Date: 8/2/2022

1. What is cron job

A cron job is a Linux command used for scheduling tasks to be executed sometime in the future. This is normally used to schedule a job that is executed periodically – for example, to send out a notice every morning

2. SOLID principles with used cases

The following five concepts make up our SOLID principles:

1. **Single Responsibility Principle**: This principle states that “a class should have only one reason to change” which means every class should have a single responsibility or single job or single purpose.
2. **Open/Closed Principle**: This principle states that “software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification” which means you should be able to extend a class behavior, without modifying it.
3. **Liskov’s Substitution Principle**: The principle was introduced by Barbara Liskov in 1987 and according to this principle “Derived or child classes must be substitutable for their base or parent classes“. This principle ensures that any class that is the child of a parent class should be usable in place of its parent without any unexpected behavior.
4. **Interface Segregation Principle**: This principle is the first principle that applies to Interfaces instead of classes in SOLID and it is similar to the single responsibility principle. It states that “do not force any client to implement an interface which is irrelevant to them“. Here your main goal is to focus on avoiding fat interface and give preference to many small client-specific interfaces. You should prefer many client interfaces rather than one general interface and each interface should have a specific responsibility.
5. **Dependency Inversion Principle**: Now two key points are here to keep in mind about this principle:
    a. High-level modules/classes should not depend on low-level modules/classes. Both should depend upon abstractions.
    b. Abstractions should not depend upon details. Details should depend upon abstractions.

    The above lines simply state that if a high module or class will be dependent more on low-level modules or class then your code would have tight coupling and if you will try to make a change in one class it can break another class which is risky at the production level. So always try to make classes loosely coupled as much as you can and you can achieve this through abstraction. The main motive of this principle is decoupling the dependencies so if class A changes the class B doesn’t need to care or know about the changes.

3. Bob's design patterns book

Robert Cecil Martin

- Clean Code
- Design Principles and Design Patterns
- Agile Software Development, Principles, Patterns, and Practices

4. Distributions' types with examples

a. Discrete Distributions:

i. Uniform Distribution : When you roll a fair die, the outcomes are 1 to 6. The probabilities of getting these outcomes are equally likely and that is the basis of a uniform distribution. Unlike Bernoulli Distribution, all the n number of possible outcomes of a uniform distribution are equally likely.

ii. Bernoulli Distribution: has only two possible outcomes, namely 1 (success) and 0 (failure), and a single trial. So the random variable X which has a Bernoulli distribution can take value 1 with the probability of success, say p, and the value 0 with the probability of failure, say q or 1-p. Here, the occurrence of a head denotes success, and the occurrence of a tail denotes failure. Probability of getting a head = 0.5 = Probability of getting a tail since there are only two possible outcomes.

iii. Binomial Distribution: A distribution where only two outcomes are possible, such as success or failure, gain or loss, win or lose and where the probability of success and failure is same for all the trials. The properties of a Binomial Distribution are:

1. Each trial is independent.
2. There are only two possible outcomes in a trial- either a success or a failure.
3. A total number of n identical trials are conducted.
4. The probability of success and failure is same for all trials.

iv. Poisson's Distribution: used to test out how unusual an event frequency is for a given interval. Suppose you work at a call center, approximately how many calls do you get in a day? It can be any number. Now, the entire number of calls at a call center in a day is modeled by Poisson distribution. Poisson Distribution is applicable in situations where events occur at random points of time and space wherein our interest lies only in the number of occurrences of the event.

A distribution is called Poisson distribution when the following assumptions are valid:

1. Any successful event should not influence the outcome of another successful event.

2. The probability of success over a short interval must equal the probability of success over a longer interval.

3. The probability of success in an interval approaches zero as the interval becomes smaller.

b. Continuous Distributions:

i. Normal Distribution: represents the behavior of most of the situations in the universe. The large sum of (small) random variables often turns out to be normally distributed, contributing to its widespread application. Any distribution is known as Normal distribution if it has the following characteristics:

1. The mean, median and mode of the distribution coincide.
2. The curve of the distribution is bell-shaped and symmetrical about the line x=μ.
3. The total area under the curve is 1.
4. Exactly half of the values are to the left of the center and the other half to the right.

ii. T-Distribution: a small sample approximation of normal distribution

iii. Chi-squared: asymmetric, consists of only non-negative values, doesn't mirror real-life events, often used in hypothesis testing to help determine goodness of fit.

iv. Exponential Distribution: for events rapidly changing early on. Exponential distribution is widely used for survival analysis. From the expected life of a machine to the expected life of a human, exponential distribution successfully delivers the result.

v. Logistic Distribution: useful in forecast analysis for determining a cut-off point for a successful outcome

5. Central Limit theorem

In probability theory, the central limit theorem (CLT) states that the distribution of a sample variable approximates a normal distribution (i.e., a "bell curve") as the sample size becomes larger, assuming that all samples are identical in size, and regardless of the population's actual distribution shape.

Put another way, CLT is a statistical premise that, given a sufficiently large sample size from a population with a finite level of variance, the mean of all sampled variables from the same population will be approximately equal to the mean of the whole population. Furthermore, these samples approximate a normal distribution, with their variances being approximately equal to the variance of the population as the sample size gets larger, according to the law of large numbers.

6. Difference between hypothesis and statistical testing

-Hypothesis testing is a formal procedure for investigating our ideas about the world using statistics. It is most often used by scientists to test specific predictions, called hypotheses, that arise from theories. There are 5 main steps in hypothesis testing:

1. State your research hypothesis as a null hypothesis (Ho) and alternate hypothesis (Ha or H1).
2. Collect data in a way designed to test the hypothesis.

3. Perform an appropriate statistical test.
4. Decide whether to reject or fail to reject your null hypothesis.
5. Present the findings in your results and discussion section.

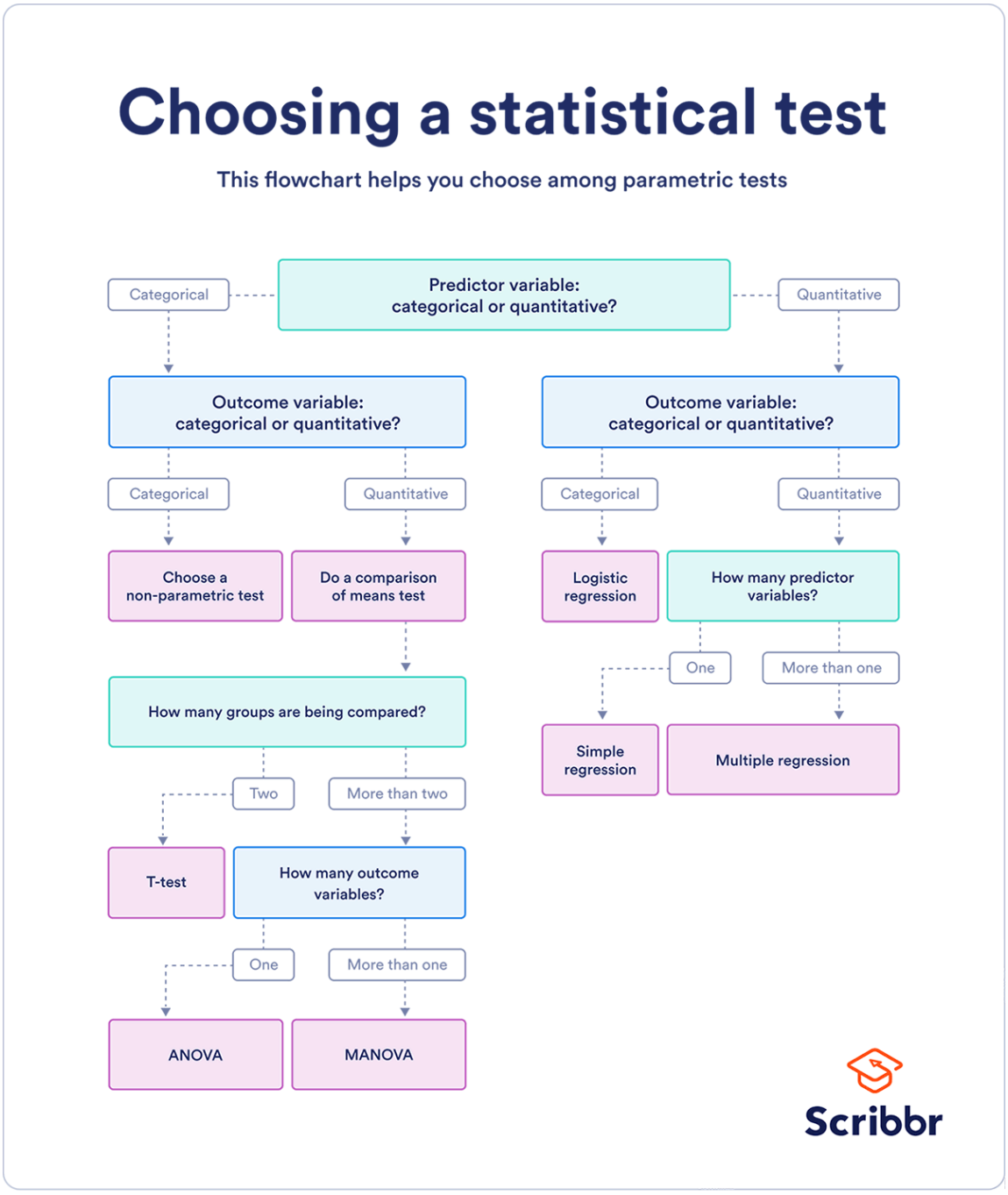
-Statistical tests are used in hypothesis testing. They can be used to:

- determine whether a predictor variable has a statistically significant relationship with an outcome variable.
- estimate the difference between two or more groups.

Statistical tests assume a null hypothesis of no relationship or no difference between groups. Then they determine whether the observed data fall outside of the range of values predicted by the null hypothesis.

	Predictor variable	Outcome variable	Use in place of…
Spearman's *r*	• Quantitative	• Quantitative	Pearson's *r*
Chi square test of independence	• Categorical	• Categorical	Pearson's *r*
Sign test	• Categorical	• Quantitative	One-sample *t*-test
Kruskal–Wallis *H*	• Categorical • 3 or more groups	• Quantitative	ANOVA
ANOSIM	• Categorical • 3 or more groups	• Quantitative • 2 or more outcome variables	MANOVA
Wilcoxon Rank-Sum test	• Categorical • 2 groups	• Quantitative	Independent t-test

		• groups come from different populations	
Wilcoxon Signed-rank test	• Categorical • 2 groups	• Quantitative • groups come from the same population	Paired t-test



7. Average collected data per day

In 2018, more than 2.5 quintillion (10^18) bytes of data were created every day.

8. Research Labs in computer science

Advancing Machine and Human Reasoning (AMHR) Lab
AMHR is a cross-disciplinary lab dedicated to answering the following guiding research questions: How can artificial intelligence make people better reasoners? How can we create better artificially intelligent reasoners? And how can we advance our knowledge of logic and other cognitive-level reasoning processes in order to produce better conclusions, justifications, and arguments? We're

devoted to not only creating smarter AI, but ensuring that these advances help improve, rather than replace, human reasoners. We believe that one day, we can do this with advances in computational, logical, rational, and justifiable reasoning / argumentation.

ARGUS Cybersecurity Lab

The focus of this lab is on the defense aspect of the cyber space, and the philosophy is to start from real problems, and create solutions that last. Research attempts to address the root causes of the various cybersecurity problems. The lab works closely with industry to ensure that its work both addresses the most pressing problems of the time, and provides the scientific basis for solutions that can stand the test of time.

Biorobotics Lab

Research at the USF BioRobotics Lab covers various areas of robotics and biologically-inspired systems. The lab is involved in various domestic and international research collaborations related to behavioral and neural modeling of spatial cognition in animals. Robot research focuses on autonomous mobile platforms, including single and multi-robot wheeled robots, humanoid robots, aerial and marine systems. The biorobotics lab is also home to the USF RoboBulls autonomous robot soccer team.

Computing Education Research & Evolutionary Algorithms Lab (CEREAL)

This lab gathers researchers from diversified disciplines who share an interest in; 1) identifying and understanding the learning barriers encountered by students of the computing disciplines and developing and evaluating innovative, technology-supported pedagogies to address them, 2) developing Evolutionary techniques that are able to tackle challenging application domains that require very significant adaptive capabilities (e.g., interactive or time-dependent optimization problems), and 3) applying our experience with the above to develop Evolutionary-Aided Teaching and Learning approaches (e.g., autonomous design of practice problems, automated discovery of concept inventories). As a result of these interests, our work spans both the Computing Education research and the Evolutionary Computation fields.

Computer Vision and Pattern Recognition Group

The Computer Vision and Pattern Recognition Group Invents Technologies Resulting in Commercial Products that Enhance the Security, Health and Quality of Life. The lab leverages USF's strengths in Video and Image Analysis Technology, Biometric Technology, Affective Computing, Classification and Knowledge Discovery, and Medical Data Analysis Technology to impact domestic security, quality of life, and healthcare.

Cryptographic Engineering and Hardware Security Lab

Cryptographic Engineering and Hardware Security Lab conducts research on the secure hardware design, implementation, and optimization of crypto-systems. In addition, the lab's focus is on emerging topics in side-channel analysis attacks and countermeasures. The research conducted in this lab includes a number of abstraction levels, including hardware micro architecture and platform specific deeply-embedded hardware systems. The research work platforms utilized are broad, e.g., ASIC/FPGA and embedded processors. Moreover, the lab conducts research on pre- and post-quantum cryptography and security of a number of sensitive and constrained applications including low-power and energy-constrained RFID/NFC technology-based applications and wireless nano-sensors.

Cyber Identity and Behavior Research (CiBeR) Lab

In the Cyber Identity and Behavior Research Lab, we focus on creating dynamic and robust solutions to person identification in online environments. To this end, our research efforts are largely centered upon understanding and modeling cyber behaviors for the validation of patterns of behavior as behavioral biometric modalities. We also aim to uncover signatures of behavior

which persist across physical and cyber domains to better interpret and make use of the concept of an identity. Our work spans multiple research areas, including person identification and verification, pattern recognition, computer vision, and cybersecurity. Applications for our research are in forensics, homeland security, online marketing, healthcare, and more.

Distributed Systems Group

The Distributed Systems Group focuses on research in computational sociology and distributed systems, with current emphasis on understanding behavior in global-scale social networked systems. Research in the group quantitatively characterizes socio-technical phenomena at scale, models them, applies new understandings to the design of distributed systems, and experimentally measures performance. Interdisciplinary collaborations are highly regarded, as the group often uses results and theories from sociology, psychology and political science to build better interpretations of quantitative observations and to inform design and experiments.

-**Intelligent Systems Lab (ISL)**

ISL research is focused on learning high quality models from data. Unlabeled data is modeled with clustering algorithms. Mixtures of labeled and unlabeled data are addressed with semi-supervised learning approaches. Of particular interest is big data for which Deep Neural Networks are often useful. Ensembles of different (and the same) types of models are considered, imbalanced data is continually addressed. Imprecision in intelligent systems is also a research topic. Some recent work focuses on learning prognostic models from medical images and clinical data, learning models of activity in very large information networks and clustering data in a network environment. An overall goal is to be able to group large sets of unlabeled data in useful ways, uncovering small, but important groups where they exist. Another goal is to be able to make accurate predictions from potentially large (at least partially) labeled data sets.

-**Neuro-Machine Interaction Lab (NMIL)**

The Neuro-Machine Interaction Lab studies mediums of new human interaction with physiological sensors, specifically sensors that acquire brain data. The purpose is to investigate novel methods to use the brain as a third arm to assist users perform daily activities. Also, to measure and decode the affective, cognitive, and emotional state of a person to further understand the brain's behavior during human-machine interaction. The work we do is interdisciplinary and benefits from areas like Psychology, Neuroscience, Computer Science, Electrical Engineering, Arts, and others. General research areas within NCIL are Affective-BCI, Brain-Controlled Drones, and Artistic BCI.

-**Robot Perception and Action Lab (RPAL)**

People in RPAL have been working on bringing intelligent robots into our daily living life. Their research focuses include robotic grasping, structured knowledge representation, motion generating, visual SLAM for medical robotics, unmanned aerial vehicle (UAV), and neurorobotics. The lab collaborate with many professors, researchers, and medical doctors in USF College of Engineering, Medical School, Psychology Department, and other prestigious institutions and companies around the world.

-**Social Computing Research Lab**

The Social Computing Research Lab conducts theoretical and experimental research to address critical emerging problems when societies and computing technologies interact closely with each other. There is a significant emphasis within the group on addressing big-data challenges via effective data mining, data fusion and machine learning techniques. The group's research is strongly multi-disciplinary involving collaborators in computer science, engineering, behavioral

sciences, clinical psychiatry and education. Practical applications of our research are in cyber security, smart healthcare, disaster management, environmental sustenance and more. Students are constantly encouraged to innovate and transition outcomes from the lab to industry.

-**Software Security and Programming Languages Lab**
This lab conducts research on: software attacks (such as buffer overflows and SQL injections), runtime defense mechanisms (such as CFI, firewalls, and other monitors), theories of security (such as security models, approaches to composing security policies, and enforceability theory), strong type systems for programming languages, and tools for specifying and managing complex security policies.

-**Ubiquitous Sensing Lab**
Ubiquitous sensing is a new area of research that encompasses the integration of smartphones with senors and Internet technologies to address large-scale societal as well as individual problems.

9. Relational vs non-relational database

A **relational database** is a collection of data items with pre-defined relationships between them. These items are organized as a set of tables with columns and rows. Tables are used to hold information about the objects to be represented in the database. Each column in a table holds a certain kind of data and a field stores the actual value of an attribute. The rows in the table represent a collection of related values of one object or entity. Each row in a table could be marked with a unique identifier called a primary key, and rows among multiple tables can be made related using foreign keys. This data can be accessed in many different ways without reorganizing the database tables themselves.

A **non-relational database** is a database that does not use the tabular schema of rows and columns found in most traditional database systems. Instead, non-relational databases use a storage model that is optimized for the specific requirements of the type of data being stored. For example, data may be stored as simple key/value pairs, as JSON documents, or as a graph consisting of edges and vertices.

10. When are outliers kept in the data

11. Business tools for data science operations

1.**SAS** : (Statistical Analysis System) is one of the oldest Data Science tools in the market. One can perform granular analysis of textual data and can generate insightful reports via SAS. Besides data analysis, SAS is also used to access/retrieve data from various sources. It is widely used for multiple Data Science activities like data mining, time series analysis, econometrics, business intelligence, etc. SAS is platform-independent and is also used for remote computing. One can't ignore the role of SAS in quality improvement and application development.

2. APACHE HADOOP: is an open-source software widely used for the parallel processing of data. Any large file is distributed/split into chunks and then handed over to various nodes. The clusters of nodes are then used for parallel processing by Hadoop. Hadoop consists of a distributed file

system responsible for dividing the data into chunks and distributing it to various nodes. Besides the Hadoop File Distribution System, many other Hadoop components are used to parallelly process data, such as Hadoop YARN, Hadoop MapReduce, and Hadoop Common.

3. TABLEAU: is a data visualization tool that assists in decision-making and data analysis. You can represent data visually in less time by Tableau so that everyone can understand it. Advanced data analytics problems can be solved in less time using Tableau. Founded in 2003, Tableau has transformed the way data scientists used to approach Data Science problems. One can make the most of their dataset using Tableau and can generate insightful reports.

4. TENSORFLOW: is widely used with various new-age technologies like Data Science, Machine Learning, Artificial Intelligence, etc. TensorFlow is a Python library that you can use for building and training Data Science models. You can take data visualization to the next level with the aid of TensorFlow. TensorFlow is easy to use as it is written in Python and is widely used for differential programming. One can deploy Data Science models across various devices using TensorFlow. TensorFlow uses an N-dimensional array as its data type, which is also called a tensor.

5. BIGML: is used for building datasets and then sharing them easily with other systems. Initially developed for Machine Learning (ML), BigML is widely used for creating practical Data Science algorithms. You can easily classify data and find the anomalies/outliers in the data set using BigML. The interactive data visualization process of BigML makes it easy for data scientists to make decisions. The Scalable BigML platform is also used for time series forecasting, topic modeling, association discovery tasks, and much more. You can operate on large sets of data using BigML.

6. KNIME: is one of the widely used Data Science tools for data reporting, mining, and analysis. Its ability to perform data extraction and transformation makes it one of the essential tools used in Data Science. The Knime platform is open-source and free to use in various parts of the world. It uses ‘Lego of Analytics,’ a data pipelining concept for integrating various components of Data Science. The easy-to-use GUI (Graphical User Interface) of Knime helps perform data science tasks with minimum programming expertise. The visual data pipelines of Knime are used to create interactive views for the given dataset.

7. RAPIDMINER: is a widely used Data Science software tool due to its capacity to provide a suitable environment for data preparation. Any Data Science/ML model can be prepared from scratch using RapidMiner. Data scientists can track data in real-time using RapidMiner and can perform high-end analytics. RapidMiner can perform various other Data Science chores like text mining, predictive analysis, model validation, comprehensive data reporting, etc. The high scalability and security features offered by RapidMiner are also remarkable. Commercial Data Science applications can be developed from scratch using RapidMiner.

8. EXCEL: Part of Microsoft’s Office tools, Excel is one of the best tools for Data Science freshers. It also helps in understanding the basics of Data Science before moving into high-end analytics. It is one of the essential tools used by data scientists for data visualization. Excel represents the data in a simple way using rows and columns to be understood even by non-technical users. Excel also offers various formulas for Data Science calculations like concatenation, find average

data, summation, etc. Its ability to process large data sets makes it one of the critical tools used for Data Science.

9. **APACHE FLINK**: it is one of the best Data Science tools 2020/2021 offered by the Apache Software Foundation. Apache Flink can quickly carry out real-time data analysis. Apache Flink is an open-source distributed framework that can perform scalable Data Science computations. Flink offers both pipeline and parallel execution of dataflow diagrams at low latency. An unbounded data stream that does not have a fixed start and endpoint can also be processed using Apache Flink. Apache has a reputation for providing Data Science tools and techniques that can speed up the analysis process. Flink helps data scientists in reducing complexity while real-time data processing.

10. POWERBI: is also one of the essential tools of Data Science integrated with business intelligence. You can combine it with other Microsoft Data Science tools for performing data visualization. You can generate rich and insightful reports from a given dataset using PowerBI. Users can also create their data analytics dashboard using PowerBI. The incoherent sets of data can be turned into coherent sets using PowerBI. You can develop a logically consistent dataset that will generate rich insights using PowerBI. One can generate eye-catching visual reports using PowerBI that can be understood by non-technical professionals too.

11. DATAROBOT: is one of the valuable tools required for Data Science operations integrated with ML and Artificial Intelligence. You can drag and drop a dataset quickly on the DataRobot user interface. Its easy-to-use GUI makes data analytics possible for freshers as well as expert data scientists. You can build and deploy more than 100 Data Science models at once via DataRobot and can get rich insights. Enterprises also use it to provide high-end automation to their users/customers. The efficient predictive analysis offered by DataRobot can help you in making intelligent data-based decisions.

12. APACHE SPARK: is designed for performing Data Science calculations with low latency. Based on the Hadoop MapReduce, Apache Spark can handle interactive queries and stream processing. It has become one of the best Data Science tools in the market due to its in-memory cluster computing. Its in-memory computing can increase the processing speed significantly. Apache Spark supports SQL queries so that you can derive various relationships among your dataset. Spark also provides various APIs in Java, Scala, and Python for developing Data Science applications.

13. SAP HANA: is a relational database management system that makes data storage and retrieval easy. It is a handy tool in Data Science due to its in-memory and column-based data management system. If a database has objects stored in a geometrical space (spatial data), it can be processed using Sap Hana. Various other Data Science activities can be performed using Sap Hana like text search and analytics, graph data processing, predictive analysis, etc. Its in-memory data storage stores data in the main memory besides keeping it in any disk, which offers enhanced querying and data processing.

14. MONGODB: is a high-performance database and is one of the top Data Science tools in the market. One can store large volumes of data in a collection (MongoDB documents) offered by MongoDB. It provides all the capabilities of SQL and supports dynamic queries. MongoDB stores

data in the form of JSON-style documents and offers high data replications capability. Managing Big data is much easier with MongoDB as it provides high data availability. Besides basic database queries, MongoDB can also perform advanced analytics. The high scalability of MongoDB also makes it one of the widely used Data Science tools.

15. PYTHON: The Data Science tools and technologies are not limited to databases and frameworks. Choosing the right programming language for Data Science is of utmost importance. Python is used by a lot of data scientists for web scraping. Python offers various libraries designed explicitly for Data Science operations. You can efficiently perform various mathematical, statistical, and scientific calculations with Python. Some of the widely used Python libraries for Data Science are NumPy, SciPy, Matplotlib, Pandas, Keras, etc.

16. TRIFACTA: is one of the widely used Data Science tools for data cleaning and preparation. A cloud data lake that includes a mix of structured and unstructured data can be cleaned using Trifacta. The data preparation process is significantly paced via Trifacta as compared to other platforms. One can easily identify the errors, outliers, etc., in the dataset using Trifacta. You can prepare data in less time across a multi-cloud environment, too, using Trifacta. You can automate the data visualization process and data pipeline management using Trifacta.

17. MINITAB: is a software package and is widely used for data manipulation and analysis. Minitab will help you in identifying trends and patterns in an unstructured dataset. The dataset which is going to be the input for data analysis can be simplified using Minitab. Minitab also helps data scientists to automate Data Science calculations and graph generation. While using Minitab, descriptive statistics are displayed based on the entered dataset highlighting various key points in data like mean, median, standard deviation, etc. Besides creating multiple types of graphs with Minitab, you can also perform regression analysis.

18. R: provides a scalable software environment for statistical analysis and is one of the many popular programming languages used in the Data Science sector. Data clustering and classification can be performed in less time using R. Various statistical models can be created using R, supporting both linear and nonlinear modeling. You can perform data cleaning and visualization efficiently via R. R represents the data visually in simple ways so that everyone can understand it. R offers various add-ons for Data Science like DBI, RMySQL, dplyr, ggmap, xtable, etc.

19. APACHE KAFKA: Apache Kafka is a distributed messaging system used to transfer large volumes of data from one application to another. Real-time data pipelines can be constructed in less time using Apache Kafka. Known for its fault tolerance and scalability, Kafka will provide you with zero data loss while transferring data over applications. Apache Kafka works on the publish-subscribe messaging system where publishers convey messages within topics to subscribers. The subscribers can consume all the messages in a topic via the publish-subscribe messaging system.

20. QLIKVIEW: is one of the widely used Data Science tools and is also concerned with business intelligence. QlikView helps data scientists to derive relationships between unstructured data and perform data analysis. You can also demonstrate a visual representation of data relationships via QlikView. One can perform data aggregation and compression via QlikView in less time. You do

not have to spend time determining the relationships between data entities as QlikView does it for you automatically. Its in-memory data processing provides faster results as compared to other Data Science tools in the market.

21. MICROSTRATEGY: is used by data scientists who are also into business intelligence. Besides enhanced data visualizations and discovery, MicroStrategy offers a wide range of data analytics capabilities. You can connect MicroStrategy to various data warehouses and relational systems to access data, thus adding to its data accessibility/discovery capabilities. You can break unstructured and complex data into smaller chunks of data for better analysis via MicroStrategy. Better data analytics reports can be generated with the help of MicroStrategy, along with monitoring data in real-time.

22. GOOGLE ANALYTICS: Data scientists are spread across various industries/streams, and one of them is digital marketing. It is one of the top Data Science tools used in the digital marketing industry. A web admin can access, visualize, and analyze the website data via Google Analytics to understand the way users interact with the website. The data trail that users leave behind while using a website can be recognized and used to generate better marketing decisions via Google Analytics. Non-technical professionals can also use it to perform data analytics with its high-end analytics and easy-to-use interface.

23. JULIA: is another programming language specifically designed for Data Science. With its JIT (Just-in-Time) compilation, Julia can match the speed of popular programming languages like C, C++ during Data Science operations. Julia allows you to perform complex statistical calculations related to Data Science in less time. You can manually control the garbage collection process via Julia, and you don't even have to worry about memory management. Its math-friendly syntax and automatic memory management have made it one of the most preferred programming languages for Data Science.

24. SPSS: (Statistical Package for the Social Sciences) used to perform statistical data analysis. SPSS can also be used to process and analyze survey data in less time. One can build predictive models via the Modeler program offered by SPSS. Surveys contain text data, and SPSS can determine the insights from text data in a survey. You can also create various types of data visualizations via SPSS like density chart, radial boxplot, etc.

25. MATLAB: is one of the popular Data Science tools used by organizations/enterprises. It is a programming platform designed for data scientists and helps them access data from flat files, databases, cloud platforms, etc. You can perform feature engineering on a given dataset efficiently with MATLAB. The data types of MATLAB are designed explicitly for Data Science and reduce a lot of time in data pre-processing.

12. Binary search with an example

Binary search is an efficient algorithm for finding an item from a sorted list of items. It works by repeatedly dividing in half the portion of the list that could contain the item, until you've narrowed down the possible locations to just one. One of the most common ways to use binary search is to find an item in an array.

For example, the Tycho-2 star catalog contains information about the brightest 2,539,913 stars in our galaxy. Suppose that you want to search the catalog for a particular star, based on the star's name. If the program examined every star in the star catalog in order starting with the first, an algorithm called linear search, the computer might have to examine all 2,539,913 stars to find the star you were looking for, in the worst case. If the catalog were sorted alphabetically by star names, binary search would not have to examine more than 22 stars, even in the worst case.

13. 7Vs concept of Big data

Volume: When discussing Big Data volumes, almost unimaginable sizes and unfamiliar numerical terms are required: Each day, the world produces 2.5 quintillion bytes of data. That is 2.3 trillion gigabytes.By 2020, we will have created 40 zettabytes of data, which is 43 trillion gigabytes. Most companies already have, on average, 100 terabytes of data stored each. Facebook users upload that many data daily.Walmart alone processes over a million transactions per hour.

Velocity: Underlying the volume numbers is an even larger trend, which is that 90 percent of extant data have been created in just the last two years. The speed at which data are generated, accumulated and analyzed is on a steep acceleration curve. Although most data are warehoused before analysis, there is an increasing need for real-time processing of these enormous volumes, such as the 200 million emails, 300,000 tweets and 100 hours of Youtube videos that are passing by every minute of the day. Real-time processing reduces storage requirements while providing more responsive, accurate and profitable responses.

Variety: Another challenge of Big Data processing goes beyond the massive volumes and increasing velocities of data but also in manipulating the enormous variety of these data. Taken as a whole, these data appear as an indecipherable mass without structure. Consisting of natural language, hashtags, geo-spatial data, multimedia, sensor events and so much more, the extraction of meaning from such diversity requires ever-increasing algorithmic and computational power.

Variability: Furthermore, the intrinsic meanings and interpretations of these conglomerations of raw data depends on its context. This is especially true with natural language processing. A single word may have multiple meanings. New meanings are created and old meanings discarded over time. Interpreting connotations is, for instance, essential to gauging and responding to social media buzz. The boundless variability of Big Data therefore presents a unique decoding challenge if one is to take advantage of its full value.

Veracity: Understanding what Big Data is telling you is one thing. However, it is useless if the data being analyzed are inaccurate or incomplete. This situation arises when data streams originate from diverse sources presenting a variety of formats with varying signal-to-noise ratios. By the time these data arrive at a Big Data analysis stage, they may be rife with accumulated errors that are difficult to sort out. It almost goes without saying that the veracity of the final analysis is degraded without first cleaning up the data it works with.

Visualization: A core task for any Big Data processing system is to transform the immense scale of it into something easily comprehended and actionable. For human consumption, one of the best methods for this is converting it into graphical formats. Spreadsheets and even three-dimensional visualizations are often not up to the task, however, due to the attributes of velocity and variety. There

may be a multitude of spatial and temporal parameters and relationships between them to condense into visual forms. Solving these problems is the main impetus behind AT&T's Nanocubes visual representation package.

Value: No one doubts that Big Data offers an enormous source of value to those who can deal with its scale and unlock the knowledge within. Not only does Big Data offer new, more effective methods of selling but also vital clues to new products to meet previously undetected market demands. Many industries utilize Big Data in the quest for cost reductions for their organizations and their customers. Those who offer the tools and machines to handle Big Data, its analysis and visualization also benefit hugely, albeit indirectly.