LECTURE NOTES

ON

DATA PREPARATION AND ANALYSIS (BCSB13)

Prepared by,

G. Sulakshana, Assistant Professor, CSE Dept.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal, Hyderabad- 500043

MODULE -I

DATA GATHERING AND PREPARATION

BIG DATA ANALYTICS:

The volume of data that one has to deal has exploded to unimaginable levels in the past decade, and at the same time, the price of data storage has systematically reduced. Private companies and research institutions capture terabytes of data about their users' interactions, business, social media, and also sensors from devices such as mobile phones and automobiles. The challenge of this era is to make sense of this sea of data. This is where **big data analytics** comes into picture.

Big Data Analytics largely involves collecting data from different sources, mange it in a way that it becomes available to be consumed by analysts and finally deliver data products useful to the organization business.

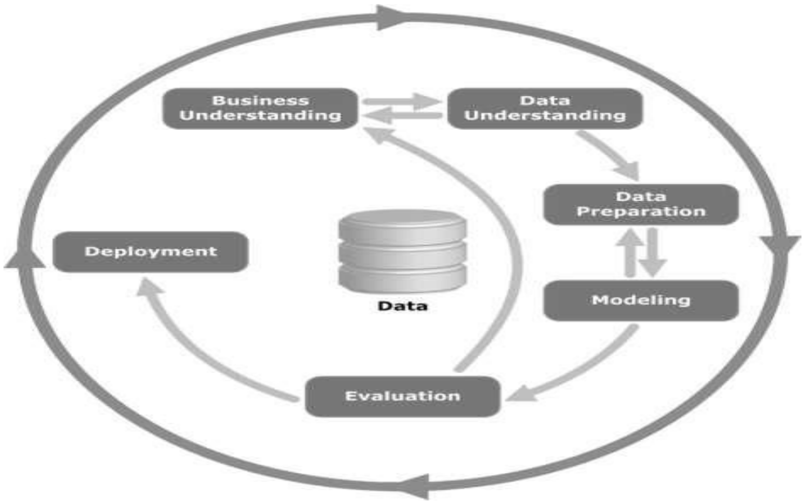
Big Data Analytics - Data Life Cycle: Traditional Data Mining Life Cycle:

In order to provide a framework to organize the work needed by an organization and deliver clear insights from Big Data, it's useful to think of it as a cycle with different stages. It is by no means linear, meaning all the stages are related with each other. This cycle has superficial similarities with the more traditional data mining cycle as described in CRISP methodology.

CRISP-DM Methodology:

The **CRISP-DM methodology** that stands for Cross Industry Standard Process for Data Mining is a cycle that describes commonly used approaches that data mining experts use to tackle problems in traditional BI data mining. It is still being used in traditional BI data mining teams.

Take a look at the following illustration. It shows the major stages of the cycle as described by the CRISP-DM methodology and how they are interrelated.



CRISP-DM was conceived in 1996 and the next year, it got underway as a European Union project under the ESPRIT funding initiative. The project was led by five companies: SPSS, Terradata, Daimler AG, NCR Corporation, and OHRA (an insurance company). The project was finally incorporated into SPSS. The methodology is extremely detailed oriented in how a data mining project should be specified.

Let us now learn a little more on each of the stages involved in the CRISP-DM life cycle −

- **Business Understanding** − This initial phase focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition. A preliminary plan is designed to achieve the objectives. A decision model, especially one built using the Decision Model and Notation standard can be used.

- **Data Understanding** − The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information.
- **Data Preparation** − The data preparation phase covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. Tasks include table, record, and attribute selection as well as transformation and cleaning of data for modeling tools.
- **Modeling** − In this phase, various modeling techniques are selected and applied and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques have specific requirements on the form of data. Therefore, it is often required to step back to the data preparation phase.
- **Evaluation** − At this stage in the project, you have built a model (or models) that appears to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to evaluate the model thoroughly and review the steps executed to construct the model, to be certain it properly achieves the business objectives.

  A key objective is to determine if there is some important business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.
- **Deployment** − Creation of the model is generally not the end of the project. Even if the purpose of the model is to increase knowledge of the data, the knowledge gained will need to be organized and presented in a way that is useful to the customer.

  Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data scoring (e.g. segment allocation) or data mining process.

In many cases, it will be the customer, not the data analyst, who will carry out the deployment steps. Even if the analyst deploys the model, it is important for the customer to understand upfront the actions which will need to be carried out in order to actually make use of the created models.

SEMMA Methodology:

SEMMA is another methodology developed by SAS for data mining modeling. It stands for **S**ample, **E**xplore, **M**odify, **M**odel, and **A**sses. Here is a brief description of its stages −

- **Sample** − The process starts with data sampling, e.g., selecting the dataset for modeling. The dataset should be large enough to contain sufficient information to retrieve, yet small enough to be used efficiently. This phase also deals with data partitioning.
- **Explore** − This phase covers the understanding of the data by discovering anticipated and unanticipated relationships between the variables, and also abnormalities, with the help of data visualization.
- **Modify** − The Modify phase contains methods to select, create and transform variables in preparation for data modeling.
- **Model** − In the Model phase, the focus is on applying various modeling (data mining) techniques on the prepared variables in order to create models that possibly provide the desired outcome.
- **Assess** − The evaluation of the modeling results shows the reliability and usefulness of the created models.

The main difference between CRISM–DM and SEMMA is that SEMMA focuses on the modeling aspect, whereas CRISP-DM gives more importance to stages of the cycle prior to modeling such as understanding the business problem to be solved, understanding and preprocessing the data to be used as input, for example, machine learning algorithms.

Big Data Life Cycle:

In today’s big data context, the previous approaches are either incomplete or suboptimal. For example, the SEMMA methodology disregards completely data collection and preprocessing of different data sources. These stages normally constitute most of the work in a successful big data project.

A big data analytics cycle can be described by the following stage −

- Business Problem Definition
- Research
- Human Resources Assessment
- Data Acquisition
- Data Munging
- Data Storage

- Exploratory Data Analysis
- Data Preparation for Modeling and Assessment
- Modeling
- Implementation

In this section, we will throw some light on each of these stages of big data life cycle.

Business Problem Definition

This is a point common in traditional BI and big data analytics life cycle. Normally it is a non-trivial stage of a big data project to define the problem and evaluate correctly how much potential gain it may have for an organization. It seems obvious to mention this, but it has to be evaluated what are the expected gains and costs of the project.

Research

Analyze what other companies have done in the same situation. This involves looking for solutions that are reasonable for your company, even though it involves adapting other solutions to the resources and requirements that your company has. In this stage, a methodology for the future stages should be defined.

Human Resources Assessment

Once the problem is defined, it’s reasonable to continue analyzing if the current staff is able to complete the project successfully. Traditional BI teams might not be capable to deliver an optimal solution to all the stages, so it should be considered before starting the project if there is a need to outsource a part of the project or hire more people.

Data Acquisition

This section is key in a big data life cycle; it defines which type of profiles would be needed to deliver the resultant data product. Data gathering is a non-trivial step of the process; it normally involves gathering unstructured data from different sources. To give an example, it could involve writing a crawler to retrieve reviews from a website. This involves dealing with text, perhaps in different languages normally requiring a significant amount of time to be completed.

Data Munging

Once the data is retrieved, for example, from the web, it needs to be stored in an easyto-use format. To continue with the reviews examples, let’s assume the data is retrieved from different sites where each has a different display of the data.

Suppose one data source gives reviews in terms of rating in stars, therefore it is possible to read this as a mapping for the response variable **$y \in \{1, 2, 3, 4, 5\}$**. Another data source gives reviews using two arrows system, one for up voting and the other for down voting. This would imply a response variable of the form y ∈ {positive, negative}.

In order to combine both the data sources, a decision has to be made in order to make these two response representations equivalent. This can involve converting the first data source response representation to the second form, considering one star as negative and five stars as positive. This process often requires a large time allocation to be delivered with good quality.

Data Storage

Once the data is processed, it sometimes needs to be stored in a database. Big data technologies offer plenty of alternatives regarding this point. The most common alternative is using the Hadoop File System for storage that provides users a limited version of SQL, known as HIVE Query Language. This allows most analytics task to be done in similar ways as would be done in traditional BI data warehouses, from the user perspective. Other storage options to be considered are MongoDB, Redis, and SPARK.

This stage of the cycle is related to the human resources knowledge in terms of their abilities to implement different architectures. Modified versions of traditional data warehouses are still being used in large scale applications. For example, teradata and IBM offer SQL databases that can handle terabytes of data; open source solutions such as postgreSQL and MySQL are still being used for large scale applications.

Even though there are differences in how the different storages work in the background, from the client side, most solutions provide a SQL API. Hence having a good understanding of SQL is still a key skill to have for big data analytics.

This stage a prioriseems to be the most important topic, in practice, this is not true. It is not even an essential stage. It is possible to implement a big data solution that would be working with real-time data, so in this case, we only need to gather data to develop the model and then implement it in real time. So there would not be a need to formally store the data at all.

Exploratory Data Analysis

Once the data has been cleaned and stored in a way that insights can be retrieved from it, the data exploration phase is mandatory. The objective of this stage is to understand the data, this is normally done with statistical techniques and also plotting the data. This is a good stage to evaluate whether the problem definition makes sense or is feasible.

Data Preparation for Modeling and Assessment

This stage involves reshaping the cleaned data retrieved previously and using statistical preprocessing for missing values imputation, outlier detection, normalization, feature extraction and feature selection.

Modeling

The prior stage should have produced several datasets for training and testing, for example, a predictive model. This stage involves trying different models and looking forward to solving the business problem at hand. In practice, it is normally desired that the model would give some insight into the business. Finally, the best model or combination of models is selected evaluating its performance on a left-out dataset.

Implementation

In this stage, the data product developed is implemented in the data pipeline of the company. This involves setting up a validation scheme while the data product is working, in order to track its performance. For example, in the case of implementing a predictive model, this stage would involve applying the model to new data and once the response is available, evaluate the model.

Big Data Analytics – Methodology:

In terms of methodology, big data analytics differs significantly from the traditional statistical approach of experimental design. Analytics starts with data. Normally we model the data in a way to explain a response. The objectives of this approach are to predict the response behavior or understand how the input variables relate to a response. Normally in statistical experimental designs, an experiment is developed and data is retrieved as a result. This allows generating data in a way that can be used by a statistical model, where certain assumptions hold such as independence, normality, and randomization.

In big data analytics, we are presented with the data. We cannot design an experiment that fulfills our favorite statistical model. In large-scale applications of analytics, a large amount of work (normally 80% of the effort) is needed just for cleaning the data, so it can be used by a machine learning model.

We don’t have a unique methodology to follow in real large-scale applications. Normally once the business problem is defined, a research stage is needed to design the methodology to be used. However general guidelines are relevant to be mentioned and apply to almost all problems.

One of the most important tasks in big data analytics is statistical modeling, meaning supervised and unsupervised classification or regression problems. Once the data is cleaned and preprocessed, available for modeling, care should be taken in evaluating different models with reasonable loss metrics and then once the model is implemented, further evaluation and results should be reported. A common pitfall in predictive modeling is to just implement the model and never measure its performance.

Big Data Analytics - Core Deliverables:

As mentioned in the big data life cycle, the data products that result from developing a big data product are in most of the cases some of the following −

- **Machine learning implementation** − This could be a classification algorithm, a regression model or a segmentation model.
- **Recommender system** − The objective is to develop a system that recommends choices based on user behavior. **Netflix** is the characteristic example of this data product, where based on the ratings of users, other movies are recommended.
- **Dashboard** − Business normally needs tools to visualize aggregated data. A dashboard is a graphical mechanism to make this data accessible.
- **Ad-Hoc analysis** − Normally business areas have questions, hypotheses or myths that can be answered doing ad-hoc analysis with data.

Big Data Analytics - Key Stakeholders:

In large organizations, in order to successfully develop a big data project, it is needed to have management backing up the project. This normally involves finding a way to show the business advantages of the project. We don’t have a unique solution to the problem of finding sponsors for a project, but a few guidelines are given below −

- Check who and where are the sponsors of other projects similar to the one that interests you.
- Having personal contacts in key management positions helps, so any contact can be triggered if the project is promising.
- Who would benefit from your project? Who would be your client once the project is on track?
- Develop a simple, clear, and exiting proposal and share it with the key players in your organization.

The best way to find sponsors for a project is to understand the problem and what would be the resulting data product once it has been implemented. This understanding will give an edge in convincing the management of the importance of the big data project.

Big Data Analytics - Data Analyst:

A data analyst has reporting-oriented profile, having experience in extracting and analyzing data from traditional data warehouses using SQL. Their tasks are normally either on the side of data storage or in reporting general business results. Data warehousing is by no means simple, it is just different to what a data scientist does.

Many organizations struggle hard to find competent data scientists in the market. It is however a good idea to select prospective data analysts and teach them the relevant skills to become a data scientist. This is by no means a trivial task and would normally involve the person doing a master degree in a quantitative field, but it is definitely a viable option. The basic skills a competent data analyst must have are listed below −

- Business understanding
- SQL programming
- Report design and implementation
- Dashboard development

Big Data Analytics - Data Scientist:

The role of a data scientist is normally associated with tasks such as predictive modeling, developing segmentation algorithms, recommender systems, A/B testing frameworks and often working with raw unstructured data.

The nature of their work demands a deep understanding of mathematics, applied statistics and programming. There are a few skills common between a data analyst and a data scientist, for example, the ability to query databases. Both analyze data, but the decision of a data scientist can have a greater impact in an organization.

Here is a set of skills a data scientist normally needs to have −

- Programming in a statistical package such as: R, Python, SAS, SPSS, or Julia
- Able to clean, extract, and explore data from different sources
- Research, design, and implementation of statistical models
- Deep statistical, mathematical, and computer science knowledge

In big data analytics, people normally confuse the role of a data scientist with that of a data architect. In reality, the difference is quite simple. A data architect defines the tools and the architecture the data would be stored at, whereas a data scientist uses this architecture. Of course, a data scientist should be able to set up new tools if needed for ad-hoc projects, but the infrastructure definition and design should not be a part of his task.

Big Data Analytics - Problem Definition:

Through this tutorial, we will develop a project. Each subsequent chapter in this tutorial deals with a part of the larger project in the mini-project section. This is thought to be an applied tutorial section that will provide exposure to a real-world problem. In this case, we would start with the problem definition of the project.

Project Description

The objective of this project would be to develop a machine learning model to predict the hourly salary of people using their curriculum vitae (CV) text as input.

Using the framework defined above, it is simple to define the problem. We can define $X = \{x_1, x_2, ..., x_n\}$ as the CV's of users, where each feature can be, in the simplest way possible, the amount of times this word appears. Then the response is real valued, we are trying to predict the hourly salary of individuals in dollars.

These two considerations are enough to conclude that the problem presented can be solved with a supervised regression algorithm.

Problem Definition

Problem Definition is probably one of the most complex and heavily neglected stages in the big data analytics pipeline. In order to define the problem a data product would solve, experience is mandatory. Most data scientist aspirants have little or no experience in this stage.

Most big data problems can be categorized in the following ways −

- Supervised classification
- Supervised regression
- Unsupervised learning
- Learning to rank

Let us now learn more about these four concepts.

Supervised Classification

Given a matrix of features $X = \{x_1, x_2, ..., x_n\}$ we develop a model M to predict different classes defined as $y = \{c_1, c_2, ..., c_n\}$. For example: Given transactional data of customers in an insurance company, it is possible to develop a model that will predict if a client would churn or not. The latter is a binary classification problem, where there are two classes or target variables: churn and not churn.

Other problems involve predicting more than one class, we could be interested in doing digit recognition, and therefore the response vector would be defined as: $y = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, a-state-of-the-art model would be convolution neural network and the matrix of features would be defined as the pixels of the image.

Supervised Regression

In this case, the problem definition is rather similar to the previous example; the difference relies on the response. In a regression problem, the response $y \in \Re$, this means the response is real valued. For example, we can develop a model to predict the hourly salary of individuals given the corpus of their CV.

Unsupervised Learning

Management is often thirsty for new insights. Segmentation models can provide this insight in order for the marketing department to develop products for different segments. A good approach for developing a segmentation model, rather than thinking of algorithms, is to select features that are relevant to the segmentation that is desired.

For example, in a telecommunications company, it is interesting to segment clients by their cell phone usage. This would involve disregarding features that have nothing to do with the segmentation objective and including only those that do. In this case, this would be selecting features as the number of SMS used in a month, the number of inbound and outbound minutes, etc.

Big Data Analytics - Data Collection:

Data collection plays the most important role in the Big Data cycle. The Internet provides almost unlimited sources of data for a variety of topics. The importance of this area depends on the type of business, but traditional industries can acquire a diverse source of external data and combine those with their transactional data.

For example, let’s assume we would like to build a system that recommends restaurants. The first step would be to gather data, in this case, reviews of restaurants from different websites and store them in a database. As we are interested in raw text, and would use that for analytics, it is not that relevant where the data for developing the model would be stored. This may sound contradictory with the big data main technologies, but in order to implement a big data application, we simply need to make it work in real time.

Data formats:

Data can mean many different things, and there are many ways to classify it. Two of the more common are:

- Primary and Secondary: Primary data is data that you collect or generate. Secondary data is created by other researchers, and could be their primary data, or the data resulting from their research.
- Qualitative and Quantitative: Qualitative refers to text, images, video, sound recordings, observations, etc. Quantitative refers to numerical data.

There are typically five main categories that it can be sorted into for management purposes. The category that you choose will then have an effect upon the choices that you make throughout the rest of your data management plan.

Observational

- Captured in real-time
- Cannot be reproduced or recaptured. Sometimes called ‘unique data’.
- Examples include sensor readings, telemetry, survey results, images, and human observation.

Experimental

- Data from lab equipment and under controlled conditions
- Often reproducible, but can be expensive to do so
- Examples include gene sequences, chromatograms, magnetic field readings, and spectroscopy.

Simulation

- Data generated from test models studying actual or theoretical systems
- Models and metadata where the input more important than the output data
- Examples include climate models, economic models, and systems engineering.

Derived or compiled

- The results of data analysis, or aggregated from multiple sources
- Reproducible (but very expensive)
- Examples include text and data mining, compiled database, and 3D models

Reference or canonical

- Fixed or organic collection datasets, usually peer-reviewed, and often published and curate
- Examples include gene sequence databanks, census data, chemical structures.

Data can come in many forms. Some common ones are text, numeric, multimedia, models, audio, code, software, discipline specific (i.e., FITS in astronomy, CIF in chemistry), video, and instrument.

Data format in information technology may refer to:

- Data type, constraint placed upon the interpretation of data in a type system
- Signal (electrical engineering), a format for signal data used in signal processing
- Recording format, a format for encoding data for storage on a storage medium
- File format, a format for encoding data for storage in a computer file
  - Container format (digital), a format for encoding data for storage by means of a standardized audio/video codecs file format
- Content format, a format for representing media content as data
  - Audio format, a format for encoded sound data

Video format, a format for encoded video data

Recommended Digital Data Formats:

Text, Documentation, Scripts: XML, PDF/A, HTML, Plain Text.
Still Image: TIFF, JPEG 2000, PNG, JPEG/JFIF, DNG (digital negative), BMP, GIF.
Geospatial: Shapefile (SHP, DBF, SHX), GeoTIFF, NetCDF.
Graphic Image:

- raster formats: TIFF, JPEG2000, PNG, JPEG/JFIF, DNG, BMP, GIF.
- vector formats: Scalable vector graphics, AutoCAD Drawing Interchange Format, Encapsulated Postscripts, Shape files.
- cartographic: Most complete data, GeoTIFF, GeoPDF, GeoJPEG2000, Shapefile.

Audio: WAVE, AIFF, MP3, MXF, FLAC.
Video: MOV, MPEG-4, AVI, MXF.
Database: XML, CSV, TAB.

Parsing and Transformation:

In data transformation process data are transformed from one format to another format, that is more appropriate for data mining.

Some Data Transformation Strategies:-

Smoothing: Smoothing is a process of removing noise from the data.

Aggregation: Aggregation is a process where summary or aggregation operations are applied to the data.

Generalization: In generalization low-level data are replaced with high-level data by using concept hierarchies climbing.

Normalization: Normalization scaled attribute data so as to fall within a small specified range, such as 0.0 to 1.0.

Attribute Construction: In Attribute construction, new attributes are constructed from the given set of attributes.

Scalability:

Scalability is the capability of a system, network, or process to handle a growing amount of work, or its potential to be enlarged to accommodate that growth.For example, a system is considered scalable if it is capable of increasing its total output under an increased load when resources (typically hardware) are added. An analogous meaning is implied when the word is used in an economic context, where a company's scalability implies that the underlying businessmodel offers the potential for economicgrowth within the company.

Scalability, as a property of systems, is generally difficult to defineand in any particular case it is necessary to define the specific requirements for scalability on those dimensions that are deemed important. It is a highly significant issue in electronics systems, databases, routers, and networking. A system, whose performance improves after adding hardware, proportionally to the capacity added, is said to be a ***scalable system***.

Scalability can be measured in various dimensions, such as:

- ***Administrative scalability*****:** The ability for an increasing number of organizations or users to easily share a single distributed system.
- *Functional scalability*: The ability to enhance the system by adding new functionality at minimal effort.
- ***Geographic scalability*****:** The ability to maintain performance, usefulness, or usability regardless of expansion from concentration in a local area to a more distributed geographic pattern.
- ***Load scalability*****:** The ability for a distributed system to easily expand and contract its resource pool to accommodate heavier or lighter loads or number of inputs. Alternatively, the ease with which a system or component can be modified, added, or removed, to accommodate changing load.
- ***Generation scalability*****:** The ability of a system to scale up by using new generations of components. Thereby, *heterogeneous scalability* is the ability to use the components from different vendors.

Scalability issues:

- A routing protocol is considered scalable with respect to network size, if the size of the necessary routing table on each node grows as O(log N), where N is the number of nodes in the network.

- A scalable online transaction processing system or database management system is one that can be upgraded to process more transactions by adding new processors, devices and storage, and which can be upgraded easily and transparently without shutting it down.
- Some early peer-to-peer (P2P) implementations of Gnutella had scaling issues. Each node query flooded its requests to all peers. The demand on each peer would increase in proportion to the total number of peers, quickly overrunning the peers' limited capacity. Other P2P systems like BitTorrent scale well because the demand on each peer is independent of the total number of peers. There is no centralized bottleneck, so the system may expand indefinitely without the addition of supporting resources (other than the peers themselves).
- The distributed nature of the Domain Name System allows it to work efficiently even when all hosts on the worldwide Internet are served, so it is said to "scale well".

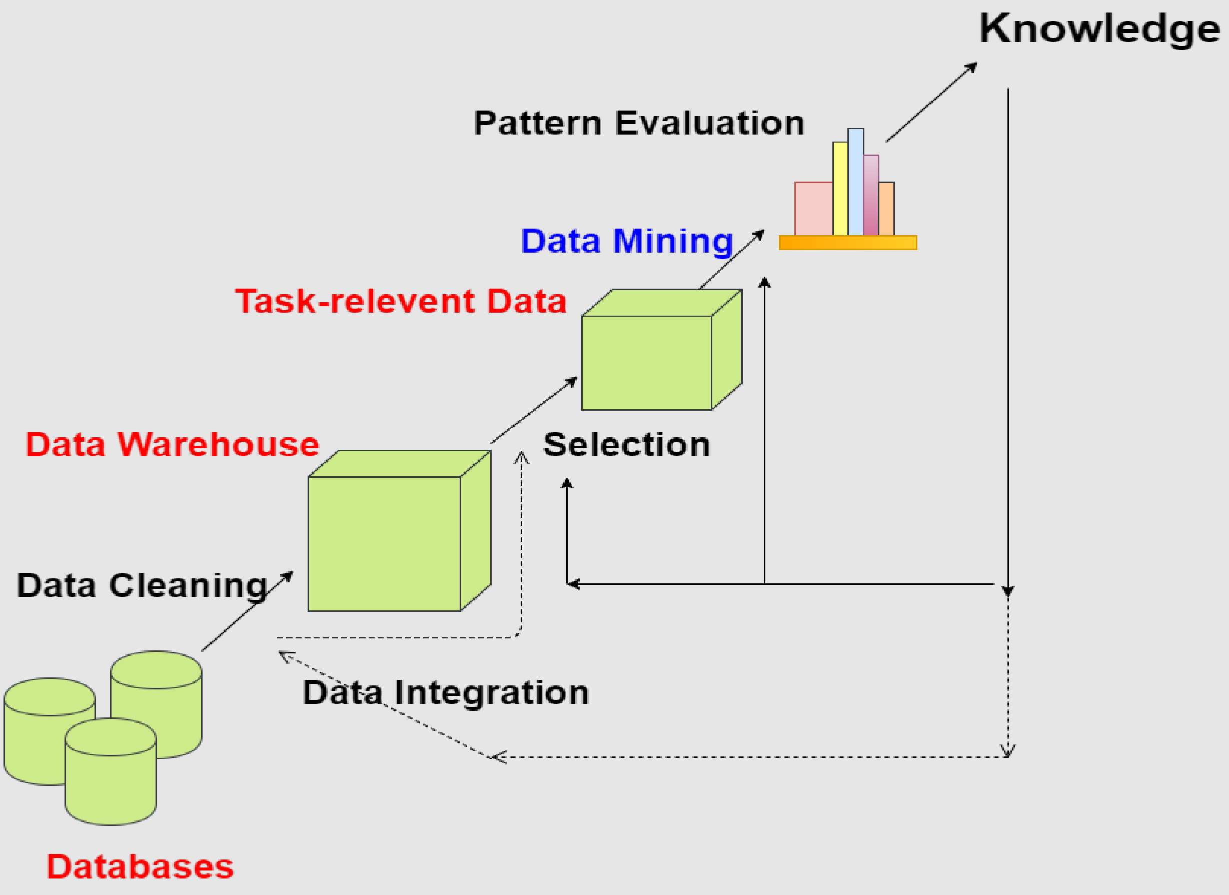
MODULE -II

DATA CLEANING

Data Cleaning:

Data cleaning is a technique that is applied to remove the noisy data and correct the inconsistencies in data. Data cleaning involves transformations to correct the wrong data. Data cleaning is performed as a data preprocessing step while preparing the data for a data warehouse.

KDP is a process of finding knowledge in data; it does this by using data mining methods (algorithms) in order to extract demanding knowledge from large amount of data.



Knowledge Discovery Process (KDP)

Data cleansing or **data cleaning** is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. Data cleansing may be performed interactively with datawrangling tools, or as batchprocessing through scripting.

After cleansing, a dataset should be consistent with other similar data sets in the system. The inconsistencies detected or removed may have been originally caused by user entry errors, by corruption in transmission or storage, or by different datadictionary definitions of similar entities in different stores. Data cleaning differs from datavalidation in that validation almost invariably means data is rejected from the system at entry and is performed at the time of entry, rather than on batches of data.

The actual process of data cleansing may involve removing typographicalerrors or validating and correcting values against a known list of entities. The validation may be strict (such as rejecting any address that does not have a valid postalcode) or fuzzy (such as correcting records that partially match existing, known records). Some data cleansing solutions will clean data by cross checking with a validated data set. A common data cleansing practice is data enhancement, where data is made more complete by adding related information. For example, appending addresses with any phone numbers related to that address. Data cleansing may also involve activities like, harmonization of data, and standardization of data. For example, harmonization of short codes (st, rd, etc.) to actual words (street, road, and etcetera). Standardization of data is a means of changing a reference data set to a new standard, ex, use of standard codes.

- Data cleansing is a valuable process that can help companies save time and increase their efficiency. Datacleansingsoftware tools are used by various organizations to remove duplicate data, fix and amend badly-formatted, incorrect and amend incomplete data from marketing lists, databases and CRM's. They can achieve in a short period of time what could take days or weeks for an administrator working manually to fix. This means that companies can save not only time but money by acquiring data cleaning tools.

Data cleansing is of particular value to organizations that have vast swathes of data to deal with. These organizations can include banks or government organizations but small to medium enterprises can also find a good use for the programs. In fact, it's suggested by many sources that any firm that works with and hold data should invest in cleansing tools. The tools should also be used on a regular basis as inaccurate data levels can grow quickly, compromising database and decreasing business efficiency.

CONSISTENCY:

Data consistency refers to the usability of data:

> The degree to which a set of measures are equivalent in across systems (Consistency). Inconsistency occurs when two data items in the data set contradict each other: e.g., a customer is recorded in two different systems as having two different current addresses, and only one of them can be correct. Fixing inconsistency is not always possible: it requires a variety of strategies - e.g., deciding which data were recorded more recently, which data source is likely to be most reliable (the latter knowledge may be specific to a given organization), or simply trying to find the truth by testing both data items (e.g., calling up the customer).

Point-in-time consistency:

Point-in-time consistency is an important property of backup files and a critical objective of software that creates backups. It is also relevant to the design of disk memory systems, specifically relating to what happens when they are unexpectedly shut down.

As a relevant backup example, consider a website with a database such as the online encyclopedia Wikipedia, which needs to be operational around the clock, but also must be backed up with regularity to protect against disaster. Portions of Wikipedia are constantly being updated every minute of every day, meanwhile, Wikipedia's database is stored on servers in the form of one or several very large files which require minutes or hours to back up.

These large files - as with any database - contain numerous data structures which reference each other by location. For example, some structures are indexes which permit the database subsystem to quickly find search results. If the data structures cease to reference each other properly, then the database can be said to be corrupted.

Counter example:

The importance of point-in-time consistency can be illustrated with what would happen if a backup were made without it.

Assume Wikipedia's database is a huge file, which has an important index located 20% of the way through, and saves article data at the 75% mark. Consider a scenario where an editor comes and creates a new article at the same time a backup is being performed, which is being made as a simple "file copy" which copies from the beginning to the end of the large file(s) and doesn't consider data consistency - and at the time of the article edit, it is 50% complete. The new article is added to the article space (at the 75% mark) and a corresponding index entry is added (at the 20% mark).

Because the backup is already halfway done and the index already copied, the backup will be written with the article data present, but with the index reference missing. As a result of the inconsistency, this file is considered corrupted.

In real life, a real database such as Wikipedia's may be edited thousands of times per hour, and references are virtually always spread throughout the file and can number into the millions, billions, or more. A sequential "copy" backup would literally contain so many small corruptions that the backup would be completely unusable without a lengthy repair process which could provide no guarantee as to the completeness of what has been recovered.

A backup process which properly accounts for data consistency ensures that the backup is a snapshot of how the entire database looked at a single moment. In the given Wikipedia example, it would ensure that the backup was written *without* the added article at the 75% mark, so that the article data would be consistent with the index data previously written.

Disk caching systems:

Point-in-time consistency is also relevant to computer disk subsystems.

Specifically, operating systems and file systemsare designed with the expectation that the computer system they are running on could lose power, crash, fail, or otherwise cease operating at any time. When properly designed, they ensure that data will not be unrecoverably corrupted if the power is lost. Operating systems and file systems do this by ensuring that data is written to a hard disk in a certain order, and rely on that in order to detect and recover from unexpected shutdowns.

On the other hand, rigorously writing data to disk in the order that maximizes data integrity also impacts performance. A process of write caching is used to consolidate and re-sequence write operations such that they can be done faster by minimizing the time spent moving disk heads.

Data consistency concerns arise when write caching changes the sequence in which writes are carried out, because it there exists the possibility of an unexpected shutdown that violates the operating system's expectation that all writes will be committed sequentially.

For example, in order to save a typical document or picture file, an operating system might write the following records to a disk in the following order:

1. Journal entry saying file XYZ is about to be saved into sector 123.
2. The actual contents of the file XYZ are written into sector 123.
3. Sector 123 is now flagged as occupied in the record of free/used space.
4. Journal entry noting the file completely saved, and its name is XYZ and is located in sector 123.

The operating system relies on the assumption that if it sees item #1 is present (saying the file is about to be saved), but that item #4 is missing (confirming success), that the save operation was unsuccessful and so it should undo any incomplete steps already taken to save it (e.g. marking sector 123 free since it never was properly filled, and removing any record of XYZ from the file directory). It relies on these items being committed to disk in sequential order.

Suppose a caching algorithm determines it would be fastest to write these items to disk in the order 4-3-1-2, and starts doing so, but the power gets shut down after 4 get written, before 3, 1 and 2, and so those writes never occur. When the computer is turned back on, the file system would then show it contains a file named XYZ which is located in sector 123, but this sector really does not contain the file. (Instead, the sector will contain garbage, or zeroes, or a random portion of some old file - and that is what will show if the file is opened).

Further, the file system's free space map will not contain any entry showing that sector 123 is occupied, so later; it will likely assign that sector to the next file to be saved, believing it is available. The file system will then have two files both unexpectedly claiming the same sector (known as a cross-linked file). As a result, a write to one of the files will overwrite part of the other file, invisibly damaging it.

A disk caching subsystem that ensures point-in-time consistency guarantees that in the event of an unexpected shutdown, the four elements would be written one of only five possible ways: completely (1-2-3-4), partially (1, 1-2, 1-2-3), or not at all.

High-end hardware disk controllers of the type found in servers include a small **battery back-up unit** on their cache memory so that they may offer the performance gains of write caching while mitigating the risk of unintended shutdowns. The battery back-up unit keeps the memory powered even during a shutdown so that when the computer is powered back up, it can quickly complete any writes it has previously committed. With such a controller, the operating system may request four writes (1-2-3-4) in that order, but the controller may decide the quickest way to write them is 4-3-1-2. The controller essentially *lies* to the operating system and reports that the writes have been completed in order (a lie that improves performance at the expense of data corruption if power is lost), and the battery backup hedges against the risk of data corruption by giving the controller a way to silently fix any and all damage that could occur as a result.

If the power gets shut off after element 4 has been written, the battery backed memory contains the record of commitment for the other three items and ensures that they are written ("flushed") to the disk at the next available opportunity.

Transaction consistency:

Consistency (database systems) in the realm of Distributed databasesystems refers to the property of many ACIDdatabases to ensure that the results of a Database transaction are visible to all nodes simultaneously. That is, once the transaction has been committed all parties attempting to access the database can see the results of that transaction simultaneously.

A good example of the importance of transaction consistency is a database that handles the transfer of money. Suppose a money transfer requires two operations: writing a debit in one place, and a credit in another. If the system crashes or shuts down when one operation has completed but the other has not, and there is nothing in place to correct this, the system can be said to lack transaction consistency. With a money transfer, it is desirable that either the entire transaction completes, or none of it completes. Both of these scenarios keep the balance in check.

Transaction consistency ensures just that - that a system is programmed to be able to detect incomplete transactions when powered on, and undo (or "roll back") the portion of any incomplete transactions that are found.

Application consistency:

Application Consistency, similar to Transaction consistency, is applied on a grander scale. Instead of having the scope of a single transaction, data must be consistent within the confines of many different transaction streams from one or more applications. An application may be made up of many different types of data, various types of files and data feeds from other applications. Application consistency is the state in which all related files and databases are synchronized representing the true status of the application.

In statistics, consistency of procedures, suchas computing confidence intervalsor conducting hypothesis tests, is a desired property of their behavior as the number of items in the data set to which they are applied increases indefinitely. In particular, consistency requires that the outcome of the procedure with unlimited data should identify the underlying truth. Use of the term in statistics derives from Sir Ronald Fisher in 1922.

Use of the terms *consistency* and *consistent* in statistics is restricted to cases where essentially the same procedure can be applied to any number of data items. In complicated applications of statistics, there may be several ways in which the number of data items may grow. For example, records for rainfall within an area might increase in three ways: records for additional time periods; records for additional sites with a fixed area; records for extra sites obtained by extending the size of the area. In such cases, the property of consistency may be limited to one or more of the possible ways a sample size can grow.

HETEROGENEOUS:

Once the data is collected, we normally have diverse data sources with different characteristics. The most immediate step would be to make these data sources homogeneous and continue to develop our data product. However, it depends on the type of data. We should ask ourselves if it is practical to homogenize the data.

Maybe the data sources are completely different, and the information loss will be large if the sources would be homogenized. In this case, we can think of alternatives. Can one data source help me build a regression model and the other one a classification model? Is it possible to work with the heterogeneity on our advantage rather than just lose information? Taking these decisions are what make analytics interesting and challenging.

In the case of reviews, it is possible to have a language for each data source. Again, we have two choices −

- **Homogenization** − It involves translating different languages to the language where we have more data. The quality of translations services is acceptable, but if we would like to translate massive amounts of data with an API, the cost would be significant. There are software tools available for this task, but that would be costly too.

- **Heterogenization** − would it be possible to develop a solution for each language? As it is simple to detect the language of a corpus, we could develop a recommender for each language. This would involve more work in terms of tuning each recommender according to the amount of languages available but is definitely a viable option if we have a few languages available.

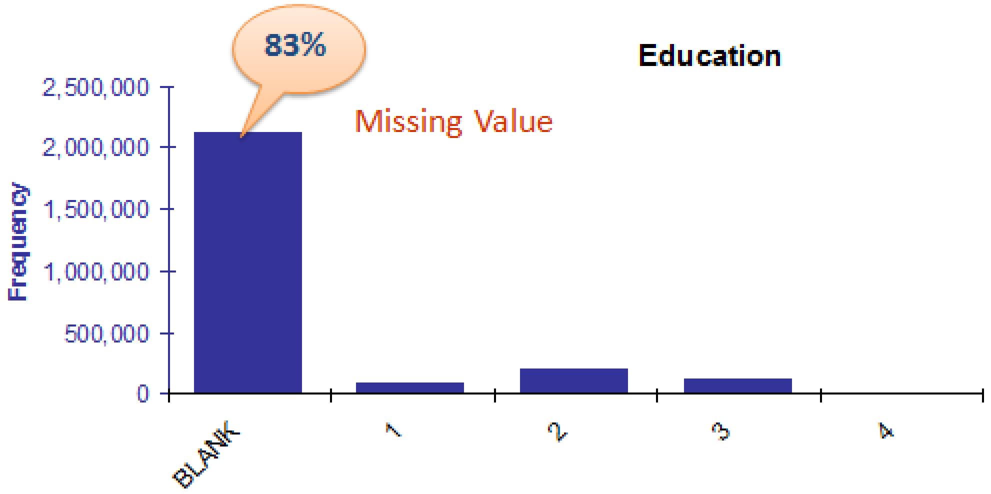
MISSING DATA:

Missing data is always a problem in real life scenarios. Areas like machine learning and data mining face severe issues in the accuracy of their model predictions because of poor quality of data caused by missing values. In these areas, missing value treatment is a major point of focus to make their models more accurate and valid.

When and Why Is Data Missed?

Let us consider an online survey for a product. Many a times, people do not share all the information related to them. Few people share their experience, but not how long they are using the product; few people share how long they are using the product, their experience but not their contact information. Thus, in some or the other way a part of data is always missing, and this is very common in real time.

Missing values are a common occurrence, and you need to have a strategy for treating them. A missing val signify a number of different things in your data. Perhaps the data was not available or not applicable or the ev not happen. It could be that the person who entered the data did not know the right value, or missed filling in mining methods vary in the way they treat missing values. Typically, they ignore the missing values, or exclu records containing missing values, or replace missing values with the mean, or infer missing values from e values.



Missing Values Replacement Policies:

- Ignore the records with missing values.
- Replace them with a global constant (e.g., “?”).
- Fill in missing values manually based on your domain knowledge.
- Replace them with the variable mean (if numerical) or the most frequent value (if categorical).

- Use modeling techniques such as nearest neighbors, Bayes' rule, decision tree, or EM algorithm.

In statistics, **missing data**, or **missing values**, occur when no data value is stored for the variable in an observation. Missing data are a common occurrence and can have a significant effect on the conclusions that can be drawn from the data.

Missing data can occur because of nonresponse: no information is provided for one or more items or for a whole unit ("subject"). Some items are more likely to generate a nonresponse than others: for example items about private subjects such as income. Attrition is a type of missingness that can occur in longitudinal studies—for instance studying development where a measurement is repeated after a certain period of time. Missingness occurs when participants drop out before the test ends and one or more measurements are missing.

Data often are missing in research in economics, sociology, and political science because governments or private entities choose not to, or fail to, report critical statistics, or because the information is not available. Sometimes missing values are caused by the researcher for example, when data collection is done improperly or mistakes are made in data entry.

These forms of missingness take different types, with different impacts on the validity of conclusions from research: Missing completely at random, missing at random, and missing not at random. Missing data can be handled similarly as censored data.

Types of missing data:

Understanding the reasons why data are missing is important for handling the remaining data correctly. If values are missing completely at random, the data sample is likely still representative of the population. But if the values are missing systematically, analysis may be biased. For example, in a study of the relation between IQ and income, if participants with an above-average IQ tend to skip the question 'What is your salary?', analyses that do not take into account this missing at random may falsely fail to find a positive association between IQ and salary. Because of these problems, methodologists routinely advise researchers to design studies to minimize the occurrence of missing values. Graphical models can be used to describe the missing data mechanism in detail.

Missing completely at random:

Values in a data set are **Missing Completely at Random (MCAR)** if the events that lead to any particular data-item being missing are independent both of observable variables and of unobservable parameters of interest, and occur entirely at random.When data are MCAR, the analysis performed on the data is unbiased; however, data are rarely MCAR.

In the case of MCAR, the missingness of data is unrelated to any study variable: thus, the participants with completely observed data are in effect a random sample of all the participants assigned a particular intervention. With MCAR, the random assignment of treatments is assumed to be preserved, but that is usually an unrealistically strong assumption in practice.

Missing at random:

Missing at random (MAR) occurs when the missingness is not random, but where missingness can be fully accounted for by variables where there is complete information.MAR is an assumption that is impossible to verify statistically, we must rely on its substantive reasonableness.[8] An example is that males are less likely to fill in a depression survey but this has nothing to do with their level of depression, after accounting for maleness. Depending on the analysis method, these data can still induce parameter bias in analyses due to the contingent emptiness of cells (male, very high depression may have zero entries). However, if the parameter is estimated with Full Information Maximum Likelihood, MAR will provide asymptotically unbiased estimates.

Missing not at random:

Missing not at random (MNAR) (also known as nonignorable nonresponse) is data that is neither MAR nor MCAR (i.e. the value of the variable that's missing is related to the reason it's missing).To extend the previous example, this would occur if men failed to fill in a depression survey *because*of their level of depression.

Techniques of dealing with missing data:

Missing data reduces the representativeness of the sample and can therefore distort inferences about the population. Generally speaking, there are three main approaches to handle missing data: *Imputation*—where values are filled in the place of missing data, *omission*where samples with invalid data are discarded from further analysis and *analysis*—by directly applying methods unaffected by the missing values.

In some practical application, the experimenters can control the level of missingness, and prevent missing values before gathering the data. For example, in computer questionnaires, it is often not possible to skip a question. A question has to be answered; otherwise one cannot continue to the next. So missing values due to the participant are eliminated by this type of questionnaire, though this method may not be permitted by an ethics board overseeing the research. In survey research, it is common to make multiple efforts to contact each individual in the sample, often sending letters to attempt to persuade those who have decided not to participate to change their minds.However, such techniques can either help or hurt in terms of reducing the negative inferential effects of missing data, because the kind of people who are willing to be persuaded to participate after initially refusing or not being home are likely to be significantly different from the kinds of people who will still refuse or remain unreachable after additional effort.

In situations where missing values are likely to occur, the researcher is often advised on planning to use methods of data analysis methods that are robustto missingness. An analysis is robust when we are confident that mild to moderate violations of the technique's key assumptions will produce little or no bias, or distortion in the conclusions drawn about the population.

Imputation:

Imputation (statistics):

Some data analysis techniques are not robust to missingness, and require to "fill in", or impute the missing data. Rubin argued that repeating imputation even a few times (5 or less) enormously improves the quality of estimation. For many practical purposes, 2 or 3 imputations capture most of the relative efficiency that could be captured with a larger number of imputations. However, a too-small number of imputations can lead to a substantial loss of statistical power, and some scholars now recommend 20 to 100 or more.Any multiply-imputed data analysis must be repeated for each of the imputed data sets and, in some cases, the relevant statistics must be combined in a relatively complicated way.

The expectation-maximization algorithmis an approach in which values of the statistics which would be computed if a complete dataset were available are estimated (imputed), taking into account the pattern of missing data. In this approach, values for individual missing data-items are not usually imputed.

Interpolation:

In the mathematical field of numerical analysis, interpolation is a method of constructing new data points within the range of a discrete set of known data points.

In the comparison of two paired samples with missing data, a test statistic that uses all available data without the need for imputation is the partially overlapping samples t-test.This is valid under normality and assuming MCAR

Partial deletion:

Methods which involve reducing the data available to a dataset having no missing values include:

- Listwise deletion/casewise deletion
- Pairwise deletion

Full analysis:

Methods which take full account of all information available, without the distortion resulting from using imputed values as if they were actually observed:

- Generative approaches:
  - The expectation-maximization algorithm
  - full information maximum likelihoodestimation
- Discriminative approaches:
  - Max-margin classification of data with absent features

Model-based techniques:

Model based techniques, often using graphs, offer additional tools for testing missing data types (MCAR, MAR, MNAR) and for estimating parameters under missing data conditions. For example, a test for refuting MAR/MCAR reads as follows:

For any three variables X,Y, and Z where Z is fully observed and X and Ypartially observed, the data should satisfy: .

In words, the observed portion of X should be independent on the missingness status of Y, conditional on every value of Z. Failure to satisfy this condition indicates that the problem belongs to the MNAR category.

Remark: These tests are necessary for variable-based MAR which is a slight variation of event-based MAR.

When data falls into MNAR category techniques are available for consistently estimating parameters when certain conditions hold in the model.[3] For example, if Y explains the reason for missingness in X and Y itself has missing values, the joint probability distribution of X and Y can still be estimated if the missingness of Y is random. The estimand in this case will be:

Different model structures may yield different estimands and different procedures of estimation whenever consistent estimation is possible. The preceding estimand calls for first estimating from complete data and multiplying it by estimated from cases in which Y is observed regardless of the status of X. Moreover, in order to obtain a consistent estimate it is crucial that

the first term be as opposed to In many cases model based techniques permit the model structure to undergo refutation tests.Any model which implies the independence between a partially

observed variable X and the missingness indicator of another variable Y (i.e.), conditional

on can be submitted to the following refutation test: .

Finally, the estimands that emerge from these techniques are derived in closed form and do not require iterative procedures such as Expectation Maximization that are susceptible to local optima.

A special class of problems appears when the probability of the missingness depends on time. For example, in the trauma databases the probability to lose data about the trauma outcome depends on the day after trauma. In these cases various non-stationary Markov chain models are applied.

Data Transformation:

Data transformation is the process of converting data or information from one format to another, usually from the format of a source system into the required format of a new destination system. The usual process involves converting documents, but data conversions sometimes involve the conversion of a program from one computer language to another to enable the program to run on a different platform. The usual reason for this data migration is the adoption of a new system that's totally different from the previous one.

In computing, data transformation is the process of converting data from one format or structure into another format or structure. It is a fundamental aspect of most data integrationand data management tasks such as data wrangling, data warehousing, data integration and application integration.

Data transformation can be simple or complex based on the required changes to the data between the source (initial) data and the target (final) data. Data transformation is typically performed via a mixture of manual and automated steps. Tools and technologies used for data transformation can vary widely based on the format, structure, complexity, and volume of the data being transformed.

A master data recast is another form of data transformation where the entire database of data values is transformed or recast without extracting the data from the database. All data in a well-designed database is directly or indirectly related to a limited set of master database tables by a network of foreign key constraints. Each foreign key constraint is dependent upon a unique database index from the parent database table. Therefore, when the proper master database table is recast with a different unique index, the directly and indirectly related data are also recast or restated. The directly and indirectly related data may also still be viewed in the original form since the original unique index still exists with the master data. Also, the database recast must be done in such a way as to not impact the applications architecture software.

When the data mapping is indirect via a mediating data model, the process is also called **data mediation**.

Data Transformation Process:

Data transformation can be divided into the following steps, each applicable as needed based on the complexity of the transformation required.

- Data discovery
- Data mapping
- Code generation
- Code execution
- Data review

These steps are often the focus of developers or technical data analysts who may use multiple specialized tools to perform their tasks.

The steps can be described as follows:

Data discovery is the first step in the data transformation process. Typically the data is profiled using profiling tools or sometimes using manually written profiling scripts to better understand the structure and characteristics of the data and decide how it needs to be transformed.

Data mapping is the process of defining how individual fields are mapped, modified, joined, filtered, and aggregated etc. to produce the final desired output. Developers or technical data analysts traditionally perform data mapping since they work in the specific technologies to define the transformation rules (e.g. visual ETL tools, transformation languages).

Code generation is the process of generating executable code (e.g. SQL, Python, R, or other executable instructions) that will transform the data based on the desired and defined data mapping rules.[4] Typically, the data transformation technologies generate this codebased on the definitions or metadata defined by the developers.

Code execution is the step whereby the generated code is executed against the data to create the desired output. The executed code may be tightly integrated into the transformation tool, or it may require separate steps by the developer to manually execute the generated code.

Data review is the final step in the process, which focuses on ensuring the output data meets the transformation requirements. It is typically the business user or final end-user of the data that performs this step. Any anomalies or errors in the data that are found and communicated back to the developer or data analyst as new requirements to be implemented in the transformation process

Types of Data Transformation:

Batch Data Transformation:

Traditionally, data transformation has been a bulk or batch process,whereby developers write code or implement transformation rules in a data integration tool, and then execute that code or those rules on large volumes of data.This process can follow the linear set of steps as described in the data transformation process above.

Batch data transformation is the cornerstone of virtually all data integration technologies such as data warehousing, data migration and application integration.

When data must be transformed and delivered with low latency, the term “microbatch” is often used.[6] This refers to small batches of data (e.g. a small number of rows or small set of data objects) that can be processed very quickly and delivered to the target system when needed.

Benefits of Batch Data Transformation:

Traditional data transformation processes have served companies well for decades. The various tools and technologies (data profiling, data visualization, data cleansing, data integration etc.) have matured and most (if not all) enterprises transform enormous volumes of data that feed internal and external applications, data warehouses and other data stores.

Limitations of Traditional Data Transformation:

This traditional process also has limitations that hamper its overall efficiency and effectiveness.

The people who need to use the data (e.g. business users) do not play a direct role in the data transformation process.Typically, users hand over the data transformation task to developers who have the necessary coding or technical skills to define the transformations and execute them on the data.

This process leaves the bulk of the work of defining the required transformations to the developer. The developer interprets the business user requirements and implements the related code/logic. This has the potential of introducing errors into the process (through misinterpreted requirements), and also increases the time to arrive at a solution.

This problem has given rise to the need for agility and self-service in data integration (i.e. empowering the user of the data and enabling them to transform the data themselves interactively).

There are companies that provide self-service data transformation tools. They are aiming to efficiently analyze, map and transform large volumes of data without the technical and process complexity that currently exists. While these companies use traditional batch transformation, their tools enable more interactivity for users through visual platforms and easily repeated scripts.

Interactive Data Transformation:

Interactive data transformation (IDT)is an emerging capability that allows business analysts and business users the ability to directly interact with large datasets through a visual interface,understand the characteristics of the data (via automated data profiling or visualization), and change or correct the data through simple interactions such as clicking or selecting certain elements of the data.

Although IDT follows the same data integration process steps as batch data integration, the key difference is that the steps are not necessarily followed in a linear fashion and typically don't require significant technical skills for completion.

A number of companies, primarily start-ups such as Trifacta, Alteryx and Paxata provide interactive data transformation tools. They are aiming to efficiently analyze, map and transform large volumes of data without the technical and process complexity that currently exists.

IDT solutions provide an integrated visual interface that combines the previously disparate steps of data analysis, data mapping and code generation/execution and data inspection.IDT interfaces incorporate visualization to show the user patterns and anomalies in the data so they can identify erroneous or outlying values.

Once they've finished transforming the data, the system can generate executable code/logic, which can be executed or applied to subsequent similar data sets.

By removing the developer from the process, IDT systems shorten the time needed to prepare and transform the data, eliminate costly errors in interpretation of user requirements and empower business users and analysts to control their data and interact with it as needed.

Transformational languages:

There are numerous languages available for performing data transformation. Many transformation languages require a grammar to be provided. In many cases, the grammar is structured using something closely resembling Backus–Naur Form (BNF). There are numerous languages available for such purposes varying in their accessibility (cost) and general usefulness. Examples of such languages include:

- AWK - one of the oldest and popular textual data transformation language;
- Perl - a high-level language with both procedural and object-oriented syntax capable of powerful operations on binary or text data.
- Template languages - specialized to transform data into documents (see also template processor);
- TXL - prototyping language-based descriptions, used for source code or data transformation.
- XSLT - the standard XML data transformation language (suitable by XQuery in many applications);

Additionally, companies such as Trifacta and Paxata have developed domain-specific transformational languages (DSL) for servicing and transforming datasets. The development of domain-specific languages has been linked to increased productivity and accessibility for non-technical users. Trifacta's "Wrangle" is an example of such a domain specific language.

Another advantage of the recent DSL trend is that a DSL can abstract the underlying execution of the logic defined in the DSL, but it can also utilize that same logic in various processing engines, such as Spark, MapReduce, and Dataflow. With a DSL, the transformation language is not tied to the engine.

Although transformational languages are typically best suited for transformation, something as simple as regular expressions can be used to achieve useful transformation. A text editorlike emacsor TextPad supports the use of regular expressions with arguments. This would allow all instances of a particular pattern to be replaced with another pattern using parts of the original pattern. For

In other words, all instances of a function invocation of foo with three arguments, followed by a function invocation with two arguments would be replaced with a single function invocation using some or all of the original set of arguments.

Another advantage to using regular expressions is that they will not fail the null transform test. That is, using your transformational language of choice, run a sample program through a transformation that doesn't perform any transformations. Many transformational languages will fail this test.

Data segmentation:

Data segmentation is the process of taking your **data** and **segmenting** it so that you can use it more efficiently within marketing and operations.

How can data segmentation help?

Data segmentation will allow you to communicate a relevant and targeted message to each segment identified. By segmenting your data, you will be able to identify different levels of your customer database and allow messaging to be tailored and sophisticated to suit your target market.

Single Customer View: A single customer view is a consolidated, consistent and holistic representation of the data known by an organization about its customers.

Data Quality Blo: Read all about the latest trends in the market and expert insights around Data Quality.

Data segmentation can help with personalization:

Even with all of the personalization advances, the improvement is not without its challenges. According to the recent study, 94 percent of respondents have challenges related to personalization.

The biggest challenge is gaining insight quickly enough. This is followed by having enough data and inaccurate information.

What do we mean by segmentation?

Within our overall customer database, which ones have something in common. We need to find the groups of people, understand them and make some commercial value from the different groups. The three types of segmentation:

- **Demographic** – You can segment visitors by demographic but when we are looking at websites, this information is not great as we can't make great use of it
- **Attitudinal** – You won't know if customers are happy or unhappy until they complete a survey online but that isn't going to be helpful initially
- **Behavioral** – In the online world, segmenting visitors by behaviour is key. We can optimise a website experience a lot faster with this type of segmentation

If you run a campaign which ends up bringing in more first time customers, your overall conversion rate will drop. If you had segmented your data first and understood how your new vs. returning customers convert first, you would look to create a campaign to increase sales from the returning customers as they convert at a higher rate.

The Data Mining Process

This is a simple analytical process which you will continuously need to refine. Once you get to one stage, you will almost certainly find that you need to go back a step and refine some more before you finally get the data into a format that you can use.

Business Understanding > Data Understanding > Data Preparation > Analysis and Modeling> Evaluation > Deployment

To Summarise:

- Segmentation is critical in extracting insight from any digital data
- Means are generally meaningless and useless
- Never underestimate the data cleaning and integration aspects of any data mining

- To get meaningful and useful segments takes a lot of iteration, a lot of iteration, a lot of iteration….

MODULE -III

EXPLORATORY ANALYSIS

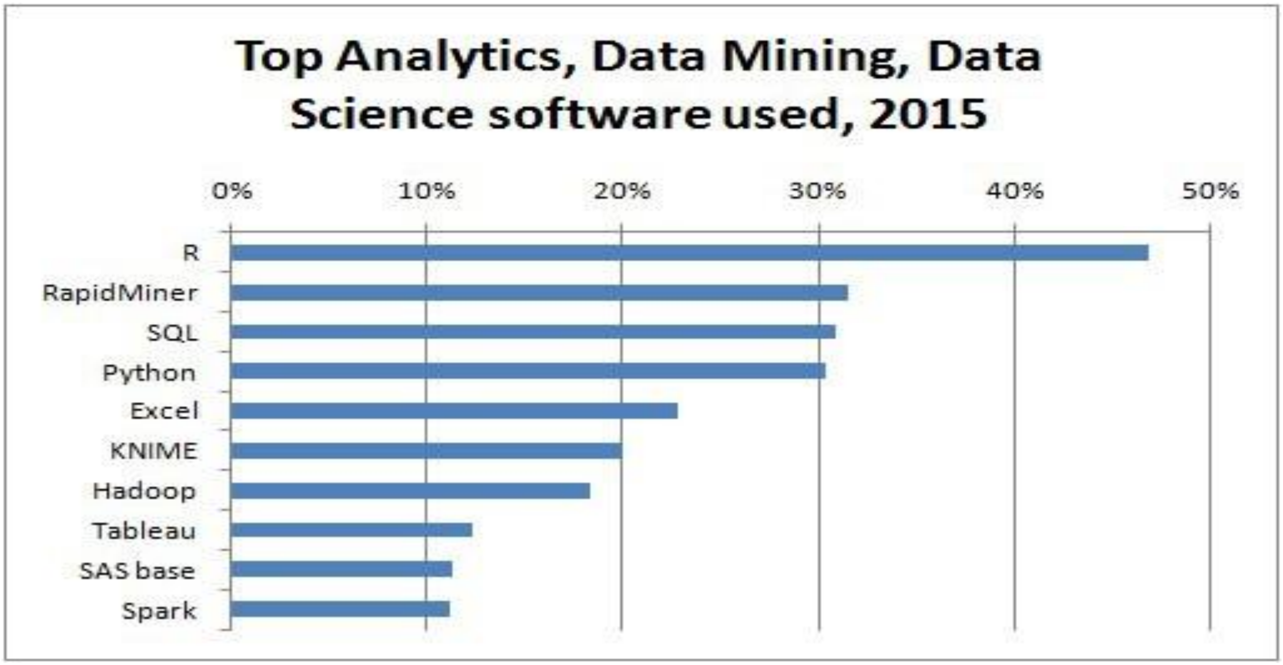
Exploratory data analysis:

In statistics, exploratory data analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics, often with visual methods. A statistical model can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modeling or hypothesis testing task.

Exploratory data analysis is a concept developed by John Tuckey (1977) that consists on a new perspective of statistics. Tuckey's idea was that in traditional statistics, the data was not being explored graphically, it was just being used to test hypotheses. The first attempt to develop a tool was done in Stanford, the project was called prim9. The tool was able to visualize data in nine dimensions, therefore it was able to provide a multivariate perspective of the data.

In recent days, exploratory data analysis is a must and has been included in the big data analytics life cycle. The ability to find insight and be able to communicate it effectively in an organization is fueled with strong EDA capabilities.

Based on Tuckey's ideas, Bell Labs developed the S programming languagein order to provide an interactive interface for doing statistics. The idea of S was to provide extensive graphical capabilities with an easy-to-use language. In today's world, in the context of Big Data, R that is based on the Sprogramming language is the most popular software for analytics.



The following program demonstrates the use of exploratory data analysis.

In statistics, exploratory data analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics, often with visual methods. A statistical model can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modeling or hypothesis testing task. Exploratory data analysis was promoted by John Tuckey to encourage statisticians to explore the data, and possibly formulate hypotheses that could lead to new data collection and experiments. EDA is different from initial data analysis (IDA), which focuses more narrowly on checking assumptions required for model fitting and hypothesis testing, and handling missing values and making transformations of variables as needed. EDA encompasses IDA.

Exploratory Data Analysis in Tuckey held that too much emphasis in statistics was placed on statistical hypothesis testing(confirmatory data analysis); more emphasis needed to be placed on using data to suggest hypotheses to test. In particular, he held that confusing the two types of analyses and employing them on the same set of data can lead to systematicbias owing to the issues inherent in testing hypotheses suggested by the data.

The objectives of EDA are to:

- Suggest hypotheses about the causes of observed phenomena
- Assess assumptions on which statistical inference will be based
- Support the selection of appropriate statistical tools and techniques
- Provide a basis for further data collection through surveys or experiments

Many EDA techniques have been adopted into data mining, as well as into big data analytics.They are also being taught to young students as a way to introduce them to statistical thinking.

DESCRIPTIVE ANALYSIS:

Descriptive statistics are used to describe the basic features of the data in a study. They provide simple summaries about the sample and the measures. Together with simple graphics analysis, they form the basis of virtually every quantitative analysis of data.

Descriptive statistics are typically distinguished from inferential statistics. With descriptive statistics you are simply describing what is or what the data shows. With inferential statistics, you are trying to reach conclusions that extend beyond the immediate data alone. For instance, we use inferential statistics to try to infer from the sample data what the population might think. Or, we use inferential statistics to make judgments of the probability that an observed difference between groups is a dependable one or one that might have happened by chance in this study. Thus, we use inferentialstatistics to make inferences from our data to more general conditions; we use descriptive statistics simply to describe what's going on in our Exploratory data analysis data.

Descriptive Statistics are used to present quantitative descriptions in a manageable form. In a research study we may have lots of measures. Or we may measure a large number of people on any measure. Descriptive statistics help us to simplify large amounts of data in a sensible way. Each descriptive statistic reduces lots of data into a simpler summary. For instance, consider a simple number used to summarize how well a batter is performing in baseball, the batting average. This single number is simply the number of hits divided by the number of times at bat (reported to three significant digits). A batter who is hitting .333 is getting a hit one time in every three at bats. One batting .250 is hitting one time in four. The single number describes a large number of discrete events. Or, consider the scourge of many students, the Grade Point Average (GPA). This single number describes the general performance of a student across a potentially wide range of course experiences.

Every time you try to describe a large set of observations with a single indicator you run the risk of distorting the original data or losing important detail. The batting average doesn't tell you whether the batter is hitting home runs or singles. It doesn't tell whether she's been in a slump or on a streak. The GPA doesn't tell you whether the student was in difficult courses or easy ones, or whether they were courses in their major field or in other disciplines. Even given these limitations, descriptive statistics provide a powerful summary that may enable comparisons across people or other units.

Univariate Analysis:

Univariate analysis involves the examination across cases of one variable at a time. There are three major characteristics of a single variable that we tend to look at:

- the distribution
- the central tendency
- the dispersion

In most situations, we would describe all three of these characteristics for each of the variables in our study.

The Distribution: The distribution is a summary of the frequency of individual values or ranges of values for a variable. The simplest distribution would list every value of a variable and the number of persons who had each value. For instance, a typical way to describe the distribution of college students is by year in college, listing the number or percent of students at each of the four years. Or, we describe gender by listing the number or percent of males and females. In these cases, the variable has few enough values that we can list each one and summarize how many sample cases had the value. But what do we do for a variable like income or GPA? With these variables there can be a large number of possible values, with relatively few people having each one. In this case, we group the raw scores into categories according to ranges of values. For instance, we might look at GPA according to the letter grade ranges. Or, we might group income into four or five ranges of income values.

Frequency distribution table.

One of the most common ways to describe a single variable is with a frequency distribution. Depending on the particular variable, all of the data values may be represented, or you may group the values into categories first (e.g., with age, price, or temperature variables, it would usually not be sensible to determine the frequencies for each value. Rather, the value are grouped into ranges and the frequencies determined.). Frequency distributions can be depicted in two ways, as a table or as a graph. Table 1 shows an age frequency distribution with five categories of age ranges defined. The same frequency distribution can be depicted in a graph as shown in Figure 1. This type of graph is often referred to as a histogram or bar chart.

Frequency distribution bar chart.

Distributions may also be displayed using percentages. For example, you could use percentages to describe the:

- percentage of people in different income levels
- percentage of people in different age ranges
- percentage of people in different ranges of standardized test scores

Central Tendency: The central tendency of a distribution is an estimate of the "center" of a distribution of values. There are three major types of estimates of central tendency:

- Mean
- Median
- Mode

The Mean or average is probably the most commonly used method of describing central tendency. To compute the mean all you do is add up all the values and divide by the number of values. For example, the mean or average quiz score is determined by summing all the scores and dividing by the number of students taking the exam. For example, consider the test score values:

15, 20, 21, 20, 36, 15, 25, 15

The sum of these 8 values is 167, so the mean is 167/8 = 20.875.

The Median is the score found at the exact middle of the set of values. One way to compute the median is to list all scores in numerical order, and then locate the score in the center of the sample. For example, if there are 500 scores in the list, score #250 would be the median. If we order the 8 scores shown above, we would get:

15, 15,15,20,20,21,25,36

There are 8 scores and score #4 and #5 represent the halfway point. Since both of these scores are 20, the median is 20. If the two middle scores had different values, you would have to interpolate to determine the median.

The mode is the most frequently occurring value in the set of scores. To determine the mode, you might again order the scores as shown above, and then count each one. The most frequently occurring value is the mode. In our example, the value 15 occurs three times and is the model. In some distributions there is more than one modal value. For instance, in a bimodal distribution there are two values that occur most frequently.

Notice that for the same set of 8 scores we got three different values -- 20.875, 20, and 15 -- for the mean, median and mode respectively. If the distribution is truly normal (i.e., bell-shaped), the mean, median and mode are all equal to each other.

Dispersion: Dispersion refers to the spread of the values around the central tendency. There are two common measures of dispersion, the range and the standard deviation. The range is simply the highest value minus the lowest value. In our example distribution, the high value is 36 and the low is 15, so the range is 36 - 15 = 21.

The Standard Deviation is a more accurate and detailed estimate of dispersion because an outlier can greatly exaggerate the range (as was true in this example where the single outlier value of 36 stands apart from the rest of the values. The Standard Deviation shows the relation that set of scores has to the mean of the sample. Again let’s take the set of scores:

15, 20,21,20,36,15,25,15

to compute the standard deviation, we first find the distance between each value and the mean.

Comparative analysis:

comparative analysis as comparison analysis: Use comparison analysis to measure the financial relationships between variables over two or more reporting periods. Businesses use comparative analysis as a way to identify their competitive positions and operating results over a defined period. Larger organizations may often comprise the resources to perform financial comparative analysis monthly or quarterly, but it is recommended to perform an annual financial comparison analysis at a minimum.

Financial Comparatives:

Financial statements outline the financial comparatives, which are the variables defining operating activities, investing activities and financing activities for a company. Analysts assess company financial statements using percentages, ratios and amounts when making financial comparative analysis. This information is the business intelligence decision maker's use for determining future businessdecisions. A financial comparison analysis may also be performed to determine companyprofitability and stability. For example, management of a new venture may make a financial comparison analysis periodically to evaluate company performance. Determining losses prematurely and redefining processes in a shorter period will favor compared to unforeseen annual losses.

Comparative Format:

The comparative format for comparative analysis in accounting is a side by side view of the financial comparatives in the financial statements. Comparative analysis accounting identifies an organization's financial performance. For example, income statements identify financial comparables such as company income, expenses, and profit over a period of time. A comparison analysis report identifies where a business meets or exceeds budgets. Potential lenders will also utilize this information to determine a company's credit limit.

Comparative Analysis in Business:

Financial statements play a pivotal role in comparative analysis in business. By analyzing financial comparatives, businesses are able to pinpoint significant trends and project future trends with the identification of considerable or abnormal changes. Businesscomparative analysis against others in their industry allows a company to evaluate industry results and gauge overall companyperformance. Different factors such as political events, economics changes, or industry changes influence the changes in trends. Companies may often document significant events in their financial statements that have a major influence on a change in trends.

CLUSTERING:

Cluster Analysis

Cluster is a group of objects that belongs to the same class. In other words, similar objects are grouped in one cluster and dissimilar objects are grouped in another cluster.

What is Clustering?

Clustering is the process of making a group of abstract objects into classes of similar objects.

Points to Remember

- A cluster of data objects can be treated as one group.
- While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups.
- The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups.

Applications of Cluster Analysis

- Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
- Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
- In the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations.
- Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
- Clustering also helps in classifying documents on the web for information discovery.
- Clustering is also used in outlier detection applications such as detection of credit card fraud.
- As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.

Requirements of Clustering in Data Mining

The following points throw light on why clustering is required in data mining −

- Scalability − we need highly scalable clustering algorithms to deal with large databases.

- Ability to deal with different kinds of attributes − Algorithms should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data.
- Discovery of clusters with attribute shape − the clustering algorithm should be capable of detecting clusters of arbitrary shape. They should not be bounded to only distance measures that tend to find spherical cluster of small sizes.
- High dimensionality − the clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space.
- Ability to deal with noisy data − Databases contain noisy, missing or erroneous data. Some algorithms are sensitive to such data and may lead to poor quality clusters.
- Interpretability − the clustering results should be interpretable, comprehensible, and usable.

Clustering Methods:

Clustering methods can be classified into the following categories −

- Partitioning Method
- Hierarchical Method
- Density-based Method
- Grid-Based Method
- Model-Based Method
- Constraint-based Method

Partitioning Method

Suppose we are given a database of ‘n’ objects and the partitioning method constructs ‘k’ partition of data. Each partition will represent a cluster and k ≤ n. It means that it will classify the data into k groups, which satisfy the following requirements −

- Each group contains at least one object.
- Each object must belong to exactly one group.

Points to remember −

- For a given number of partitions (say k), the partitioning method will create an initial partitioning.

- Then it uses the iterative relocation technique to improve the partitioning by moving objects from one group to other.

Hierarchical Methods

This method creates a hierarchical decomposition of the given set of data objects. We can classify hierarchical methods on the basis of how the hierarchical decomposition is formed. There are two approaches here −

- Agglomerative Approach
- Divisive Approach

Agglomerative Approach

This approach is also known as the bottom-up approach. In this, we start with each object forming a separate group. It keeps on merging the objects or groups that are close to one another. It keep on doing so until all of the groups are merged into one or until the termination condition holds.

Divisive Approach

This approach is also known as the top-down approach. In this, we start with all of the objects in the same cluster. In the continuous iteration, a cluster is split up into smaller clusters. It is down until each object in one cluster or the termination condition holds. This method is rigid, i.e., once a merging or splitting is done, it can never be undone.

Approaches to Improve Quality of Hierarchical Clustering

Here are the two approaches that are used to improve the quality of hierarchical clustering −

- Perform careful analysis of object linkages at each hierarchical partitioning.
- Integrate hierarchical agglomeration by first using a hierarchical agglomerative algorithm to group objects into micro-clusters, and then performing macro-clustering on the micro-clusters.

Density-based Method

This method is based on the notion of density. The basic idea is to continue growing the given cluster as long as the density in the neighborhood exceeds some threshold, i.e., for each data point within a given cluster, the radius of a given cluster has to contain at least a minimum number of points.

Grid-based Method

In this, the objects together form a grid. The object space is quantized into finite number of cells that form a grid structure.

Advantages

- The major advantage of this method is fast processing time.
- It is dependent only on the number of cells in each dimension in the quantized space.

Model-based methods

In this method, a model is hypothesized for each cluster to find the best fit of data for a given model. This method locates the clusters by clustering the density function. It reflects spatial distribution of the data points.

This method also provides a way to automatically determine the number of clusters based on standard statistics, taking outlier or noise into account. It therefore yields robust clustering methods.

Constraint-based Method

In this method, the clustering is performed by the incorporation of user or application-oriented constraints. A constraint refers to the user expectation or the properties of desired clustering results. Constraints provide us with an interactive way of communication with the clustering process. Constraints can be specified by the user or the application requirement.

ASSOCIATION:

Association Rule Mining, as the name suggests, association rules are simple If/Then statements that help discover relationships between seemingly independent relational databases or other data repositories.

Most machine learning algorithms work with numeric datasets and hence tend to be mathematical. However, association rule mining is suitable for non-numeric, categorical data and requires just a little bit more than simple counting.

Association rule mining is a procedure which aims to observe frequently occurring patterns, correlations, or associations from datasets found in various kinds of databases such as relational databases, transactional databases, and other forms of repositories.

An association rule has two parts:

- an antecedent (if) and
- a consequent (then).

An antecedent is something that's found in data, and a consequent is an item that is found in combination with the antecedent. Have a look at this rule for instance:

“If a customer buys bread, he’s 70% likely of buying milk.”

In the above association rule, bread is the antecedent and milk is the consequent. Simply put, it can be understood as a retail store’s association rule to target their customers better. If the above rule is a result of thorough analysis of some data sets, it can be used to not only improve customer service but also improve the company’s revenue.

Association rules are created by thoroughly analyzing data and looking for frequent if/then patterns. Then, depending on the following two parameters, the important relationships are observed:

- Support: Support indicates how frequently the if/then relationship appears in the database.
- Confidence: Confidence tells about the number of times these relationships have been found to be true.

So, in a given transaction with multiple items, Association Rule Mining primarily tries to find the rules that govern how or why such products/items are often bought together. For example, peanut butter and jelly are frequently purchased together because a lot of people like to make PB&J sandwiches.

Association Rule Mining is sometimes referred to as “Market Basket Analysis”, as it was the first application area of association mining. The aim is to discover associations of items occurring together more often than you’d expect from randomly sampling all the possibilities. The classic anecdote of Beer and Diaper will help in understanding this better.

The story goes like this: young American men who go to the stores on Fridays to buy diapers have a predisposition to grab a bottle of beer too. However unrelated and vague that may sound to us laymen, association rule mining shows us how and why!

Let’s do a little analytics ourselves, shall we?

Suppose an X store’s retail transactions database includes the following data:

- Total number of transactions: 600,000
- Transactions containing diapers: 7,500 (1.25 percent)
- Transactions containing beer: 60,000 (10 percent)
- Transactions containing both beer and diapers: 6,000 (1.0 percent)

From the above figures, we can conclude that if there was no relation between beer and diapers (that is, they were statistically independent), then we would have got only 10% of diaper purchasers to buy beer too.

However, as surprising as it may seem, the figures tell us that 80% (=6000/7500) of the people who buy diapers also buy beer.

This is a significant jump of 8 over what was the expected probability. This factor of increase is known as Lift – which is the ratio of the observed frequency of co-occurrence of our items and the expected frequency.

Simply by calculating the transactions in the database and performing simple mathematical operations.

So, for our example, one plausible association rule can state that the people who buy diapers will also purchase beer with a Lift factor of 8. If we talk mathematically, the lift can be calculated as the ratio of the joint probability of two items x and y, divided by the product of their probabilities.

$Lift = P(x,y)/[P(x)P(y)]$

However, if the two items are statistically independent, then the joint probability of the two items will be the same as the product of their probabilities. Or, in other words,

$P(x,y)=P(x)P(y)$,

Which makes the Lift factor = 1. An interesting point worth mentioning here is that anti-correlation can even yield Lift values less than 1 – which corresponds to mutually exclusive items that rarely occur together.

Association Rule Mining has helped data scientists find out patterns they never knew existed.

1. ***Market Basket Analysis:***
   This is the most typical example of association mining. Data is collected using barcode scanners in most supermarkets. This database, known as the “market basket” database, consists of a large number of records on past transactions. A single record lists all the items bought by a customer in one sale. Knowing which groups are inclined towards which set of items gives these shops the freedom to adjust the store layout and the store catalogue to place the optimally concerning one another.

2. ***Medical Diagnosis***:
   Association rules in medical diagnosis can be useful for assisting physicians for curing patients. Diagnosis is not an easy process and has a scope of errors which may result in unreliable end-results. Using relational association rule mining, we can identify the probability of the occurrence of an illness concerning various factors and symptoms. Further, using learning techniques, this interface can be

extended by adding new symptoms and defining relationships between the new signs and the corresponding diseases.

3. ***Census Data:***

   Every government has tonnes of census data. This data can be used to plan efficient public services(education, health, transport) as well as help public businesses (for setting up new factories, shopping malls, and even marketing particular products). This application of association rule mining and data mining has immense potential in supporting sound public policy and bringing forth an efficient functioning of a democratic society.

4. ***Protein Sequence:***

   Proteins are sequences made up of twenty types of amino acids. Each protein bears a unique 3D structure which depends on the sequence of these amino acids. A slight change in the sequence can cause a change in structure which might change the functioning of the protein. This dependency of the protein functioning on its amino acid sequence has been a subject of great research. Earlier it was thought that these sequences are random, but now it's believed that they aren't. *Nitin Gupta, Nitin Mangal, Kamal Tiwari, and Pabitra Mitra* have deciphered the nature of associations between different amino acids that are present in a protein. Knowledge and understanding of these association rules will come in extremely helpful during the synthesis of artificial proteins.

Hypothesis Generation:

In a nutshell, hypothesis generation is what helps you come up with new ideas for what you need to change. Sure, you can do this by sitting around in a room and brainstorming new features, but reaching out and learning from your users is a much faster way of getting the right data.

Imagine you were building a product to help people buy shoes online. Hypothesis generation might includethingslike:

- Talking to people who buy shoes online to explore what their problems are

- Talking to people who don’t buy shoes online to understand why
- Watching people attempt to buy shoes both online and offline in order to understand what their problems really are rather than what they tell you they are
- Watching people use your product to figure out if you’ve done anything particularly confusing that is keeping them from buying shoes from you

As you can see, you can do hypothesis generation at any point in the development of your product. For example, before you have any product at all, you need to do research to learn about your potential users’ habits and problems. Once you have a product, you need to do hypothesis generation to understand how people are using your product and what problems you’ve caused. To be clear, the research itself does not generate hypotheses. YOU do that. The goal is not to just go out and have people tell you exactly what they want and then build it. The goal is to gain an understanding of your users or your product to help you think up clever ideas for what to build next. Good hypothesis generation almost always involves qualitative research.

At some point, you need to observe people or talk to people in order to understand them better. However, you can sometimes use data mining or other metrics analyzation to begin to generate a hypothesis. For example, you might look at your registration flow and notice a severe drop off half way through. This might give you a clue that you have some sort of user problem half way through your registration process that you might want to look into with some qualitative research.

Hypothesis Validation:

Hypothesis validation is different. In this case, you already have an idea of what is wrong, and you have an idea of how you might possibly fix it. You now have to go out and do some research to figure out if your assumptions and decisions were correct.

For our fictional shoe-buying product, hypothesis validation might look something like:

- Standard usability testing on a proposed new purchase flow to see if it goes more smoothly than the old one

- Showing mockups to people in a particular persona group to see if a proposed new feature appeals to that specific group of people
- A/B testing of changes to see if a new feature improves purchase conversion

Hypothesis validation also almost always involves some sort of tangible thing that is getting tested. That thing could be anything from a wireframe to a prototype to an actual feature, but there's something that you're testing and getting concrete data about.

Hypothesis:

Simply put, a hypothesis is a possible view or assertion of an analyst about the problem he or she is working upon. It may be true or may not be true.

For example, if you are asked to build a credit risk model to identify which customers are likely to lapse are which are not, these can a possible set of hypothesis:

- Customers with poor credit history in past are more likely to default in future
- Customers with high (loan_value / income) are likely to default more than those with low ratio
- Customers doing impulsive shopping are more likely to be at a higher credit risk

At this stage, you don't know which out of these hypothesis would be true.

Hypothesis generation important:

Now, the natural question which arises is why is an upfront hypothesis generation important? Let us try and understand the 2 broad approaches and their contrast:

Approach 1: Non-hypothesis driven data analysis (i.e. boiling the ocean)

In today's world, there is no end to what data you can capture and how much time you can spend in trying to find out more variables / data. For example, in this particular case mentioned above, if you don't form initial hypothesis, you will try and understand every possible variable available to you. This would include Bureau variables (which will have hundreds of variables), the company's internal experience variables, and other external data sources. So, you are already talking about analyzing 300 -

500 variables. As an analyst, you will take a lot of time to do this and the value in doing that is not much. Why? Because, even if you understand the distribution of all 500 variables, you would need to understand their correlation and a lot of other information, which can take hell of a time. This strategy is typically known as boiling the ocean. So, you don't know exactly what you are looking for and you are exploring every possible variable and relationship in a hope to use all - very difficult and time consuming.

Approach2:Hypothesisdrivenanalysis

In this case, you list down a comprehensive set of analysis first - basically whatever comes to your mind. Next, you see which out of these variables are readily available or can be collected. Now, this list should give you a set of smaller, specific individual pieces of analysis to work on. For example, instead of understanding all 500 variables first, you check whether the bureau provides number of past defaults or not and use it in your analysis. This saves a lot of time and effort and if you progress on hypothesis in order of your expected importance, you will be able to finish the analysis in fraction of time.

If you have read through the examples closely, the benefit of hypothesis driven approach should be pretty clear. You can further read books “The McKinsey Way” and"The Pyramid Principle" for gaining more insight into this process.

MODULE – IV

VISUALIZATION – 1

Data Visualization:

In order to understand data, it is often useful to visualize it. Normally in Big Data applications, the interest relies in finding insight rather than just making beautiful plots. The following are examples of different approaches to understanding data using plots.

To start analyzing the flights data, we can start by checking if there are correlations between numeric variables.

Visualization or **visualization** (is any technique for creating images, diagrams, or animations to communicate a message. Visualization through visual imagery has been an effective way to communicate both abstract and concrete ideas since the dawn of humanity. Examples from history include cave paintings, Egyptian hieroglyphs, Greek geometry, and Leonardo da Vinci's revolutionary methods of technical drawing for engineering and scientific purposes.

Visualization today has ever-expanding applications in science, education, engineering (e.g., product visualization), interactive multimedia, medicine, etc. Typical of a visualization application is the field of computer graphics. The invention of computer graphics may be the most important development in visualization since the invention of central perspective in the Renaissance period. The development of animation also helped advance visualization.

Designing for Data Visualization:

Data Visualization at IBM:

Our clients are from various industries as well as organizations of all sizes, from large institutions to lean start-ups. But regardless of size or industry, our users all have the same goal. They have data, they have questions, and they need an analytics tools that will help them make sense of their data and turn it into useful business insights, while reducing uncertainty.

When it comes to designing the details of a data driven product, there are a few things that we keep in mind to create the best possible experience for our user.

1.What is the power of data visualization?

Consider this: you receive a postcard from a friend in Venice. The glossy photo contains a typical Venetian scene—a view of the Grand Canal, a gondola navigated by a man in a white shirt who appears to be singing, and stone bridges that fade into the horizon. Your friend writes about how beautiful it is and ends the note with “you simply have to see it for yourself!”

Suddenly you’re overcome with excitement and you begin trolling travel booking sites looking for cheap flights and accommodation. One postcard just isn’t enough of an experience for you. You want to go there yourself, explore the tunnels across the stone bridges, and hear the sounds of the gondoliers singing as you venture off the Grand Canal down the backstreets of Venice. A snapshot is simply not enough to satisfy your need to explore and see things for yourself. You’ve heard about St. Mark’s Square and while it may not be pictured on the postcard, you really want to see it and it can’t wait any longer.

It’s the same with data visualizations. Users are not typically satisfied with simple postcards no matter how picturesque they may be. They need an experience that is as immersive as possible, while making it easy for them to uncover deeper insights and drill deeper into their data in order to make better business decisions.

2.How can you experience your data?

Our users are looking for a tool that not only presents a static view of their business, but one that also enables them to interact with that data in real time. Offering data visualizations that are flexible and change with the user's thought process allows for true exploration.

For example, in Cognos Analytics we offer an experience in the user dashboard that provides side-by-side data comparisons and methods to quickly see how these data points relate to each other and what these discoveries mean.

The first dashboard of Cognos Analytics gives a good overview of data from Bikeshare Chicago's overall ridership. By using the sorting, filtering, and brushing features, the city manager was able to see which neighborhoods have the highest percentage of subscribers in the 39 to 55 year age range from the previous year.

These type of user goals show that data exploration isn't the end, but rather a means to gain as much use out of the data as possible, whether it be applying it to business models to maximize profit, or to creating more accurate troubleshooting techniques. This application of data demonstrates the true power of data analysis tools.

Thinking back to our Venice analogy, a tool like Cognos Analytics allows the user to really dive-in and and get immersed in the data, rather than only being able to see it at surface level.

Guided exploration with cognitive analytics:

Insights have more value when you can act upon them, especially in business. The tools we build enable users to quickly uncover interesting patterns and relationships in their data without the need for any coding. But endless exploration can sometimes lead to analysis paralysis, where the user is continuing to search for every possible data correlation. This may be fun for some data scientists, but is not something that most businesses can afford.

To overcome this, a well-designed data exploration tool not only helps users explore freely, but also navigates them towards the insights they're really looking to make.

Watson Analytics allows easy data exploration through natural language processing, which means users can ask simple questions about their data regardless of their analytics expertise. In this day and age, as designers we need to reduce the gulf between man and machine. With the predictive capabilities of

Watson Analytics, users can ask questions like “what drives sales?” and swiftly be presented with key sales analytics to help them make decisions.

In this example, Sam, an airport operations manager, asks Watson Analytics “What’s driving overall satisfaction?” in English. Watson does the number crunching, creates a predictive model and returns a number of different fields and graphics associated with levels of satisfaction for airport customers, in addition to displaying their predictive strength. The user is easily able to obtain this level of data analysis without needing a statistics degree.

Again thinking of Venice, guided exploration in Watson Analytics is like having your very own personal tour guide who not only knows the best local restaurants to eat at, but also knows what to order and how to order it in Venetian.

3. Analytics your way

When it comes to analytics, not all users’ needs are equal. One of the challenges of designing a data visualization tool is making it intuitive to use for anyone. To define universal experiences, we believe in designing for diversity.

When designing our Business Analytics portfolio, we have multiple personas in mind that range from a novice analyst to a power user. This means the design needs to strike a balance between reducing the learning curve while conveying powerful analytic capabilities. Our customer experience strategy is focused on providing users with the tools and resources they need to be successful, regardless of their existing expertise in data analytics.

Our research process includes many direct and indirect partnerships with our clients and users, including deep observational studies to learn about their workflow as well as how they want to use a data analytics tool. All of this hands-on research helps ensure that we design meaningful experiences, with embedded support and guidance to help users succeed.

Analytics isn't easy and it's not as intuitive as booking a trip to Venice. Our users have many different levels of skill, experience and understanding. They consist of new managers who want to get a better understanding of the business, to power users who want to look under the hood to find out what statistical model was used in the analysis. In Watson Analytics we offer layers in our products that progressively disclose as much or as little of the magic that is used to generate visualizations.

Watson Analytics doesn't simply show a chart of his data, it highlights statistically significant numbers and results, saving him the trouble of doing the calculation. It also surfaces a series of insights and follow-on chart suggestions in the Discovery Panel on the right. A data scientist working for a business professional can open up the Statistical Details panel to confirm and investigate more closely the models and parameters behind the results.

Design plays a key role:

Designing data visualization is not just about the visuals, but why those visuals matter in the data analysis process and how they can be of actual use for the user. We work on designing for iterative data exploration, a guided experience that helps the business user get to their business answers as quickly as possible, and a flexible work flow that supports analytics experts and novices alike. Design work in this field can have powerful implications for data users and effect on how businesses operate.

5 Steps to Designing an Information Visualization:

The overall process is simple and once you've reviewed the process, it should feel like common sense:

1. Define the problem
2. Define the data to be represented
3. Define the dimensions required to represent the data
4. Define the structures of the data
5. Define the interaction required from the visualization

1. Define the Problem

As with any user experience work; the first step is to define the problem that your information visualization will solve. This will usually require some user research to answer the questions; "what does my user need from this?" and "how will they work with it?"

You may be trying to explain something to a user or you may be trying to enable them to make new connections or observations; it might even be that the user is trying to prove a theory.

You should also take into account any specific factors that are unique to the user base during this research. What is their level of education or ability with data handling? What kind of experience do they have with the data in the past? This will guide the level of complexity of the output and clarify the overall needs of the user.

2. Define the Data to be represented

There are three main types of data that can be represented through information visualization and the way that they are mapped can vary dramatically – so it pays to have it clear in your mind before you start designing, what data will you use?

1. **Quantitative data** – this data which is numerical.
2. **Ordinal data** – data which doesn't have numbers but does have an intrinsic form of order. (Think days of the week, for example.)
3. **Categorical data** – data which has neither numbers nor intrinsic order. (Such as business names or place names).

3. Define the Dimensions Required to Represent the Data

The number of dimensions or attributes of a data set must be considered carefully as it will determine, to a great extent, the possible information visualizations that can be used to represent the data.

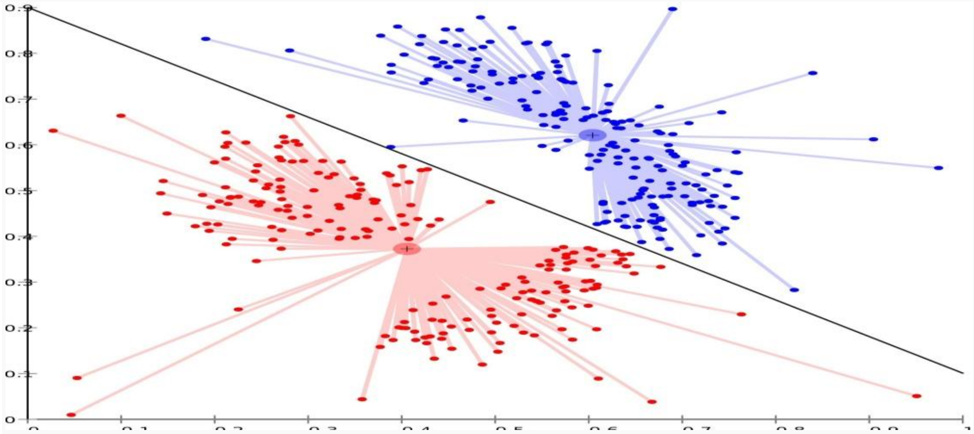
The more dimensions that are represented in the data – the more confusing it can be to comprehend the information visualization. Thus it's worth noting that the data with large numbers of dimensions may well benefit from using a highly interactive representation rather than a static one.

Dimensions can be either dependent or independent of each other. It is the dependent dimensions which vary and which we would expect to need to analyze with respect to the independent dimensions.

There are four types of analysis which can be conducted based on the number of dependent dimensions to be studied:

1. **Univariate analysis** – where a single dependent variable is studied against independent variables
2. **Bivariate analysis** – where two dependent variables are studied against independent variables
3. **Trivariate analysis** – where three dependent variables are studied against independent variables

4. **Multivariate analysis** – where more than three dependent variables are studied against independent variables

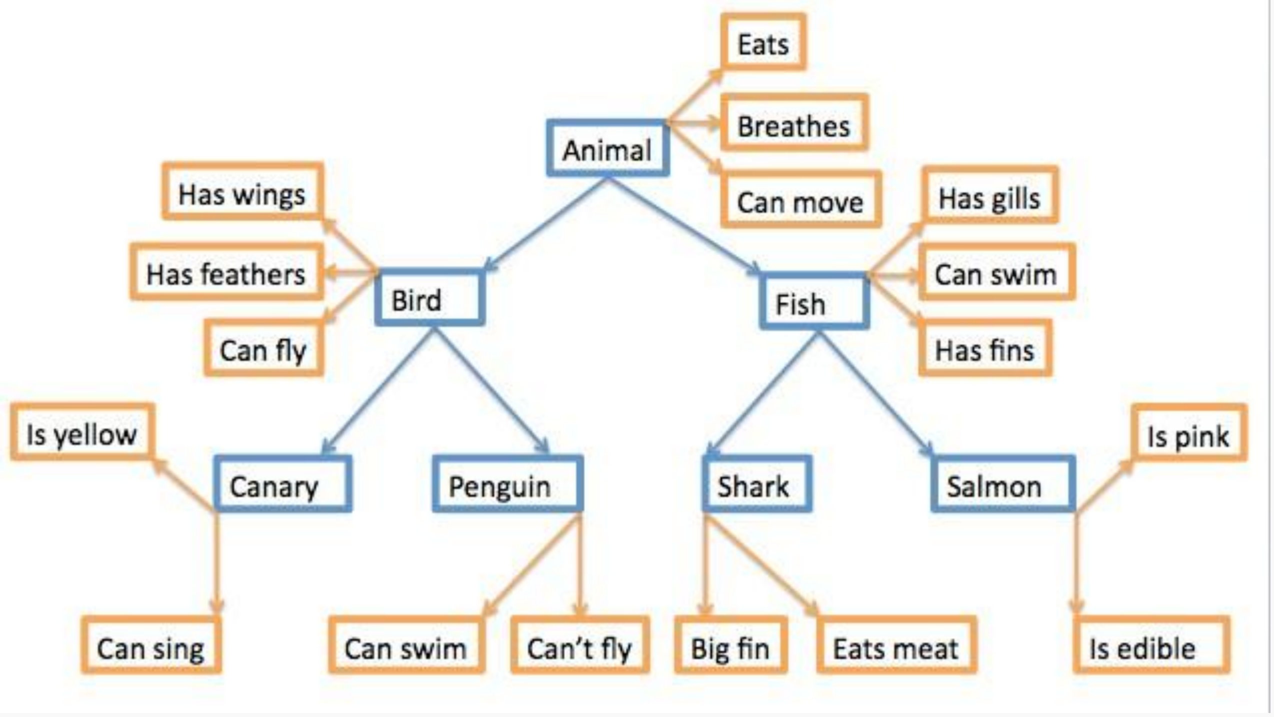


An image of multi-variate analysis where relationships between data points are numerous and dependant.

4. Define the Structures of the Data

This is all about examining how the data sets will relate to each other, common relationship structures include:

- **Linear relationships** – where data can be shown in linear formats such as tables, vectors, etc.
- **Temporal relationships** – where data changes over the passage of time
- **Spatial relationships** – data that relates to the real world (such as map data or an office floor plan) this is sometimes also known as a geographical relationship
- **Hierarchical relationships** – data that relates to positions in a defined hierarchy (from an office management structure to a simple flowchart)
- **Networked relationships** – where the data relates to other entities within the same data



An example of a hierarchical network model is shown above.

5. Define the Interaction Required from the Visualization

The final part of the design process requires that you understand the level of interaction required from the information visualization by the user. There are three categories of interaction:

1. **Static models** – these models are presented "as is" such as maps in a Road Atlas that you keep in a car. They cannot be modified by the user.
2. **Transformable models** – these models enable the user to transform or modify data. They may allow the user to vary parameters for analysis or choose a different form of visual mapping for the data set.
3. **Manipulable models** – these models give the user control over the generation of views. For example; they may allow a user to zoom in or zoom out on a model or to rotate 3-dimensional models in space for viewing from other angles.

It's worth noting that you can combine transformable and manipulable models to create the highest level of interaction in information visualization.

TIME SERIES:

Essentially, these are visualizations that track time series data — the performance of an indicator over a period of time — also known as temporal visualizations.

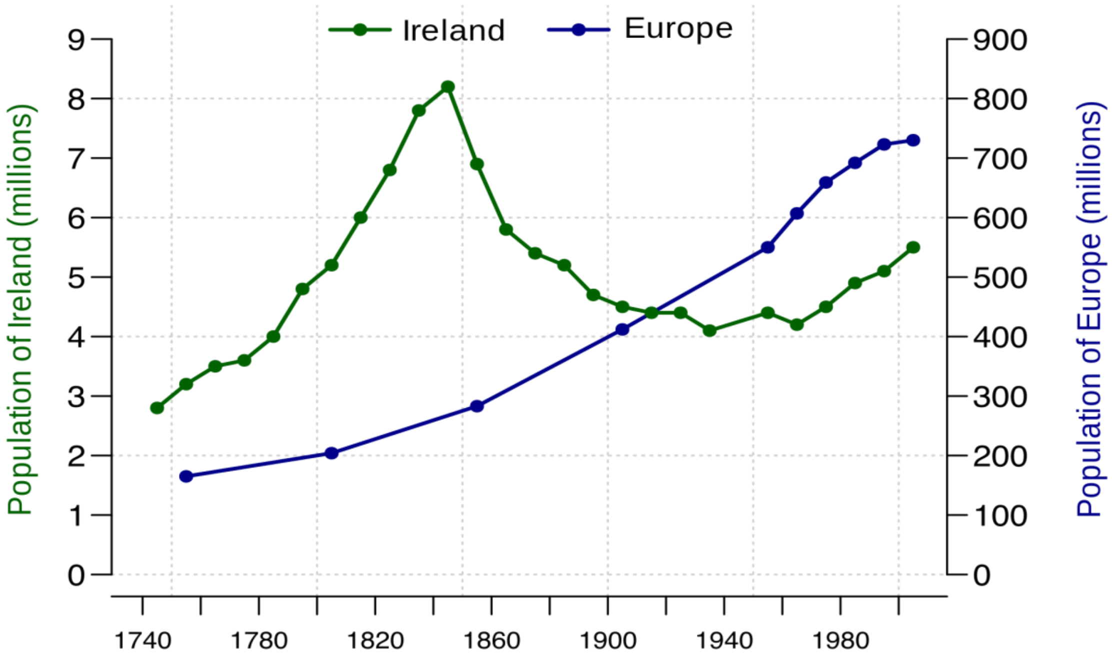
Temporal visualizations are one of the simplest, quickest ways to represent important time series data. In this blog, we have put together 7 handy temporal visualization styles for your time series data. Explore and let us know which is your favorite!

1. Line Graph

A line graph is the simplest way to represent time series data. It is intuitive, easy to create, and helps the viewer get a quick sense of how something has changed over time.

A line graph uses points connected by lines (also called trend lines) to show how a dependent variable and independent variable changed. An independent variable, true to its name, remains unaffected by other parameters, whereas the dependent variable depends on how the independent variable changes. For temporal visualizations, time is always the independent variable, which is plotted on the horizontal axis. Then the dependent variable is plotted on the vertical axis.

In the graph below, the populations of Europe and Ireland are the dependent variables and time is the independent variable.



This graph captures the population growth in Europe and Ireland from 1740 to around 2010. It clearly highlights the sudden drop in Ireland's population in the 1840s. History books will tell you this was the result of the devastating Irish Potato Famine, a period of mass starvation, disease, and emigration in Ireland between 1845 and 1852.

Note that this graph uses different y-axis scales for its two dependent variables — the populations of Europe and Ireland. If the viewer doesn't pay attention to the difference in the scales, they could be led to the conclusion that until about 1920, Ireland's population was greater than that of Europe!

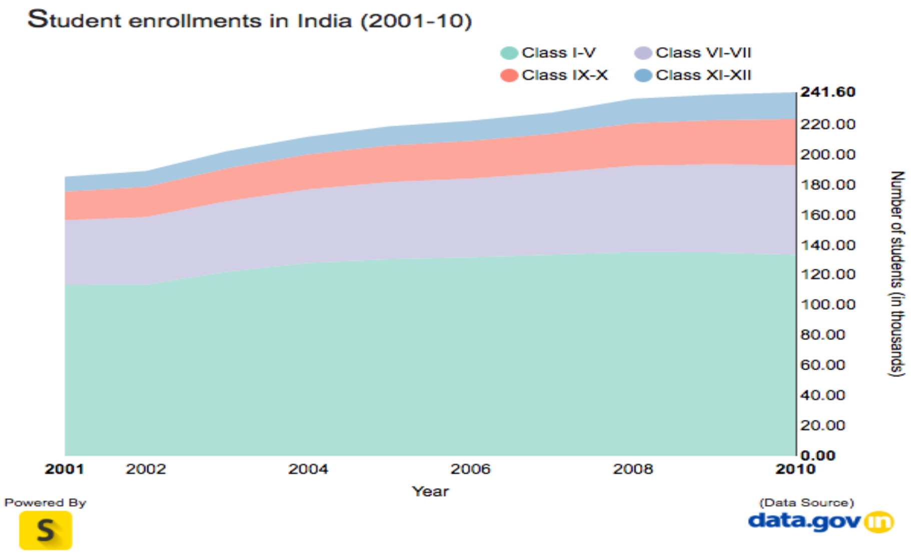
Use different scales with care and only when absolutely necessary. If you need to represent multiple variables on a line graph, try to use the same y-axis for all dependent variables to avoid confusion. If you can't do this, like in the chart above, make sure both y-axes use the same number of increments and use color to show which y-axis belongs to which line.

As a good rule of thumb, don't represent more than four variables on a line graph. With that many variables, the axis scales can become difficult to understand.

2. Stacked Area Chart

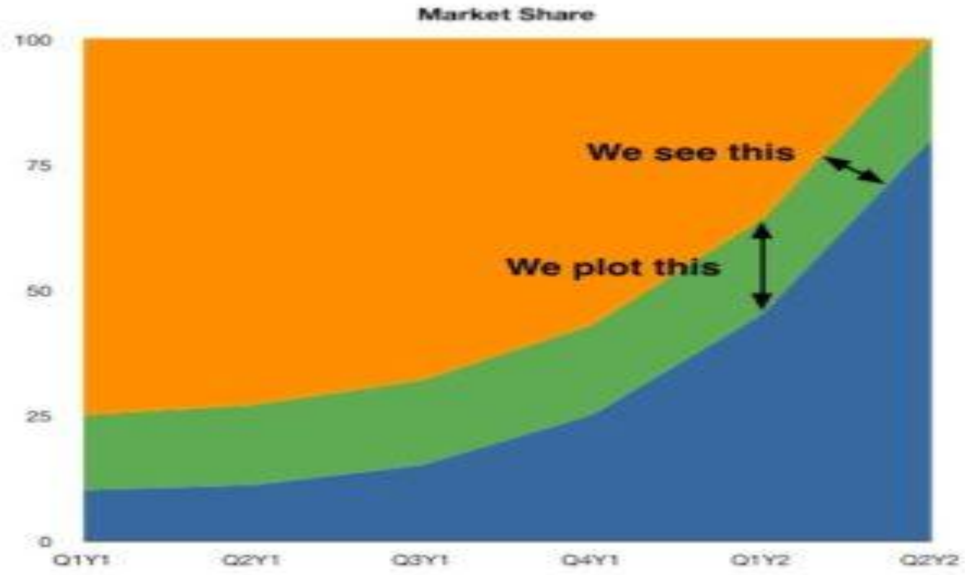
An area chart is similar to a line chart in that it has points connected by straight lines on a two-dimensional chart. It also puts time as the independent variable on the x-axis and the dependent variable on the y-axis. However, in an area chart, multiple variables are "stacked" on top of each other, and the area below each line is colored to represent each variable.

This is a stacked area chart showing time series data of student enrollments in India from 2001-10.



Stacked area charts are useful to show how both a cumulative total and individual components of that total changed over time.

The order in which we stack the variables is crucial because there can sometimes be a difference in the actual plot versus human perception. The chart plots the value vertically whereas we perceive the value to be at right angles to the general direction of the chart. For instance, in the case below, a bar graph would be a cleaner alternative.



3. Bar Charts

Bar charts represent data as horizontal or vertical bars. The length of each bar is proportional to the value of the variable at that point in time. A bar chart is the right choice for you when you wish to look at how the variable moved over time or when you wish to compare variables versus each other. Grouped or stacked bar charts help you combine both these purposes in one chart while keeping your visualization simple and intuitive.

For instance, this grouped bar chart in this interactive visualization of number of deaths by disease type in India not only lets you compare the deaths due to diarrhea, malaria, and acute respiratory disease across time, but also lets you compare the number of deaths by these three diseases in a given year.

GEO LOCATED DATA:

The digital era has ushered in new and exciting ways for us to process and envision information. Geographical data visualization has quickly evolved from static representations on paper to interactive experiences that are more engaging and memorable for the user. With emerging technologies and software products, data can be organized and analyzed without having to invest vast amounts of time typing complex data into every cell of a spreadsheet. Quantitative information can be quickly disseminated, attached to points, lines, and shapes, and transformed visually. This allows the creator to discover exciting insights and trends without having to graph and draw a new map for each train of information.

This blog will cover a methodology for developing successful data visualizations using geographic location(s), as well as techniques, useful resources, and some tips and tricks.

STEP 1: A CLEAR GOAL

Every single successful data visualization begins with a question, a curiosity, followed by dedicated research. Data sets can be found in a large variety of places. Census tracts are useful and free resources that can benefit almost any topographic endeavor, especially those focused on discovering correlations between demographic, behavioral, and geographic data to learn more about the people which the data represents. There are scores of resources that maintain lists of usable information, some which are part of the open data movement, others that are only accessible for a fee. Another option for finding information, sometimes overlooked, is asking questions on forums, in blog comments, etc. The analyst should always use caution before beginning a project to ensure a reliable dataset.

STEP 2: CLEANING

The next step, especially when constructing something to publish, is to clean the data. Capitalization should be consistent, spelling errors and extra spaces eradicated, and formatting developed with purpose. You may have 100,000 rows of data or more, so no time to waste!

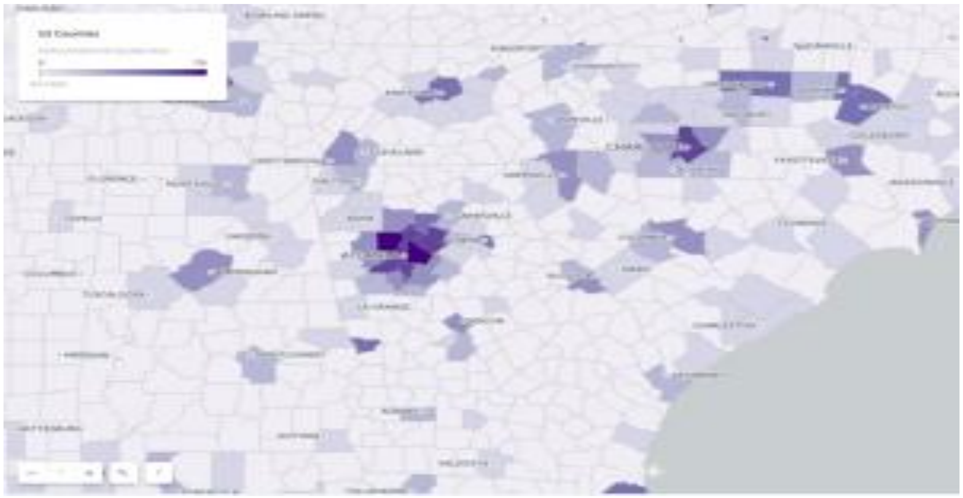
We call this technique “data wrangling,” and there are many useful tools to help in this sort of effort. One such tool is an open source application called Open Refine. OpenRefineallows the user to group the data using pieces of information, such as a string (a sequence of characters), a numeric value, or any other similarity that the data share. This is helpful for discovering typos, capitalization patterns, or other differences within a cell of data. This application can transform the data into other formats; combine it with other data sets, and more. Open Refine fits most of CATMEDIA’s data wrangling needs. There are tools available for a fee that can help with other specific needs.

STEP 3: DISCOVERY

The next step in this method is to import the cleaned data into applicable geographical data visualization software, and begin analysis. Today, there are many innovative mapping software projects and presentation tools to choose from, making it difficult to recommend just one. I'll let you in a little secret. The most important part of creating successful data visualization is uncovering a story, trend, or insight from the data using curiosity, ingenuity, and creativity. This part is fun and exciting, full of adventure, and unexpected surprises. Thoughtful experimentation will help guide this process and reveal insight from what was once a series of numbers.

There are tried and true methods for representing geographical data that were developed and implemented long before digital interactive mapping was a glimmer in a cartographer's eyes. For the purposes of this blog, I have laid out a brief description of a few types of thematic mapping. Thematic maps are usually used as tools to present statistical data connected with geographical locations, as opposed to general maps, like atlases. Here are some of the most widely used types of thematic maps.

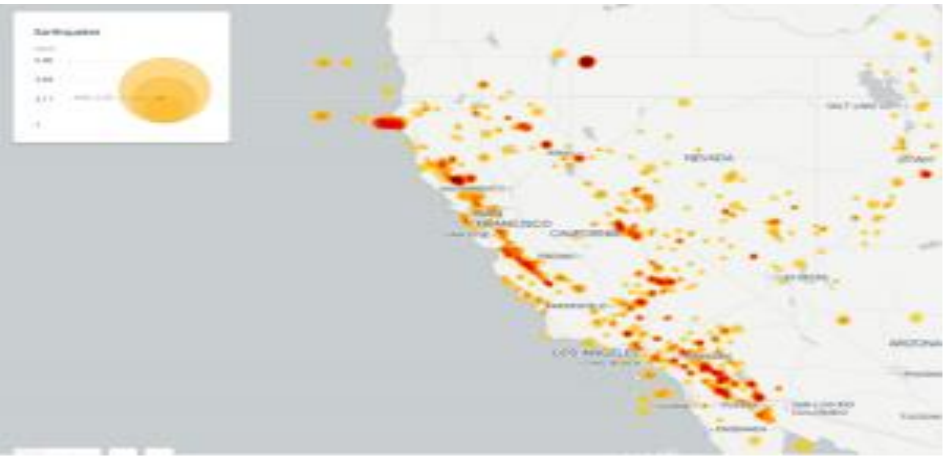
- Choropleth maps: use color to represent rates, quantities, values, etc. over a geographic range. Here, the linguistic stems are "choro," meaning area, and "plethos," meaning multitude.



- Dot maps: use markings of the same shape, size, etc. to represent a singular unit of data. Trends can be identified by determining patterns, or lack thereof, among the symbols.

Single Family Units
51,039
Multi-Family Properties
1,365
8,054
5,357
471
Area of Opportunity

- Graduated and proportional symbols: use markings of different sizes to represent different values. For example, a larger value might utilize a larger symbol.



The deciding factor regarding which type of map to choose is which map will help the analyst to uncover and present the story of the data most clearly. On occasion when appropriate, maps can and should be layered atop one another.

STEP 4: VISUALIZATION

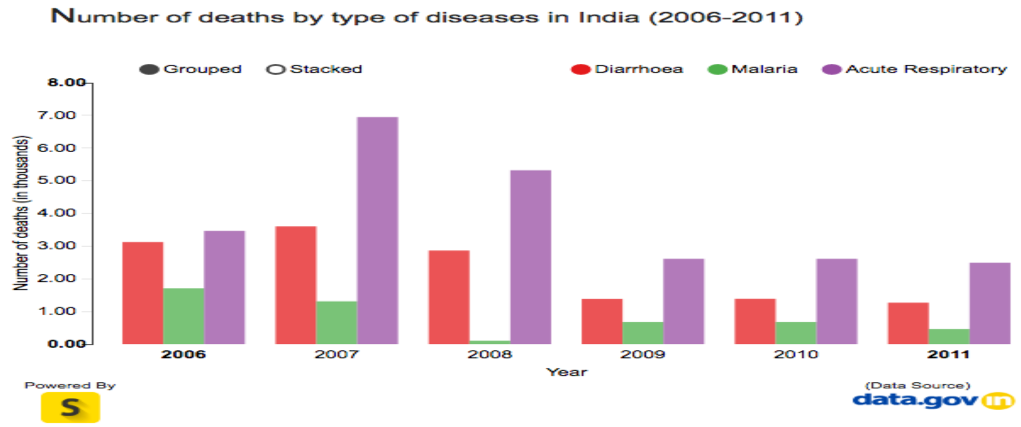
The principles of design are important aspects of any data visualization. A distinct and meaningful visual hierarchy will illuminate the data. What is the most important part of the story being told or the insight being represented? We present the story with contrast; varying font sizes and style, the use of space, and color. These aspects can help demonstrate important themes such as conflict, transition, and density. Font and color choices along with the careful use of space will also help set the tone and purpose of the map. A map designed to be hung on a wall and look pretty may have an artsy script and many differing, yet cohesive colors. Maps created to inform may make use of a sans serif and a simpler color palette, so that the user may quickly digest and understand the information.

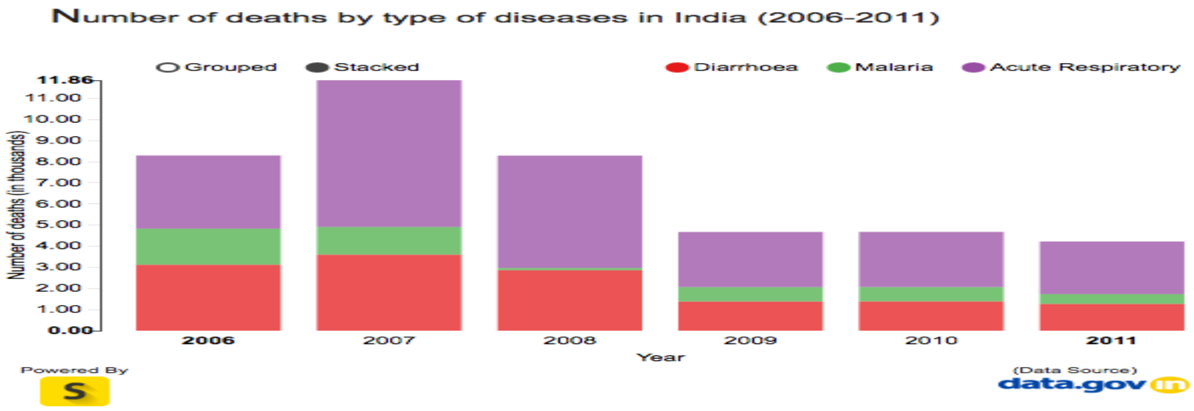
There are many tools to help demonstrate the effects that color and typography have on a map, such as ColorBrewer and TypeBrewer. These are great resources to find inspiration from, but I caution you, they are just places to start. I always recommend experimenting with and adapting the colors and typographic elements of a map to meet specific needs of the project.

Other more advanced visualization techniques involve elements that are inherent to interactivity. In the age of zooming in as close as you want to a map, enough to see an individual's home, it is important to realize that the elements of the map need to scale as they become larger. A proportionate scale is not always appropriate. Code can be used to tell an interactive map when to leave out elements to conserve space, what size fonts and symbols should be at what zoom level, and how much detail to include in the background layer. This is important for the end user experience and ensuring your visual hierarchy is intact no matter how the user interacts with the map. Another exciting interactive mapping feature is including layers that the user can turn on and off. This helps a person disseminate the data. They can compare parts at a time, and draw conclusions at their own pace.

While there are multitudes of considerations and important decisions to be made while selecting which presentation tools, methods, and techniques to use, the keys to good geographical data visualization are attention to detail, dedication, and creativity. Any good mapping software you choose can make a choropleth map, a dot map, etc.

Only people can provide the personal insight and perspective it takes to turn a long list of numbers into an interesting and useful interactive map that is fit to share with the world.



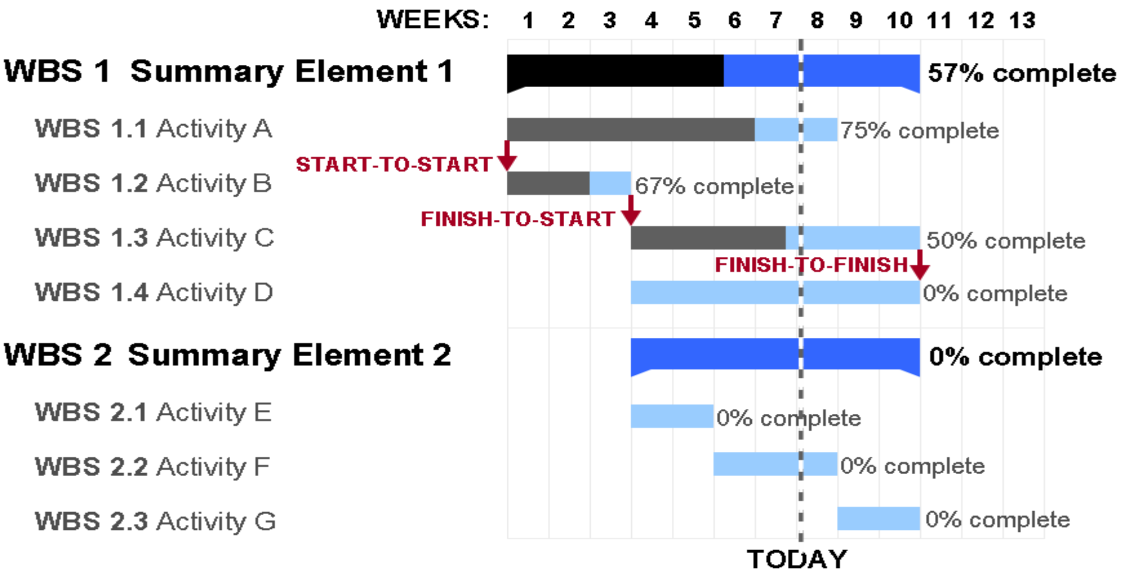


By switching to the stacked bar chart view, you get an intuitive sense of the proportion of deaths caused by each disease.

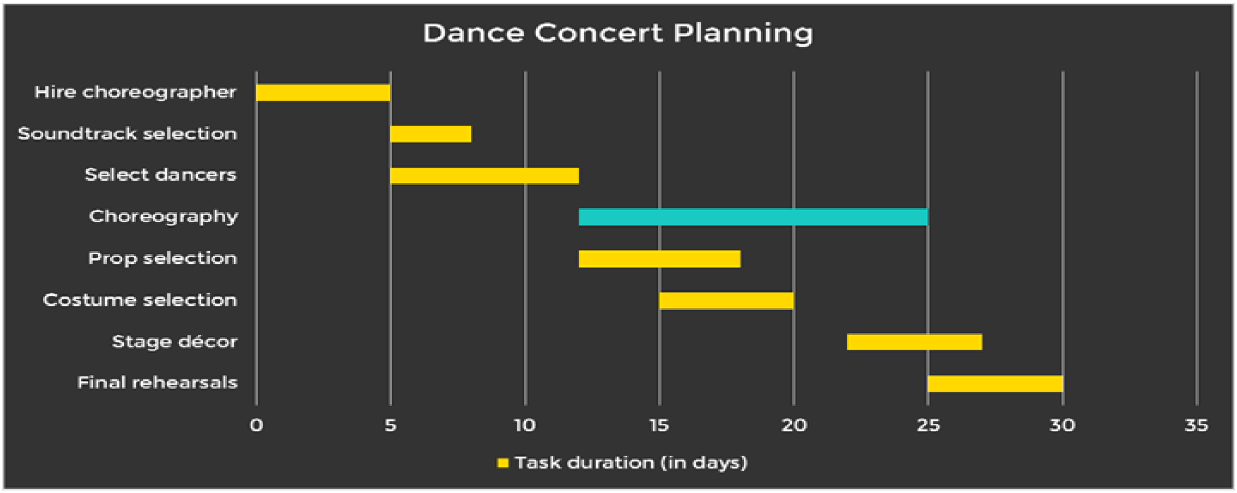
To avoid clutter and confusion, make sure to not use more than 3 variables in a stacked or group bar chart. It is also a good practice to use consistent bold colors and leave appropriate space between two bars in a bar chart. Also, check out our blog on 5 common mistakes that lead to bad data visualizationto learn why the base axis for your bar charts should start from zero.

4. Gantt Chart

A Gantt chart is a horizontal bar chart showing work completed in a certain period of time with respect to the time allocated for that particular task. It is named after the American engineer and management consultant Henry Gantt who extensively used this framework for project management.



Assume you're planning the logistics for a dance concert. There are lots of activities to be completed, some of which will take place simultaneously while some can be done only after another activity has been completed. For instance, the choreographers, soundtrack, and dancers need to be finalized before the choreography can begin. However, the costumes, props, and stage decor can be planned at the same time as the choreography. With careful preparation, Gantt charts can help you plan for complex, long-term projects that are likely to undergo several revisions and have various resource and task dependencies.

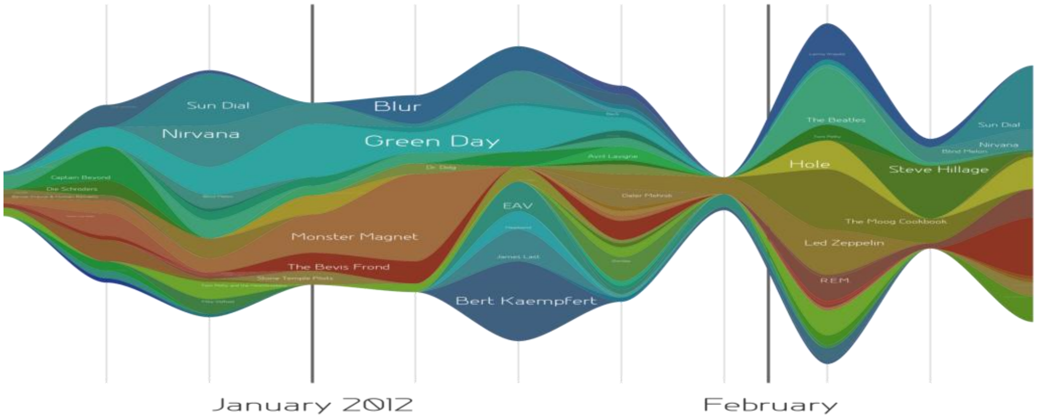


Gantt charts are a popular project management tool since they present a concise snapshot of various tasks spread across various phases of the project. You can show additional information such as the correlation between individual tasks, resources used in each task, overlapping resources, etc., by the use of colors and placement of bars in a Gantt chart.

5. Stream Graph

A stream graph is essentially a stacked area graph, but displaced around a central horizontal axis. The stream graph looks like flowing liquid, hence the name.

Below is a stream graph showing a randomly chosen listener’s last.fm music-listening habits over time.

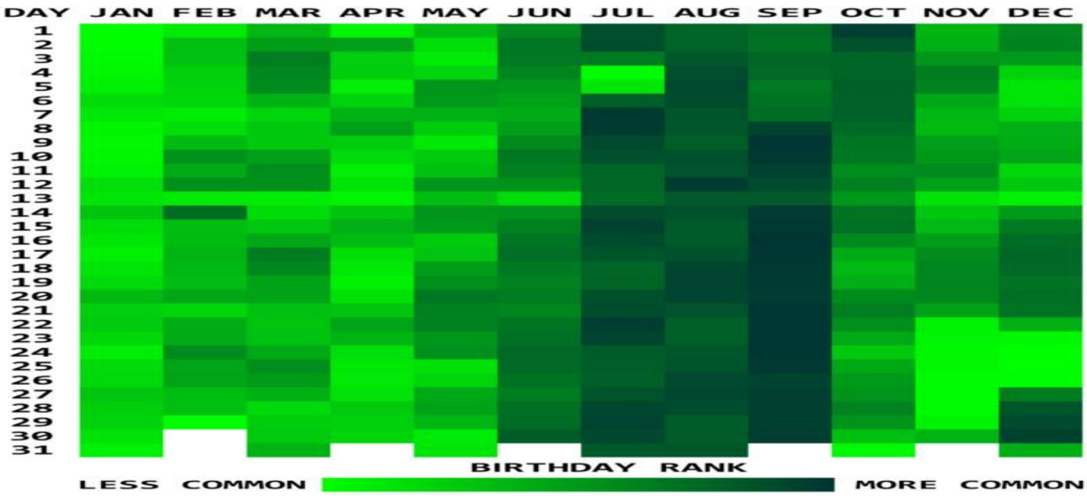


Stream graphs are great to represent and compare time series data for multiple variables. Stream graphs are, thus, apt for large data sets. Remember that choice of colors is very important, especially when there are lots of variables. Variables that do not have significantly high values might tend to get drowned out in the visualization if the colors are not chosen well.

6. Heat Map

Geospatial visualizations often use heat maps since they quickly help identify “hot spots” or regions of high concentrations of a given variable. When adapted to temporal visualizations, heat maps can help us explore two levels of time in a 2D array.

This heat map visualizes birthdays for babies born in the United States between 1973 and 1999. The vertical axis represents the 31 days in a month while the horizontal axis represents the 12 months in a year. This chart quickly helps us identify that a large number of babies were born in the later half of July, August, and September.

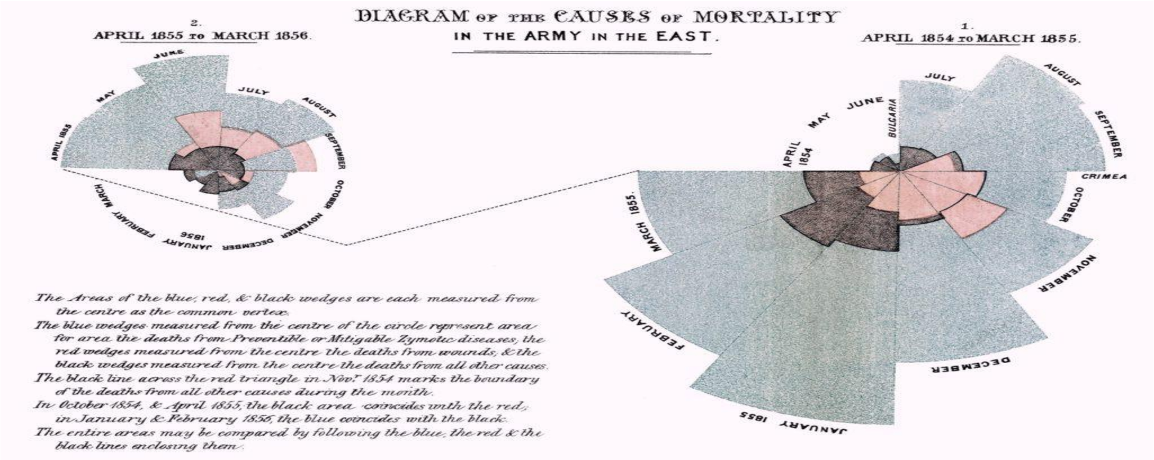


Heat maps are perfect for a two-tiered time frame — for instance, 7 days of the week spread across 52 weeks in the year, or 24 hours in a day spread across 30 days of the month, and so on. The limitation, though, is that only one variable can be visualized in a heat map. Comparison between two or more variables is very difficult to represent.

7. Polar Area Diagram

Think beyond the straight line! Sometimes, time series data can be cyclical — a season in a year, time of the day, and so on. Polar area diagrams help represent the cyclical nature time series data cleanly. A polar diagram looks like a traditional pie chart, but the sectors differ from each other not by the size of their angles but by how far they extend out from the centre of the circle.

This popular polar area diagram created by Florence Nightingale shows causes of mortality among British troops in the Crimean War. Each color in the diagram represents a different cause of death. (Check out the the text legend for more details.)



The Areas of the blue, red, & black wedges are each measured from the centre as the common vertex.
The blue wedges measured from the centre of the circle represent area for area the deaths from Preventible or Mitigable Zymotic diseases, the red wedges measured from the centre the deaths from wounds, & the black wedges measured from the centre the deaths from all other causes.
The black line across the red triangle in Nov.r 1854 marks the boundary of the deaths from all other causes during the month.
In October 1854, & April 1855, the black area coincides with the red, in January & February 1856, the blue coincides with the black.
The entire areas may be compared by following the blue, the red & the black lines enclosing them.

Polar area diagrams are useful for representing seasonal or cyclical time series data, such as climate or seasonal crop data. Multiple variables can be neatly stacked in the various sectors of the pie.

It is crucial to clarify whether the variable is proportional to the area or radius of the sector. It is a good practice to have the area of the sectors proportional to the value being represented. In that case, the radius should be proportional to the square root of the value of the variable (since area of a circle is proportional to the square of the radius).

Polar area diagrams or pie charts in general, must be made with a lot of care to avoid misrepresentation. For more tips, check out this blog on 5 things you should know before you make a pie chart.

Visualize correlations and connections:

Correlation is one of the most widely used tools in statistics. The correlation coefficient summarizes the association between two variables. In this visualization I show a scatter plot of two variables with a given correlation. The variables are samples from the standard normal distribution, which are then transformed to have a given correlation by using Cholesky decomposition. By moving the slider you will see how the shape of the data changes as the association becomes stronger or weaker. You can also look at the Venn diagram to see the amount of shared variance between the variables. It is also possible drag the data points to see how the correlation is influenced by outliers.

interactive visualization

Correlation is one of the most widely used tools in statistics.The **correlation** coefficient summarizes the association between two variables. In this **visualization** I show a scatter plot of two variables with a given**correlation**.

MODULE - V

VISUALIZATION – 2

Visualizing Hierarchies:

From a data visualization standpoint, I feel that this graphic does a tremendous job of showing the hierarchical nature of the religion data and the relative magnitudes of each level of the hierarchy.

1. Treemap

The most basic and most common way to visualize hierarchical data is through use of a tree map. A standard tree map typically uses a rectangular layout. The first level of the hierarchy is shown in rectangles, the size determined by some measure (in my case, the number of adherents to the given religious tradition). Then, each rectangle is further sub-divided into smaller rectangles for the next level of the hierarchy and so on. Color is typically used to highlight one of the hierarchical levels. Due to its rectangular nature, tree maps are able to remain very compact and use the space on screen to its fullest. Theoretically, there is no limit to the number of hierarchical levels that can be visualized with a tree map, but in practice, the data becomes very difficult to see the more levels that you add.

2. Sunburst

A sunburst is really just a tree map which uses a radial layout (thus the alternative name, “Radial Tree map”). Sunbursts are a series of rings, which represent the different hierarchical levels. The innermost ring is the first level, followed by the second level which shows a breakdown of the components of the first, and so on. Like the rectangular tree map, the size of the arc represents the magnitude of a metric and color is often used to distinguish some level of the hierarchy.

3. Packed Bubbles

I hesitated on whether or not to include packed bubbles here, as I feel it does not truly represent the hierarchies we’re trying to display. This type of visualization essentially shows the lowest level of the hierarchical data (not necessarily the *last* level, but the last level available for a specific record). Though you can use color to identify one of the parent categories, It fails to accurately represent the size of any of those parent categories. It also fails to demonstrate the hierarchical relationship between different levels.

4. Circle Packing

Circle packing is another type of tree map (also known as a “Circular Tree map”). This type of visualization is similar to packed bubbles, but it addresses the inability to see the relationships between the hierarchical levels, as well as the problem of showing the magnitude of parent levels. Unfortunately, circle packing charts suffer from other problems. The use of circles means that there is a lot of wasted space. And, as shown in the example below, the labels can be quite difficult to read, particularly as the number of levels increase.

5. Dendrogram

Dendrograms are entirely different than the visualization shown above. Dendrograms are essentially tree diagrams that show the hierarchical relationship of a data set. The most straightforward style is a cluster dendrogram, which shows the data horizontally, as shown in the following chart I created using RAW.

Dendrograms are better than tree maps at showing the hierarchical nature of the data, but they do not show the magnitude of any single item on the chart

6. Foam tree

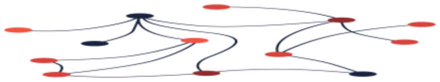
Foam tree is not exactly a different type of visualization, but rather a product with a very interesting implementation of tree maps. Foam tree, which was created by a Polish company, called *Carrot Search*, implements a Voronoi tree map, which uses polygons instead of rectangles or circles. Like rectangles, polygons make much better use of the space available on screen. But, the beauty of this implementation is how it mitigates the problem of visibility when there are numerous levels to the hierarchy. As an interactive tool, it allows you to drill down on a level of the hierarchy and zoom into the component parts within the next level. You can continue doing this, going as deep as you like, with almost no limitations to the number of levels.

Network Visualization:

Network Visualization (also called **Network Graph**) is often used to **visualize**complex relationships between a huge amount of elements. ... This type of **visualization** illuminates relationships between entities. Entities are displayed as round nodes and lines **show** the relationships between them.

Network Visualization (also called Network Graph) is often used to visualize complex relationships between a huge amounts of elements. Networkvisualization displays undirected and directed graph structures. This type of visualization illuminates relationships between entities. Entities are displayed as round nodes and lines show the relationships between them. The vivid display of network nodes can highlight non-trivial data discrepancies that may be

Otherwise be overlooked**.**



INTERACTIVITY:

Interactivevisualization or **interactivevisualisation** isa branch of graphic visualization in computer science that involves studying how humans interact with computers to create graphic illustrations of information and how this process can be made more efficient.

For a visualization to be considered interactive it must satisfy two criteria:

- **Human input**: control of some aspect of the visual representation of information, or of the information being represented, must be available to a human, and
- **Response time**: changes made by the human must be incorporated into the visualization in a timely manner. In general, interactive visualization is considered a soft real-timetask.

One particular type of interactive visualization is virtual reality (VR), where the visual representation of information is presented using an immersive display device such as a stereo projector (see stereoscopy). VR is also characterized by the use of a spatial metaphor, where some aspect of the information is represented in three dimensions so that humans can explore the information as if it were present (where instead it was remote), sized appropriately (where instead it was on a much smaller or larger scale than humans can sense directly), or had shape (where instead it might be completely abstract).

Another type of interactive visualization is collaborative visualization, in which multiple people interact with the same computer visualization to communicate their ideas to each other or to explore information cooperatively. Frequently, collaborative visualization is used when people are physically separated. Using several networked computers, the same visualization can be presented to each person simultaneously. The people then make annotations to the visualization as well as communicate via audio (i.e., telephone), video (i.e., a video-conference), or text (i.e., IRC) messages.

Human control of visualization:

The Programmer's Hierarchical Interactive Graphics System (PHIGS) was one of the first programmatic efforts at interactive visualization and provided an enumeration of the types of input humans provide. People can:

1. *Pick* some part of an existing visual representation;
2. *Locate* a point of interest (which may not have an existing representation);
3. *Stroke* a path;
4. *Choose* an option from a list of options;
5. *Valuate* by inputting a number; and
6. *Write* by inputting text.

All of these actions require a physical device. Input devices range from the common – keyboards, mice, graphics tablets, trackballs, and touchpads – to the esoteric – wired gloves, boom arms, and even omnidirectional treadmills.

These input actions can be used to control both the information being represented or the way that the information is presented. When the information being presented is altered, the visualization is usually part of a feedback loop. For example, consider an aircraft avionics system where the pilot inputs roll, pitch, and yaw and the visualization system provides a rendering of the aircraft's new attitude. Another example would be a scientist who changes a simulation while it is running in response to a visualization (see Visulation) of its current progress. This is called *computational steering*.

More frequently, the representation of the information is changed rather than the information itself (see Visualization (graphic)).

Rapid response to human input:

Experiments have shown that a delay of more than 20 ms between when input is provided and a visual representation is updated is noticeable by most people. Thus it is desirable for an interactive visualization to provide a rendering based on human input within this time frame. However, when large amounts of data must be processed to create visualization, this becomes hard or even impossible with current technology. Thus the term “interactive visualization” is usually applied to systems that provide feedback to users within several seconds of input. The term *interactive framerate* is often used to measure how interactive visualization is. Framerates measure the frequency with which an image (a frame) can be generated by a visualization system. A framerate of 50 frames per second (frame/s) is considered good while 0.1 frames would be considered poor. The use of framerates to characterize interactivity is slightly misleading however, since framerate is a measure of bandwidth while humans are more sensitive to latency. Specifically, it is possible to achieve a good frame rateof 50 frames but if the images generated refer to changes to the visualization that a person made more than 1 second ago, it will not feel interactive to a person.

The rapid response time required for interactive visualization is a difficult constraint to meet and there are several approaches that have been explored to provide people with rapid visual feedback based on their input. Some include

1. *Parallel rendering* – where more than one computer or video card is used simultaneously to render an image. Multiple frames can be rendered at the same time by different computers and the results transferred over the network for display on a single monitor. This requires each computer to hold a copy of all the information to be rendered and increases bandwidth, but also increases latency. Also, each computer can render a different region of a single frame and send the results over a network for display. This again requires each computer to hold all of the data and can lead to a load imbalance when one computer is responsible for rendering a region of the screen with more information than other computers. Finally, each computer can render an entire frame containing a subset of the information. The resulting images plus the associated depth buffer can then be sent across the network and merged with the images from other computers. The result is a single frame containing all the information to be rendered, even though no single computer's memory held all of the information. This is called *parallel depth compositing* and is used when large amounts of information must be rendered interactively.
2. *Progressive rendering* – where a framerate is guaranteed by rendering some subset of the information to be presented and providing incremental (progressive) improvements to the rendering once the visualization is no longer changing.
3. *Level-of-detail (LOD) rendering* – where simplified representations of information are rendered to achieve a desired framerate while a person is providing input and then the full representation is used to generate a still image once the person is through manipulating the visualization. One common variant of LOD rendering is *subsampling*. When the information being represented is stored in a topologically rectangular array (as is common with digital photos, MRI scans, and finite difference simulations), a lower resolution version can easily be generated by

skipping *n* points for each 1 point rendered. Subsampling can also be used to accelerate rendering techniques such as volume visualization that require more than twice the computations for an image twice the size. By rendering a smaller image and then scaling the image to fill the requested screen space, much less time is required to render the same data.

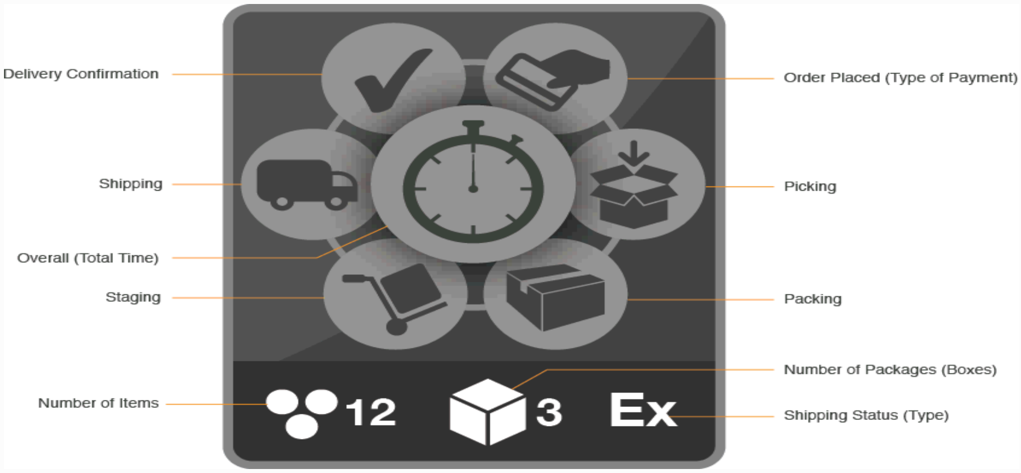
4. *Frameless rendering* – where the visualization is no longer presented as a time series of images, but as a single image where different regions are updated over time.

.The following are five key properties of Interactive Visualization:

1. The Novice User. Even novices must be able to examine data and find patterns, distributions, correlations, and/or anomalies. They must be able to build and use tools that enable faster decisions based on real-time information. As the National Research Council of the National Academies of Sciences states, even "naïve users" should be able to "carry out massive data analysis without a full understanding of systems and statistical uses." *Frontiers in Massive Data Analysis*(National Academy of Sciences 2013). And while data scientists play an indispensable role in today's corporation, business line executives should not have to rely on them to run analytics and make the inferences that are the basis for decisions. As McKinsey puts it, "sophisticated analytics solutions . . . must be embedded in frontline tools so simple and engaging that managers and frontline employees will be eager to use them daily." *Mobilizing Your C-Suite For Big Data Analytics* (McKinsey & Company 2013).
2. Driving Processes. The solution must allow the user to establish KPIs that provide the rules that drive processes. These must be displayed visually—for example, by color—in real time based on defined thresholds. Likes its architecture, Interactive Visualization is a means to an end – to stimulate informed action. Thus, for example, when a fire engulfs the third floor of a company's office space, triggers are set off that alert proper actors such as the municipal fire department. Interactive Visualization displays the department's efforts through phases of the process—discovery, initial actions, mitigation, stabilization, and recovery. As each phase is completed, analytics-based data is represented in real time in green, thereby ending with the most intuitive color cycle we know: red (danger; take action); yellow (pause; remediation is underway); and green (problem solved; situation clear). With icons changing color based on pre-defined thresholds (rules) run against multiple data streams by an analytics engine, data can be understood equally by

management and analysts with no need for technical translation. It is important that the status of a process (fire), a person (fireman), or a physical asset (fire truck) must be depicted visually either independent of one another or as they correlate. Each representation must be both simple and highly granular, allowing a user to understand huge amounts of data with little or no training.

3. Data Must Tell A Story. An intuitive, visual workplace that it easy to master is based on easily digestible interactive patterns. Data must tell a story that instantly relates the performance of a business and its assets. Almost every Interactive Visualization narrative takes place across multiple layers. Users must thus be able to select data elements and filters, and then highlight and modify options to change data perspectives – from high-tech overviews down to the most granular detail. For example, one might overlay data on top of maps, diagrams such as building schematics, or even atop steps underway in a process. A story will emerge that places data in context and in real time as needed. Visual Cue is one company that has taken telling business process stories to a new level. An example of one of their active business tiles is shown here. According to CEO Kerry Gilger, "users want a complicated story made simple so that they act on it. The story needs to unfold simply, in real time, and in intuitive diagrams that can prompt immediate



4. Data Correlation. The user should immediately know not only of hot spots that require attention, but also effortlessly find trends based on the dynamic relationship between multiple data streams and the data derived from them by means of predictive analytics.

5. Prescriptions: "What should happen next?" According to Gartner, analytics evolves through four phases. The third and most discussed phase in today's market is predictive analytics – the application of rules and algorithms against data streams in order to yield actionable intelligence. This answers the question: "What is going to happen?" World-class Interactive Visualization and underlying analytics capabilities surpass that standard

by offering prescriptive analytics(“What should happen next?”) to drive real-time asset behavior modification. This is the pinnacle of Gartner’s evolution of analytics. It is closely linked to the need to drive processes discussed above. Recommendations may range from (i) re-routing a cargo ship to a different port based on the ratio of fuel loss to cargo weight, to (ii) suggesting additional training for underperforming members of a call center.

Conclusion

Interactive Visualization is an intuitive way to enable data trending and ‘manipulation’ to review findings in and across myriad dimensions. It is not a stretch to say that mission-critical Interactive Visualization can be easy (and fun) and empower the user to explore trends he could not have known existed, whether over space, time, processes, or assets, to name a few indices. Interactive Visualization does not stand alone as a mere presentation layer. It requires rich underlying data and predictive analytics to see how assets will behave and to make recommendations as to how they should behave.