept of relations, tuples and attributes. Example – MongoDB

# SQL Comments

Used to clarify portions of SQL statements and to prevent SQL statements from being executed.

Single Line Comments: It starts with two consecutive hyphens (-)

Multi-line Comments: It starts with /\* and ends with \*/

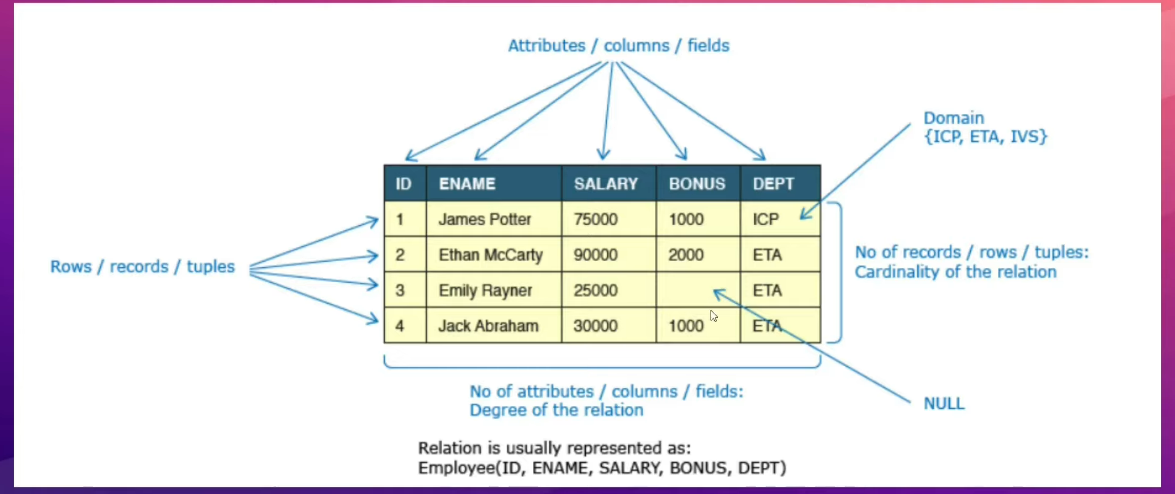
# Cursor

A database cursor is a control structure that allows for the traversal of records in a database. Cursors, facilitates processing after traversal, such as retrieval, addition, and deletion of database records. They can be viewed as a pointer to one row in a set of rows.

# Function of DBMS

# Types of Databases

1. Relational – Tabular structure (MySQL, Oracle, Microsoft Access, PostgreSQL)
2. NoSQL – Key value pairs (MongoDB)
3. Network – Graph structure
4. Hierarchal – Tree structure



# Data Integrity and Constraints

Data integrity refers to maintaining and assuring the accuracy and consistency of data over its entire life-cycle and is a critical aspect of the design, implementation, and usage of any system which stores, processes, or retrieves data.

Database systems ensure data integrity through constraints which are used to restrict data that can be entered or modified in the database. Database systems offer three types of integrity constraints.

|  |  |  |
| --- | --- | --- |
| Integrity Type | Definition | Enforced Through |
| Entity Integrity | Each table must have a column or a set of columns through which we can uniquely identify a row. These columns cannot have null values. | Primary Key |
| Domain Integrity | All attributes in a table must have a defined domain i.e., a finite set of values which have to be used. When we assign a data type to a column, we limit the values that it can contain. In addition, we can also have value restriction as per business rules e.g., gender must be M or F. | Data types, check constraint |
| Referential Integrity | Every value of a column in a table must exist as a value of another column in a different (or the same) table. | Foreign Key |

# Database Keys

A DBMS key is an attribute or set of attributes which helps us uniquely identify a row in a relation/table.

# Need for keys

1. To uniquely identify a row
2. To enforce data integrity and constraints
3. To establish relationship between tables

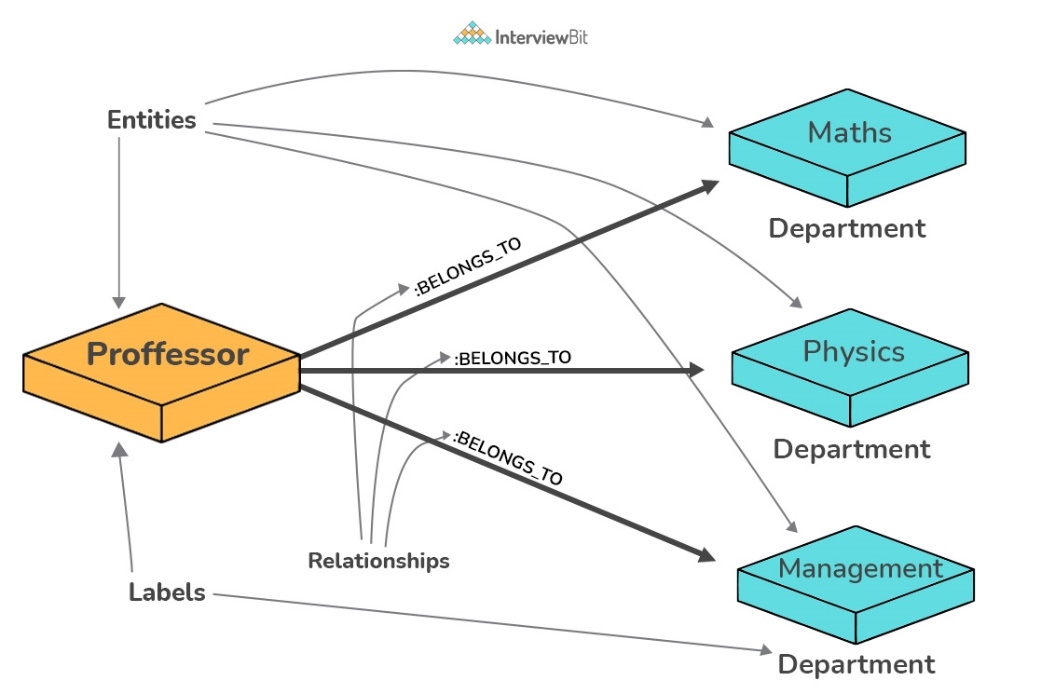
# Types of Keys

|  |  |
| --- | --- |
| Super Key | Foreign Key |
| Candidate Key | Composite Key |
| Primary Key | Compound Key |
| Alternate Key | Surrogate Key |

# Entities and Relationships

An entity can be a real-world object, either tangible or intangible, that can be easily identifiable. For example, in a college database, students, professors, workers, departments, and projects can be referred to as entities. Each entity has some associated properties that provide it an identity.

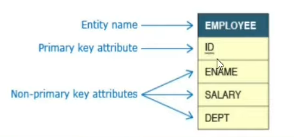
Relations or links between entities that have something to do with each other. For example, the employee’s table in a company’s database can be associated with the salary table in the same database.



# Entity Relationship Model (E-R Model)

ER model is a graphical representation of entities and their relationships which helps in understanding data independent of the actual database implementation.

|  |  |  |
| --- | --- | --- |
| Term | Definition | Examples |
| Entity | Real world objects, either tangible or intangible, which have an independent existence and about which we intend to collect data. | Employee, Computer |
| Attribute | A property that describes an entity. | Name, Salary |

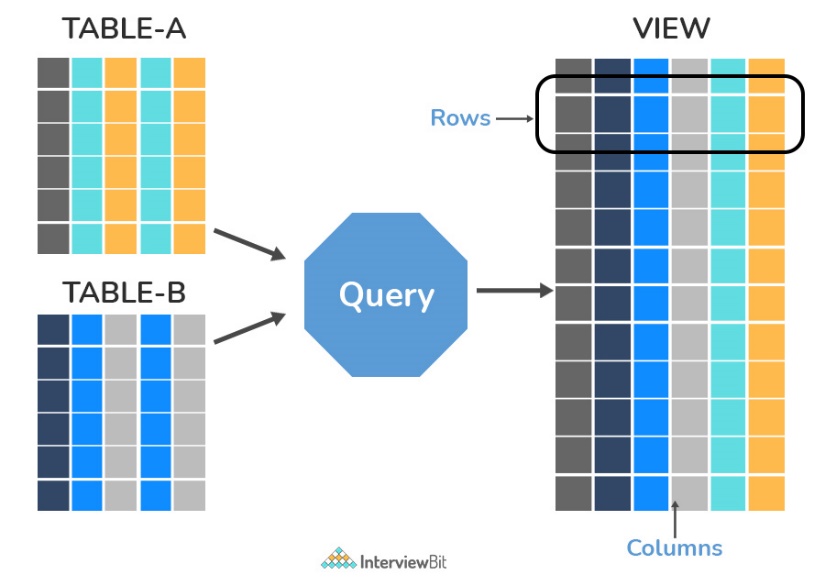


# Cardinality of relationships

Cardinality of relationships is the number of instances in one entity which is associated to the number of instances in another.

# View

A view is SQL is a virtual table based on the result-set of an SQL statement. A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.



# Normalization and Denormalization

Normalization represents the way of organizing structured data in the database efficiently. It includes the creation of tables, establishing relationships between them, and defining rules for those relationships. Inconsistency and redundancy can be kept in check based on these rules, hence, adding flexibility to the database.

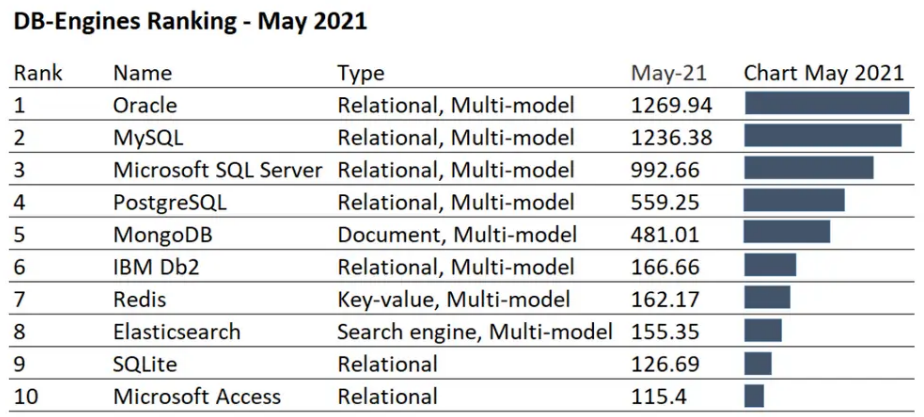
If the same information is repeated in multiple places in the database, there is a risk that it is updated in one place but not the other, leading to data corruption. By having a database with normalization errors, you open the risk of getting invalid or corrupt data into the database.

Denormalization is the inverse process of normalization, where the normalized schema is converted into a schema that has redundant information. The performance is improved by using redundancy and keeping the redundant data consistent. The reason for performing denormalization is the overheads produced in the query processor by an over-normalized structure.

# Prevalence of SQL

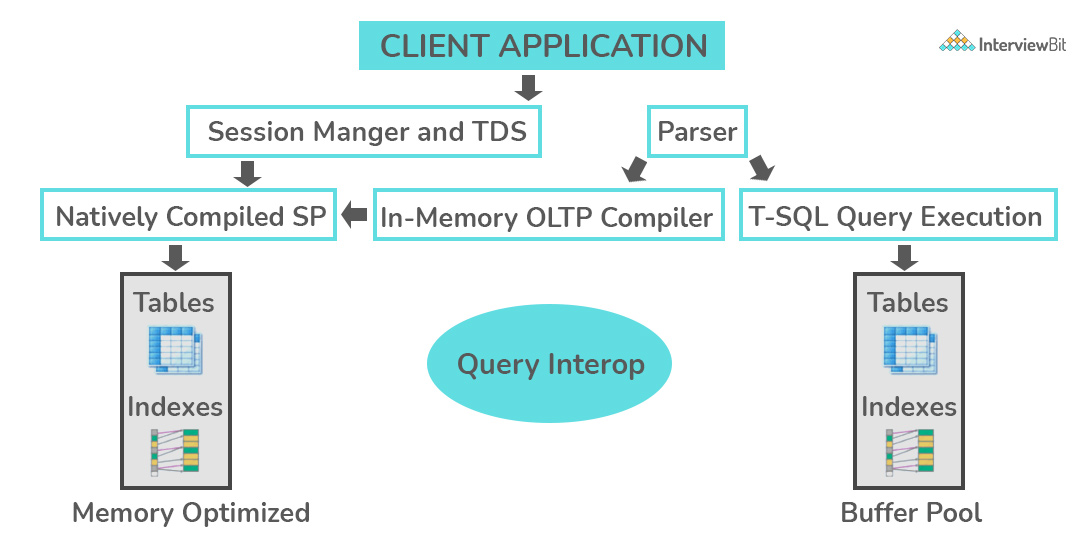
It is easy to visualize databases as large spreadsheets with millions and millions or rows and columns. SQL allows users to quickly manipulate these tables to access information and present the results in the most common formats, tables and associated graphs and visualizations. It is easy to use, syntax is easy to understand.

While No-SQL databases like MongoDB, Cassandra, etc have gained traction with the requirements of Big-Data and real time applications and increasing prevalence of unstructured data, SQL databases still hold the top seven positions of the top ten most popular database engines.



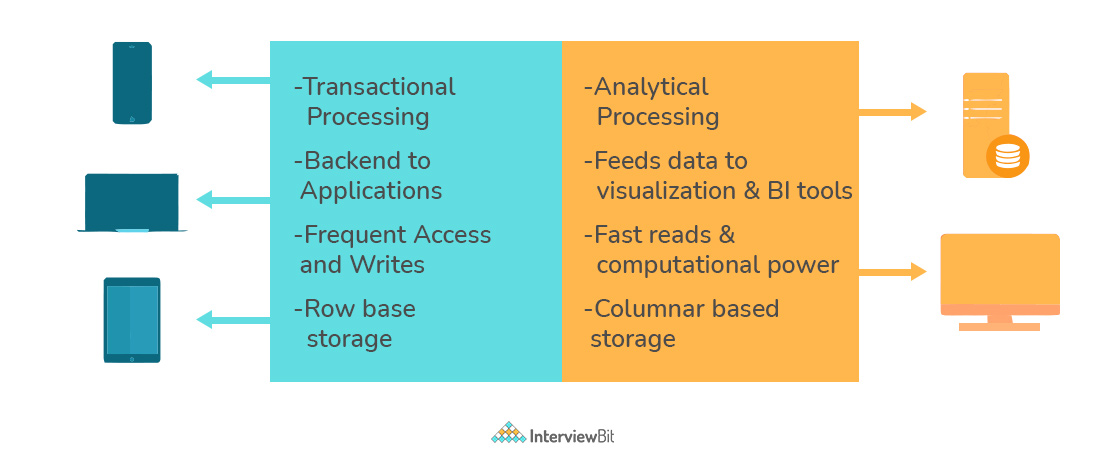
# OLTP

Online Transaction Processing, is a class of software applications capable of supporting transaction-oriented programs. An essential attribute of an OLTP system is its ability to maintain concurrency. To avoid single points of failure, OLTP systems are often decentralized. These systems are usually designed for a large number of users who conduct short transactions. Database queries are usually simple, require sub-second response times, and return relatively few records.



# OLAP

Online Analytical Processing, is a class of software programs that are characterized by the relatively low frequency of online transactions. Queries are often too complex and involve a bunch of aggregations. For OLAP systems, the effectiveness measure relies highly on response time. Such systems are widely used for data mining or maintain aggregated, historical data, usually in multi-dimensional schemas.

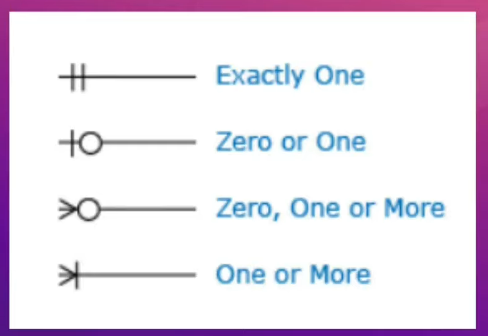


# Types of Relationships

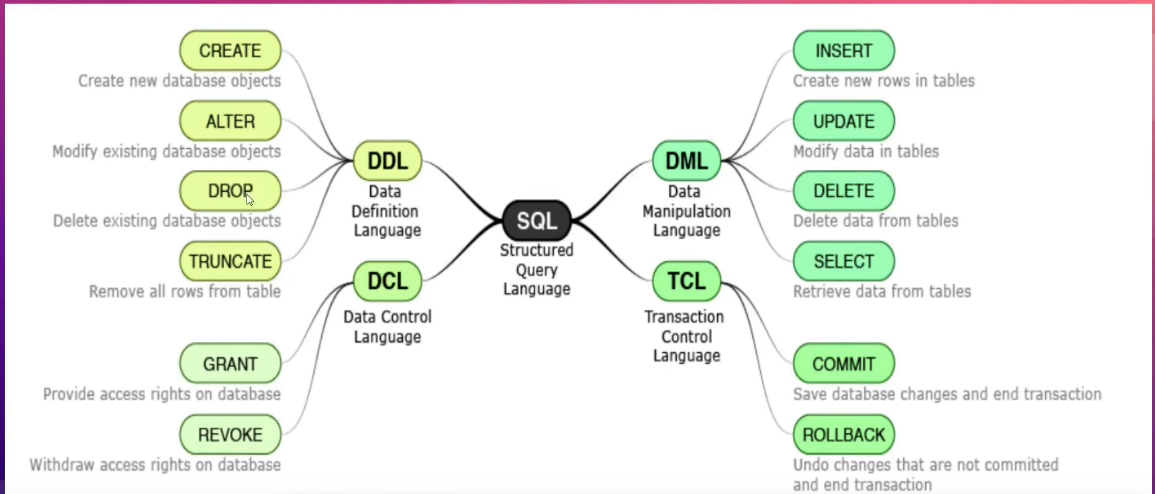
* One-to-One: This can be defined as the relationship between two tables where each record in one table is associated with the maximum of one record in the other table.
* One-to-Many & Many-to-One: Most commonly used relationship where a record in a table is associated with multiple records in the other table.
* Many-to-Many: Used in cases when multiple instances on both sides are needed for defining a relationship.
* Self-Referencing Relationships: Used when a table needs to define a relationship with itself.



# Crow-foot Notation



# Types of SQL Commands



# MySQL Data Types

In MySQL there are three main data types: string, numeric, date and time

String Data Types

|  |  |
| --- | --- |
| **Data type** | **Description** |
| CHAR(size) | A FIXED length string (can contain letters, numbers, and special characters). The size parameter specifies the column length in characters - can be from 0 to 255. Default is 1 |
| VARCHAR(size) | A VARIABLE length string (can contain letters, numbers, and special characters). The size parameter specifies the maximum column length in characters - can be from 0 to 65535 |
| BINARY(size) | Equal to CHAR(), but stores binary byte strings. The size parameter specifies the column length in bytes. Default is 1 |
| VARBINARY(size) | Equal to VARCHAR(), but stores binary byte strings. The size parameter specifies the maximum column length in bytes. |
| TINYBLOB | For BLOBs (Binary Large OBjects). Max length: 255 bytes |
| TINYTEXT | Holds a string with a maximum length of 255 characters |
| TEXT(size) | Holds a string with a maximum length of 65,535 bytes |
| BLOB(size) | For BLOBs (Binary Large OBjects). Holds up to 65,535 bytes of data |
| MEDIUMTEXT | Holds a string with a maximum length of 16,777,215 characters |
| MEDIUMBLOB | For BLOBs (Binary Large OBjects). Holds up to 16,777,215 bytes of data |
| LONGTEXT | Holds a string with a maximum length of 4,294,967,295 characters |
| LONGBLOB | For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data |
| ENUM(val1, val2, val3, ...) | A string object that can have only one value, chosen from a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. The values are sorted in the order you enter them |
| SET(val1, val2, val3, ...) | A string object that can have 0 or more values, chosen from a list of possible values. You can list up to 64 values in a SET list |

Numeric Data Types

|  |  |
| --- | --- |
| **Data type** | **Description** |
| BIT(size) | A bit-value type. The number of bits per value is specified in size. The size parameter can hold a value from 1 to 64. The default value for size is 1. |
| TINYINT(size) | A very small integer. Signed range is from -128 to 127. Unsigned range is from 0 to 255. The size parameter specifies the maximum display width (which is 255) |
| BOOL | Zero is considered as false, nonzero values are considered as true. |
| BOOLEAN | Equal to BOOL |
| SMALLINT(size) | A small integer. Signed range is from -32768 to 32767. Unsigned range is from 0 to 65535. The size parameter specifies the maximum display width (which is 255) |
| MEDIUMINT(size) | A medium integer. Signed range is from -8388608 to 8388607. Unsigned range is from 0 to 16777215. The size parameter specifies the maximum display width (which is 255) |
| INT(size) | A medium integer. Signed range is from -2147483648 to 2147483647. Unsigned range is from 0 to 4294967295. The size parameter specifies the maximum display width (which is 255) |
| INTEGER(size) | Equal to INT(size) |
| BIGINT(size) | A large integer. Signed range is from -9223372036854775808 to 9223372036854775807. Unsigned range is from 0 to 18446744073709551615. The size parameter specifies the maximum display width (which is 255) |
| FLOAT(size, d) | A floating point number. The total number of digits is specified in size. The number of digits after the decimal point is specified in the d parameter. This syntax is deprecated in MySQL 8.0.17, and it will be removed in future MySQL versions |
| FLOAT(p) | A floating point number. MySQL uses the p value to determine whether to use FLOAT or DOUBLE for the resulting data type. If p is from 0 to 24, the data type becomes FLOAT(). If p is from 25 to 53, the data type becomes DOUBLE() |
| DOUBLE(size, d) | A normal-size floating point number. The total number of digits is specified in size. The number of digits after the decimal point is specified in the d parameter |
| DOUBLE PRECISION(size, d) |  |
| DECIMAL(size, d) | An exact fixed-point number. The total number of digits is specified in size. The number of digits after the decimal point is specified in the d parameter. The maximum number for size is 65. The maximum number for d is 30. The default value for size is 10. The default value for d is 0. |
| DEC(size, d) | Equal to DECIMAL(size,d) |

Date and Time Data Types

|  |  |
| --- | --- |
| **Data type** | **Description** |
| DATE | A date. Format: YYYY-MM-DD. The supported range is from '1000-01-01' to '9999-12-31' |
| DATETIME(fsp) | A date and time combination. Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. Adding DEFAULT and ON UPDATE in the column definition to get automatic initialization and updating to the current date and time |
| TIMESTAMP(fsp) | A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC. Automatic initialization and updating to the current date and time can be specified using DEFAULT CURRENT\_TIMESTAMP and ON UPDATE CURRENT\_TIMESTAMP in the column definition |
| TIME(fsp) | A time. Format: hh:mm:ss. The supported range is from '-838:59:59' to '838:59:59' |
| YEAR | A year in four-digit format. Values allowed in four-digit format: 1901 to 2155, and 0000. MySQL 8.0 does not support year in two-digit format. |

# SQL Operators

Arithmetic Operators

|  |  |
| --- | --- |
| Operator | Description |
| + | Add |
| - | Subtract |
| \* | Multiply |
| / | Divide |
| % | Modulo |

Bitwise Operators

|  |  |
| --- | --- |
| Operator | Description |
| & | Bitwise AND |
| | | Bitwise OR |
| ^ | Bitwise exclusive OR |

Comparison Operators

|  |  |
| --- | --- |
| Operator | Description |
| = | Equal to |
| > | Greater than |
| < | Less than |
| >= | Greater than or equal to |
| <= | Less than or equal to |
| <> | Not equal to |

Compound Operators

|  |  |
| --- | --- |
| Operator | Description |
| += | Add equals |
| -= | Subtract equals |
| \*= | Multiply equals |
| /= | Divide equals |
| %= | Modulo equals |
| &= | Bitwise AND equals |
| ^-= | Bitwise exclusive equals |
| |\*= | Bitwise OR equals |

Logical Operators

|  |  |
| --- | --- |
| **Operator** | **Description** |
| ALL | TRUE if all of the subquery values meet the condition |
| AND | TRUE if all the conditions separated by AND is TRUE |
| ANY | TRUE if any of the subquery values meet the condition |
| BETWEEN | TRUE if the operand is within the range of comparisons |
| EXISTS | TRUE if the subquery returns one or more records |
| IN | TRUE if the operand is equal to one of a list of expressions |
| LIKE | TRUE if the operand matches a pattern |
| NOT | Displays a record if the condition(s) is NOT TRUE |
| OR | TRUE if any of the conditions separated by OR is TRUE |
| SOME | TRUE if any of the subquery values meet the condition |

# Comments

/\* This is a

Multiline comment

\*/

This is a single line comment

# DDL SQL Queries

1. CREATE DATABASE IF NOT EXISTS dbname
2. DROP DATABASE dbname
3. CREATE TABLE IF NOT EXISTS users(

col1 datatype,

col2 datatype,

email varchar(255) UNIQUE,

name varchar(255),

UNIQUE(name),

CONSTRAINT U\_email UNIQUE (email),

id integer NOT NULL PRIMARY KEY AUTO\_INCREMENT,

PRIMARY KEY (id, email),

age integer CHECK (age > 6 AND age < 25),

gender varchar(255) DEFAULT “Others”

journey\_date datetime DEFAULT CURRENT\_TIMESTAMP

)

1. DROP TABLE tablename
2. TRUNCATE TABLE tablename
3. CREATE TABLE orders(

user\_id integer,

FOREIGN KEY(user\_id) REFERNCES users(id)

)

1. ALTER TABLE students ADD COLUMN college varchar(255) NOT NULL
2. ALTER TABLE students DROP COLUMN age
3. ALTER TABLE students MODIFY COLUMN sname integer
4. ALTER TABLE students DROP CONSTRAINT U\_email
5. ALTER TABLE passenger ADD CONSTRAINT P\_email UNIQUE(email)
6. ALTER TABLE passenger DROP CONSTRAINT P\_email

# DML SQL Queries

1. INSERT INTO students (sid, sname, email, college) VALUES (1, “Nitish”, [abc@gmail.com](mailto:abc@gmail.com), “HIT”)
2. INSERT INTO students VALUES (5, “Rahul”, [abc@gmail.com](mailto:abc@gmail.com), “IEM”)
3. INSERT INTO students VALUES (6, “ewg”, “wrg”, “wrg”), (7, “wrgw”, “wrgw”, “wrgw”)
4. SELECT \* FROM tablename
5. SELECT colname1, colname2 FROM train
6. SELECT Name AS PassengerName, Sex AS Gender, Survived FROM train
7. SELECT Name, SibSp + Parch AS family FROM train
8. SELECT Name, Age + 102 AS CurrentAge FROM train
9. SELECT Name, 100000 AS Compensation FROM train
10. SELECT DISTINCT Sex FROM train
11. SELECT DISTINCT Pclass, Embarked FROM train
12. SELECT \* FROM train WHERE Survived = 0
13. SELECT \* FROM train WHERE Pclass = 3 AND Survived = 0
14. SELECT \* FROM train WHERE Age BETWEEN 10 AND 15
15. SELECT title FROM movies WHERE genre LIKE ‘Comedy’ OR genre LIKE ‘Action’
16. SELECT title, genre FROM movies WHERE genre IN (‘Action’, ‘Horror’, ‘Drama’)
17. SELECT title, genre FROM movies WHERE genre NOT IN (‘Action’, ‘Horror’, ‘Drama’)
18. SELECT title FROM movies WHERE title LIKE ‘A%’
19. SELECT title FROM table\_name WHERE title LIKE ‘%man%’
20. SELECT title FROM movies WHERE actor LIKE ‘%Khan%’ OR actor LIKE ‘%Kapoor%’
21. SELECT title FROM movies WHERE title LIKE ‘\_\_\_\_’
22. SELECT title FROM movies WHERE title LIKE ‘A\_\_\_’
23. UPDATE passenger SET name = ‘Rahul’’
24. UPDATE passenger SET name = ‘Rohit’ WHERE email LIKE ‘%gmail’
25. UPDATE passenger SET name = ‘Ankit’, email =’abc@gmail.com’ WHERE email LIKE ‘%yahoo%’
26. DELETE FROM passenger WHERE id = 1
27. DELETE FROM passenger WHERE id > 2 AND email LIKE ‘%yahoo%’
28. DELETE FROM passenger WHERE 1
29. SELECT title, ABS((india\_gross – budget)) AS profit FROM movies
30. SELECT title, ROUND((runtime/60)) AS runtime\_hrs FROM movies
31. SELECT title, ROUND((runtime/60), 1) AS runtime\_hrs FROM movies
32. SELECT title, CEIL((runtime/60)) AS runtime\_hrs FROM movies
33. SELECT title, FLOOR((runtime/60)) AS runtime\_hrs FROM movies
34. SELECT UPPER(title) FROM movies
35. SELECT title, CONCAT(actor, ‘ ‘, director) AS crew FROM movies
36. SELECT title, LENGTH(title) AS length FROM movies
37. SELECT title, SUBSTR(title, 1, 5) AS short FROM movies
38. SELECT MAX(budget) FROM movies
39. SELECT MIN(india\_gross) FROM movies
40. SELECT SUM(india\_gross) FROM movies
41. SELECT AVG (india\_gross) FROM movies
42. SELECT COUNT (\*) FROM movies
43. SELECT COUNT(DISTINCT(actor)) FROM movies
44. SELECT title, (worldwide\_gross – budget) AS profit FROM movies ORDER BY profit DESC LIMIT 5
45. SELECT title, (worldwide\_gross – budget) AS profit FROM movies ORDER BY profit ASC LIMIT 5
46. SELECT \* FROM movies ORDER BY genre, title DESC
47. SELECT actor, COUNT(\*) AS num\_movies FROM movies GROUP BY actor ORDER BY num\_movies DESC LIMIT 5
48. SELECT genre, SUM (worldwide\_gross – budget) AS total\_profit FROM movies GROUP BY genre ORDER BY total\_profit DESC LIMIT 5
49. SELECT director, AVG (worldwide\_gross – budget) AS avg\_profit FROM movies GROUP BY director ORDER BY avg\_profit DESC LIMIT 5
50. SELECT title, budget FROM movies GROUP BY title ORDER BY budget DESC LIMIT 5
51. SELECT actor, director, SUM (worldwide\_gross – budget) AS profit FROM movies GROUP BY actor, director ORDER BY profit DESC LIMIT 5
52. SELECT actor, SUM (worldwide\_gross – budget) AS total\_profit FROM movies GROUP BY actor ORDER BY total\_profit DESC LIMIT 10
53. SELECT actor, AVG (screens) AS opening

FROM movies

GROUP BY actor

HAVING opening > 1000

ORDER BY opening DESC

1. SELECT title, (worldwide\_gross - budget) profit,

CASE

WHEN (worldwide\_gross – budget) > 100000 THEN “SUPER HIT”

ELSE “FLOP”

END AS verdict

FROM movies

# SQL Joins

1. SELECT \* FROM users CROSS JOIN groups
2. SELECT \* FROM membership m JOIN users u ON m.uid = u.id

# What is

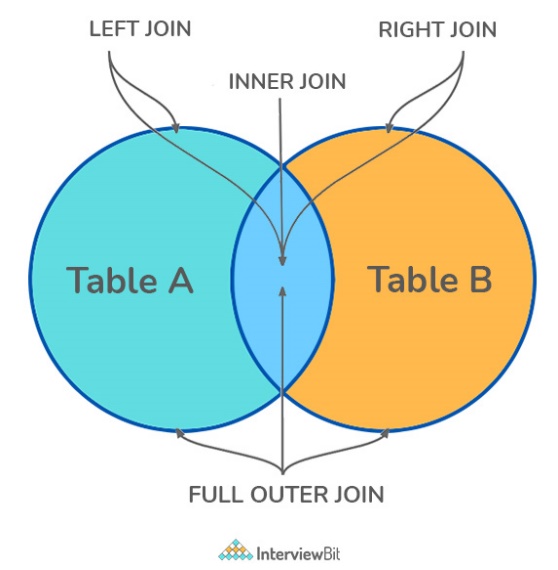
puts lines from queries after each other. removes duplicate records (where all columns in the results are the same). Example

23 bah

45 bah

# What is a

The SQL clause is used to combine records (rows) from two or more tables in a SQL database based on a related column between the two.



There are four different types of JOINs in SQL:

* : Retrieves records that have matching values in both tables involved in the join. This is the widely used join for queries.

Table\_A

Table\_B

Table\_A

Table\_B

* : Retrieves all the records/rows from the left and the matched records/rows from the right table.

Table\_A A

Table\_B B

A.col = B.col

* : Retrieves all the records/rows from the right and the matched records/rows from the left table.

Table\_A A

Table\_B B

A.col = B.col

* : Retrieves all the records where there is a match in either the left or right table.

Table\_A A

Table\_B B

A.col = B.col

# What is Self-Join

A self-JOIN is a case of regular join where a table is joined to itself based on some relation between its own columns. Self-join uses the or clause and a table alias is used to assign different names to the table within the query.

A.emp\_id “Emp\_ID”, A.emp\_name “Employee”,

B.emp\_id “Sup\_ID”, B.emp\_name “Supervisor”

employee A, employee B

A.emp\_sup = B.emp\_id

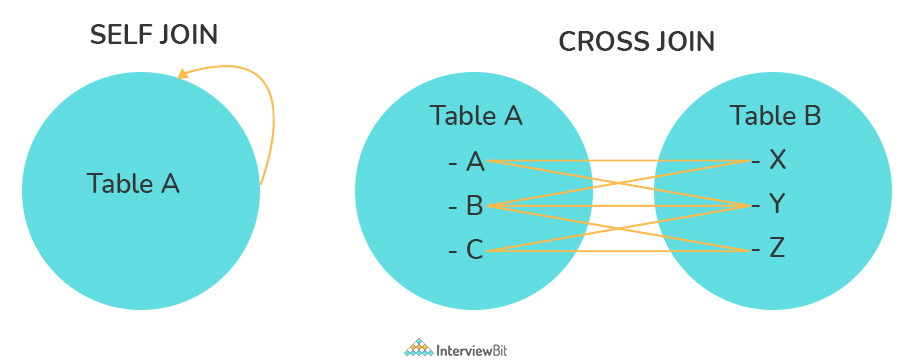
# What is a

Cross join can be defined as a cartesian product of the two tables included in the join. The table after join contains the same number of rows as in the cross-product of the number of rows in the two tables.

stu.name, sub.subject

students stu

subjects sub



# What is an Index

A database index is a data structure that provides a quick lookup of data in a column or columns of a table. It enhances the speed of operations accessing data from a database table at the cost of additional writes and memory to maintain the index data structure.

INDEX index\_name

table\_name (column\_1, column\_2)

INDEX index\_name

There are different types of indexes that can be created for different purposes:

* Unique and Non-unique Index

Unique indexes are indexes that help maintain data integrity by ensuring that no two rows of data in a table have identical key values. Once a unique index has been defined for a table, uniqueness is enforced whenever keys are added or changed within the index.

Non-unique indexes, are not used to enforce constraints on the tables with which they are associated. Instead, non-unique indexes are used solely to improve query performance by maintaining a sorted order of data values that are used frequently.

INDEX myIndex

students (enroll\_no)

* Clustered and Non-Clustered Index

Clustered indexes are those whose order of the rows in the database corresponds to the order of the rows in the index. This is why only one clustered index can exist in a given table, whereas, multiple non-clustered indexes can exist in the table.

The only difference between clustered and non-clustered indexes is that the database manager attempts to keep the data in the database in the same order as the corresponding keys appear in the clustered index.

Clustering indexes can improve the performance of most query operations because they provide a linear-access path to data stored in the database.

# What is a Query

It is a request for data or information from a database table or combination of tables. A database query can be either a select query or an action query.

fname, lname /\* select query \*/

myDb.students

student\_id = 1

myDB.students /\* action query \*/

fname = ‘Captain’, lname = ‘America’

student\_id = 1

# What is a Subquery

A subquery is a query within another query, also known as nested query or inner query. It is used to restrict or enhance the data to be queried by the main query, thus restricting or enhancing the output of the main query respectively. For example, here we fetch the contact information for students who have enrolled for the maths subject:

name, email, mob, address

myDb.contacts

roll\_no ( roll\_no myDb.students subject = ‘Maths’)

There are two types of subqueries – Correlated and Non-Correlated

* A correlated subquery cannot be considered as an independent query, but it can refer to the column in a table listed in the of the main query.
* A non-correlated subquery can be considered as an independent query and the output of the subquery is substituted in the main query.

# What is the statement

operator in SQL is used to select data from a database. The data returned is stored in a result table, called the result-set.

myDB.students

Some common SQL clauses used in conjunction with a query are as follows:

* clause in SQL is used to filter records that are necessary, based on specific conditions
* clause in SQL is used to sort the records based on some fields in ascending () or descending order ()
* clause is used to group records with identical data and can be used in conjunction with some aggregation functions to produce summarized results from the database.
* clause is used to filter records in combination with the . It is different from , since the clause cannot filter aggregated records

myDB.students

graduation\_year = 2019

studentID

COUNT(studentID), country

myDB.students

country != “INDIA”

country

COUNT(studentID) > 5

# What are , and commands

The operator combines and returns the result-set retrieved by two or more statements.

The operator is used to remove duplicates from the result-set.

The clause in SQL combines the result-set fetched by the two statements where records from one match the other and then returns this intersection of result-sets.

name Students

name Contacts

name Students

name Contacts

name Students

name Contacts

name Students

name Contacts

# What is an Alias in SQL

It is a temporary name assigned to the table or table column for the purpose of a particular SQL query. Aliasing can be employed as an obfuscation technique to secure the real names of database fields. A table alias is also called a correlation name.

An alias is represented explicitly by the AS keyword.

A.emp\_name “Employee”

B.emp\_name “Supervisor”

employee A, employee B

A.emp\_sup = B.emp\_id

# What are DELETE, TRUCATE and DROP statements

statement is used to delete rows from a table based on the condition given in the clause or deletes all the rows from the table if no condition is specified. But it does not free the space containing the table.

command is used to delete all the rows from the table and free the space containing the table.

command is used to remove an object from the database. IF you drop a table, all the rows in the table are deleted and the table structure is removed from the database.

Candidates

CandidateID > 1000

Candidates

Candidates

# What is User-defined function

The user-defined functions in SQL are functions that accept parameters, perform complex calculations, and return a value. There are two types of SQL user-defined functions:

* Scalar Function: Return a single scalar value based on the input value.

– Calculates the total length of the given field (column)

– Converts a collection of string values to uppercase characters

– Converts a collection of string values to uppercase characters

– Extracts substrings from a collection of string values in a table

– Concatenates two or more strings

– Generates a random collection of numbers of a given length

– Calculates the round-off integer value for a numeric field (or decimal point value)

– Returns the current date and time

– Sets the format to display a collection of values

* Table-Valued Function: Return a table as output

Inline: returns a table data type based on a single statement.

Multi-statement: returns a tabular result-set but, unlike inline, multiple statements can be used inside the function body.

# What is Aggregate function

An aggregate function performs operations on a collection of values to return a single scalar value. Aggregate funcitons are often used with and clauses of the statement. Following is the widely used SQL aggregate functions

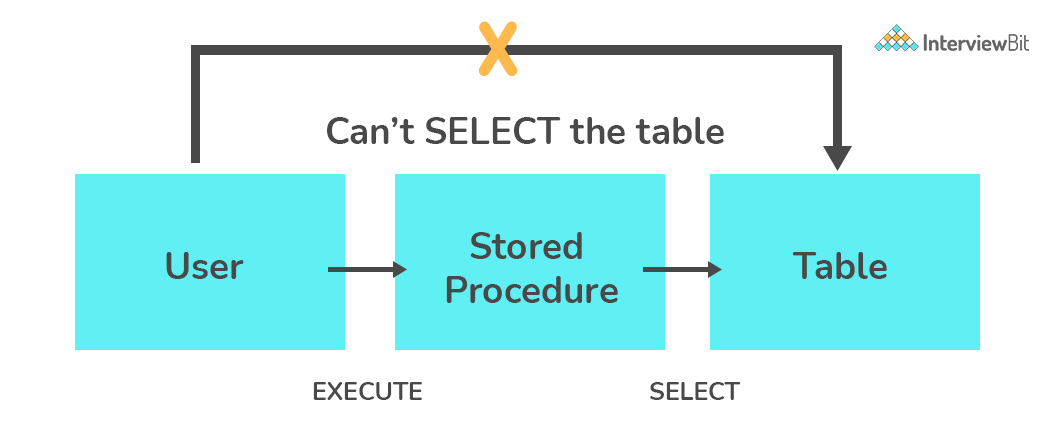
* – Calculates the mean of a collection of values
* – Counts the total number of records in a specific table or view
* – Calculates the minimum of a collection of values
* – Calculates the maximum of a collection of values
* – Fetches the first element in a collection of values
* – Fetches the last element in a collection of values

# What is a Stored Procedure

A stored procedure is a subroutine available to applications that access a RDBMS. Such procedures are stored in the database data dictionary. Disadvantage of stored procedure is that it can be executed nowhere except in the database and occupies who more memory in the database server. It also provides a sense of security and functionality as users who can’t access the data directly can be granted access via stored procedures.

FetchAllStudents()

myDB.students



# What is a Recursive Stored Procedure

A stored procedure that calls itself until a boundary condition is reached, is called a recursive stored procedure. This recursive function helps the programmers to deploy the same set of code several times as and when required. Some SQL programming languages limit the recursion depth to prevent an infinite loop of procedure calls from causing a stack overflow, which slows down the system and may lead to system crashes.

# What is Pattern Matching in SQL

SQL pattern matching provides for pattern search in data. This SQL query uses wildcards to match a string pattern, rather than writing the exact word. The operator is used in conjunction with SQL Wildcards to fetch the required information.

* Using a % wildcard to perform a simple search

The % wildcard matches zero or more characters of any type and can be used to define wildcards both before and after the pattern.

Example: Search a student in database with the first name beginning with the letter K

students

first\_name ‘K%’

* Omitting the patterns using the NOT keyword

Use the keyword to select records that don’t match the pattern.

Example: This query returns all students whose first name does not begin with K

students

first\_name ‘K%’

* Matching a pattern anywhere using the % wildcard twice

Example: Search for a student in database where he/she has a K in his/her first name

students

first\_name ‘%Q%’

* Using the \_ wildcard to match the pattern at a specific position

The \_ wildcard matches exactly one character of any type. It can be used in conjunction with % wildcard. This query fetches all students with letter K at the third position in their first name

students

first\_name ‘\_\_K%’

* Matching patterns for a specific length

The \_ wildcard plays an important role as a limitation when it matches exactly one character. It limits the length and position of the matched results.

/\* Matches first names with three or more letters \*/

students

first\_name ‘\_\_%’

students

first\_name ‘\_\_’

# What is the difference between DDL, DML, DCL, and TCL

These represents four categories into which the SQL commands have been separated. The following are the four significant subsets of the SQL.

* Data Definition Language (DDL) – Involves SQL commands used to define data structures – CREATE, ALTER, TRUNCATE, and DROP
* Data Manipulation Language (DML) – Involves SQL commands used to manipulate data – SELECT, INSERT, UPDATE, DELETE.
* Data Control Language (DCL) – Involves SQL commands used commonly by database administrators (DBAs) to manage permissions – GRANT, REVOKE

GRANT: It enables system administrators to assign privileges and roles to the specific user accounts to perform specific tasks on the database.

REVOKE: It enables system administrators to revoke privileges and roles from the user accounts so that they cannot use the previously assigned permission on the database.

* Transaction Control Language (TCL) – To ensure the transactions that occur in the database to happen in such a way that minimizes the danger of suffering from data loss. The commands in this category are COMMIT, ROLLBACK, SET TRANSACTION, SAVEPOINT, etc.

# References

* Mlstack.cafe

Data Structures

# **What is data?**

Data is the collection of different numbers, symbols, and alphabets to represent information.

# **What is data structure?**

Group of data elements that provides the easiest way to store, organize and perform different actions on the data of the computer so it can be used effectively. The idea is to reduce the space and time complexities of different tasks.

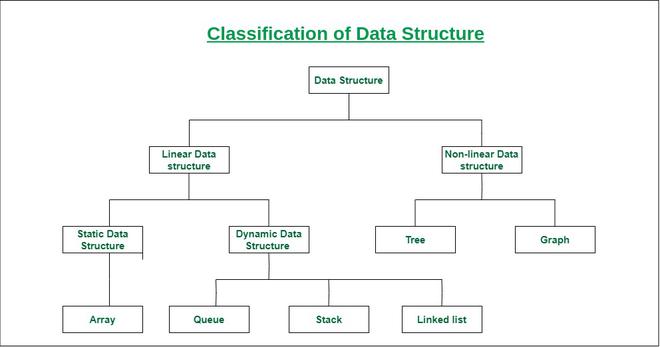
The choice of a good data structure makes it possible to perform a variety of critical operations effectively. An efficient data structure also uses minimum memory space and execution time to process the structure.

**Linear Data Structure** - Elements are arranged in one dimension. Example: lists, stack, queue.

**Non-Linear Data Structure** – Elements are arranged in one-many, many-one, many-many dimensions. Example: tree, graph, table, etc.

**Dynamic data structure** - Dynamic data structures have the feature where they expand and contract as a program runs. It provides a very flexible method of data manipulation.

Fields where data structures are used – Operating system, Graphics, Computer Design, Blockchain, Genetics, Image Processing, Simulation etc.



# Can a class inherit the constructor of its base class?

Whenever a child class extends parent class, the subclass inherits state and behaviour in the form of variables and methods from its superclass but it does not inherit constructor of super class.

# What are Arrays?

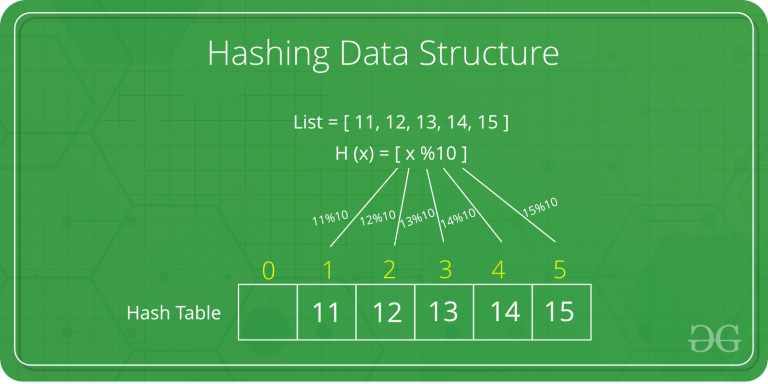
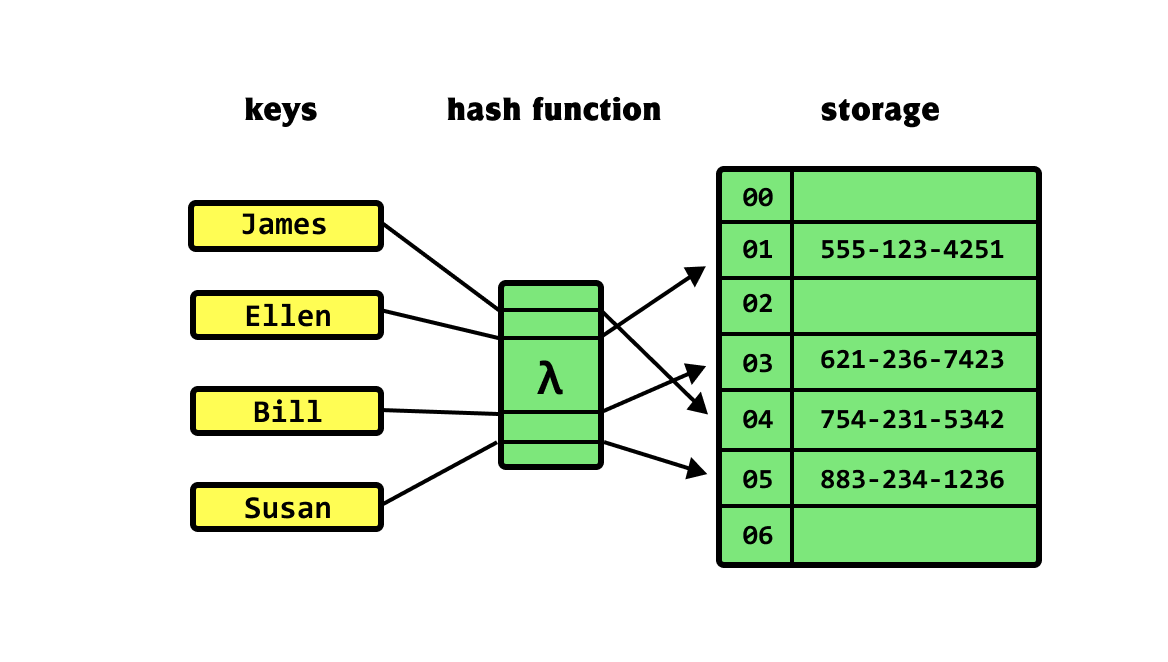
A collection of data items stored at contiguous memory locations. The idea is to store multiple items of the same type together. This makes it easier to calculate the position of each element by simply adding an offset to a base value, i.e., the memory location of the first element of the array (generally denoted by the name of the array).

**Real-Life applications of array:**

* An array is frequently used to store data for mathematical computations.
* It is used in image processing.
* It is also used in record management.
* Book pages are also real-life examples of an array.
* It is used to ordering boxes also.

# Hash tables and hashing

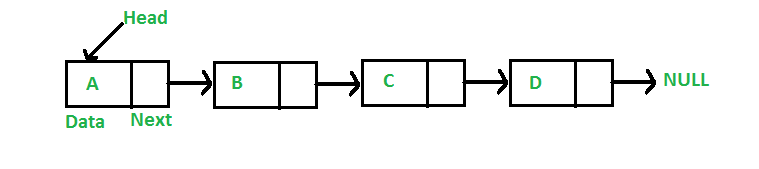
Hash tables are the data structures that are used to store **key-value pairs** in a structured way. The hashing function is used by a hash table to compute an index that contains all of the details regarding the keys that are mapped to their associated values.

Hashing is designed to use a special function called the **Hash function** which is used to map a given value with a particular key for faster access of elements. The efficiency of mapping depends on the efficiency of the hash function used. Let a hash function H(x) maps the value x at the index x%10 in an array. For example, if the list of values is [11, 12, 13, 14, 15] it will be stored at positions {1, 2, 3, 4, 5} in the array or Hash table respectively.

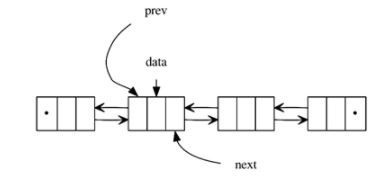
# What are Linked List?

Linked List is a linear data structure. Linked list elements are not stored at a contiguous location; the elements are linked using pointers.

Types of Linked List

* Singly-linked list
* Doubly-linked list - Doubly linked lists are categorized as a special type of linked list in which traversal across the data elements can be done in both directions.
* Circular-linked list
* Doubly circular linked list

Real-Life applications of a Linked list:

* A linked list is used in Round-robin scheduling, to keep track of the turn in multi-player games.
* It is used in image viewer. Previous and the next images are linked, hence can be accessed by the next and previous buttons.
* In the Music playlist, songs are linked to the previous and next songs.

# What are Stacks?

Stack is a linear data structure which follows a particular order in which the operations are performed. The order may be LIFO (Last in First Out) or FILO (First in Last Out). In stack, all insertion and deletion are permitted at only one end of the list.

LIFO is Last in First Out. It is a way of accessing, storing and retrieving data. It extracts the data that was stored last first.

Basic operations performed in the stack:

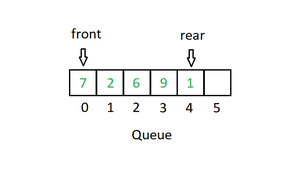
* Initialize: Make a stack empty.
* Push**:** Adds an item in the stack. If the stack is full, then it is said to be an Overflow condition.
* Pop**:** Removes an item from the stack. The items are popped in the reversed order in which they are pushed. If the stack is empty, then it is said to be an Underflow condition.
* Peek or Top**:** Returns top element of the stack.
* isEmpty**:** Returns true if the stack is empty, else false.

Real life applications of Stack

* Browsers uses stack data structure to keep track of previously visited sites.
* Call log in mobile also uses stack data structure.

# What are Queues?

Queue is a linear structure which follows a particular order in which the operations are performed. The order is First in First Out (FIFO). In the queue, items are inserted at one end and deleted from the other end. A good example of the queue is any queue of consumers for a resource where the consumer that came first is served first. The difference between stacks and queues is in removing. In a stack we remove the item the most recently added; in a queue, we remove the item the least recently added.

Basic operations are performed on queue:

* Enqueue: Adds an item to the queue. If the queue is full, then it is said to be an overflow condition.
* Dequeue: Removes an item from the queue. The items are popped in the same order in which they are pushed. If the queue is empty, then it is said to be an Underflow condition.
* Front: Get the front item from the queue.
* Rear: Get the last item from the queue.

Real life applications of queue

* Single lane one way road, where the vehicle enters first will exit first.
* The queue at the ticket windows.
* Cashier in a store is also example of queue.
* People on an escalator.

# What is Tree?

A tree is non-linear and hierarchal data arranged in a tree-like structure. In the tree, the topmost node is called the root node. Each node contains some data, and data can be of any type. It consists of a central node, structural nodes, and sub-nodes which are connected via edges. Different tree data structures allow quicker and easier access to the data as it is a non-linear data structure. the tree has various terminologies like Node, Root, Edge, Height of the tree, Degree of a tree, etc.

Tree is an efficient information retrieval data structure. Using Tree, search complexities can be brought to an optimal limit (key length). If we store keys in the binary search tree, a well-balanced BST will need time proportional to M \* log N, where M is maximum string length and N is the number of keys in the tree. Using Trie, we can search the key in O(M) time. However, the penalty is on Trie storage requirements.

The process of visiting all the nodes of a tree is known as tree traversal. Some of the algorithms to traverse a binary tree are as follows:

* Pre-order Traversal
* In order Traversal
* Post order Traversal
* Breadth First Search
* Zigzag Traversal

There are different types of Trees

* Binary Tree
* Binary Search Tree
* AVL Tree
* B-Tree

Real life applications of Tree

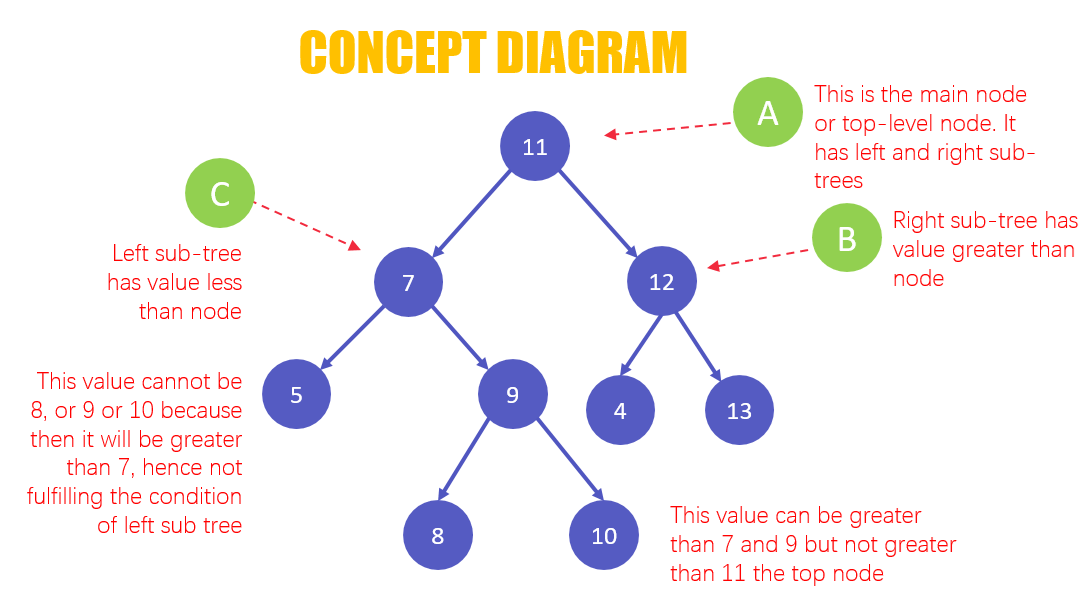
* Helps in Game Development.
* Helps in indexing of database.
* Efficient machine learning tool, commonly used in decision analysis.
* Tree helps to understand the data using flow chart.
* Domain name server also uses tree data structure.
* Use case of a tree is in any social networking site

**Binary Tree**: Trees are hierarchical data structures. A binary tree is a tree data structure in which each node has at most two children, which are referred to as the left child and the right child. It is implemented mainly using Links. A Binary Tree is represented by a pointer to the topmost node in the tree. If the tree is empty, then the value of root is NULL.

* Data
* Pointer to left child
* Pointer to right child

**Binary Search Tree**: In Binary Search Tree is a Binary Tree with the following additional properties.

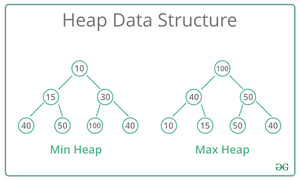
* The left subtree of a node contains only nodes with keys less than the node’s key.
* The right subtree of a node contains only nodes with keys greater than the node’s key.
* The left and right subtree each must also be a binary search tree.



# What is heap?

Heap is a special tree-based data structure in which the tree is a **complete binary tree**. Heaps can be of two types:

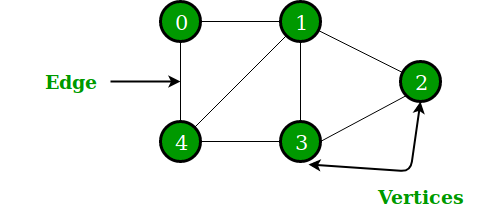
* Max-Heap**:** In a Max-Heap the key present at the root node must be greatest among the keys present at all of its children. The same property must be recursively true for all sub-trees in that Binary Tree.
* Min-Heap**:** In a Min-Heap the key present at the root node must be minimum among the keys present at all of its children. The same property must be recursively true for all sub-trees in that Binary Tree.



# What is Graph?

A graph is a non-linear data structure that consists of nodes and edges. A graph consists of finite sets of vertices and sets of edges that connect a pair of nodes. The graph is used to solve the most challenging and complex programming problems. The graph has different terminologies which are Path, Degree, Adjacent vertices, Connected components, etc.

**Real-Life Applications of Graph:**

* Google map where cities are located as vertices and path is located as edges of the graph.
* Social Network is also one real-world example of a graph where every person on the network is a node, and all the friendships on Instagram form the edges of the graph.
* A graph is also used to study molecules in physics and chemistry.

ALGORTIHMS

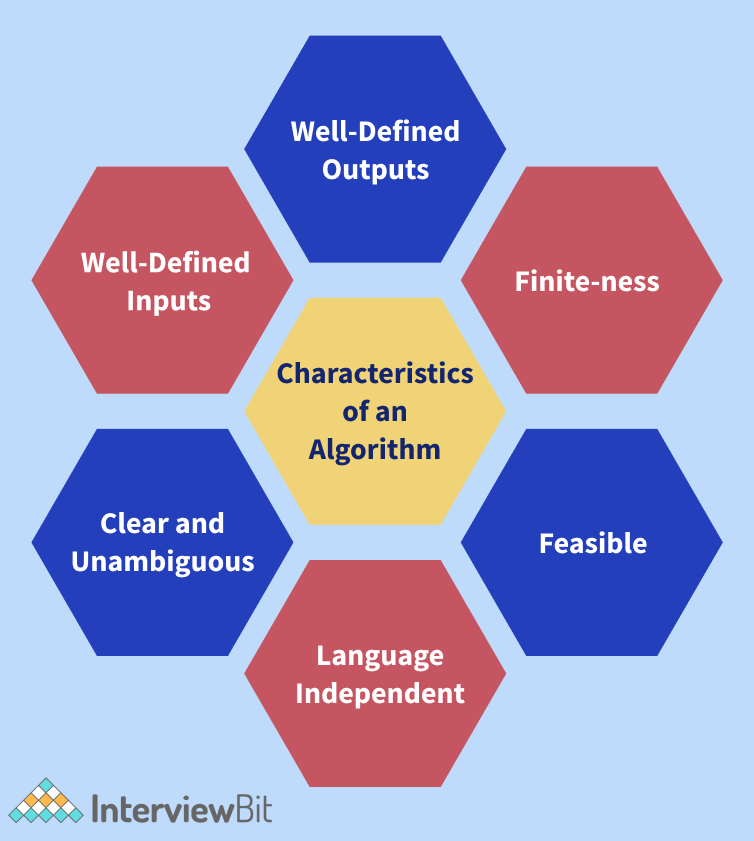
# What is an Algorithm?

An algorithm is a finite sequence of well-defined instructions used to solve a class of problems or conduct a computation in mathematics and computer science. Algorithms are used to specify how calculations, data processing, automated reasoning, automated decision making, and other tasks should be done.

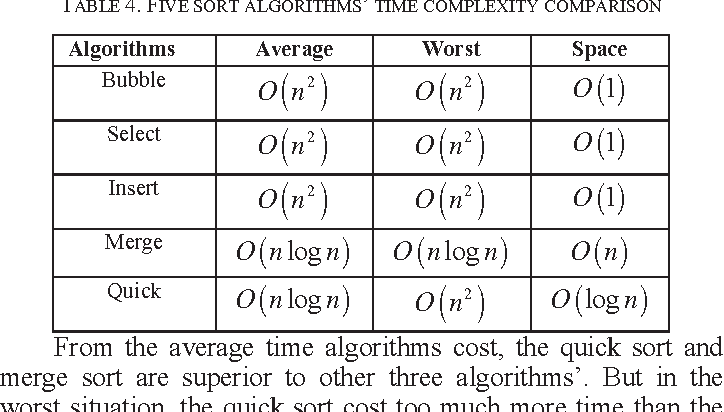
# Need for algorithms?

* Algorithms boost the effectiveness of an existing method.
* It is easy to compare an algorithm’s performance to those of other approaches using various methods (Time Complexity, Space Complexity, etc)
* Algorithms provide the designers with a detailed description of the criteria and goals of the problems.
* They enable a reasonable comprehension of the program’s flow.
* Algorithms evaluate how well the approaches work in various scenarios (Best cases, worst cases, average cases).
* Algorithm also determines which resources (I/O, memory) cycles are necessary.
* We can quantify and assess the problem’s complexity in terms of time and space using an algorithm.
* The cost of design is also reduced if proper algorithms are used.

# Best Case, Worst Case and Average Case of an algorithm?

The mathematical foundation/framing of an algorithm's run time performance is defined by asymptotic analysis. We can determine the best case, average case, and worst-case scenarios of an algorithm using asymptotic analysis.

* **Best Case Scenario of an Algorithm**: The data arrangement in which the algorithm performs the best. Take a binary search, for example, where the best-case scenario is if the target value is in the very centre of the data we are looking for. The best-case scenario for binary search would have a time complexity of O (1) or constant time complexity.
* **Worst Case Scenario of an Algorithm**: The worst collection of input for a given algorithm is referred to as the worst-case scenario of an Algorithm. For example, quicksort can perform poorly if the pivot value is set to the largest or smallest element of a sub-list. Quicksort will degenerate into an algorithm with a time complexity of O(n2), where n is the size of the list to be sorted.
* **Average Case Scenario of an Algorithm**:The amount of some computational resource (usually time) used by the process, averaged over all possible inputs, according to computational complexity theory. For example, the average-case complexity of the randomised quicksort algorithm is O(n\*log(n)), where n is the size of the list to be sorted.



# What are Asymptotic Notations?

Asymptotic analysis is a technique that is used for determining the efficiency of an algorithm that does not rely on machine-specific constants and avoids the algorithm from comparing itself to the time-consuming approach. For asymptotic analysis, asymptotic notation is a mathematical technique that is used to indicate the temporal complexity of algorithms.

The following are the three most common asymptotic notations.

* **Big Theta Notation (θ Notation):** The exact asymptotic behaviour is defined using the theta (θ) Notation. It binds functions from above and below to define behaviour. Dropping low order terms and ignoring leading constants is a convenient approach to get Theta notation for an expression.
* **Big O Notation:** It defines an upper bound for an algorithm by bounding a function from above. Consider the situation of insertion sort: in the best-case scenario, it takes linear time, and in the worst case, it takes quadratic time. Insertion sort has a time complexity O(n^2). It is useful when we just have an upper constraint on an algorithm's time complexity.
* **Big Omega (Ω) Notation:** It provides an asymptotic lower bound on a function, just like Big O notation does. It is useful when we have a lower bound on an algorithm's time complexity.

# Divide and Conquer Algorithm Paradigm?

Divide and conquer is an algorithm paradigm, not an algorithm itself. It is set up in such a way that it can handle a large amount of data, split it down into smaller chunks, and determine the solution to the problem for each of the smaller chunks. It combines all of the piecewise solutions of the smaller chunks to form a single global solution. This is known as the divide and conquer technique. The Divide and Conquer algorithmic paradigm employ the steps given below:

* Divide: The algorithm separates the original problem into a set of subproblems in this step.
* Conquer: The algorithm solves each subproblem individually in this step.
* Combine: In this step, the algorithm combines the solutions to the subproblems to obtain the overall solution.

Some of the algorithms which use the Divide and Conquer Algorithmic paradigm are as follows:

* Binary Search
* Merge Sort
* Strassen's Matrix Multiplication
* Quick Sort
* Closest pair of points

# What is Greedy Algorithm?

A greedy algorithm is an algorithmic method that aims to choose the best optimal decision at each sub-step, eventually leading to a globally optimal solution. This means that the algorithm chooses the best answer available at the time, regardless of the consequences. In other words, when looking for an answer, an algorithm always selects the best immediate, or local, option. Greedy algorithms may identify less than perfect answers for some cases of other problems while finding the overall, ideal solution for some idealistic problems.

The Greedy algorithm is used in the following algorithms to find their solutions:

* Prim's Minimal Spanning Tree Algorithm
* Kruskal's Minimal Spanning Tree Algorithm
* Travelling Salesman Problem
* Fractional Knapsack Problem
* Dijkstra's Algorithm
* Job Scheduling Problem
* Graph Map Colouring
* Graph Vertex Cover

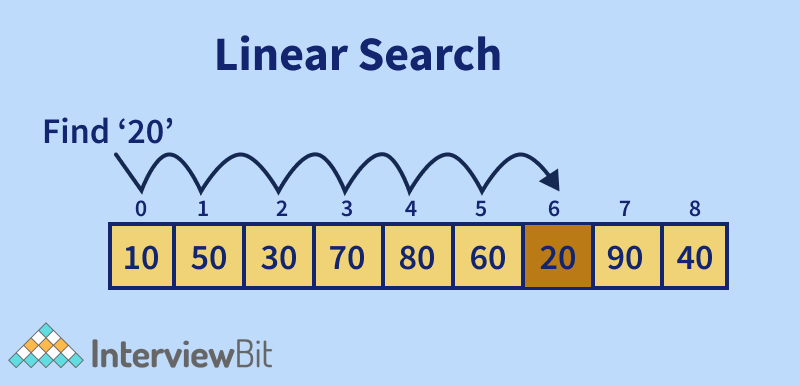
# What is Searching Algorithm?

Searching Algorithms are used to look for an element or get it from a data structure (usually a list of elements). These algorithms are divided into two categories based on the type of search operation:

* Sequential Search: This method traverses the list of elements consecutively, checking each element and reporting if the element to be searched is found. E.g., Linear Search.
* Interval Search: These algorithms were created specifically for searching sorted data structures. Because they continually target the centre of the search structure and divide the search space in half, these types of search algorithms are far more efficient than Sequential Search algorithms. E.g., Binary Search.

# Describe Linear Search Algorithm?

To find an element in a group of elements, the linear search can be used. It works by traversing the list of elements from the beginning to the end and inspecting the properties of all the elements encountered along the way. Let us consider the case of an array containing some integer elements. We want to find out and print all of the elements' positions that match a particular value (also known as the "key" for the linear search). The linear search works in a flow here, matching each element with the number from the beginning to the end of the list, and then printing the element's location if the element at that position is equal to the key. The time complexity of the Linear Search Algorithm is O(n) where n is the size of the list of elements and its space complexity is constant, that is, O (1).

Given below is an algorithm describing Linear Search:

* Step 1: Using a loop, traverse the list of elements given.
* Step 2: In each iteration, compare the target value (or key-value) to the list's current value.
* Step 3: If the values match, print the array's current index.
* Step 4: Move on to the next array element if the values do not match.
* Step 5: Repeat Steps 1 to 4 till the end of the list of elements is reached.

# Describe Binary Search Algorithm?

To apply binary search on a list of elements, the prerequisite is that the list of elements should be sorted. It is based on the Divide and Conquers Algorithmic paradigm. In the Binary Search Algorithm, we divide the search interval in half periodically to search the sorted list. We begin by creating an interval that spans the entire list. If the search key's value is less than the item in the interval's midpoint, the interval should be narrowed to the lower half. Otherwise, we limit it to the upper half of the page. We check for the value until it is discovered or the interval is empty. Given below is an algorithm describing Binary Search: (Let us assume that the element to be searched is x and the array of elements is sorted in ascending order)

* Step 1: x should be firstly compared to the middle element.
* Step 2: We return the middle element's index if x matches the middle element.
* Step 3: Else If x is greater than the middle element, x can only be found after the middle element in the right half subarray since the array is sorted in the ascending order. As a result, we repeat the process for the right half.
* Step 4: Otherwise, we repeat for the left half (x is smaller).
* Step 5: If the interval is empty, we terminate the binary search.

The time complexity of the Binary Search Algorithm is O(log(n)) where n is the size of the list of elements and its space complexity is constant, O (1).

# What is string matching algorithm?

* Naïve String-Matching Algorithm
* Rabin Karp Algorithm
* Knuth Morris Pratt Algorithm
* Boyer Moore Algorithm

# What is Dynamic Programming (DP)?

DP is primarily a recursion optimization. We can use DP to optimise any recursive solution that involves repeated calls for the same inputs.

The goal is to simply save the results of subproblems so that we do not have to recalculate them later.

The time complexity of this simple optimization is reduced from exponential to polynomial. For example, if we create a simple recursive solution for Fibonacci Numbers, the time complexity is exponential, but if we optimise it by storing subproblem answers using DP, the time complexity is linear.

# What is BFS (Breadth First Search) algorithm?

BFS is a graph traversal technique. It begins by traversing the graph from the root node and explores all the nodes in the immediate vicinity. It chooses the closest node and then visits all of the nodes that have yet to be visited. Until it reaches the objective node, the algorithm repeats the same method for each of the closest nodes.

* Set status = 1 as the first step for all the nodes (ready state).
* Set the status of the initial node A to 2, that is, waiting state.
* Repeat steps 4 and 5 until the queue is not empty.
* Dequeue and process node N from the queue, setting its status to 3, that is, the processed state.
* Put all of N's neighbours in the ready state (status = 1) in the queue and set their status to 2 (waiting state)
* Exit

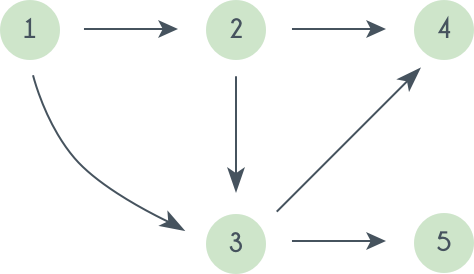
# What is DFS (Depth First Search) Algorithm?

DFS is a technique for traversing or exploring data structures such as trees and graphs. The algorithm starts at the root node (in the case of a graph, any random node can be used as the root node) and examines each branch as far as feasible before retracing. So, the basic idea is to start at the root or any arbitrary node and mark it, then advance to the next unmarked node and repeat until there are no more unmarked nodes. After that, go back and check for any more unmarked nodes to cross. Finally, print the path's nodes.

* Step1: Create a recursive function that takes the node's index and a visited array as input.
* Step 2: Make the current node a visited node and print it.
* Step 3: Call the recursive function with the index of the adjacent node after traversing all nearby and unmarked nodes

# What is Topological sorting?

Topological sorting of vertices of a Directed Acyclic Graph is an ordering of the vertices in such a way, that if there is an edge directed towards vertex v from vertex u, then u comes before v.



A topological sorting of this graph is: 1 2 3 4 5.

There is multiple topological sorting possible for a graph. In order to have a topological sorting the graph must not contain any cycles.

# Explain working of encryption algorithm?

The process of transforming plaintext into a secret code format known as "Ciphertext'' is known as encryption. For calculations, this technique uses a string of bits known as "keys" to convert the text. The larger the key, the more potential patterns for producing ciphertext there are. Most encryption algorithms use fixed blocks of input with lengths ranging from 64 to 128 bits, while others use the stream technique. A few of the most widely used cryptographic algorithms are as follows:

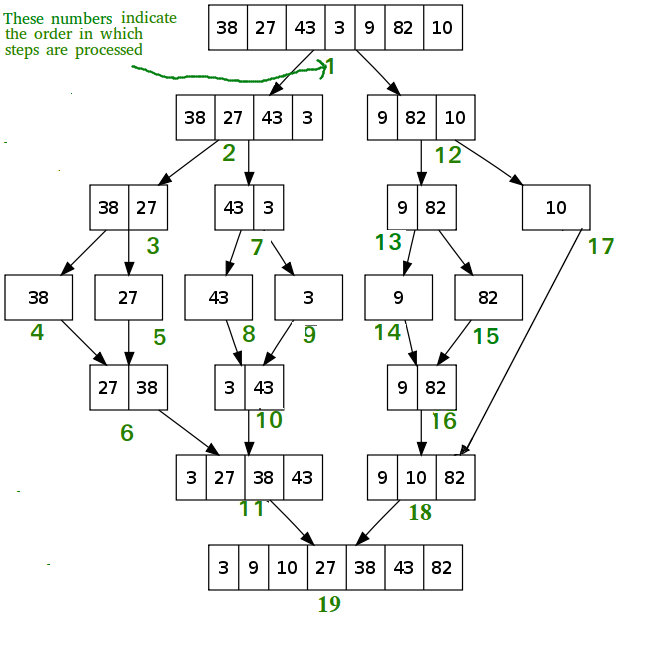
* IDEA
* CAST
* CMEA
* 3-way
* Blowfish
* GOST
* LOKI
* DES and Triple DES

# Explain Merge Sort Algorithm?

Merge sort is a comparison-based sorting algorithm. It is a **stable sort**, which indicates that the order of equal elements in the input and output is the same. It is a **divide and conquer** algorithm.

* Separate the unsorted list into n sub-lists, each with one element (a list of one element is considered sorted).
* Merge sub lists repeatedly to create new sorted sub-lists until only one sub-list remains.

The time complexity of the Merge Sort Algorithm is O (n \* log(n)) where n is the size of the list of the elements to be sorted while the space complexity of the Merge Sort Algorithm is O(n), that is, linear space complexity.



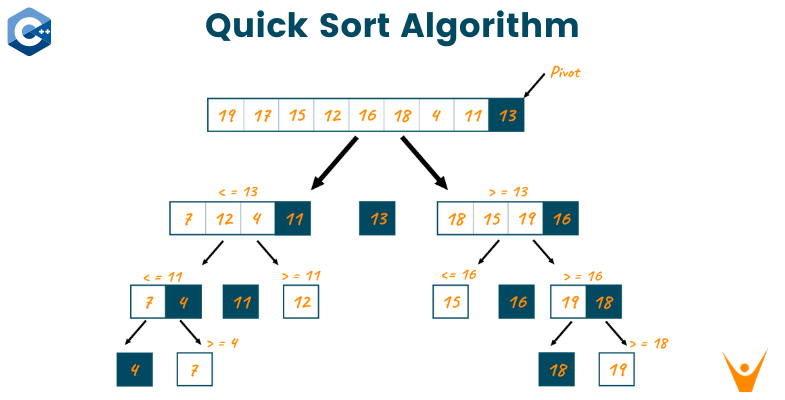
# Explain Quick Sort Algorithm?

Quicksort is a **in place** (in-place algorithm transforms input using no auxiliary data structure). It can be somewhat quicker than merge sort and two or three times faster than heapsort.

Quicksort is based on the **divide and conquer** algorithm. It operates by picking a 'pivot' element from the array and separating the other elements into two subarrays based on whether they are greater or less than the pivot. As a result, it is also known as **partition** **exchange sort**. The subarrays are then recursively sorted. This can be done in place, with only a little amount of additional RAM required for sorting.

Quicksort is a comparison sorting algorithm, which means it can sort objects of any type that have a "less-than" relation (technically, a total order) declared for them. Quicksort is not a stable sort, which means that the relative order of equal sort items is not retained in efficient implementations. Quicksort (like the partition method) must be written in such a way that it can be called for a range within a bigger array, even if the end purpose is to sort the entire array, due to its recursive nature.

* If there are less than two elements in the range, return immediately.
* Otherwise, choose a pivot value, which is a value that occurs in the range (the precise manner of choice depends on the partition routine, and can involve randomness).
* Partition the range by reordering its elements while determining a point of division so that all elements with values less than the pivot appear before the division and all elements with values greater than the pivot appear after it; elements with values equal to the pivot can appear in either direction. Most partition procedures ensure that the value that ends up at the point of division is equal to the pivot, and is now in its ultimate location because at least one instance of the pivot is present (but termination of quicksort does not depend on this, if sub-ranges strictly smaller than the original are produced).
* Apply the quicksort recursively to the sub-range up to the point of division and the sub-range after it, optionally removing the element equal to the pivot at the point of division from both ranges. (If the partition creates a potentially bigger sub-range near the boundary with all elements known to be equal to the pivot, these can also be omitted)



# Explain bubble sort algorithm?

Bubble sort, a.k.a. sinking sort, is a basic sorting algorithm that iterates through a list, comparing neighbouring elements and swapping them if they are out of order. The list is sent through again and again until it is sorted. The comparison sort method is named from the manner that smaller or larger components "bubble" to the top of the list. This simplistic method performs badly in real-world situations and is mostly used as a teaching aid.

Let us assume that the array to be sorted is (50 10 40 20 80). The various passes or rounds of bubble sort are given below:

**First Pass:**

* (50 10 40 20 80) –> (10 50 40 20 80), Since 50 > 10, the algorithm compares the first two elements and swaps them.
* (10 50 40 20 80) –> (10 40 50 20 80), Since 50 > 40, the algorithm swaps the values at the second and third positions.
* (10 40 50 20 80) –> (10 40 20 50 80), Since 50 > 3, the algorithm swaps the third and fourth elements.
* (10 40 20 50 80) -> (10 40 20 50 80), The method does not swap the fourth and fifth elements because they are already in order (80 > 50).

**Second Pass:**

* (10 40 20 50 80) –> (10 40 20 50 80), Elements at first and second position are in order so now swapping.
* (10 40 20 50 80) –> (10 20 40 50 80), Since 40 > 20, the algorithm swaps the values at the second and third positions.
* (10 20 40 50 80) –> (10 20 40 50 80), Elements at the third and fourth position are in order so now swapping.
* (10 20 40 50 80) –> (10 20 40 50 80), Elements at fourth and fifth position are in order so now swapping.

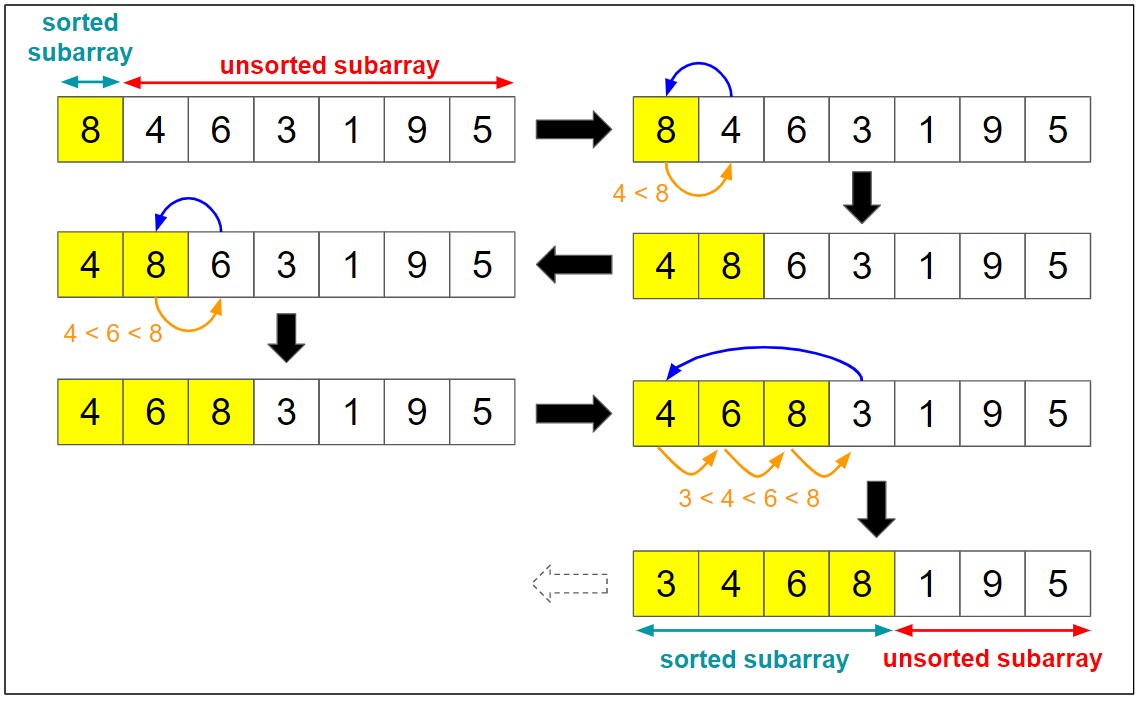
The array is now sorted, but our algorithm is unsure whether it is complete. To know if the algorithm is sorted, it must complete one complete pass without any swaps.

**Third Pass:**

* (10 20 40 50 80) –> (10 20 40 50 80), Elements at the first and second position are in order so now swapping.
* (10 20 40 50 80) –> (10 20 40 50 80), Elements at the second and third position are in order so now swapping.
* (10 20 40 50 80) –> (10 20 40 50 80), Elements at the third and fourth position are in order so now swapping.
* (10 20 40 50 80) –> (10 20 40 5 80), Elements at the fourth and fifth position are in order so now swapping.

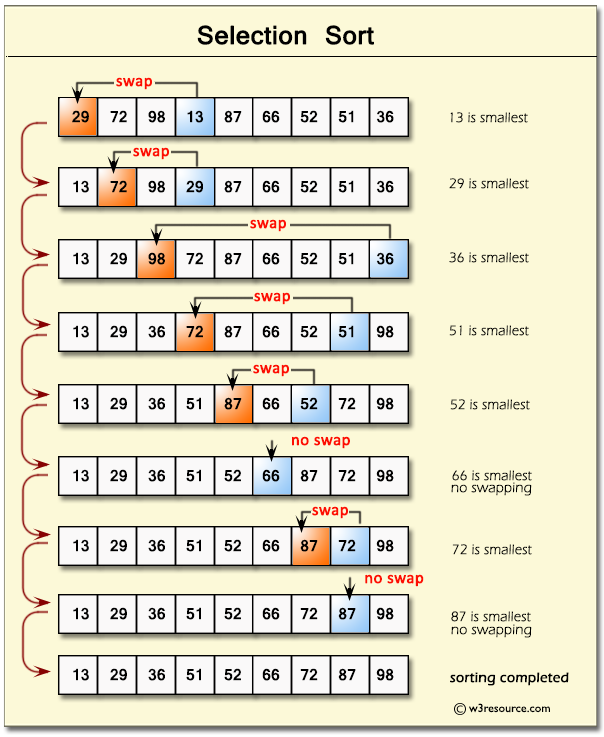
# Define insertion sort and selection sort?

**Insertion sort:** Insertion sort separates the list into sorted and unsorted sub-lists. It inserts one element at a time into the proper spot in the sorted sub-list. After insertion, the output is a sorted sub-list. It iteratively works on all the elements of an unsorted sub-list and inserts them into a sorted sub-list in order.



**Selection sort:** Selection sort is an **in-place** sorting technique. It separates the data collection into sorted and unsorted sub-lists. The minimum element from the unsorted sub-list is then selected and placed in the sorted list. This loops until all the elements in the unsorted sub-list have been consumed by the sorted sub-list.

Note: Both sorting strategies keep two sub-lists, sorted and unsorted, and place one element at a time into the sorted sub-list. Insertion sort takes the currently selected element and places it in the sorted array at the right point while keeping the insertion sort attributes. Selection sort, looks for the smallest element in an unsorted sub-list and replaces it with the current element.



# What is radix sort?

Linear sorting algorithm used for integers. There is digit by digit sorting performed that starts from the least significant digit to the most significant digit.

The process of radix sort works similar to the sorting of student’s names, according to the alphabetical order. In this case, there are 26 radixes formed due to the 26 alphabets in English. In the first pass, the names of students are grouped according to the ascending order of the first letter of their names. After that, in the second pass, their names are grouped according to the ascending order of the second letter of their name. And the process continues until we find the sorted list.

Best Case: O(n+k)

Average Case: O(nk)

Worst Case: O(nk)

Space complexity: O(n+k)

Stable: Yes

# Can we use the binary search algorithm for linked lists?

No. Because random access is not allowed in linked lists, reaching the middle element is constant or O (1) time is impossible. As a result, the usage of a binary search algorithm on a linked list is not possible.

# What are recursive algorithms and their rules?

Recursive algorithm is a way of tackling a difficult problem by breaking it down into smaller and smaller subproblems until the problem is small enough to be solved quickly. It usually involves a function that calls itself (property of recursive functions).

The three laws which must be followed by all recursive algorithms are as follows:

* There should be a base case.
* It is necessary for a recursive algorithm to call itself.
* The state of a recursive algorithm must be changed for it to return to the base case.

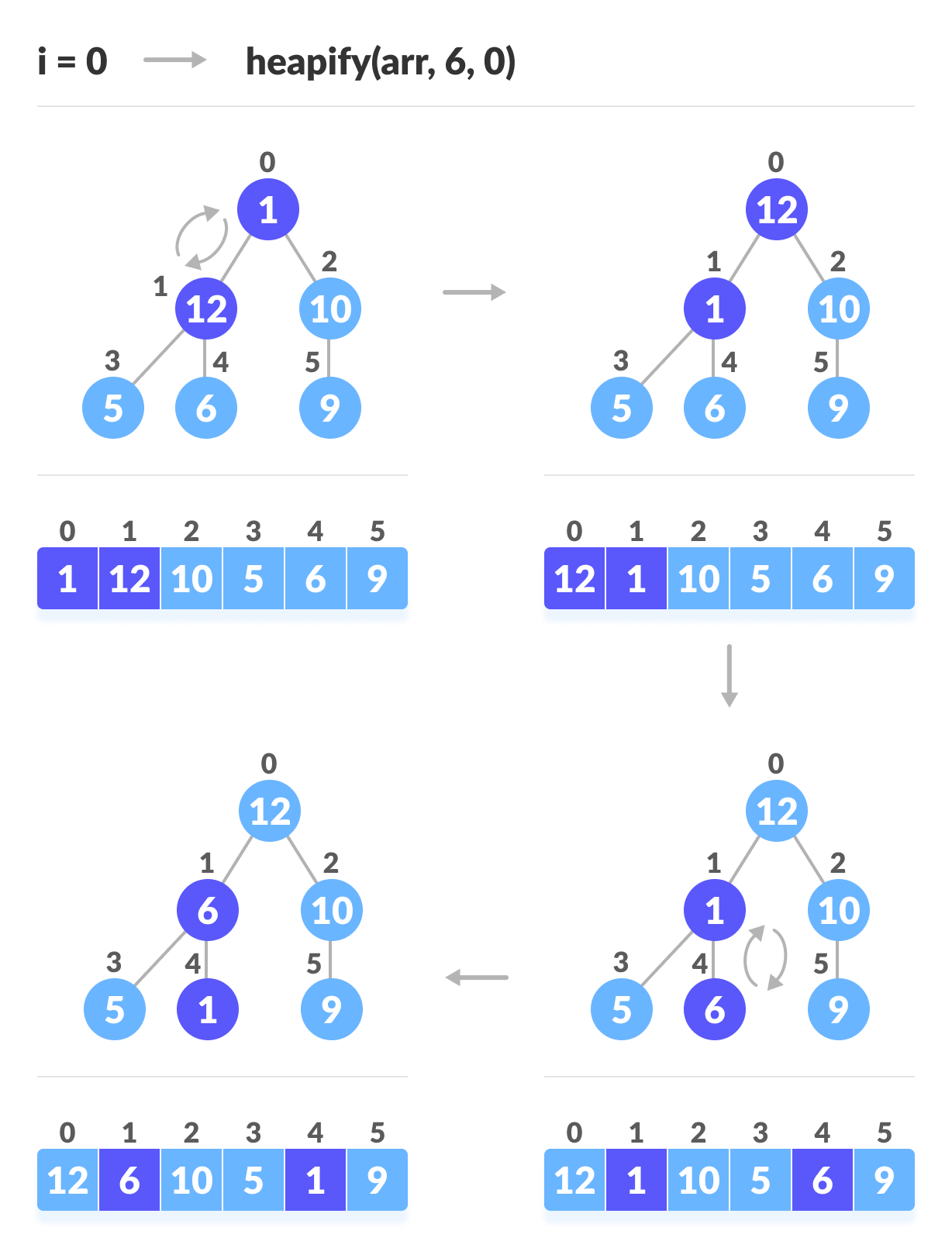
# Describe the heap sort algorithm?

Heap sort is a comparison-based sorting algorithm. Heapsort is like selection sort in that it separates its input into a sorted and an unsorted region, then successively decreases the unsorted part by taking the largest element from it and putting it into the sorted region. Unlike selection sort, heapsort does not waste time scanning the unsorted region in linear time; instead, heap sort keeps the unsorted region in a heap data structure to identify the largest element in each step more rapidly. Let us look at the heap sort algorithm:

The Heapsort algorithm starts by converting the list to a max heap. The algorithm then swaps the first and last values in the list, reducing the range of values considered in the heap operation by one, and filters the new first value into its heap place. This process is repeated until the range of values considered is only one value long.

* On the list, use the buildMaxHeap() function. This function, also known as heapify(), creates a heap from a list in O(n) operations.
* Change the order of the list's first and last elements. Reduce the list's considered range by one.
* To sift the new initial member to its appropriate index in the heap, use the siftDown() function on the list.
* Unless the list's considered range is one element, proceed to step 2.

Note: The buildMaxHeap() operation runs only one time with a linear time complexity or O(n) time complexity. The siftDown() function works in O(log n) time complexity, and is called n times. Therefore, the overall time complexity of the heap sort algorithm is O (n + n log (n)) = O (n log n).



Pandas

# Features that make pandas a reliable option to store tabular data

* Memory Efficient
* Data Alignment
* Reshaping
* Merge and Join
* Time series

# What is a Series in pandas?

Series is defined as a 1-D array that is capable of storing various data types. The row labels of the series are called the index. By using a ‘series’ method, we can easily convert the list, tuple, and dictionary into series. A series cannot contain multiple columns.

# What is Time Series in Pandas?

A time series is an ordered sequence of data which basically represents how some quantity changes over time. Pandas contain extensive capabilities and features for working with time series data for all domains.

# What is vectorization in Pandas?

Vectorization is the process of running operations on the entire array. This is done to reduce the amount of iteration performed by the functions. Pandas have a number of vectorized functions like aggregations, and string functions that are optimized to operate specifically on series and DataFrames. So, it is preferred to use the vectorized panda’s functions to execute the operations quickly.

# What is ‘groupby’ in Pandas

GroupBy is used to split the data into groups. It groups the data based on some criteria. Grouping also provides a mapping of labels to the group names. It has a lot of variations that can be defined with the parameters and makes the task of splitting the data quick and easy.

# What is the meaning of axis = 0 and axis = 1

Axis = 0 is meant for reading rows, axis = 1 is meant for reading columns.

# Difference between the two data series df[‘Name’] and df.loc[:, ‘Name’]

df[‘Name’] is a view of the original data frame and second one is a copy of the original dataframe.

* df.head(n)

Returns the first ‘n’ rows, default 5. If n is larger than the number of rows, it returns all rows.

* df.tail(n)

Returns the last ‘n’ rows, default 5. For negative values of ‘n’, this function returns all rows except the first |n| rows, equivalent to df[|n|:]

* df.shape[0]

Returns a tuple representing the dimensionality of the DataFrame. Gives number of row count.

* df.shape[1]

Gives number of column count.

* df.columns

The column labels of the DataFrame

* df.dtypes

Returns the datatype of each column

* df[‘column\_name’].isna().sum()

Count NaN values under a single DataFrame column

* df.isnull().sum().sum()

Count NaN values under an entire DataFrame

* df.loc[[index\_value]].isna().sum().sum()

Count NaN values across a single DataFrame row

* df.nunique(axis = 0, dropna = True)

Count number of distinct elements in specified axis. Axis = 0 for rows, axis = 1 for columns.

* df.info()

Prints information about the DataFrame.

* df.copy(deep = True)

Make a copy of object’s indices and data.

When deep=True (default), a new object will be created with a copy of the calling object’s data and indices. Modifications to the data or indices of the copy will not be reflected in the original object (see notes below).

When deep=False, a new object will be created without copying the calling object’s data or index (only references to the data and index are copied). Any changes to the data of the original will be reflected in the shallow copy (and vice versa).

* df.describe()

Returns description of the data in the DataFrame.

* df.loc[input]

Access a group of rows and columns by labels or a boolean array.

Input can be a single label, a list or array of labels etc.

df[‘column\_name’].astype(‘category’)

* df.corr(method = ‘pearson’)

Compute pairwise correlation of columns, excluding null values.

df.select\_dtypes([‘category’]).columns

df[‘column\_name’].value\_counts()

df.loc[df.column\_name == ‘category’]

* df.count(axis = 0)

Count non-NA cells for each column or row.

df.column\_name

df.column\_name.mean()

pd.crosstab()

df.groupby(by = label, axis = 0)