MU

Sem 6

AIDS, CSE (DS), CSE (AIML), AIML, DE

(Course Code : CSC601) (Compulsory Subject)

DATA ANALYTICS AND VISUALISATION

Prof. Baphana R. M.
Adjunct Faculty at C.O.E.P. (Pune) Teaching Ph. D & M. Tech Students (Artificial Intelligence & Robotics)

Prof. Yogesh Mali
G H Raisoni College of Engineering, Wagholi, Pune

TECH-NEO PUBLICATIONS
Where Authors Inspire Innovation
A Sachin Shah Venture
www.techneobooks.in
info@techneobooks.in

- Simple and Easy Language.
- Concepts explained with Solved Numericals

University of Mumbai

Data Analytics and Visualization

(Course Code : CSC601) (Compulsory Subject)

Semester 6

- Computer Science and Engineering (Data Science)
- Computer Science & Engineering (Artificial Intelligence and Machine Learning)
- Artificial Intelligence and Data Science
- Artificial Intelligence and Machine Learning
- Data Engineering

Strictly as per the New Syllabus (REV-2019 'C' Scheme) of Mumbai University w.e.f. academic year 2022-2023

Prof. R. M. Baphana

Adjunct Faculty,

Government College of Engineering, Pune (C.O.E.P)

(Teaching M. Tech & Ph.D students)

Prof. Yogesh Mali

M.E. Computer Engineering

Assistant Professor,

Computer Engineering Department,

G H Raisoni College of Engineering, Wagholi, Pune

TECH-NEO PUBLICATIONS

Where Authors Inspire Innovation

A Sachin Shah Venture

M6-128

Syllabus...

Mumbai University

CSC601 : Data Analytics and Visualization

Course Code	Course Name	Credit
CSC601	Compulsory Subject : Data Analytics and Visualization	03

Prerequisite : Basic statistics and Maths, Python programming

Course Objectives : The course aims :

1. To Introduce the concept of Data Analytics Lifecycle.
2. To Develop Mathematical concepts required for advance regression.
3. To Understand data modeling in time series and its process.
4. To create awareness about Text analytics and its applications.
5. To provide overview of Data analytics and visualization with R.
6. To provide overview of Data analytics and visualization with Python.

Course Outcomes : After successful completion of the course students will be able to :

1. Comprehend basics of data analytics and visualization.
2. Apply various regression models on given data set and perform prediction.
3. Demonstrate advance understanding of Time series concepts and analysis of data using various time series models.
4. Analyze Text data and gain insights.
5. Experiment with different analytics techniques and visualization using R.
6. Experiment with different analytics techniques and visualization using Python.

Course Contents

Module		Detailed Content	Hours
1		**Introduction to Data Analytics and Life Cycle**	
	1.1	Data Analytics Lifecycle overview : Key Roles for a Successful Analytics, Background and Overview of Data Analytics Lifecycle Project. **Phase 1 :** Discovery : Learning the Business Domain, Resources Framing the Problem, Identifying Key Stakeholders. Interviewing the Analytics Sponsor, Developing Initial Hypotheses Identifying Potential Data Sources. **Phase 2 :** Data Preparation : Preparing the Analytic Sandbox, Performing ETLT, Learning About the Data, Data Conditioning, Survey and visualize, Common Tools for the Data Preparation Phase. **Phase 3 :** Model Planning : Data Exploration and Variable Selection, Model Selection, Common Tools for the Model Planning Phase. **Phase 4 :** Model Building : Common Tools for the Model Building Phase. **Phase 5 :** Communicate Results. **Phase 6 :** Operationalize. **(Refer Chapter 1)**	5

Module		Detailed Content	Hours
2		**Regression Models**	8
	2.1	Introduction to simple Linear Regression : The Regression Equation, Fittedvalue and Residuals, Least Square. Introduction to Multiple Linear Regression : Assessing the Model, Cross-Validation, Model Selection and Stepwise Regression, Prediction using Regression.	
	2.2	Logistic Regression : Logistic Response function and logit, Logistic Regression and GLM, Generalized Linear model, Predicted values from Logistic Regression, Interpreting the coefficients and odds ratios, Linear and Logistic Regression : Similarities and Differences, Assessing the models. **(Refer Chapter 2)**	
3		**Time Series**	7
		Overview of Time Series Analysis Box-Jenkins Methodology, ARIMA Model Autocorrelation Function (ACF), Autoregressive Models, Moving Average Models, ARMA and ARIMA Models, Building and Evaluating an ARIMA Model, Reasons to Choose and Cautions. **(Refer Chapter 3)**	
4		**Text Analytics**	7
	4.1	History of text mining, Roots of text mining, overview of seven practices of text analytic, Application and use cases for Text mining : extracting meaning from unstructured text, Summarizing Text.	
		Text Analysis Steps, A Text Analysis Example, Collecting Raw Text, Representing Text, Term Frequency-Inverse Document Frequency (TFIDF), Categorizing Documents by Topics, Determining Sentiments, Gaining Insights. **(Refer Chapter 4)**	
5		**Data Analytics and Visualization with R**	6
	5.1	Introduction to R : Data Import and Export, Attribute and Data type, Descriptive statistics. Exploratory Data Analysis : Visualization before analysis, DirtyData, visualizing single variable, examining Multiple variable, Data Exploration versus presentation. **(Refer Chapter 5)**	
6		**Data Analytics and Visualization with Python**	6
	6.1	Essential Data Libraries for data analytics : Pandas, NumPy, SciPy. Plotting and visualization with python : Introduction to Matplotlib, Basic Plotting with Matplotlib, Create Histogram, BarChart, Pie chart, Box Plot, violin plot using Matplotlib.	
	6.2	Introduction to seaborn Library, MultiplePlots, Regressionplot, regplot. **(Refer Chapter 6)**	
		Total	39

Assessment

Internal Assessment

Assessment consists of two class tests of 20 marks each. The first-class test is to be conducted when approx. 40% syllabus is completed and second-class test when additional 40% syllabus is completed. Duration of each test shall be one hour.

End Semester Theory Examination

1. Question paper will consist of 6 questions, each carrying 20 marks.
2. The students need to solve a total of 4 questions.
3. Question No.1 will be compulsory and based on the entire syllabus.
4. Remaining question (Q.2 to Q.6) will be selected from all the modules.

□□□

Index

□□□

Module 1

CHAPTER 1

Introduction to Data Analytics & life Cycle

University Prescribed Syllabus

Data Analytics Lifecycle overview : Key Roles for a Successful Analytics, Background and Overview of Data Analytics Lifecycle Project

Phase 1 : Discovery: Learning the Business Domain, Resources Framing the Problem, Identifying Key Stakeholders. Interviewing the Analytics Sponsor, Developing Initial Hypotheses Identifying Potential Data Sources

Phase 2 : Data Preparation: Preparing the Analytic Sandbox, Performing ETLT, Learning About the Data, DataConditioning, Survey and visualize, Common Tools for the Data Preparation Phase

Phase 3 : Model Planning: Data Exploration and Variable Selection, Model Selection ,Common Tools for the Model Planning Phase

Phase 4 : Model Building: Common Tools for the Model Building Phase

Phase 5 : Communicate Results

Phase 6: Operationalize.

1.1 DATA ANALYTICS LIFECYCLE OVERVIEW

GQ. Explain Data Analytics Lifecycle ?

- The Data analytics lifecycle was designed to address Big Data problems and data science projects. The process is repeated to show the real projects.
- To address the specific demands for conducting analysis on Big Data, the step-by-step methodology is required to plan the various tasks associated with the acquisition, processing, analysis, and recycling of data.
- The Data Analytics Life Cycle covers the process of generating, collecting, processing, using, and analyzing data to achieve corporate objectives. It provides a systematic method for managing data to convert it into information that can be used to achieve organizational and project goals. The process gives guidance and strategies for extracting information from data and moving forward on the proper path to achieve corporate objectives.

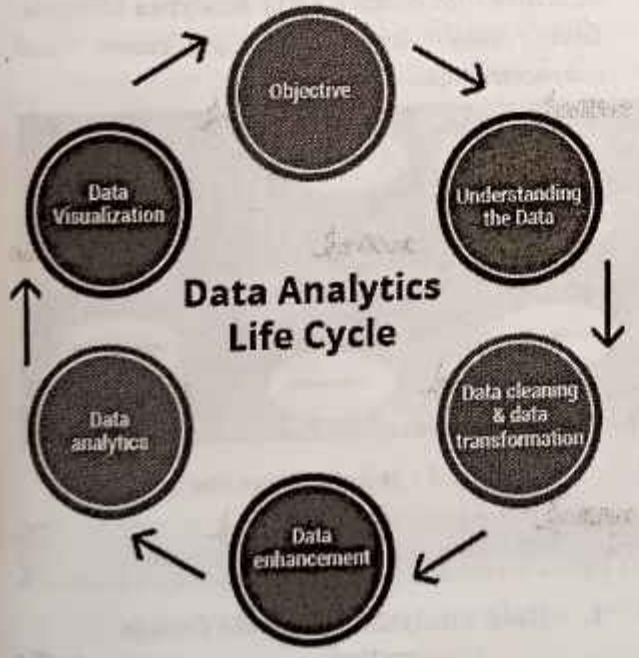


Fig. 1.1.1 : Data Analytics Lifecycle

- Data professionals use the circular nature of the Life Cycle to go ahead or backward with data analytics.
- Based on the new information, they can decide whether to continue with their current research or abandon it and redo the entire analysis. Throughout the process, they are guided by the Data Analytics Life Cycle.

1.1.1 Key Roles for a Successful Analytics

- There are certain key roles that are required for the complete and fulfilled functioning of the data science team to execute projects on analytics successfully. The key roles are seven in number.
- Each key plays a crucial role in developing a successful analytics project. There is no hard and fast rule for considering the listed seven roles, they can be used fewer or more depending on the scope of the project, skills of the participants, and organizational structure.

Example

- For a small, versatile team, these listed seven roles may be fulfilled by only three to four people but a large project on the contrary may require 20 or more people for fulfilling the listed roles.

Key Roles for a Data analytics project

(1) Business User

- The business user is the one who understands the main area of the project and is also basically benefited from the results.
- This user gives advice and consult the team working on the project about the value of the results obtained and how the operations on the outputs are done.
- The business manager, line manager, or deep subject matter expert in the project mains fulfills this role.

(2) Project Sponsor

- The Project Sponsor is the one who is responsible to initiate the project. Project Sponsor provides the actual requirements for the project and presents the basic business issue.
- He generally provides the funds and measures the degree of value from the final output of the team working on the project. This person introduce the prime concern and brooms the desired output.

(3) Project Manager

This person ensures that key milestone and purpose of the project is met on time and of the expected quality.

(4) Business Intelligence Analyst

- Business Intelligence Analyst provides business domain perfection based on a detailed and deep understanding of the data, key performance indicators (KPIs), key matrix, and business intelligence from a reporting point of view.
- This person generally creates fascia and reports and knows about the data feeds and sources.

(5) Database Administrator (DBA)

- DBA facilitates and arrange the database environment to support the analytics need of the team working on a project.
- His responsibilities may include providing permission to key databases or tables and making sure that the appropriate security stages are in their correct places related to the data repositories or not.

(6) Data Engineer

- Data engineer grasps deep technical skills to assist with tuning SQL queries for data management and data extraction and provides support for data intake into the analytic sandbox.
- The data engineer works jointly with the data scientist to help build data in correct ways for analysis.

(7) Data Scientist

- Data scientist facilitates with the subject matter expertise for analytical techniques, data modelling, and applying correct analytical techniques for a given business issues.
- He ensures overall analytical objectives are met. Data scientists outline and apply analytical methods and proceed towards the data available for the concerned project.

1.1.2 Background and Overview of Data Analytics Lifecycle Project

- In today's digital-first world, data is of immense importance. It undergoes various stages throughout its life, during its creation, testing, processing, consumption, and reuse.
- Data Analytics Lifecycle maps out these stages for professionals working on data analytics projects.
- These phases are arranged in a circular structure that forms a Data Analytics Lifecycle. Each step has its significance and characteristics.

Fig. 1.1.2 : Data analytics lifecycle

GQ. Why is Data Analytics Lifecycle Essential?

1. Data Analytics Lifecycle Design

- The Data Analytics Lifecycle is designed to be used with significant big data projects.

- It is used to portray the actual project correctly; the cycle is iterative.
- A step-by-step technique is needed to arrange the actions and tasks involved in gathering, processing, analyzing, and reusing data to explore the various needs for assessing the information on big data.
- Data analysis is modifying, processing, and cleaning raw data to obtain useful, significant information that supports business decision-making.

2. Importance of Data Analytics Lifecycle

- Data Analytics Lifecycle defines the roadmap of how data is generated, collected, processed, used, and analyzed to achieve business goals. It offers a systematic way to manage data for converting it into information that can be used to fulfill organizational and project goals. The process provides the direction and methods to extract information from the data and proceed in the right direction to accomplish business goals.
- Data professionals use the lifecycle's circular form to proceed with data analytics in either a forward or backward direction. Based on the newly received insights, they can decide whether to proceed with their existing research or scrap it and redo the complete analysis. The Data Analytics lifecycle guides them throughout this process.

3. Data Analytics Lifecycle Phases

- There's no defined structure of the phases in the life cycle of Data Analytics; thus, there may not be uniformity in these steps.
- There can be some data professionals that follow additional steps, while there may be some who skip some stages altogether or work on different phases simultaneously.
- Let us discuss the various phases of the data analytics life cycle.

1.2 PHASE 1: DISCOVERY

GQ. Explain Phase 1?

- This phase is all about defining the data's purpose and how to achieve it by the end of the data analytics lifecycle. The stage consists of identifying critical objectives a business is trying to discover by mapping out the data.
- During this process, the team learns about the business domain and checks whether the business unit or organization has worked on similar projects to refer to any learnings.

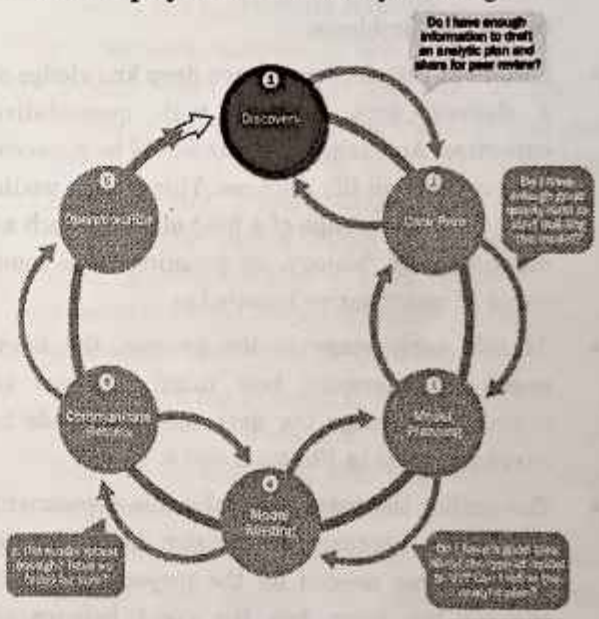


Fig. 1.2.1 : Phase 1

- The team also evaluates technology, people, data, and time in this phase. For example, the team can use Excel while dealing with a small dataset.
- However, heftier tasks demand more rigid tools for data preparation and exploration. The team will need to use Python, R, Tableau Desktop or Tableau Prep, and other data-cleaning tools in such scenarios.

- This phase's critical activities include framing the business problem, formulating initial hypotheses to test, and beginning data learning

1.2.1 Learning the Business Domain

- Understanding the domain area of the problem is essential. In many cases, data scientists will have deep computational and quantitative knowledge that can be broadly applied across many disciplines. An example of this role would be someone with an advanced degree in applied mathematics or statistics.
- These data scientists have deep knowledge of the methods, techniques, and ways for applying heuristics to a variety of business and conceptual problems.
- Others in this area may have deep knowledge of a domain area, coupled with quantitative expertise. An example of this would be someone with a Ph.D. in life sciences. This person would have deep knowledge of a field of study, such as oceanography, biology, or genetics, with some depth of quantitative knowledge.
- At this early stage in the process, the team needs to determine how much business or domain knowledge the data scientist needs to develop models in Phases 3 and 4.
- The earlier the team can make this assessment the better, because the decision helps dictate the resources needed for the project team and ensures the team has the right balance of domain knowledge and technical expertise.

1.2.2 Resources Framing the Problem

- As part of the discovery phase, the team needs to assess the resources available to support the project. In this context, resources include technology, tools, systems, data, and people.
- During this scoping, consider the available tools and technology the team will be using and the types of systems needed for later phases to operationalize the models.
- In addition, try to evaluate the level of analytical sophistication within the organization and gaps that may exist related to tools, technology, and skills. For instance, for the model being developed to have longevity in an organization, consider what types of skills and roles will be required that may not exist today.
- For the project to have long-term success, what types of skills and roles will be needed for the recipients of the model being developed? Does the requisite level of expertise exist within the organization today, or will it need to be cultivated?
- Answering these questions will influence the techniques the team selects and the kind of implementation the team chooses to pursue in subsequent phases of the Data Analytics Lifecycle.
- In addition to the skills and computing resources, it is advisable to take inventory of the types of data available to the team for the project. Consider if the data available is
- sufficient to support the project's goals. The team will need to determine whether it must collect additional data, purchase it from outside sources, or transform existing data.
- Often, projects are started looking only at the data available. When the data is less than hoped for, the size and scope of the project is reduced to work within the constraints of the existing data.

1.2.3 Identifying Key Stakeholders

- Framing the problem well is critical to the success of the project. Framing is the process of stating the analytics problem to be solved.
- At this point, it is a best practice to write down the problem statement and share it with the key stakeholders.

- Each team member may hear slightly different things related to the needs and the problem and have somewhat different ideas of possible solutions. For these reasons, it is crucial to state the analytics problem, as well as why and to whom it is important.
- Essentially, the team needs to clearly articulate the current situation and its main challenges.
- As part of this activity, it is important to identify the main objectives of the project, identify what needs to be achieved in business terms, and identify what needs to be done to meet the needs.
- Additionally, consider the objectives and the success criteria for the project. What is the team attempting to achieve by doing the project, and what will be considered "good enough" as an outcome of the project? This is critical to document and share with the project team and key stakeholders.
- It is best practice to share the statement of goals and success criteria with the team and confirm alignment with the project sponsor's expectations.
- Perhaps equally important is to establish failure criteria. Most people doing projects prefer only to think of the success criteria and what the conditions will look like when the participants are successful.
- However, this is almost taking a best-case scenario approach, assuming that everything will proceed as planned and the project team will reach its goals.
- However, no matter how well planned, it is almost impossible to plan for everything that will emerge in a project. The failure criteria will guide the team in understanding when it is best to stop trying or settle for the results that have been gleaned from the data.
- Many times people will continue to perform analyses past the point when any meaningful insights can be drawn from the data. Establishing criteria for both success and failure helps the participants avoid unproductive effort and remain aligned with the project sponsors

1.2.4 Interviewing the Analytics Sponsor

- The team should plan to collaborate with the stakeholders to clarify and frame the analytics problem.
- At the outset, project sponsors may have a predetermined solution that may not necessarily realize the desired outcome. In these cases, the team must use its knowledge and expertise to identify the true underlying problem and appropriate solution.
- For instance, suppose in the early phase of a project, the team is told to create a recommender system for the business and that the way to do this is by speaking with three people and integrating the product recommender into a legacy corporate system. Although this may be a valid approach, it is important to test the assumptions and develop a clear understanding of the problem.
- The data science team typically may have a more objective understanding of the problem set than the stakeholders, who may be suggesting solutions to a given problem.
- Therefore, the team can probe deeper into the context and domain to clearly define the problem and propose possible paths from the problem to a desired outcome.
- In essence, the data science team can take a more objective approach, as the stakeholders may have developed biases over time, based on their experience. Also, what may have been true in the past may no longer be a valid working assumption.
- One possible way to circumvent this issue is for the project sponsor to focus on clearly defining the requirements, while the other members of the data science team focus on the methods needed to achieve the goals.

- When interviewing the main stakeholders, the team needs to take time to thoroughly interview the project sponsor, who tends to be the one funding the project or providing the high-level requirements.
- This person understands the problem and usually has an idea of a potential working solution. It is critical to thoroughly understand the sponsor's perspective to guide the team in getting started on the project.
- Here are some tips for interviewing project sponsors: Prepare for the interview; draft questions, and review with colleagues. Use open-ended questions; avoid asking leading questions.

Probe for details and pose follow-up questions.

1. Avoid filling every silence in the conversation;

- Give the other person time to think.
- Let the sponsors express their ideas and ask clarifying questions, such as "Why? Is that correct? Is this idea on target? Is there anything else?"

2. Use active listening techniques

- Repeat back what was heard to make sure the team heard it correctly, or reframe what was said.
- Try to avoid expressing the team's opinions, hich can introduce bias; instead, focus on listening.
- Be mindful of the body language of the interviewers and stakeholders; use eye contact where appropriate, and be attentive.

3. Minimize distractions.

- Document what the team heard, and review it with the sponsors.

1.2.5 Developing Initial Hypotheses Identifying Potential Data Sources

- Developing a set of IHs is a key facet of the discovery phase. This step involves forming ideas that the team can test with data.
- Generally, it is best to come up with a few primary hypotheses to test and then be creative about developing several more.
- These IHs form the basis of the analytical tests the team will use in later phases and serve as the foundation for the findings in Phase 5. As a result, the team will have a much richer set of observations to choose from and more choices for agreeing upon the most impactful conclusions from a project.
- Another part of this process involves gathering and assessing hypotheses from stakeholders and domain experts who may have their own perspective on what the problem is, what the solution should be, and how to arrive at a solution. These stakeholders would know the domain area well and can offer suggestions on ideas to test as the team formulates hypotheses during this phase.
- The team will likely collect many ideas that may illuminate the operating assumptions of the stakeholders. These ideas will also give the team opportunities to expand the project scope into adjacent spaces where it makes sense or design experiments in a meaningful way to address the most important interests of the stakeholders.
- As part of this exercise, it can be useful to obtain and explore some initial data to inform discussions with stakeholders during the hypothesis-forming stage.
- As part of the discovery phase, identify the kinds of data the team will need to solve the problem. Consider the volume, type, and time span of the data needed to test the hypotheses.

- Ensure that the team can access more than simply aggregated data. In most cases, the team will need the raw data to avoid introducing bias for the downstream analysis.
- The main characteristics of the data, with regard to its volume, variety, and velocity of change. A thorough diagnosis of the data situation will influence the kinds of tools and techniques to use in Phases 2-4 of the Data Analytics Lifecycle.
- In addition, performing data exploration in this phase will help the team determine the amount of data needed, such as the amount of historical data to pull from existing systems and the data structure.
- Develop an idea of the scope of the data needed, and validate that idea with the domain experts on the project.
- The team should perform five main activities during this step of the discovery phase:
- Identify data sources: Make a list of candidate data sources the team may need to test the initial hypotheses outlined in this phase. Make an inventory of the datasets currently available and those that can be purchased or otherwise acquired for the tests the team wants to perform. Capture aggregate data sources: This is for previewing the data and providing high-level understanding. It enables the team to gain a quick overview of the data and perform further exploration on specific areas. It also points the team to possible areas of interest within the data.
- Review the raw data: Obtain preliminary data from initial data feeds. Begin understanding the interdependencies among the data attributes, and become familiar with the content of the data, its quality, and its limitations.
- Evaluate the data structures and tools needed: The data type and structure dictate which tools the team can use to analyze the data. This evaluation gets the team thinking about which technologies may be good candidates for the project and how to start getting access to these tools.

1.3 PHASE 2: DATA PREPARATION

GQ. Explain Phase 2?

- The second phase of the Data Analytics Lifecycle involves data preparation, which includes the steps to explore, preprocess, and condition data prior to modeling and analysis.
- In this phase, the team needs to create a robust environment in which it can explore the data that is separate from a production environment. Usually, this is done by preparing an analytics sandbox.
- To get the data into the sandbox, the team needs to perform ETLT, by a combination of extracting, transforming, and loading data into the sandbox. Once the data is in the sandbox, the team needs to learn about the data and become familiar with it.
- Understanding the data in detail is critical to the success of the project. The team also must decide how to condition and transform data to get it into a format to facilitate subsequent analysis.
- The team may perform data visualizations to help team members understand the data, including its trends, outliers, and relationships among data variables. Each of these steps of the data preparation phase is discussed throughout this section.
- Data preparation tends to be the most labor-intensive step in the analytics lifecycle. In fact, it is common for teams to spend at least 50% of a data science project's time in this critical phase. If the team cannot obtain enough data of sufficient quality, it may be unable to perform the subsequent steps in the lifecycle process.

- Fig. 1.3.1 shows an overview of the Data Analytics Lifecycle for Phase 2. The data preparation phase is generally the most iterative and the one that teams tend to underestimate most often. This is because most teams and leaders are anxious to begin
- analyzing the data, testing hypotheses, and getting answers to some of the questions posed in Phase 1. Many tend to jump into Phase 3 or Phase 4 to begin rapidly developing models and algorithms without spending the time to prepare the data for modeling. Consequently, teams come to realize the data they are working with does not allow them to execute the models they want, and they end up back in Phase 2 anyway.

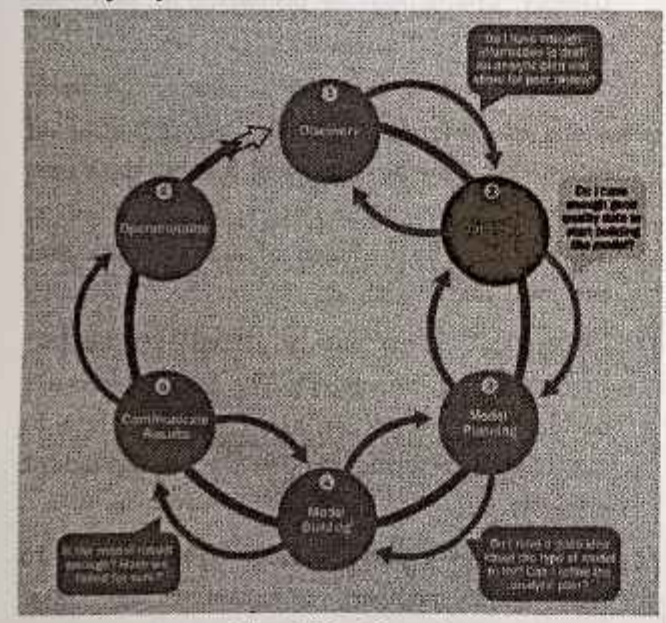


Fig. 1.3.1 : Phase 2

1.3.1 Preparing the Analytic Sandbox

- The first subphase of data preparation requires the team to obtain an analytic sandbox (also commonly referred to as a workspace), in which the team can explore the data without interfering with live production databases.
- Consider an example in which the team needs to work with a company's financial data.
- The team should access a copy of the financial data from the analytic sandbox rather than interacting with the production version of the organization's main database, because that will be tightly controlled and needed for financial reporting.
- When developing the analytic sandbox, it is a best practice to collect all kinds of data there, as team members need access to high volumes and varieties of data for a Big Data analytics project.
- This can include everything from summary-level aggregated data, structured data, raw data feeds, and unstructured text data from call logs or web logs, depending on the kind of analysis the team plans to undertake.
- This expansive approach for attracting data of all kind differs considerably from the approach advocated by many information technology (IT) organizations.
- Many IT groups provide access to only a particular subsegment of the data for a specific purpose.
- Often, the mindset of the IT group is to provide the minimum amount of data required to allow the team to achieve its objectives. Conversely, the data science team wants access to everything.
- From its perspective, more data is better, as oftentimes data science projects are a mixture of purpose-driven analyses and experimental approaches to test a variety of ideas.
- In this context, it can be challenging for a data science team if it has to request access to each and every dataset and attribute one at a time.
- Because of these differing views on data access and use, it is critical for the data science team to collaborate with IT, make clear what it is trying to accomplish, and align goals.

1.3.2 Performing ETLT

- As the team looks to begin data transformations, make sure the analytics sandbox has ample bandwidth and reliable network connections to the underlying data sources to enable uninterrupted read and write.
- In ETL, users perform extract, transform, load processes to extract data from a datastore, perform data transformations, and load the data back into the datastore. However, the analytic sandbox approach differs slightly; it advocates extract, load, and then transform. In this case, the data is extracted in its raw form and loaded into the datastore, where analysts can choose to transform the data into a new state or leave it in its original, raw condition.
- The reason for this approach is that there is significant value in preserving the raw data and including it in the sandbox before any transformations take place.
- For instance, consider an analysis for fraud detection on credit card usage. Many times, outliers in this data population can represent higher-risk transactions that may be indicative of fraudulent credit card activity.
- Using ETL, these outliers may be inadvertently filtered out or transformed and cleaned before being loaded into the datastore. In this case, the very data that would be needed to evaluate instances of fraudulent activity would be inadvertently cleansed, preventing the kind of analysis that a team would want to do.
- Following the ELT approach gives the team access to clean data to analyze after the data has been loaded into the database and gives access to the data in its original form for finding hidden nuances in the data. This approach is part of the reason that the analytic sandbox can quickly grow large.
- The team may want clean data and aggregated data and may need to keep a copy of the original data to compare against or look for hidden patterns that may have existed in the data before the cleaning stage. This process can be summarized as ETLT to reflect the fact that a team may choose to perform ETL in one case and ELT in another.

1.3.3 Learning About the Data

- A critical aspect of a data science project is to become familiar with the data itself.
- Spending time to learn the nuances of the datasets provides context to understand what constitutes a reasonable value and expected output versus what is a surprising finding.
- In addition, it is important to catalog the data sources that the team has access to and identify additional data sources that the team can leverage but perhaps does not have access to today.
- Some of the activities in this step may overlap with the initial investigation of the datasets that occur in the discovery phase. Doing this activity accomplishes several goals.
- Clarifies the data that the data science team has access to at the start of the project .
- Highlights gaps by identifying datasets within an organization that the team may find useful but may not be accessible to the team today. As a consequence, this activity can trigger a project to begin building relationships with the data owners and finding ways to share data in appropriate ways.
- In addition, this activity may provide an impetus to begin collecting new data that benefits the organization or a specific longterm project.
- Identifies datasets outside the organization that may be useful to obtain, through open APIs, data sharing, or purchasing data to supplement already existing datasets.

Dataset	Data Available and Accessible	Data Available, but not Accessible	Data to Collect	Data to Obtain from Third Party Sources
Products shipped	*			
Product Financials		*		
Product Call Center Data		*		
Live Product Feedback Surveys			*	
Product Sentiment from Social Media				*

Table Simple DataSet Inventory

1.3.4 Data Conditioning

- Data conditioning refers to the process of cleaning data, normalizing datasets, and performing transformations on the data. A critical step within the Data Analytics
- Lifecycle, data conditioning can involve many complex steps to join or merge datasets or otherwise get datasets into a state that enables analysis in further phases.
- Data conditioning is often viewed as a preprocessing step for the data analysis because it involves many operations on the dataset before developing models to process or analyze the data.
- This implies that the data-conditioning step is performed only by IT, the data owners, a DBA, or a data engineer. However, it is also important to involve the data scientist in this step because many decisions are made in the data conditioning phase that affect subsequent analysis.
- Part of this phase involves deciding which aspects of particular datasets will be useful to analyze in later steps.
- Because teams begin forming ideas in this phase about which data to keep and which data to transform or discard, it is important to involve multiple team members in these decisions.
- Leaving such decisions to a single person may cause teams to return to this phase to retrieve data that may have been discarded.
- What are the data sources? What are the target fields (for example, columns of the tables)?

How clean is the data?

- How consistent are the contents and files? Determine to what degree the data contains missing or inconsistent values and if the data contains values deviating from normal.
- Assess the consistency of the data types. For instance, if the team expects certain data to be numeric, confirm it is numeric or if it is a mixture of alphanumeric strings and text.
- Review the content of data columns or other inputs, and check to ensure they make sense. For instance, if the project involves analyzing income levels, preview the data to confirm that the income values are positive or if it is acceptable to have zeros or negative values.
- Look for any evidence of systematic error. Examples include data feeds from sensors or other data sources breaking without anyone noticing, which causes invalid, incorrect, or missing data values.
- In addition, review the data to gauge if the definition of the data is the same over all measurements. In some cases, a data column is repurposed, or the column stops being populated, without this change being annotated or without others being notified.

1.3.5 Survey and Visualize

- After the team has collected and obtained at least some of the datasets needed for the subsequent analysis, a useful step is to leverage data visualization tools to gain an overview of the data.
- Seeing high-level patterns in the data enables one to understand characteristics about the data very quickly.
- One example is using data visualization to examine data quality, such as whether the data contains many unexpected values or other indicators of dirty data.
- Another example is skewness, such as if the majority of the data is heavily shifted toward one value or end of a continuum.

- Review data to ensure that calculations remained consistent within columns or across tables for a given data field. For instance, did customer lifetime value change at some point in the middle of data collection? Or if working with financials, did the interest calculation change from simple to compound at the end of the year?
- Does the data distribution stay consistent over all the data? If not, what kinds of actions should be taken to address this problem?
- Assess the granularity of the data, the range of values, and the level of aggregation of the data.
- Does the data represent the population of interest? For marketing data, if the project is focused on targeting customers of child-rearing age, does the data represent that, or is it full of senior citizens and teenagers?
- For time-related variables, are the measurements daily, weekly, monthly? Is that good enough? Is time measured in seconds everywhere? Or is it in milliseconds in some places? Determine the level of granularity of the data needed for the analysis, and assess whether the current level of timestamps on the data meets that need.
- Is the data standardized/normalized? Are the scales consistent? If not, how consistent or irregular is the data?

1.3.6 Common Tools for the Data Preparation Phase

Several tools are commonly used for this phase

- **Hadoop** can perform massively parallel ingest and custom analysis for web traffic parsing, GPS location analytics, genomic analysis, and combining of massive unstructured data feeds from multiple sources.
- **Alpine Miner** provides a Graphical User Interface (GUI) for creating analytic workflows, including data manipulations and a series of analytic events such as staged data-mining techniques (for example, first select the top 100 customers, and then run descriptive statistics and clustering) on Postgres SQL and other Big Data sources.
- **OpenRefine** (formerly called Google Refine) is "a free, open source, powerful tool for working with messy data." It is a popular GUI-based tool for performing data transformations, and it's one of the most robust free tools currently available
- Similar to OpenRefine, **Data Wrangler** is an interactive tool for data cleaning and transformation.
- Wrangler was developed at Stanford University and can be used to perform many transformations on a given dataset.
- In addition, data transformation outputs can be put into Java or Python.
- The advantage of this feature is that a subset of the data can be manipulated in Wrangler via its GUI, and then the same operations can be written out as Java or Python code to be executed against the full, larger dataset offline in a local analytic sandbox.

1.4 MODEL PLANNING

GQ. Explain Phase 3?

- In Phase 3, the data science team identifies candidate models to apply to the data for clustering, classifying, or finding relationships in the data depending on the goal of the project, as shown in Fig. 1.4.1.
- It is during this phase that the team refers to the hypotheses developed in Phase 1, when they first became acquainted with the data and understanding the business problems or domain area.

These hypotheses help the team frame the analytics to execute in Phase 4 and select the right methods to achieve its objectives.

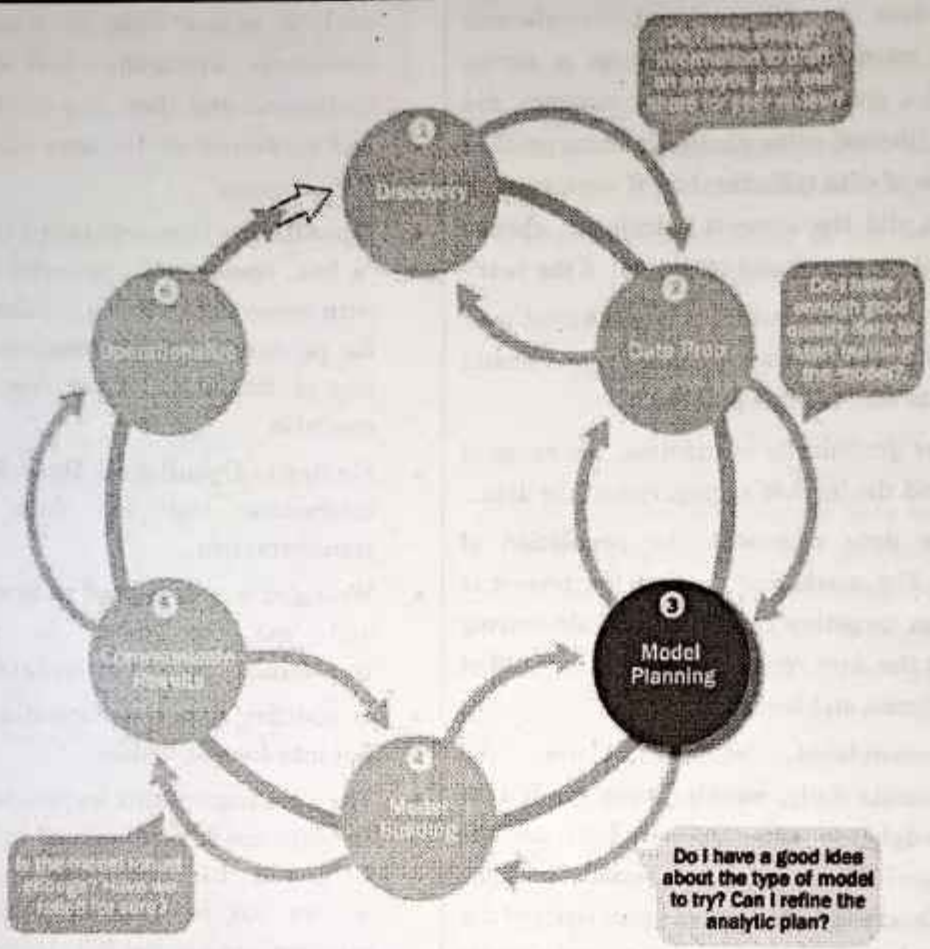


Fig. 1.4.1 : Phase 3

1.4.1 Data Exploration and Variable Selection

- Although some data exploration takes place in the data preparation phase, those activities focus mainly on data hygiene and on assessing the quality of the data itself.
- In Phase 3, the objective of the data exploration is to understand the relationships among the variables to inform selection of the variables and methods and to understand the problem domain.
- As with earlier phases of the Data Analytics Lifecycle, it is important to spend time and focus attention on this preparatory work to make the subsequent phases of model selection and execution easier and more efficient.
- A common way to conduct this step involves using tools to perform data visualizations.
- Approaching the data exploration in this way aids the team in previewing the data and assessing relationships between variables at a high level.
- In many cases, stakeholders and subject matter experts have instincts and hunches about what the data science team should be considering and analyzing.
- Likely, this group had some hypothesis that led to the genesis of the project.
- Often, stakeholders have a good grasp of the problem and domain, although they may not be aware of the subtleties within the data or the model needed to accept or reject a hypothesis.

- Other times, stakeholders may be correct, but for the wrong reasons (for instance, they may be correct about a correlation that exists but infer an incorrect reason for the correlation).
- Meanwhile, data scientists have to approach problems with an unbiased mind-set and be ready to question all assumptions.
- As the team begins to question the incoming assumptions and test initial ideas of the project sponsors and stakeholders, it needs to consider the inputs and data that will be needed, and then it must examine whether these inputs are actually correlated with the outcomes that the team plans to predict or analyze.
- Some methods and types of models will handle correlated variables better than others.
- Depending on what the team is attempting to solve, it may need to consider an alternate method, reduce the number of data inputs, or transform the inputs to allow the team to use the best method for a given business problem.

1.4.2 Model Selection

- In the model selection subphase, the team's main goal is to choose an analytical technique, or a short list of candidate techniques, based on the end goal of the project.
- For the context of this book, a model is discussed in general terms. In this case, a model simply refers to an abstraction from reality. One observes events happening in a real-world situation or with live data and attempts to construct models that emulate this behavior with a set of rules and conditions.
- In the case of machine learning and data mining, these rules and conditions are grouped into several general sets of techniques, such as classification, association rules, and clustering. When reviewing this list of types of potential models, the team can winnow down the list to several viable models to try to address a given problem.
- An additional consideration in this area for dealing with Big Data involves determining if the team will be using techniques that are best suited for structured data, unstructured data, or a hybrid approach. For instance, the team can leverage MapReduce to analyze unstructured data.
- Lastly, the team should take care to identify and document the modeling assumptions it is making as it chooses and constructs preliminary models.
- Typically, teams create the initial models using a statistical software package such as R, SAS, or Matlab.
- Although these tools are designed for data mining and machine learning algorithms, they may have limitations when applying the models to very large datasets, as is common with Big Data.
- As such, the team may consider redesigning these algorithms to run in the database itself during the pilot phase mentioned in Phase 6.
- The team can move to the model building phase once it has a good idea about the type of model to try and the team has gained enough knowledge to refine the analytics plan.
- Advancing from this phase requires a general methodology for the analytical model, a solid understanding of the variables and techniques to use, and a description or diagram of the analytic workflow.

1.4.3 Common Tools for the Model Planning Phase

- **Many tools are available to assist in this phase. Here are several of the more common ones :**
- R has a complete set of modeling capabilities and provides a good environment for building interpretive models with high-quality code.

- In addition, it has the ability to interface with databases via an ODBC connection and execute statistical tests and analyses against Big Data via an open source connection.
- These two factors make R well suited to performing statistical tests and analytics on Big Data.
- As of this writing, R contains nearly 5,000 packages for data analysis and graphical representation.
- New packages are posted frequently, and many companies are providing value-add services for R (such as training, instruction, and best practices), as well as packaging it in ways to make it easier to use and more robust.
- This phenomenon is similar to what happened with Linux in the late 1980s and early 1990s, when companies appeared to package and make Linux easier for companies to consume and deploy.
- Use R with file extracts for offline analysis and optimal performance, and use RODBC connections for dynamic queries and faster development.
- SQL Analysis services can perform in-database analytics of common data mining functions, involved aggregations, and basic predictive models.
- SAS/ACCESS provides integration between SAS and the analytics sandbox via multiple data connectors such as OBDC, JDBC, and OLE DB.
- SAS itself is generally used on file extracts, but with SAS/ACCESS, users can connect to relational databases (such as Oracle or Teradata) and data warehouse appliances (such as Greenplum or Aster), files, and enterprise applications (such as SAP and Salesforce.com).

1.5 PHASE 4: MODEL BUILDING

GQ Explain Phase 4?

- In Phase 4, the data science team needs to develop datasets for training, testing, and production purposes.
- These datasets enable the data scientist to develop the analytical model and train it ("training data"), while holding aside some of the data ("hold-out data" or "test data") for testing the model.
- During this process, it is critical to ensure that the training and test datasets are sufficiently robust for the model and analytical techniques.
- A simple way to think of these datasets is to view the training dataset for conducting the initial experiments and the test sets for validating an approach once the initial experiments and models have been run.
- In the model building phase, shown in Fig. 1.5.1, an analytical model is developed and fit on the training data and evaluated (scored) against the test data.
- The phases of model planning and model building can overlap quite a bit, and in practice one can iterate back and forth between the two phases for a while before settling on a final model
- Although the modeling techniques and logic required to develop models can be highly complex, the actual duration of this phase can be short compared to the time spent preparing the data and defining the approaches.
- In general, plan to spend more time preparing and learning the data (Phases 1–2) and crafting a presentation of the findings (Phase 5).
- Phases 3 and 4 tend to move more quickly, although they are more complex from a conceptual standpoint.

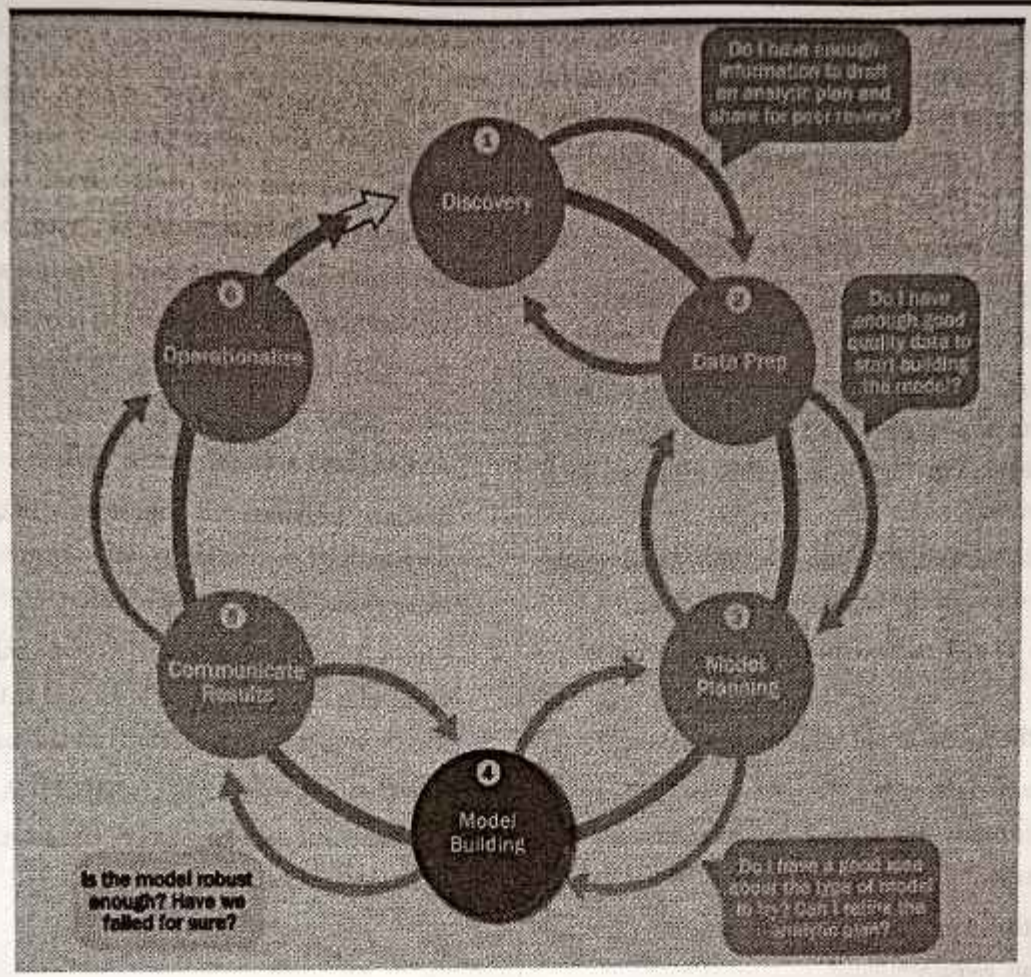


Fig. 1.5.1 Phase 4

- As part of this phase, the data science team needs to execute the models defined in Phase 3.
- During this phase, users run models from analytical software packages, such as R or SAS, on file extracts and small datasets for testing purposes.
- On a small scale, assess the validity of the model and its results.
- For instance, determine if the model accounts for most of the data and has robust predictive power.
- At this point, refine the models to optimize the results, such as by modifying variable inputs or reducing correlated variables where appropriate.
- In Phase 3, the team may have had some knowledge of correlated variables or problematic data attributes, which will be confirmed or denied once the models are actually executed.
- When immersed in the details of constructing models and transforming data, many small decisions are often made about the data and the approach for the modeling.
- These details can be easily forgotten once the project is completed.
- Therefore, it is vital to record the results and logic of the model during this phase.
- In addition, one must take care to record any operating assumptions that were made in the modeling process regarding the data or the context.

- Creating robust models that are suitable to a specific situation requires thoughtful consideration to ensure the models being developed ultimately meet the objectives outlined in Phase 1.

Questions to consider include these

(1) Does the model appear valid and accurate on the test data?

(2) Does the model output/behavior make sense to the domain experts? That is, does it appear as if the model is giving answers that make sense in this context?

(3) Do the parameter values of the fitted model make sense in the context of the domain?

(4) Is the model sufficiently accurate to meet the goal?

(5) Does the model avoid intolerable mistakes?

(6) Are more data or more inputs needed? Do any of the inputs need to be transformed or eliminated?

(7) Will the kind of model chosen support the runtime requirements?

(8) Is a different form of the model required to address the business problem?

- If so, go back to the model planning phase and revise the modeling approach.
- Once the data science team can evaluate either if the model is sufficiently robust to solve the problem or if the team has failed, it can move to the next phase in the Data Analytics Lifecycle.

1.5.1 Common Tools for the Model Building Phase

There are many tools available to assist in this phase, focused primarily on statistical analysis or data mining software. Common tools in this space include, but are not limited to, the following :

Commercial Tools

- SAS Enterprise Miner allows users to run predictive and descriptive models based on large volumes of data from across the enterprise. It interoperates with other large data stores, has many partnerships, and is built for enterprise-level computing and analytics.
- SPSS Modeler (provided by IBM and now called IBM SPSS Modeler) offers methods to explore and analyze data through a GUI.
- Matlab provides a high-level language for performing a variety of data analytics, algorithms, and data exploration.
- Alpine Miner provides a GUI front end for users to develop analytic workflows and interact with Big Data tools and platforms on the back end.
- STATISTICA and MATHEMATICA are also popular and well-regarded data mining and analytics tools.

Free or Open Source tools

- R and PL/R R was described earlier in the model planning phase, and PL/R is a procedural language for PostgreSQL with R. Using this approach means that R commands can be executed in database. This technique provides higher performance and is more scalable than running R in memory.
- Octave, a free software programming language for computational modeling, has some of the functionality of Matlab. Because it is freely available, Octave is used in major universities when teaching machine learning.
- WEKA is a free data mining software package with an analytic workbench. The functions created in WEKA can be executed within Java code.

- Python is a programming language that provides toolkits for machine learning and analysis, such as scikit-learn, numpy, scipy, pandas, and related data visualization using matplotlib.
- SQL in-database implementations, such as MADlib , provide an alterative to in-memory desktop analytical tools. MADlib provides an open-source machine learning library of algorithms that can be executed in-database, for PostgreSQL or Greenplum.

1.6 PHASE 5: COMMUNICATE RESULTS

GQ. Explain Phase 5?

- After executing the model, the team needs to compare the outcomes of the modeling to the criteria established for success and failure.
- In Phase 5, shown in Fig. 1.6.1, the team considers how best to articulate the findings and outcomes to the various team members and stakeholders, taking into account caveats, assumptions, and any limitations of the results. Because the presentation is often circulated within an organization, it is critical to articulate the results properly and position the findings in a way that is appropriate for the audience.

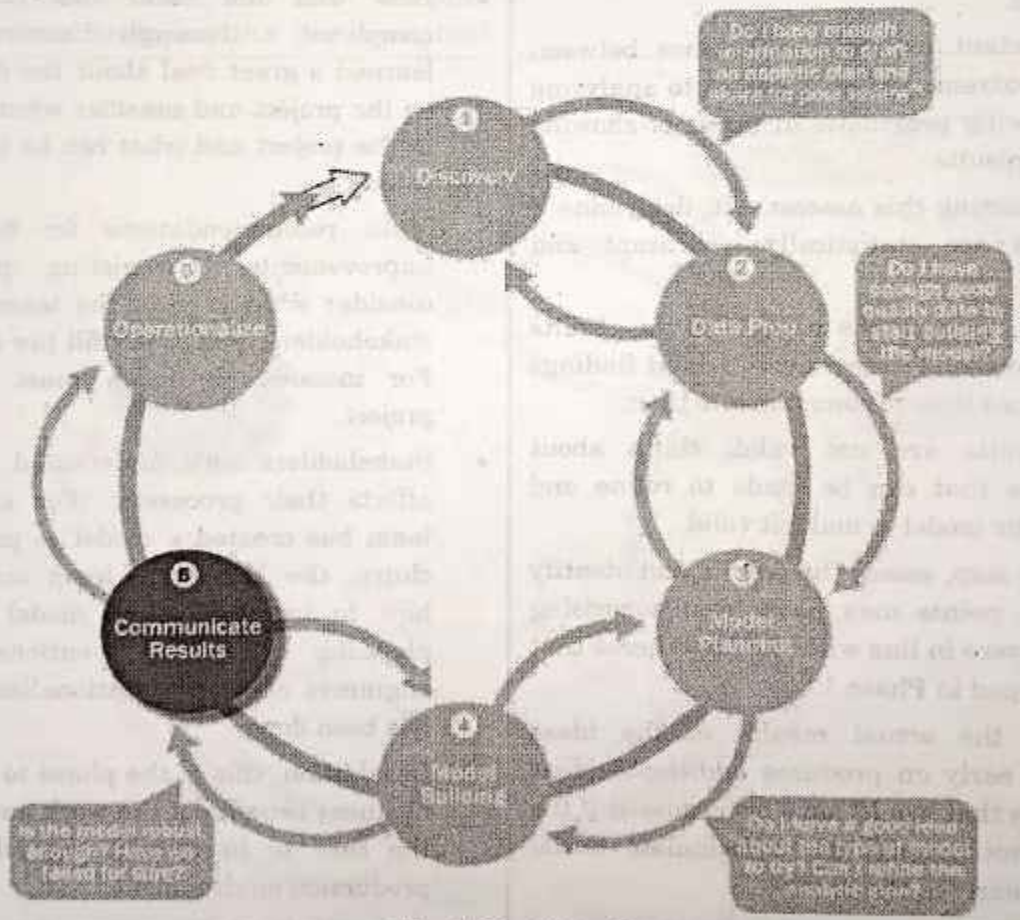


Fig. 1.6.1 : Phase 5

- As part of Phase 5, the team needs to determine if it succeeded or failed in its objectives. Many times people do not want to admit to failing, but in this instance failure should not be considered as a true failure, but rather as a failure of the data to accept or reject a given hypothesis adequately. This concept can be counter intuitive for those who have been told their whole careers not to fail.
- However, the key is to remember that the team must be rigorous enough with the data to determine whether it will prove or disprove the hypotheses outlined in Phase 1 (discovery). Sometimes teams have only done a superficial analysis, which is not robust enough to accept or reject a hypothesis. Other times, teams perform very robust analysis and are searching for ways to show results, even when results may not be there.
- It is important to strike a balance between these two extremes when it comes to analyzing data and being pragmatic in terms of showing real-world results.
- When conducting this assessment, determine if the results are statistically significant and valid.
- If they are, identify the aspects of the results that stand out and may provide salient findings when it comes time to communicate them.
- If the results are not valid, think about adjustments that can be made to refine and iterate on the model to make it valid.
- During this step, assess the results and identify which data points may have been surprising and which were in line with the hypotheses that were developed in Phase 1.
- Comparing the actual results to the ideas formulated early on produces additional ideas and insights that would have been missed if the team had not taken time to formulate initial hypotheses early in the process.
- By this time, the team should have determined which model or models address the analytical challenge in the most appropriate way.
- In addition, the team should have ideas of some of the findings as a result of the project.
- The best practice in this phase is to record all the findings and then select the three most significant ones that can be shared with the stakeholders.
- In addition, the team needs to reflect on the implications of these findings and measure the business value.
- Depending on what emerged as a result of the model, the team may need to spend time quantifying the business impact of the results to help prepare for the presentation and demonstrate the value of the findings.
- Doug Hubbard's work offers insights on how to assess intangibles in business and quantify the value of seemingly unmeasurable things.
- Now that the team has run the model, completed a thorough discovery phase, and learned a great deal about the datasets, reflect on the project and consider what obstacles were in the project and what can be improved in the future.
- Make recommendations for future work or improvements to existing processes, and consider what each of the team members and stakeholders needs to fulfill her responsibilities. For instance, sponsors must champion the project.
- Stakeholders must understand how the model affects their processes. (For example, if the team has created a model to predict customer churn, the Marketing team must understand how to use the churn model predictions in planning their interventions.) Production engineers need to operationalize the work that has been done.
- In addition, this is the phase to underscore the business benefits of the work and begin making the case to implement the logic into a live production environment.
- As a result of this phase, the team will have documented the key findings and major insights derived from the analysis.

- The deliverable of this phase will be the most visible portion of the process to the outside stakeholders and sponsors, so take care to clearly articulate the results, methodology, and business value of the findings. More details will be provided about data visualization tools.

1.7 PHASE 6 : OPERATIONALIZE

GQ. Explain Phase 6?

- In the final phase, the team communicates the benefits of the project more broadly and sets up a pilot project to deploy the work in a controlled way before broadening the work to a full enterprise or ecosystem of users.
- In Phase 4, the team scored the model in the analytics sandbox. Phase 6, shown in Fig. 1.7.1, represents the first time that most analytics teams approach deploying the new analytical methods or models in a production environment.
- Rather than deploying these models immediately on a wide-scale basis, the risk can be managed more effectively and the team can learn by undertaking a small scope, pilot deployment before a wide-scale rollout. This approach enables the team to learn about the performance and related constraints of the model in a production environment on a small scale and make adjustments before a full deployment.
- During the pilot project, the team may need to consider executing the algorithm in the database rather than with in-memory tools such as R because the run time is significantly faster and more efficient than running in-memory, especially on larger datasets.

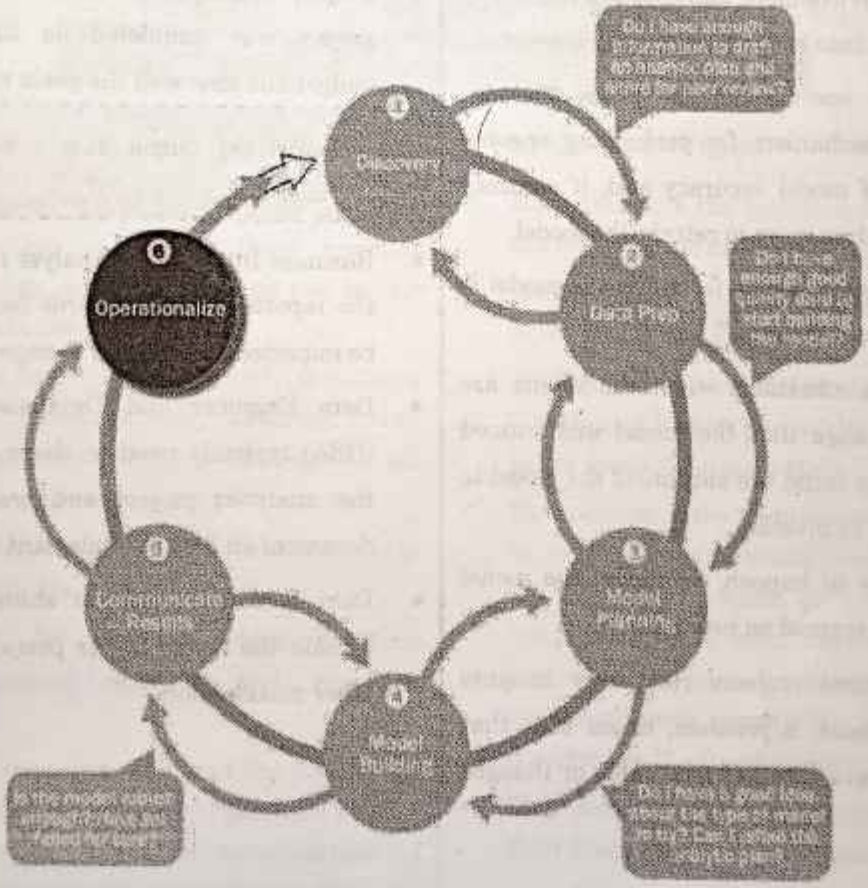


Fig. 1.7.1 : Phase 6

- While scoping the effort involved in conducting a pilot project, consider running the model in a production environment for a discrete set of products or a single line of business, which tests the model in a live setting.
- This allows the team to learn from the deployment and make any needed adjustments before launching the model across the enterprise.
- Be aware that this phase can bring in a new set of team members - usually the engineers responsible for the production environment who have a new set of issues and concerns beyond those of the core project team.
- This technical group needs to ensure that running the model fits smoothly into the production environment and that the model can be integrated into related business processes.
- Part of the operationalizing phase includes creating a mechanism for performing ongoing monitoring of model accuracy and, if accuracy degrades, finding ways to retrain the model.
- If feasible, design alerts for when the model is operating "out-of-bounds."
- This includes situations when the inputs are beyond the range that the model was trained on, which may cause the outputs of the model to be inaccurate or invalid.
- If this begins to happen regularly, the model needs to be retrained on new data.
- Often, analytical projects yield new insights about a business, a problem, or an idea that people may have taken at face value or thought was impossible to explore. Four main deliverables can be created to meet the needs of most stakeholders. This approach for developing.
- Fig. 1.7.2 portrays the key outputs for each of the main stakeholders of an analytics project and what they usually expect at the conclusion of a project.
- Business User typically tries to determine the benefits and implications of the findings to the business.
- Project Sponsor typically asks questions related to the business impact of the project, the risks and return on investment (ROI), and the way the project can be evangelized within the organization (and beyond).
- Project Manager needs to determine if the project was completed on time and within budget and how well the goals were met.

GQ Explain Key Output from a successful Analytic Project ?

- Business Intelligence Analyst needs to know if the reports and dashboards he manages will be impacted and need to change.
- Data Engineer and Database Administrator (DBA) typically need to share their code from the analytics project and create a technical document on how to implement it.
- Data Scientist needs to share the code and explain the model to her peers, managers, and other stakeholders.

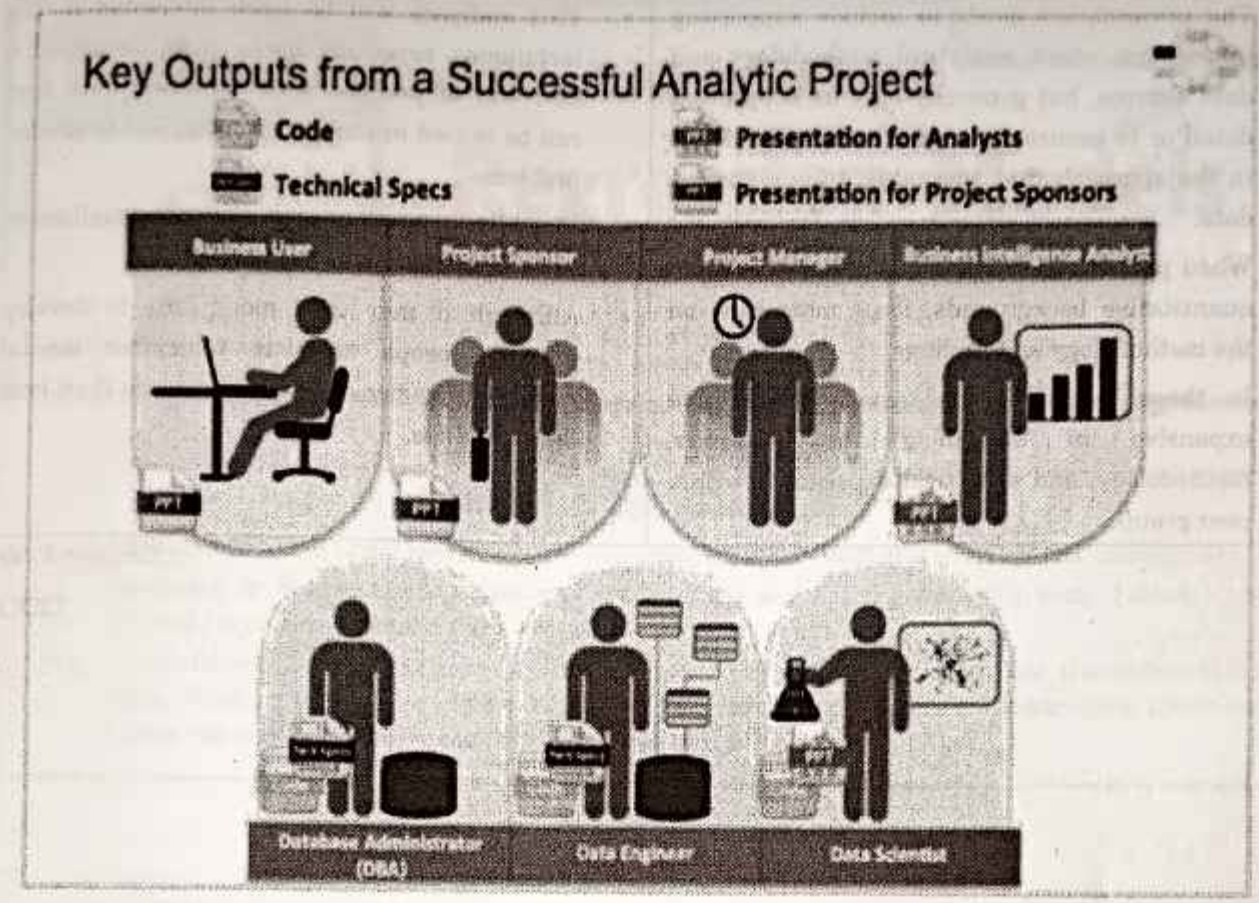


Fig 1.7.2 : Key Output from a successful Analytic Project

- Although these seven roles represent many interests within a project, these interests usually overlap, and most of them can be met with four main deliverables.
- Presentation for project sponsors: This contains high-level takeaways for executive level stakeholders, with a few key messages to aid their decision-making process.
- Focus on clean, easy visuals for the presenter to explain and for the viewer to grasp.
- Presentation for analysts, which describes business process changes and reporting changes.
- Fellow data scientists will want the details and are comfortable with technical graphs. Code for technical people. Technical specifications of implementing the code. As a general rule, the more executive the audience, the more succinct the presentation needs to be.
- Most executive sponsors attend many briefings in the course of a day or a week.
- Ensure that the presentation gets to the point quickly and frames the results in terms of value to the sponsor's organization.
- For instance, if the team is working with a bank to analyze cases of credit card fraud, highlight the frequency of fraud, the number of cases in the past month or year, and the cost or revenue impact to the bank (or focus on the reverse - how much more revenue the bank could gain if it addresses the fraud problem).
- This demonstrates the business impact better than deep dives on the methodology.

- The presentation needs to include supporting information about analytical methodology and data sources, but generally only as supporting detail or to ensure the audience has confidence in the approach that was taken to analyze the data.
- When presenting to other audiences with more quantitative backgrounds, focus more time on the methodology and findings.
- In these instances, the team can be more expansive in describing the outcomes, methodology, and analytical experiment with a peer group.
- This audience will be more interested in the techniques, especially if the team developed a new way of processing or analyzing data that can be reused in the future or applied to similar problems.
- In addition, use imagery or data visualization when possible.
- Although it may take more time to develop imagery, people tend to remember mental pictures to demonstrate a point more than long lists of bullets.

...Chapter Ends

□□□

Module 2

CHAPTER

2

Regression Models

University Prescribed Syllabus

2.1 Introduction to simple Linear Regression: The Regression Equation, Fittedvalue and Residuals, Least Square. Introduction to Multiple Linear Regression: Assessing the Model, Cross-Validation, Model Selection and Stepwise Regression, Prediction Using Regression

2.2 Logistic Regression: Logistic Response function and logit, Logistic Regression and GLM, Generalized Linear model, Predicted values from Logistic Regression, Interpreting the coefficients and odds ratios, Linear and Logistic Regression: similarities and Differences, Assessing the models.

2.1 REGRESSION

GQ. Explain regression and types of regression.

- Regression is defined as a method of estimating the value of one variable when that of the other is known and the variables are correlated.
- Regression analysis is used to predict or estimate one variable in terms of the other variable.
- It is useful in statistical estimation of demand curves, supply curves, production function, cost function etc.

2.1.1 Type of Regression

Regression is classified into two types:

(1) Simple regression and multiple regression

(2) Linear regression and nonlinear regression

1. Simple and multiple regression

- **Simple regression :** The regression analysis for studying only two variables at a time is known as simple regression.
- **Multiple regression :** The regression analysis for studying more than two variables at a time is known as multiple regression.

2. Linear and nonlinear regression

- **Linear regression :** If the regression curve is straight line, then the regression is said to be linear.
- **Nonlinear regression :** If the regression curve is not a straight line i.e. not a first-degree equation in the variables x and y, the regression is said to be nonlinear regression.

2.1.2 Lines of Regression

Definition

1. Line of regression of y on x is the line which gives the best estimate for the value of y for any specified value of x.

We have already seen that the equation of line of regression of y on x is

$$y - \bar{y} = r\left(\frac{\sigma y}{\sigma x}\right)(x - \bar{x}) \quad \ldots(i)$$

2. The line of regression of x on y is the line which gives the best estimate of x for any given value of y. And the equation of line of regression of x on y is

$$x - \bar{x} = r\left(\frac{\sigma x}{\sigma y}\right)(y - \bar{y}) \quad \ldots(ii)$$

Remarks

1. Equation (i) implies that the line of regression of y on x passes through the point $(\bar{x}, \bar{y})$.

   Similarly Equation (ii) implies that the line of regression of x on y passes through the point $(\bar{x}, \bar{y})$.

   Hence both the lines pass through the point $(\bar{x}, \bar{y})$. In other words, the mean values $(\bar{x}, \bar{y})$ can be obtained as the point of intersection of the two regression lines.

2. Why two lines of regression ?

   There are always two lines of regression on of y on x and the other of x on y.

   The line of regression of y on x is used to estimate or predict the value of y for any given value of x, i.e., when y is dependent variable and x is independent variable. The estimate so obtained will be best in the sense that it will have the minimum possible error as defined by the principle of least squares.

We can also obtain an estimate x for any given value of y by using Equation (i) but the estimate so obtained **will not be best since Equation (i) is obtained on minimising the sum of squares of errors of estimates in y and not in x.**

Hence to estimate or predict x for any given value of y, we use the regression equation of x on y i.e. equation (ii), which is derived on minimising the sum of squares of errors of estimates in x.

Here x is dependent variable and y is an independent variable.

The two regression equations are not reversible because the basis and assumptions for deriving these equations are quite different.

The regression equation of y on x is obtained or minimising the sum of square of the errors parallel to Y-axis while the regression equation of x on y is obtained on minimising the sum of squares of the errors parallel to X-axis.

3. In case of perfect correlation, i.e., $r = \pm 1$ (positive r negative), the equation of line of regression of y on x becomes

$$y - \bar{y} = \pm \frac{\sigma y}{\sigma x}(x - \bar{x})$$

$$\text{i.e. } \frac{y - \bar{y}}{\sigma y} = \pm \frac{(x - \bar{x})}{\sigma x} \quad \ldots(iii)$$

Similarly the equation of the line of regression of x on y becomes

$$x - \bar{x} = \pm \frac{\sigma x}{\sigma y}(y - \bar{y})$$

$$\text{i.e. } \frac{x - \bar{x}}{\sigma x} = \pm \frac{(y - \bar{y})}{\sigma y} \quad \ldots(iv)$$

Thus (iii) and (iv) are identical

Hence, in case of perfect correlation ($r = \pm 1$), both the lines of regression coincide and we get only one line.

2.1.3 Using Regression Lines for Prediction

The equation of the regression line is commonly used to predict the value for the dependent variable Y for a given value of the independent variable X.

For example the predicted value of Y, written as $\hat{y}_i$ when X_i given by,

$$\hat{y}_i = a + b X_i$$

Where a and b are least squares estimates given by the normal equations.

The regression equation should be used for prediction with utmost care.

Before using the lines of regression, we should test for 'goodness of fit'.

Following points to be noted while using equations of lines of regression :

(i) If the value of 'r' is significant, we can use lines of regression for estimation and prediction.

(ii) If 'r' is not significant then the linear model is not a good fit and lines of regression equations should not be used.

(iii) Even if 'r' is significant, we should not use the linear regression model to make prediction for Y corresponding to far distant values of X.

2.2 COEFFICIENT OF REGRESSION

Let us consider the line of regression of y on x :

$$y = a + bx$$

The coefficient 'b' which is the slope of the line of y on x is called 'coefficient of regression of y on x'. For convenience the slope b, i.e. coefficient of regression of y on x is written as 'by x'.

Similarly in the regression equation of x on y.

$x = A + By;$

Again the coefficient 'b' is called 'coefficient of regression of x on y. For convenience it is written as 'b_{xy}'.

Thus : Notation :

b_{yx} = coefficient of regression of y on x and

b_{xy} = coefficient of regression of x on y.

1. Now, the coefficient of regression of y on x is given by

$$b_{yx} = \frac{\text{cov}(x, y)}{\sigma_x^2} = r\frac{\sigma y}{\sigma x}$$

$[\because \text{cov}(x, y) = r\sigma x \cdot \sigma y]$ ...(i)

Similarly, the coefficient of regression of x on y is given by

$$b_{xy} = \frac{\text{cov}(x, y)}{\sigma_y^2} = r\frac{\sigma x}{\sigma y} \quad \ldots(ii)$$

Hence, the equation of line of regression of y on x is given by

$$y - \bar{y} = b_{yx}(x - \bar{x}) \quad \ldots(iii)$$

and the equation of the line of regression of x on y become

$$x - \bar{x} = b_{xy}(y - \bar{y}) \quad \ldots(iv)$$

Remarks derive regression coefficient b_{yx} and b_{yx}

1. We develop the formulae for regression coefficients b_{yx} and b_{xy} :

We have $\text{cov}(x, y) = \frac{1}{n}\sum(x - \bar{x})(y - \bar{y})$

$$= \frac{1}{n}\sum xy - \bar{x} \cdot \bar{y}$$

(on simplification)

$$= \frac{1}{n^2}[n\sum xy - (\sum x)(\sum y)]$$

and $\sigma_x^2 = \frac{1}{n}\sum(x - \bar{x})^2 = \frac{1}{n}\sum x^2 - \left(\frac{\sum x}{n}\right)^2$

$$= \frac{1}{n^2}\left[n\sum x^2 - \left(\frac{\sum x}{n}\right)^2\right]$$

$$\therefore \quad b_{yx} = \frac{\text{cov}(x, y)}{\sigma_x^2} = \frac{\sum(x - \bar{x})(y - \bar{y})}{\sum(x - \bar{x})^2}$$

$$= \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2} \quad \ldots(v)$$

Similarly,

$$\sigma_y^2 = \frac{1}{n}\sum(y - \bar{y})^2 = \frac{1}{n^2}[n\sum y^2 - (\sum x)^2]$$

$$\therefore b_{xy} = \frac{\text{cov}(x, y)}{\sigma_y^2}$$

$$\therefore b_{xy} = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum y^2 - (\sum y)^2} \quad \ldots(vi)$$

Formula (v) and (vi) can be used conveniently to find regression coefficient.

Also, we can use :

$$b_{yx} = r\frac{\sigma_y}{\sigma_x} \quad \text{and} \quad b_{xy} = r\frac{\sigma_x}{\sigma_y} \quad \ldots(vii)$$

2. Correlation coefficient between two variables x and y is a symmetrical function between x and y, i.e. $r_{yx} = r_{xy}$.

But the regression coefficients are not symmetric functions of x and y;

i.e. $b_{xy} \neq b_{yx}$

$\because b_{yx} = r\frac{\sigma_y}{\sigma_x}$ and $b_{xy} = r\frac{\sigma_x}{\sigma_y}$;

While $r_{xy} = \frac{\text{cov}(x, y)}{\sigma_x \cdot \sigma_y} = \frac{\text{cov}(y, x)}{\sigma_y \cdot \sigma_x} = r_{yx}$

3. Since $b_{yx} = \frac{\text{cov}(x, y)}{\sigma_x^2}$ and $b_{xy} = \frac{\text{cov}(x, y)}{\sigma_y^2}$

and $r_{xy} = \frac{\text{cov}(x, y)}{\sigma_x \cdot \sigma_y}$

$\because \sigma_x > 0, \ \sigma_y > 0,$

$\therefore$ 'sign' of b_{yx} and b_{xy} depend on cov (x, y)

If cov(x, y) is positive, then both b_{yx} and b_{xy} are +ve. And if cov(x, y) is negative then both b_{xy} and b_{yx} are negative.

Thus the sign of correlation coefficient is same as that of regression coefficients. If regression coefficients are positive, r is positive and if regression coefficients are negative r is negative.

Theorems on Regression coefficients

Theorem 1 : The correlation coefficient is the geometric mean between the regression coefficients, i.e.

$$r^2 = b_{yx} \cdot b_{xy}$$

Proof : We have

$$b_{yx} = \frac{\text{cov}(x, y)}{\sigma_x^2}; \ b_{xy} = \frac{\text{cov}(x, y)}{\sigma_y^2} = r\frac{\sigma x}{\sigma y} = r\frac{\sigma x}{\sigma y}$$

$$\therefore b_{yx} \cdot b_{xy} = \left(r\frac{\sigma y}{\sigma x}\right) \cdot \left(r\frac{\sigma x}{\sigma y}\right) = r^2$$

$$\therefore r = \pm\sqrt{byx \cdot bxy};$$

Hence the result

Remark

If regression coefficient are positive, we take r +ve and if regression coefficients are negative, we take r – ve.

Theorem 2 : The arithmetic mean of the modulus value of the regression coefficients is greater than the modulus value of the correlation coefficients.

Proof : We recall the result : If a and b are any two distinct positive real numbers them

i.e. $\frac{a+b}{2} > \sqrt{ab}$

$\therefore \frac{1}{2}[|b_{xy}| + |b_{yx}|] > \sqrt{|b_{xy}| + |b_{yx}|}$

$\therefore \frac{1}{2}[|b_{xy}| + |b_{yx}|] > \sqrt{|b_{xy}| \cdot |b_{yx}|} = |r|$

$\therefore \frac{1}{2}[|b_{xy}| + |b_{yx}|] > |r|$

Theorem 3 : If one of the regression coefficents is greater than unity (one), the other must be less than unity.

Proof :

since $r^2 = b_{yx} \cdot b_{xy}$

and $0 \le r^2 \le 1$. $\therefore b_{yx} \cdot b_{xy} \le 1$

If $b_{yx} > 1$, then $\frac{1}{byx} < 1$ ...(i)

$\because$ Now, $b_{xy} \le \frac{1}{byx} < 1$ ...from (i)

Theorem 4 : Regression coefficients are independent of change of origin but not of scale.

Proof

Let $u = \frac{x-a}{h}$, $V = \frac{y-b}{k}$;

Where a, b, h (> 0) and k (> 0) are constants.

Since the correlation coefficient is independent of change of origin and scale, we have

$r_{xy} = r_{uv}$ ...(i)

Also, $\sigma_x = h\,\sigma_u$, $\sigma_y = k\,\sigma_v$ ...(ii)

Since standard deviation is independent of change of origin but not of scale

$\therefore b_{xy} = r_{xy}\frac{\sigma x}{\sigma y} = r_{uv}\frac{h\,\sigma_u}{k\,\sigma_v} = \frac{h}{k}\left[r_{uv}\frac{\sigma u}{\sigma v}\right]$

$= \frac{h}{k} b_{uv}$ ...(iii)

$\therefore b_{yx} = r_{yx}\frac{\sigma y}{\sigma x}$

$= r_{vu}\frac{k\,\sigma_v}{h\,\sigma_u} = \frac{k}{h} r_{vu}\frac{\sigma_v}{\sigma_u} = \frac{k}{h} b_{vu}$ ...(iv)

$\therefore b_{xy} = \frac{h}{k} b_{uv}$ and

$b_{yx} = \frac{k}{h} b_{vu}$

Ex. 2.2.1 : The regression lines of a sample are $x + 6y = 6$ and $3x + 2y = 10$. Find (i) sample means $\bar{x}$ and $\bar{y}$ (ii) the coefficient of correlation between x and y (iii) Also, find the value of y at x = 12.

☑ **Soln. :**

(i) The regression lines pass through the point $(\bar{x}, \bar{y})$.

So, the regression lines of as sample are

$\bar{x} + 6\bar{y} = 6$

$3\bar{x} + 2\bar{y} = 10$

To solve the above equations, we get $\bar{x} = 3, \bar{y} = \frac{1}{2}$.

(ii) Consider the line $x + 6y = 6$ be the regression line of y on x. So,

$y = -\frac{1}{6}x + 1$

Compare with general form of regression line of y on x.

$b_{yx} = -\frac{1}{6}$

Again, consider the line $3x + 2y = 10$ be the regression line of x on y. So,

$x = -\frac{2}{3}y + \frac{10}{3}$

Compare with general form of regression line of x on y,

$b_{xy} = -\frac{2}{3}$

Thus,

$$r = \sqrt{b_{yx}\, b_{xy}} = \sqrt{\left(-\frac{1}{6}\right)\left(-\frac{2}{3}\right)} = \frac{1}{3}$$

Here, both b_{yx} and b_{xy} are negative. So, r is also negative.

Therefore, the coefficient of correlation is $r = -\frac{1}{3}$.

(iii) From Equation (ii), At x = 12,

$$y = -\frac{1}{6}x + 1$$

$$\therefore y = -\frac{1}{6}(12) + 1$$

$$\therefore y = -1$$

Ex. 2.2.2 : From the following results, obtain the two regression equations and estimate the yield when the rainfall is 29 cm and the rainfall, when the yield is 600 kg :

	Yield in kg	Rainfall in cm
Mean	508.4	26.7
SD	36.8	4.6

The coefficient of correlation between yield and rainfall is 0.52.

☑ **Soln. :**

Let x be the rainfall in cm and y be the yield in kg. Here,

$\bar{x} = 26.7$, $\sigma_x = 4.6$, $\bar{y} = 508.4$, $\sigma_y = 36.8$ and $r = 0.52$

The regression coefficients are

$$b_{yx} = r\frac{\sigma_y}{\sigma_x} = 0.52\frac{36.8}{4.6} = 4.16$$

$$b_{xy} = r\frac{\sigma_x}{\sigma_y} = 0.52\frac{4.6}{36.8} = 0.065$$

Now, the regression line of y on x is

$$y - \bar{y} = b_{yx}(x - \bar{x})$$

$$y - 508.4 = 4.16(x - 26.7)$$

$$\therefore y = 4.16x + 397.328$$

And the regression line of x on y is

$$x - \bar{x} = b_{xy}(y - \bar{y})$$

$$x - 26.7 = 0.065(y - 508.4)$$

$$\therefore x = 0.065y - 6.346$$

When the rainfall x is 29 cm, estimated yield y is

$$y = 4.16(29) + 397.328 = 517.968 \text{ kg}$$

When the yield y is 600 kg, estimated rainfall x is

$$x = 0.065(600) - 6.346 = 32.654 \text{ cm}$$

Ex. 2.2.3 : The following data give the experience of machine operators and their performance rating as given by the number of good parts turned out per 100 pieces.

Operators	1	2	3	4	5	6
Performance rating (x)	23	43	53	63	73	83
Experience (y)	5	6	7	8	9	10

Calculate the regression line of performance rating on experience and also estimate the probable performance if an operator has 11 years of experience.

☑ **Soln. :**

Here, n = 6

x	y	y^2	xy
23	5	25	115
43	6	36	258
53	7	49	371
63	8	64	504
73	9	81	657
83	10	100	830
$\sum x = 338$	$\sum y = 45$	$\sum y^2 = 355$	$\sum xy = 2735$

The regression coefficient of x on y is

$$b_{xy} = \frac{n\sum xy - \sum x \sum y}{n\sum y^2 - (\sum y)^2} = 11.429$$

Here, $\bar{x} = \frac{\sum x}{n} = 56.33$ and $\bar{y} = \frac{\sum y}{n} = 7.5$

So, the equation of regression line of x on y is

$$x - \bar{x} = b_{xy}(y - \bar{y})$$

$$x - 56.33 = 11.429(y - 7.5)$$

$$\therefore x = 11.429y - 29.3875$$

When the experience is 11 years of an operator, estimated performance is x = 96.33

UEx. 2.2.4 (Ref. - Q. 2(c), W-19, 7 Marks)

The number of bacterial cells (y) per unit volume in a culture at different hours (x) is given below :

x	0	1	2	3	4	5	6	7	8	9
y	43	46	82	98	123	167	199	213	245	272

Fit lines of regression of y on x and x on y. Also, estimate the number of bacterial cells after 15 hours.

☑ **Soln. :**

Here, n = 10

x	y	x^2	xy	y^2
0	43	0	0	1849
1	46	1	46	2116
2	82	4	164	6724
3	98	9	294	9604
4	123	16	492	15129
5	167	25	835	27889
6	199	36	1194	39601
7	213	49	1491	45369
8	245	64	1960	60025
9	272	81	2448	73984
$\Sigma x = 45$	$\Sigma y = 1488$	$\Sigma x^2 = 285$	$\Sigma xy = 8924$	$\Sigma y^2 = 282290$

Here, $\bar{x} = \frac{\Sigma x}{n} = 4.5$ and $\bar{y} = \frac{\Sigma y}{n} = 148.8$

The regression coefficients are

$$b_{xy} = \frac{n\Sigma xy - \Sigma x \Sigma y}{n\Sigma y^2 - (\Sigma y)^2} = 0.0366$$

and

$$b_{yx} = \frac{n\Sigma xy - \Sigma x \Sigma y}{n\Sigma x^2 - (\Sigma x)^2} = 27.0061$$

The regression line of y on x is

$$y - \bar{y} = b_{yx}(x - \bar{x})$$

$$y - 148.8 = 27.0061(x - 4.5)$$

$$\therefore y = 27.0061x + 27.2726$$

The regression line of x on y is

$$x - \bar{x} = b_{xy}(y - \bar{y})$$

$$x - 4.5 = 0.0366(y - 148.8)$$

$$\therefore x = 0.366y - 0.9461$$

Thus, at x = 15 hours, y = 432.3641

UEx. 2.2.5 (Ref. - Q. 3(c), W-19, 7 Marks)

Find the two lines of regression from the following data:

Age of Husband (x)	25	22	28	26	35	20	22	40	20	18
Age of wife (y)	18	15	20	17	22	14	16	21	15	14

Hence, estimate (i) the age of the husband when the age of the wife is 19, and (ii) the age of the wife when the age of the husband is 30.

☑ **Soln. :** Let a = 26 and b = 17 be the assumed means of x and y series respectively.

$d_x = x - a = x - 26$ and $d_y = y - b = y - 17$

Here, n = 10

x	y	d_x	d_y	d_x^2	d_y^2	$d_x d_y$
25	18	−1	1	1	1	−1
22	15	−4	−2	16	4	8
28	0	2	3	4	9	6
26	17	0	0	0	0	0
35	22	9	5	81	25	45
20	14	−6	−3	36	9	18
22	16	−4	−1	16	1	4
40	21	14	4	196	16	56
20	15	−6	−2	36	4	12
18	14	−8	−3	64	9	24
$\Sigma x = 256$	$\Sigma y = 172$	$\Sigma d_x = -4$	$\Sigma d_y = 2$	$\Sigma d_x^2 = 450$	$\Sigma d_y^2 = 78$	$\Sigma d_x d_y = 172$

Means of x and y are

$$\bar{x} = \frac{\Sigma x}{n} = 25.6 \text{ and } \bar{y} = \frac{\Sigma y}{n} = 17.2$$

The regression coefficients are

$$b_{xy} = \frac{n\Sigma d_x d_y - \Sigma d_x \Sigma d_y}{n\Sigma d_y^2 - (\Sigma d_y)^2} = 2.227$$

and

$$b_{yx} = \frac{n\Sigma d_x d_y - \Sigma d_x \Sigma d_y}{n\Sigma d_x^2 - (\Sigma d_x)^2} = 0.385$$

The regression line of y on x is

$$y - \bar{y} = b_{yx}(x - \bar{x})$$

$$y - 17.2 = 0.385(x - 25.6)$$

$$\therefore y = 0.385x + 7.344$$

The regression line of x on y is,

$$x - \bar{x} = b_{xy}(y - \bar{y})$$

$$x - 25.6 = 2.227(y - 17.2)$$

$$\therefore x = 2.227\,y - 12.704$$

When the age of the wife is 19, estimated age of the husband is

$$x = 2.227\,(19) - 12.704 = 29.601 \approx 30$$

So, Age of the husband is 30 years

When the age of husband is 30, estimated age of the wife is

$$y = 0.385(30) + 7.344 = 18.894 \approx 19$$

So, Age of the wife is 19 years.

2.2.1 Solved Examples on Coefficients of Regression

Ex. 2.2.6 : From the following data, obtain the two regression equations :

Sales :	91	97	108	121	67	124	51	73	111	57
Purchases :	71	75	69	97	70	91	39	61	80	47

Soln. :

Let us denote the sales by the variable x and the purchases by the variable y :

x	y	$u = x - \bar{x}$	$v = y - \bar{y}$	u^2	v^2	uv
91	71	1	1	1	1	1
97	75	7	5	49	25	35
108	69	18	−1	324	1	−18
121	97	31	27	961	729	837
67	70	−23	0	529	0	0
124	91	34	21	1156	441	714
51	39	−39	−31	1521	961	1209
73	61	−17	−9	289	81	153
111	80	21	10	441	100	210
57	47	−33	−23	1089	529	759

We have $\sum x = 900$, $\sum y = 700$, $\sum u = 0$, $\sum v = 0$,

$\sum u^2 = 6360$, $\sum v^2 = 2868$, $\sum uv = 3900$

We have, $\bar{x} = \frac{\sum x}{n} = \frac{900}{10} = 90$,

$$\bar{y} = \frac{\sum y}{n} = \frac{700}{10} = 70,$$

$$\text{Now, } b_{yx} = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sum (x-\bar{x})^2} = \frac{\sum u \cdot v}{\sum u^2}$$

$$b_{yx} = \frac{3900}{6360} = 0.6132$$

$$b_{xy} = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sum (y-\bar{y})^2} = \frac{\sum u \cdot v}{\sum v^2}$$

$$b_{xy} = \frac{3900}{2868} = 1.361$$

Now, Regression equations

1. Equation of line of regression of y on x is

$$y - \bar{y} = b_{yx}(x - \bar{x})$$

$$\therefore y - 70 = 0.6132\,(x - 90)$$

$$= 0.6132x - 55.188$$

$$\therefore y = 0.6132\,x + 14.812$$

2. Equation of line of Regression of x on y is :

$$x - \bar{x} = b_{xy}(y - \bar{y})$$

$$\therefore x - 90 = 1.361\,(y - 70)$$

$$= 1.361\,y - 95.27$$

$$\therefore x = 1.361\,y - 5.27$$

3. Again, $r^2 = b_{yx} \cdot b_{xy} = (0.6132)\,(1.361) = 0.8346$

$$\therefore r = \pm\sqrt{0.8346} = \pm 0.9135$$

$\because b_{yx}$ and b_{xy} both are positive

$$\therefore r = 0.9135 \quad \text{...Ans.}$$

Ex. 2.2.7 : From the data, given below :

Marks in Economics	25	28	35	32	31	36	29	38	34	32
Marks in statistics	43	46	49	41	36	32	31	30	33	39

Find : (a) The two regression coefficients

(b) two regression equation

(c) coefficients of correlation between marks in Economics states

(d) The most likely marks in statistics when marks in economics are 30.

☑ **Soln. :**

Let marks in Economics be denoted by the variable x and in statistics by y.

x	y	u	v	u^2	v^2	uv
25	43	−7	5	49	25	−35
28	46	−4	8	16	64	−32
35	49	3	11	9	121	33
32	41	0	3	0	9	0
31	36	−1	−2	1	4	2
36	32	4	−6	16	36	−24
29	31	−3	−7	9	49	21
38	30	6	−8	36	64	−48
34	33	2	−5	4	25	−10
32	39	0	1	0	1	0

We have

$$\sum x = 320, \quad \sum y = 380,$$

Also, $u = x - \bar{x}$, $\quad v = y - \bar{y}$

$$\text{and} \quad \bar{x} = \frac{\sum x}{n} = \frac{320}{10} = 32;$$

$$\bar{y} = \frac{\sum y}{n} = \frac{380}{10} = 38$$

$$\therefore u = x - 32, \quad v = y - 38$$

and, $\sum u^2 = 140, \quad \sum v^2 = 398, \quad \sum vu = -93$

(a) Regression coefficients

Coefficient of Regression of y on x

$$\frac{\sum (x-\bar{x})(y-\bar{y})}{\sum (x-\bar{x})^2} = \frac{\sum u \cdot v}{\sum u^2} = \frac{-93}{140} = -0.6643$$

Coefficients of regression of x on y

$$= \frac{\sum (x-\bar{x})(y-\bar{y})}{\sum (y-\bar{y})^2} = \frac{\sum uv}{\sum v^2} = \frac{-93}{398} = -0.2337$$

(b) Regression Equations

1. Equation of line of regression of y on x :

$$y - \bar{y} = b_{yx}(x - \bar{x})$$

$$\therefore \quad y - 38 = -0.6643\,(x - 32)$$

$$\therefore \quad y = -0.6643x + 0.6643 \times 32 + 38$$

$$= -0.6643x + 59.2576$$

2. Equation of line regression of x on y :

$$x - \bar{x} = b_{xy}(y - \bar{y})$$

$$\therefore \quad x - 32 = -0.2337\,(y - 38)$$

$$= -0.2337\,y + 0.2337\,(38)$$

$$\therefore \quad x = -0.2337y + 32 + 0.2337 \times (38)$$

$$\therefore \quad x = -0.2337y + 40.8806$$

(c) Correlation coefficient

We have

$$r^2 = b_{yx} \cdot b_{xy} = (-0.643)(-0.2377) = 0.1552$$

$$= r = \pm\sqrt{0.1552} = \pm 0.394$$

Since, both the regression coefficient are negative r must be negative

$$\therefore \quad r = -0.394$$

(d) To estimate the most likely marks in statistics (y) when marks in Economics (x) are 30; we use the line of regression of y on x,

i.e. $y = 0.6643\,x + 59.2576$

When $x = 30$; $\quad y = -0.6643\,(30) + 59.2576$

$$\therefore y = 39.3286$$

∴ Most likely marks in statistics are 39.

Ex. 2.2.8 : A panel of judges A and B graded seven debators and independently awarded the following marks

Debator	1	2	3	4	5	6	7
Marks by A	40	34	28	30	44	38	31
Marks by B	32	19	26	30	38	34	28

An eight debator was awarded 36 marks by Judge A while judge b was not present.

If judge B was also present, how many marks would you expect him to award to eight debator assuming some degree of relationship exists in judgement ?

☑ **Soln. :**

Let the marks awarded by judge 'A' be denoted by the variable X and the marks awarded by judge 'B' be the variable Y.

Debator	x	y	$u = x - \bar{x}$	$v = y - \bar{y}$	u^2	v^2	uv
1	40	32	5	2	25	4	10
2	34	39	−1	9	1	81	−9
3	28	26	−7	−4	49	16	28
4	30	30	−5	0	25	0	0
5	44	38	9	8	81	64	72
6	38	34	3	4	9	16	12
7	31	28	−4	−2	16	4	8
Total			$\sum u = 0$	$\sum v = 17$	206	185	121

$\sum u = 0$, $\sum v = 17$, $\sum u^2 = 206$, $\sum v^2 = 185$, $\sum uv = 121$

To find the marks obtained by eighth debator, we use the equation of line of regression of y on x

We have $\bar{x} = 35$, $\bar{y} = 30 + \frac{17}{7} = 32.4286$

$$\text{and } b_{yx} = b_{vu} = \frac{n\sum uv - (\sum u)(\sum v)}{n\sum u^2 - (\sum u)^2}$$

$$= \frac{7 \times 121 - 0 \times 17}{7 \times 206} = \frac{121}{206} = 0.5874$$

∴ Equation of line of regression of y on x is

$$y - \bar{y} = b_{yx}(x - \bar{x})$$

$$\therefore \quad y - 32.4286 = 0.5874\,(x - 35)$$

$$\therefore \quad y = 0.5874\,x - 0.5874\,(35) + 32.4286$$

$$= 0.5874x + 11.8696$$

When $x = 36$,

$$y = 0.5874\,(36) + 11.8696 = 33.016$$

∴ Judge B would have given 33 marks to the eight debator. **...Ans.**

Ex. 2.2.9 : A departmental store gives in-service training, to its salesman which is followed by a test. It is considering whether it should terminate the service of any salesman who does not do well in the test. The following data give the test scores and sales made by nine salesmen during a certain period

Test scores	14	19	24	21	26	22	15	20	19
Sales (1000 Rs.)	31	36	48	37	50	45	33	41	39

Calculate the coefficient of correlation between the test scores and the sales. Does it indicate that the termination of services of low test scores is justified ? It the firm wants a minimum sales volume of Rs. 30,000, what is the minimum test score that will ensure continuation of service ? Also estimate the most probable sales volume of a salesman making a score of 28.

☑ **Soln. :**

Let x denote the test scores of the salesmen and y denote their corresponding sales (in ′ 000). Rs.

To calculate regression lines, we prepare the table

Let $u = x - \bar{x} = x - 20$, $v = y - \bar{y} = y - 40$

x	y	$u = x - \bar{x} = x - 20$	$v = y - \bar{y} = y - 40$	u^2	v^2	uv
14	31	−6	−9	36	81	54
19	36	−1	−4	1	16	04
24	48	4	8	16	64	32
21	37	1	−3	1	9	−03
26	50	6	10	16	100	60
22	45	2	5	4	25	10
15	33	−5	−7	25	49	35
20	41	0	1	0	1	00
19	39	−1	−1	1	1	01
Total 180	360	0	0	120	346	193

We have

$$\bar{x} = \frac{180}{9} = 20, \quad \bar{y} = \frac{360}{9} = 40,$$

$$\sum u^2 = 120, \quad \sum v^2 = 346, \sum uv = 193$$

Now, 1. coefficient of regression of y on x

$$b_{yx} = \frac{\sum u \cdot v}{\sum u^2} = \frac{193}{120} = 1.6083$$

2. and $b_{xy} = \frac{\sum u \cdot v}{\sum v^2} = \frac{193}{346} = 0.5578$

3. Now, $r^2 = b_{yx} \cdot b_{xy} = 1.6083 \times 0.5578 = 0.8971$

$$\therefore \quad r = \sqrt{0.8971} = 0.9471$$

∵ correlation coefficient are positive, ∴ $r = 0.9471$

To find Regression Equations

1. To obtain the test score (x) for given sales (y), we use the equation of the line of regression of x on y

$$\text{i.e., } x-\bar{x} = b_{x/y}\left(y-\bar{y}\right)$$

$$\therefore x-20 = 0.5578\,(y-40) = 0.5578\,y - 0.5578\times 40$$

$$\therefore x = 0.5578y - 0.5578\times 40 + 20$$

$$= 0.5578y - 2.312$$

To ensure the continuation of service, the minimum test-score (x) corresponding to a minimum sales volume (y) of Rs. 30 and is given by

$$x = 0.5578\,(30) - 2.312 = 14.422 = 14$$

2. To estimate the sales volume (y) of a salesman with given test score (x), we use the line of regression of y on x :

$$y-\bar{y} = b_{yx}\left(x-\bar{x}\right)$$

$$\therefore y-40 = 1.6083\,(x-20)$$

$$\therefore y = 1.6083\,x - 1.6083\,(20) + 40$$

$$= 1.6083\,x + 7.8340$$

Hence the estimated sales volume of a salesman with test score of 28 is

$$y = 1.6083\,(28) + 7.8340 = 52.866$$

$$\text{i.e., } y = 52{,}866$$

Ex. 2.2.10 : The adjoining table shows the number of motor registrations in a certain territory for a term of 5 years and the sale of motor tyres by a firm in that territory for the same period

Find the regression equation to estimate the sale of tyres when motor registration is known.

Estimate sale of tyres when registration is 850.

Year	Motor Registration	No of tyres sold
1	600	1,250
2	630	1,100
3	720	1,300
4	750	1,350
5	800	1,500

Soln. :

We denote the number of registration by the random variable x and the number of tyres sold by y.

Now to find regression equation of y on x :

Hence we change the origin as well as scale to both the variables x and y.

$$\text{Let } u = \frac{x-A}{h} = \frac{x-720}{10}$$

$$v = \frac{y-B}{k} = \frac{y-1350}{50}$$

We prepare the table :

x	y	u	v	u2	uv
600	1250	−12	−2	144	24
630	1100	−9	−5	81	45
720	1300	0	−1	0	0
750	1350	3	0	9	0
800	1500	8	3	64	24

$$\therefore \sum u = -10, \quad \sum v = -5$$

$$\sum u^2 = 298, \ \sum uv = 93$$

$$\text{Now, } \bar{x} = A + h\left(\frac{\sum u}{n}\right) = 720 + 10\left(\frac{-10}{5}\right) = 700$$

$$\bar{y} = B + k\left(\frac{\sum v}{n}\right)$$

$$= 1350 + 50\left(\frac{-5}{5}\right) = 1300 \text{ And.}$$

$$b_{yx} = \frac{k}{h}\,(b_{vu}) = \frac{k}{h}\,\frac{n\sum uv - (\sum u)(\sum v)}{n\sum u^2 - (\sum u)^2}$$

$$= \frac{50}{10}\left[\frac{5\times 93 - (-10)(-5)}{5\times 298 - (-10)^2}\right] = 5\left[\frac{465}{1490} - \frac{50}{100}\right] = 1.4928$$

$\therefore$ Equation of line of regression of sale of tyres (y) on the motor registration is :

$$y-\bar{y} = b_{yx}\left(x-\bar{x}\right)$$

$$\therefore y - 1300 = 1.4928\,(x-700)$$

$$= 1.4928\,x - (1.4928)\,(700)$$

$$\therefore y = 1.4928\,x - (1.4928)\,(700) + 1300$$

$$\therefore y = 1.4928\,x + 255.04$$

When $x = 850$;

$$y = 1.4928\,(850) + 255.04$$

$$= 1523.92 = \mathbf{1524} \quad \text{...Ans.}$$

is the estimate of sale of tyres

Ex. 2.2.11 : The data about the sales and advertisement expenditure of a firm is given below :

	Sales (in crores of Rs.)	Advt. expenditure (in crores of Rs.)
Means	40	6
Standard deviations	10	1.5

Coefficient of correlation = r = 0.9

(i) estimate the likely sales for a proposed advertisement expenditure of Rs. 10 corers.

(ii) What should be the advt. expenditure if the firm propose a sales target of 60 crores of rupees ?

☑ **Soln. :**

Let x denote the sales (in crores of Rs.) and the variable y denote the advertisement expenditure (in crores of Rs.). Then from the given data,

$$\bar{x} = 40, \quad \sigma_x = 10;$$

$$\bar{y} = 6, \quad \sigma_y = 1.5,$$

$$r_{xy} = r = 0.9$$

(i) To estimate the likely sales (x) for a proposed advt. Expenditure (y), we write the regression equation of x on y :

$$x - \bar{x} = r\frac{\sigma x}{\sigma y}\left(y - \bar{y}\right)$$

$$x - 40 = (0.9)\left(\frac{10}{1.5}\right)(y - 6)$$

when $y = 10$;

$$\therefore x - 40 = (0.9)\left(\frac{10}{1.5}\right)(10 - 6) + 40$$

$$= 6 \times 4 + 40 = 64 \text{ crores of Rs.}$$

(ii) To estimate The advt. expenditure (y) for proposed sales (x), we need the equation of line of regression of y on x which is given by

$$y - \bar{y} = r\frac{\sigma y}{\sigma y}\left(x - \bar{x}\right)$$

$$\therefore y - 6 = (0.9)\left(\frac{1.5}{10}\right)(x - 40)$$

$$\therefore y = 0.135\,(x - 40) + 6$$

∴ likely advt. expenditure (y) for the proposed sale target (x) of 60 crores is

$$y = 0.135\,(60 - 40) + 6$$

$$y = 2.7 + 6 = 8.7 \text{ crores of Rs.} \ldots \textbf{Ans.}$$

Ex. 2.2.12 : Point out the inconsistency in the following statement :

The regression equation of y on x is 2y + 3x = 4 and the correlation coefficient between x and y is 0.8.

☑ **Soln. :**

The regression line of y on x is

$$2y + 3x = 4$$

$$\therefore y = -\frac{3}{2}x + 2$$

$\therefore$ by $x = -\frac{3}{2}$ = coefficient of correlation of y on x

Also, r = 0.8 (given)

$\because$ b_{yx} and r have different signs, the given statement is inconsistent.

Ex. 2.2.13 : The following is an estimated supply regression for sugar : $Y = 0.025 + 1.5\,X$

Where y is supply in kilos and X is price (Rs.) per kilo.

(i) Interpret the coefficient of variable X.

(ii) Predict the supply when the price is Rs. 20 per kilo,

(iii) Given that r (x, y) = 1 in the above case, interpret the implied relationship between price and quantity supplied.

☑ **Soln. :**

The regression equation of Y (supply in kgs) on X (price in Rupees per kg) is given to be

$$Y = 0.025 + 1.5\,X = a + bX \text{ (say)}$$

(i) The coefficient of the variable X is b = 1.5 is the coefficient of regression of Y on X.

It reflects the unit change in the value of Y, for a unit change in the corresponding value of X.

This implies that if the price of the sugar goes up by Re. 1 per kg; the estimated supply of sugar goes up by 1.5 kg.

(ii) From (i), the estimated supply of sugar when its price is Rs. 20 per kg. is given by

$$Y = 0.025 + 1.5\,(20) = 30.025 \text{ kg}$$

(iii) $\because$ r (X, Y) = 1, implies that the relationship between X and Y is exactly linear. This means that all the observed values (X, Y) lie on a straight line. ...Ans.

Ex. 2.2.14

(a) On each of 30 items, two measurements are made. The following summations are given :

$\sum X = 15$, $\sum Y = 6$, $\sum XY = 56$, $\sum X^2 = 61$ and $\sum Y^2 = 90$.

(b) Calculate the product moment correlation coefficient and the slope of the regression line of Y on X.

(c) How would your results be affected if X is replaced by $u = \frac{X-1}{2}$.

☑ **Soln. :**

(a) The product moment correlation coefficient

R = r (x, y) is,

$$r = \frac{n\sum XY - (\sum X)(\sum Y)}{n\sum X^2 - (\sum X)^2\left[n\sum Y^2 - (\sum Y)^2\right]}$$

$$= 0\,\frac{1680 + 90}{\sqrt{(1830 - 225)\,(2700 - 36)}}$$

$$= \frac{1770}{\sqrt{(1605)\,(2664)}} = 0.856$$

(b) Coefficient of correlation of Y on X

$$= \frac{n\sum XY - (\sum X)(\sum Y)}{n\sum X^2 - (\sum X)^2}$$

$$= \frac{1680 + 90}{1605} = \frac{1770}{1605} = 1.1028$$

Slope of line of regression of Y on X is 1.1028.

(c) Now, $u = \frac{X-1}{2}$

Since r (X, Y) is independent of change of origin and scale,

$\therefore$ r(X, Y) = r (u, Y) = 0.856

But the regression coefficient is independent of change of origin but not of scale. ...Ans.

$$b_{YX} = \frac{1}{2}\, b_{Yu}$$

$$\therefore\ b_{yu} = 2\, b_{YX}$$

$$= 2\,(1.1028) = 2.2056 \quad \text{...Ans.}$$

Ex. 2.2.15 : By using the following data, find out the two lines of regression and from them compute coefficient of correlation.

$\sum X = 250$, $\sum Y = 300$, $\sum XY = 7900$, $\sum X^2 = 6500$ and $\sum Y^2 = 10{,}000$, and N = 10.

☑ **Soln. :**

We have $\bar{X} = \frac{\sum X}{N} = \frac{250}{10} = 25$,

and $\bar{Y} = \frac{\sum Y}{N} = \frac{300}{10} = 30$

Now, b_{yx} = coefficient of regression of Y on X

$$= \frac{N\sum XY - (\sum X)(\sum Y)}{N\sum X^2 - (\sum X)^2}$$

$$= \frac{10 \times 7900 - (250)\,(300)}{10(6500) - (250)^2}$$

$$= \frac{7900 - 75000}{65000 - 62500} = \frac{4000}{2500} = 1.6$$

and b_{xy} = coefficient regression of X on Y

$$= \frac{N\sum XY - (\sum X)(\sum Y)}{N\sum Y^2 - (\sum Y)^2}$$

$$= \frac{10 \times 7900 - (250)\,(300)}{10(1000) - (300)^2} = \frac{4000}{10000} = 0.4$$

$\therefore$ Correlation coefficient r_{XY} between X and Y is given by

$$r_{XY}^2 = b_{yx} \cdot b_{xy} = 1.6 \times 0.4 = 0.64$$

$\therefore r_{XY} = \pm\sqrt{0.64} = \pm 0.8$

Since the regression coefficients are positive,

$\therefore r_{XY} = 0.8$

Regression Equations

1. Regression equation of Y on X is

$$Y - \bar{Y} = b_{yx}(X - \bar{X})$$

$$\therefore Y - 30 = 1.6(X - 25)$$

$$\therefore Y = 1.6X - 40 + 30$$

$$\therefore Y = 1.6 \times -10$$

2. Regression equation of X on Y is :

$$X - \bar{X} = b_{YX}(Y - \bar{Y})$$

$$\therefore Y - 25 = 0.4(Y - 30)$$

$$\therefore X = 0.4Y + 25 - 12$$

$$\therefore \mathbf{X = 0.4Y + 13}$$ **...Ans.**

Ex. 2.2.16 : In the estimation of regression equations of two variables X and Y, the following results were obtained :

$\sum X = 900$, $\sum Y = 700$, $\sum XY = 10$;

and $\sum X^2 = 6360$ and $\sum Y^2 = 2860$, $= \sum XY = 3900$

where x and y are deviations from respective means. Obtain the two regression equations.

☑ **Soln. :**

The coefficients of regression of Y on X and X on Y are given by :

$$b_{YX} = \frac{\text{cov}(x, y)}{\sigma_x^2} = ; \frac{\sum(x - \bar{x})(y - \bar{y})}{\sum(x - \bar{x})^2}$$

$$= \frac{\sum xy}{\sum x^2} = \frac{3900}{6360} = 0.6132$$

and $$b_{XY} = \frac{\text{cov}(x, y)}{\sigma_Y^2}$$

$$= \frac{\sum xy}{\sum y^2} = \frac{3900}{2860} = 1.3636$$

$$\bar{X} = \frac{\sum X}{n} = \frac{900}{10} = 90;$$

$$\bar{Y} = \frac{\sum Y}{n} = \frac{700}{10} = 70;$$

(1) Regression equations of Y on X

$$Y - \bar{Y} = b_{yx}(X - \bar{X})$$

$$\therefore Y - 70 = 0.6132(X - 90)$$

$$\therefore Y = 0.6132X - 0.6132(90) + 70$$

$$\therefore Y = (0.6132X) + 14.812$$

(2) Regression Equation of X on Y

$$X - \bar{X} = b_{XY}(Y - \bar{Y})$$

$$\therefore Y - 90 = .1.3636(Y - 70)$$

$$\therefore X = 1.3636Y - 1.3636 \times 70 + 90$$

$$\therefore X = 1.3636Y - 5.452$$ **...Ans.**

Ex. 2.2.17 : Following information regarding a distribution is given :

$n = 5, \bar{X} = 0, \bar{Y} = 20, \sum(X - 4)^2 = 100,$

$\sum(Y - 10)^2 = 160, \sum(X - 4)(Y - 10) = 80$

Find the two regression coefficients and hence the coefficient of correlation.

☑ **Soln. :**

Let $u = X - 4$, $v = Y - 10$, then we have

$n = 5, \bar{X} = 10, \bar{Y} = 20, \sum(X - 4)^2 = \sum u^2 = 100,$

$\sum(Y - 10)^2 = \sum v^2 = 160, \sum(X - 4)(Y - 10) = \sum uv = 80$

Also, $u = X - 4$, $\therefore \bar{u} = \bar{X} - 4 = 10 - 4 = 6$

$\sum u = n\bar{u} = 5 \times 6 = 30,$

$V = Y - 10$, $\therefore \bar{V} = \bar{Y} - 10 = 20 - 10 = 10,$

$\therefore \sum V = n\bar{V} = 5 \times 10 = 50,$

Since, the regression coefficients are independent of the change of origin, the regression coefficient are given by,

$$b_{YX} = b_{vu} = \frac{n\sum uv - (\sum u)(\sum v)}{n\sum u^2 - (\sum u)^2}$$

$$= \frac{5(80)-(30)(50)}{5(100)-(30)^2} = \frac{-1100}{-400} = \frac{11}{4}$$

$$\text{and } b_{XY} = b_{uv}\frac{n\sum uv-(\sum u)(\sum v)}{n\sum v^2-(\sum v)^2}$$

$$= \frac{5(80)-(30)(50)}{5(160)-(50)^2} = \frac{-1100}{-700} = \frac{11}{17}$$

$\therefore$ correlation coefficients $= \pm\sqrt{b_{YX}\cdot b_{XY}}$

$$\therefore r = \pm\sqrt{\frac{11}{4}\cdot\frac{11}{7}} = \pm 1.33$$

$\because$ regression coefficients are +ve

$\therefore r = 1.33 > 1$, which is impossible,

$\because |r| \leq 1 \therefore$ Given data is inconsistent

2.3 MULTIPLE LINEAR REGRESSION MODEL

GQ. Explain multiple linear regression in k-variables.

- Multiple Linear Regression (MLR), also known s simply 'Multiple Regression'. It is a statistical technique that uses several explanatory variables to predict the outcome of a response variable.
- The aim of multiple linear regression is to **model** the **linear relationship** between the **explanatory** (independent) variables and **response (dependent)** variables –
- In short, multiple regression is the extension of **Ordinary Least-Squares (OLS) regression** because it involves more than one explanatory variable. MLR is used extensively in econometrics and financial inference.

2.3.1 Multiple Regression

It is observed in agriculture that, the crop yield (Y) not only depends on the amount of rainfall (X_1) but also on the amount of fertilizer (X_2) applied, pesticides (X_3) used, quality of seeds (X_4), quality of soil (X_5) etc.

Thus in multiple regression, the dependent variable Y is a function of more than one independent variables, i.e.

$$Y = f(X_1, X_2, ..., X_n)$$

In multiple nonlinear regression, f is non-linear.

In multiple linear regression f is linear.

i.e., $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_k X_k$.

2.3.2 Linear Multiple Regression

Suppose Y depends on two independent variables X_1 and X_2;

$$\text{i.e., } Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \quad ...(2.3.1)$$

To estimate the coefficients β_0, β_1, β_2; we apply the least square method to minimise.

$$\sum_{i=1}^{N}\{Y_i - (b_o + b_1 X_{1i} + b_2 X_{2i})\}^2$$

This results in three normal equations given by

$$\sum_{i=1}^{N} Y_i = Nb_0 + b_1\sum_{i=1}^{N} X_{1i} + b_2\sum_{i=1}^{N} X_{2i}$$

$$\sum_{i=1}^{N} X_{1i}Y_i = b_o\sum X_{1i} + b_1\sum_{i=1}^{N} X_{1i}^2 + b_2\sum X_{1i}X_{2i}$$

$$\sum X_{2i}Y_i = b_o\sum X_{2i} + b_1\sum X_{1i}\cdot X_{2i} + b_2\sum X_{2i}^2$$

Here b_0, b_1, b_2 are the least squares estimates of β_0, β_1, β_2.

2.3.3 Linear Multiple regression in k-independent Variables

The above analysis can be generalised to fit $N(k+1)$ tuples $(X_{1i}, X_{2i}, ..., X_{ki})$ $(i = 1 \text{ to } N)$, to the equation.

$$Y = \beta_0 + b_1 X_1 + \beta_2 X_2 + ...+ \beta_k X_k.$$

The $(k+1)$ normal equations are :

$$\sum_{i=1}^{N} Y_i = Nb_0 + b_1\sum_{i=1}^{N} X_{1i} + b_2\sum X_{2i} + ...$$

$$+ b_k\sum_{i=1}^{N} X_{ki}$$

Ex. 2.3.1 : Fit a regression equation to estimate β_0, β_1, β_2 to the following data of a transport company on the weights of 6 shipments, the distances they were moved and the damage of the goods that was incurred. Estimate the damage when a shipment of 3700 kg. is moved to a distance of 260 km.

Weight X_1 (1000 kg)	4.0	3.0	1.6	1.2	3.4	4.8
Distance X_2 (100 km)	1.5	2.2	1.0	2.0	0.8	1.6
Damage Y (Rs.)	160	112	69	90	123	186

☑ **Soln. :**

Let weight X_1 and distance X_2 be independent variables and the damage y be the dependent variable.

Let the equation of regression be,

$$y = b_0 + b_1 x_1 + b_2 x_2$$

Where b_0, b_1, b_2 are estimates of β_0, β_1, β_2. The three normal equations become.

$$\sum_{i=1}^{6} Y_i = nb_0 + b_1 \sum_{i=1}^{6} x_{1i} + b_2 \sum_{i=1}^{6} x_{2i}$$

$$\sum x_{1i}\, y_i = b_0 \sum x_{1i} + b_1 \sum x_{1i}^2 + b_2 \sum x_{1i} x_{2i}$$

$$\sum x_{2i}\, y_i = b_0 \sum x_{2i} + b_1 \sum x_{1i} x_{2i} + b_2 \sum x_{2i}^2$$

We prepare the table :

	x_1 (weight) (1000 kg)	x_2 distance 100 km	y damage Rs.	x_1^2	x_2^2	$x_1 x_2$	$x_1 y$	$x_2 y$
	4.0	1.5	160	16	2.25	6.0	640	240
	3.0	2.2	112	0	4.84	6.6	336	246.4
	1.6	1.0	69	2.56	1.0	1.6	110.4	69
	1.2	2.0	90	1.44	4.0	2.4	108	180
	3.4	0.8	123	11.56	0.64	2.72	418.2	98.4
	4.8	1.6	186	23.04	2.56	7.68	892.8	297.6
Total	18	9.1	740	63.6	15.29	27	250.54	1131.4

The data is : $n = 6$, $\sum x_{1i} = 18$, $\sum x_{2i} = 9.1$,

$\sum y_i = 740$, $\sum x_{1i}^2 = 63.6$, $\sum x_{2i}^2 = 15.29$, $\sum x_{1i} x_{2i} = 27$

$\sum x_i y_i = 250.54$, $\sum x_{2i} y_i = 1131.4$

∴ Normal equations become

$$740 = 6b_0 + 18 b_1 + 9.1 b_2$$

$$250.54 = 18b_0 + 63.6 b_1 + 27 b_2$$

$$1131.4 = 9.1 b_0 + 27 b_1 + 15.29 b_2$$

Solving, we get $b_0 = 14.56$, $b_1 = 30.109$,

$b_2 = 12.16$

Thus the required regression equation is

$$y = 14.56 + 30.109 x_1 + 12.16 x_2$$

☞ **Estimate**

For a weight of 3700 kg. ($x_1 = 3.7$) and for a distance of 260 km. ($x_2 = 2.6$) the damage incurred in rupees is

$y (x_1 = 3.7, x_2 = 2.6) = 14.56 + 30.109(3.7) + 16 (2.6)$

$= 714.58 = 715$ Rs.

2.3.4 Extension of MLR to n Variables

Multiple regression analysis is the extension of the two variable regression theory to n variables X_1, X_2, ..., X_n.

The basic objectives of multiple linear regression are :

(i) To fit the plane of regression of the dependent variable, say X, on the independent variables (X_2 , X_3, ... X_n) by the **principal of least squares** and use this plane to estimate the value of the dependent variable X_1 for a given set of values of the independent variables.

(ii) To estimate and to compute the error $\left(X_1 - \hat{X}_1\right)$. This is achieved by calculating the standard error of the estimate. Problem will make this point clear.

(iii) To determine how much variation in the dependent variable is accounted for by the fitted plane of regression.

This is achieved through the **multiple coefficient** of determination.

2.3.5 Yule's Notation

Consider a trivariate distribution with three variables X_1 , X_2 , X_3. Let X_1 be dependent variable and X_2 , X_3 be independent variables.

Then the equation of the plane of regression of X_1 on X_2 and X_3 is :

$$X_1 = a + b_{12.3} X_2 + b_{13.2} X_3 \quad ...(i)$$

Since the regression coefficients (and correlation coefficients) are independent of change of origin. We assume that X_1, X_2, X_3 are measured from their respective means, hence

$$E(X_1) = E(X_2) = E(X_3) = 0 \quad ...(ii)$$

Taking expection on both sides of Equation (i),

$$E(X_1) = E(a) + b_{12.3}\ E(X_2) + b_{13.2}\ E(X_3)$$

$$\therefore \quad 0 = E(a) + 0 + 0$$

$$\therefore \quad E(a) = 0 \qquad \therefore \quad a = 0$$

$\therefore$ Equation (i) becomes,

$$X_1 = b_{12.3}\ X_2 + b_{13.2}\ X_3 \quad ...(iii)$$

Where,

$b_{12.3}$ = **Partial regression coefficient** of X_1 on X_2 (X_3 being the third variable)

and $b_{13.2}$ = Partial regression coefficient of X_1 on X_3 (X_2 being the third variable) ...(iv)

For given values of X_2 and X_3 , the estimate of X_1 given by Equation (iii) is denoted by $e_{1.23}$.

$$\therefore \quad e_{1.23} = \hat{X}_1 = b_{12.3} X_2 + b_{13.2} X_3 \quad ...(v)$$

$$\therefore \quad X_{1.23} = X_1 - \hat{X}_1 = X_1 - e_{1.23}$$

$$\therefore \quad X_{1.23} = X_1 - b_{12.3}\ X_2 - b_{13.2} X_3 \quad ...(vi)$$

is called the **residual** or **the error of estimate.**

☞ **Remarks**

(1) The subscripts before the dot are known as **primary subscripts** and those the dot are known as **secondary subscripts.**

(2) These notations can be generated to n variables X_1 , X_2, ... X_n .

The equation of plane of regression of X_1 on (X_2 , X_3, ... , X_n) is given by :

$$X_1 = b_{12.34.....n} X_2 + b_{13.24.....n} X_3 + ... + b_{1n.23.....(n-1)} X_n \quad ...(vii)$$

(3)and $X_{1.23.....n} = X_1 - \hat{X}_1$

$$= [X_1 - (b_{12.3.....n} X_2 + b_{13.24.....n} X_3 + ...+b_{1n.23...(n-1)} X_n)] \quad ...(viii)$$

Where, X_i's ; i = 1, 2, ..., n are measured from their respective means i.e.

$$E(X_i) = 0 \ ; \ i = 1, 2, ..., n \quad ...(ix)$$

2.3.6 Order of Regression Coefficients

The **order of regression coefficients** is given by the **number of secondary subscripts in it.**

For example, '$b_{12.3}$' is a regression coefficient of order 1;

$b_{12.34}$ is a regression coefficient of order 2

$b_{12.34.....n}$ is a regression coefficient of order (n – 2).

Thus, we can say : a regression coefficient with **K secondary subscripts** is called the regression coefficient of **order K.**

☞ **Remarks**

(1) In a regression coefficient, the order of the secondary subscripts is immaterial.

For example,

$$b_{12.345} = b_{12.354} = b_{12.435} = b_{12.453} = b_{12.534} = b_{12.543} \quad ...(x)$$

(2) The ordering of the primary subscripts is important.

Of the two primary subscripts, the first subscript refers to the dependent variable and the second subscript refers to the independent variable.

For example,

In $b_{12.34.....n}$, X_1 refers to the dependent variable and

X_2 refers to the independent variable under consideration.

In $b_{21.34.....n}$, X_2 refers to dependent variable and

X_1 refers to independent variable

(3) The order of a residual is also determined by the number of secondary subscripts and is independent of the permutations of the secondary subscripts.

For example, $X_{1.2}$, $X_{1.23}$, $X_{1.23.....n}$ are residuals of order 1,2,.....,(n – 1) respectively.

and also, $X_{1.234} = X_{1.243} = X_{1.342} = X_{1.324} = X_{1.432}$ ....(xi)

2.3.7 Planes of Regression

Consider the distributions with n variables $X_1, X_2, ..., X_n$.

We assume that the variables $X_1, X_2, ..., X_n$ are measured from their respective means, so that

$$E(X_i) = 0 ; \quad \text{for} \quad i = 1, 2, ..., n \qquad ...(xii)$$

Let N be the observations on each of the n variables $X_1, X_2, ... X_n$; so that

$$\sigma_i^2 = \frac{1}{N} \sum X_i^2 ; \qquad i = 1, 2, ..., n \qquad ...(xiii)$$

$$\text{and Cov}(X_i, X_j) = \frac{1}{N} \sum X_i X_j ; i \neq j = 1, 2, ..., n \qquad ...(xiv)$$

Summation is taken over N observations for each of the variables X_i, X_j (i, j = 1, 2,...., n)

❒ **Definition** : Karl Pearson's correlation coefficient r_{ij} between (X_i, X_j) is given by :

$$r_{ij} = \frac{\text{Cov}(X_i, X_j)}{\sigma_i \sigma_j} = \frac{\sum X_i X_j}{N \sigma_i \sigma_j}$$

$$\therefore \sum X_i X_j = N \sigma_i \sigma_j r_{ij} \qquad ...(xv)$$

2.3.8 Equations of Planes of Regression

Initially we consider the case of three variables X_1 , X_2 , X_3 satisfying the conditions (xii), (xiii), (xiv), (xv)

Since X_1, X_2, X_3 are measured from their means, the equation of the plane of regression of X_1 on (X_2 and X_3) is

$$X_1 = b_{12.3} X_2 + b_{13.2} X_3 \qquad ...(xvi)$$

We determine the constants $b_{12.3}$ and $b_{13.2}$ by the principle of least squares by minimizing the sum of squares of errors of **residuals**.

Thus, we determine $b_{12.3}$ and $b_{13.2}$, so that

$$E = \sum X_{1.23}^2 = \sum$$

$$(X_1 - b_{12.3} X_2 - b_{13.2} X_3)^2 \text{ is minimum} \qquad ...(xvii)$$

Using the principle of maxima and minima the **normal equations** for estimating $b_{12.3}$ and $b_{13.2}$ are

$$\frac{\partial E}{\partial b_{12.3}} = -2 \sum X_2 (X_1 - b_{12.3} X_2 - b_{13.2} X_3) = 0$$

$$\text{and } \frac{\partial E}{\partial b_{13.2}} = -2 \sum X_3 (X_1 - b_{12.3} X_2 - b_{13.2} X_3) = 0$$

$$\Rightarrow \sum X_2 X_{1.23} = 0 ; \sum X_3 X_{1.23} = 0 \qquad ...(xviii)$$

$$\text{Also, } \sum X_1 X_2 - b_{12.3} \sum X_2^2 - b_{13.2} \sum X_2 X_3 = 0$$

$$\text{and } \sum X_1 X_3 - b_{12.3} \sum X_2 X_3 - b_{13.2} \sum X_3^2 = 0 \qquad ...(xix)$$

Using Equations (xiii) and (xv); the equations simplify to :

$$\left.\begin{aligned} &r_{12} \sigma_1 \sigma_2 - b_{12.3} \sigma_2^2 - b_{13.2} r_{23} \sigma_2 \sigma_3 = 0 \\ &\text{and } r_{13} \sigma_1 \sigma_3 - b_{12.3} r_{23} \sigma_2 \sigma_3 - b_{13.2} \sigma_3^2 = 0 \end{aligned}\right\} \qquad ...(xx)$$

On Simplification,

$$\left.\begin{aligned} b_{12.3}\,\sigma_2 + b_{13.2}\,r_{23}\,\sigma_3 - r_{12}\,\sigma_1 = 0 \\ b_{12.3}\,\sigma_2\,r_{23} + b_{13.2}\,\sigma_3 - r_{13}\,\sigma_1 = 0 \end{aligned}\right\} \quad ...(xx)$$

Solving Equation (xx) by Cramer's rule for $b_{12.3}$ and $b_{13.2}$; we get

$$\frac{b_{12.3}}{r_{12}\,\sigma_1\,\sigma_3 - r_{13}\,\sigma_1\,\sigma_3\,r_{23}} = \frac{b_{13.2}}{r_{13}\,\sigma_1\,\sigma_2 - r_{12}\,r_{23}\,\sigma_1\,\sigma_2}$$

$$= \frac{1}{\sigma_2\,\sigma_3 - r_{23}^2\,\sigma_2\,\sigma_3}$$

$$\therefore \quad b_{12.3} = \frac{\sigma_1\,\sigma_3\left(r_{12} - r_{13}\cdot r_{23}\right)}{\sigma_2\,\sigma_3\left(1 - r_{23}^2\right)}$$

$$= \frac{\sigma_1}{\sigma_2}\cdot\left(\frac{r_{12} - r_{13}\cdot r_{23}}{1 - r_{23}^2}\right) \quad ...(xxi)$$

and $$b_{13.2} = \frac{\sigma_1\,\sigma_2\left(r_{13} - r_{12}\cdot r_{23}\right)}{\sigma_2\,\sigma_3\left(1 - r_{23}^2\right)}$$

$$= \frac{\sigma_1}{\sigma_3}\cdot\left(\frac{r_{13} - r_{12}\cdot r_{23}}{1 - r_{23}^2}\right) \quad ...(xxii)$$

Substituting these values in (xvi), we get the required equation of the plane of regression of X_1 on (X_2, X_3)

☞ Remark : Note on Cramer's Rule

Let, $a_1 x + b_1 y + c_1 = 0$; $a_2 x + b_2 y + c_2 = 0$

but $$\frac{x}{b_1 c_2 - b_2 c_1} = \frac{-y}{a_1 c_2 - a_2 c_1} = \frac{1}{a_1 b_2 - a_2 b_1};$$

or $$\frac{x}{b_1 c_2 - b_2 c_1} = \frac{-y}{a_2 c_1 - a_1 c_2} = \frac{1}{a_1 b_2 - a_2 b_1}$$

$$\therefore \quad x = \frac{b_1 c_2 - b_2 c_1}{a_1 b_2 - a_2 b_1} \; ; \quad y = \frac{a_2 c_1 - a_1 c_2}{a_1 b_2 - a_2 b_1}$$

2.3.9 Simpler Form of the Equation of the Plane of Regression

Let us write,

$$W = \begin{vmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{vmatrix} \quad ...(xxiii)$$

$$= 1.\begin{vmatrix} 1 & r_{23} \\ r_{12} & 1 \end{vmatrix} - r_{12}\begin{vmatrix} r_{21} & r_{23} \\ r_{31} & 1 \end{vmatrix} + r_{13}\begin{vmatrix} r_{21} & 1 \\ r_{31} & r_{32} \end{vmatrix}$$

$$= \left(1 - r_{23}^2\right) - r_{12}\left(r_{12} - r_{13}\,r_{23}\right) + r_{13}\left(r_{12}\,r_{23} - r_{13}\right)$$

$$\therefore \quad W = 1 - r_{12}^2 - r_{13}^2 - r_{23}^2 + 2\,r_{12}\,r_{13}\cdot r_{23} \quad ...(a)$$

$[\because r_{ii} = 1$ and $r_{ij} = r_{ji}$ $[i \neq j, = 1, 2, 3)]$ ...(b)

Now, Let W_{ij} = Cofactor of element in i^{th} row and j^{th} column of W.

= $(-1)^{1+j}$ [Determinant obtained on deleting the i^{th} row and j^{th} column of w] ...(xxiv)

Now, W_{11} = Cofactor of element in 1st row and 1st column

$$= (-1)^{1+1}\begin{vmatrix} 1 & r_{23} \\ r_{32} & 1 \end{vmatrix} = 1 - r_{23}^2 \quad ...(1)$$

$(\because r_{23} = r_{32})$

$$W_{12} = W_{21} = (-1)^{1+2}\begin{vmatrix} r_{21} & r_{23} \\ r_{31} & 1 \end{vmatrix}$$

$$= -\left(r_{21} - r_{31}\cdot r_{23}\right) = \left(r_{13}\,r_{23} - r_{12}\right) \quad ...(2)$$

$(\because r_{21} = r_{12}, r_{31} = r_{13})$

$$W_{13} = W_{31} = (-1)^{1+3}\begin{vmatrix} r_{21} & 1 \\ r_{31} & r_{32} \end{vmatrix} = r_{12}\,r_{23} - r_{13} \quad ...(3)$$

$$W_{22} = (-1)^{2+2}\begin{vmatrix} 1 & r_{13} \\ r_{31} & 1 \end{vmatrix} = 1 - r_{13}^2 \quad ...(4)$$

$$W_{33} = (-1)^{3+3}\begin{vmatrix} 1 & r_{12} \\ r_{21} & 1 \end{vmatrix} = 1 - r_{12}^2 \quad ...(5)$$

Substituting Equations (1) and (5) in (xxi) and (xxii), the equation of plane of regression is,

$$X_1 = -\frac{\sigma_1}{\sigma_2}\cdot\frac{W_{12}}{W_{11}}X_2 - \frac{\sigma_1}{\sigma_3}\frac{W_{13}}{W_{11}}X_3$$

$$\therefore \quad \frac{X_1}{\sigma_1}W_{11} + \frac{X_2}{\sigma_2}W_{12} + \frac{X_3}{\sigma_3}W_{13} = 0 \quad ...(xxv)$$

is the Equation of plane of regression.

2.3.10 Remarks

(1) If the variables X_1, X_2, X_3 are not measured from their respective means, then the equation of the plane of regression of X_1 on (X_2, X_3) is given by,

$$X_1 - \bar{X}_1 = b_{12.3}\left(X_2 - \bar{X}_2\right) + b_{13.2}\left(X_3 - \bar{X}_3\right) \quad ...(xxvi)$$

i.e. $$\left(\frac{X_1 - \bar{X}_1}{\sigma_1}\right) W_{11} + \left(\frac{X_2 - \bar{X}_2}{\sigma_2}\right) W_{12} + \left(\frac{X_3 - \bar{X}_3}{\sigma_3}\right) W_{13} = 0 \quad ...(xxvii)$$

(2) The equation of the plane of regression of X_2 on (X_1, X_3) is given by,

$$X_2 = b_{21.3} X_1 + b_{23.1} X_3 \quad ...(xxviii)$$

and the values $b_{21.3}$ and $b_{23.1}$ are :

$$b_{21.3} = \frac{\sigma_2}{\sigma_1}\left(\frac{r_{12} - r_{13} \cdot r_{23}}{1 - r_{13}^2}\right) = -\frac{\sigma_2}{\sigma_1}\left(\frac{W_{21}}{W_{22}}\right)$$

$$b_{23.1} = \frac{\sigma_2}{\sigma_3}\left(\frac{r_{23} - r_{12}\, r_{13}}{1 - r_{13}^2}\right) = -\frac{\sigma_2}{\sigma_3}\left(\frac{W_{23}}{W_{22}}\right)$$

Substituting in (xxviii) ; we get

$$X_2 = -\frac{\sigma_2}{\sigma_1}\left(\frac{W_{21}}{W_{22}}\right) X_1 - \frac{\sigma_2}{\sigma_3}\left(\frac{W_{23}}{W_{22}}\right) X_3$$

$$\therefore \frac{X_1}{\sigma_1} W_{21} + \frac{X_2}{\sigma_2} W_{22} + \frac{X_3}{\sigma_3} W_{23} = 0 \quad ...(xxix)$$

Similarly, the equation of the plane of regression of X_3 on (X_1, X_2) is given by,

$$\frac{X_1}{\sigma_1} W_{31} + \frac{X_2}{\sigma_2} W_{32} + \frac{\sigma_3}{\sigma_3} W_{33} = 0$$

(3) By symmetry, the equation of the plane of regression, say X_i, on all other variables X_j; $(j \neq 1, 2, ...,n)$ is given by

$$\frac{X_1}{\sigma_1} W_{i1} + \frac{X_2}{\sigma_2} W_{i2} + ... + \frac{X_i}{\sigma_i} W_{ii} + ... + \frac{X_n}{\sigma_n} W_{in} = 0$$

2.3.11 Interpretation of Partial Regression Coefficients

For a tri-variate distribution with three variables X_1, X_2 and X_3; in the plane of regression of X_1 on X_2 and X_3. We have two partial regression coefficients, i.e. $b_{12.3}$ and $b_{13.2}$.

(i) $b_{12.3}$ represents the change in the value of the variable X_1 for a unit change in the value of the variable X_2, when the variable X_3 is kept constant.

(ii) $b_{13.2}$ represents the change in the value of the variable X_1 for a unit change in the variable X_3, when the variable X_2 is kept constant.

(iii) Similar interpretations can be given to other regression coefficients,
i.e. $b_{ij.k} : i \neq j \neq k = 12.3$

2.3.12 Solved Examples on Regression Equations

Ex. 2.3.2 : Let X_1, X_2 and X_3 be the excess of heights of father, mother and son respectively in 100 samples above their respective mean values in cm. A distribution of these variables gave the following correlation coefficients r_{ij} between X_i and X_j and standard deviations σ_i for i, j = 1, 2, 3.

$r_{12} = 0.3$, $r_{23} = 0.4$, $r_{31} = 0.5$,

$\sigma_1 = 3$, $\sigma_2 = 2$, $\sigma_3 = 4$

Obtain a regression equation of X_1 on X_2 and X_3, and estimate the excess of height of father when excess of heights of mother and son are 0.7 cm and 2.1 cm respectively.

☑ **Soln. :**

▶ **Step I :** Given Data is :

$$r_{12} = 0.3, \quad r_{23} = 0.4, \quad r_{31} = 0.5,$$
$$\sigma_1 = 3, \quad \sigma_2 = 2, \quad \sigma_3 = 4 \quad ...(i)$$

Since X_1, X_2, X_3 denote the excess of heights of father, mother and son respectively above their respective mean values, they are measured from their means.

Hence, the equation of the plane of regression of X_1 on X_2 and X_3 is given by :

$$X_1 = b_{12.3}\, X_2 + b_{13.2}\, X_3 \qquad ...(ii)$$

▶ **Step II :** We have

$$b_{12.3} = \frac{\sigma_1}{\sigma_2}\left(\frac{r_{12} - r_{13}\, r_{23}}{1 - r_{23}^2}\right)$$

$$\text{and} \quad b_{13.2} = \frac{\sigma_1}{\sigma_3}\left(\frac{r_{13} - r_{12} \cdot r_{32}}{1 - r_{32}^2}\right)$$

Substituting the given values from Equation (i) and noting that $r_{ij} = r_{ji}$; we get

$$b_{12.3} = \frac{3}{2}\left[\frac{0.3 - 0.5 \times 0.4}{1 - (0.4)^2}\right]$$

$$= \frac{3 \times 0.10}{2 \times 0.84} = \frac{10}{56} = 0.1786$$

$$\text{and} \quad b_{13.2} = \frac{3}{4}\left[\frac{0.5 - 0.3 \times 0.4}{1 - (0.4)^2}\right]$$

$$= \frac{3 \times 0.38}{4 \times 0.84} = \frac{19}{56} = 0.3393$$

▶ **Step III :** Substituting in Equation (ii), the equation of regression of X_1 on X_2 and X_3 becomes

$$X_1 = 0.1786\, X_2 + 0.3393\, X_3 \qquad ...(iii)$$

The estimate of the height of father when the excess of heights of mother and son are 0.7 cm and 2.1 cm respectively, is given by

$$\hat{X}_1 = (0.1786 \times 0.7) + (0.3393 \times 2.1)$$

$$= 0.12502 + 0.71253 = 0.83755 \text{ cm}$$

$$\therefore \hat{X}_1 = \mathbf{0.83755\ cm.}$$

Ex. 2.3.3 : From heights (X_1) in inches, weights (X_2) in kg., and ages (X_3) in years of a group of students, the following means, variances and correlation coefficients were obtained :

$\bar{X}_1 = 40,\ \bar{X}_2 = 50,\ \bar{X}_3 = 20$;

$S_1 = 3,\ S_2 = 4,\ S_3 = 2$;

$r_{12} = 0.4,\ r_{23} = 0.5,\ r_{13} = 0.25$

where $\bar{X}_i$ is the mean of X_i, S_i^2 is the variance of X_i and r_{ij} is correlation coefficient between X_i and X_j for $i, j = 1, 2, 3$. Find the multiple regressive equation of X_3 (on X_1 and X_2) and estimate the value of X_3 when $X_1 = 43$ inches, $X_2 = 54$ kg.

Soln. :

▶ **Step I :** The multiple regression equation of X_3 on X_1 and X_2 is given by :

$$X_3 - \bar{X}_3 = b_{31.2}\left(X_1 - \bar{X}_1\right) + b_{32.1}\left(X_2 - \bar{X}_2\right) \qquad ...(i)$$

Given data : $\bar{X}_1 = 40,\ \bar{X}_2 = 50,\ \bar{X}_3 = 20;$

S_1 = Std. deviation $= 3, S_2 = 4,\ S_3 = 2$

$r_{12} = 0.4,\ r_{23} = 0.5,\ r_{13} = 0.25$ ...(ii)

▶ **Step II :**

$$\text{Now,} \quad b_{31.2} = \frac{S_3}{S_1}\left(\frac{r_{31} - r_{32}\, r_{12}}{1 - r_{12}^2}\right)$$

$$\text{and} \quad b_{32.1} = \frac{S_3}{S_2}\left(\frac{r_{32} - r_{31}\, r_{21}}{1 - r_{21}^2}\right) \qquad ...(iii)$$

Substituting the given values of r_{ij}'s and S_i's in Equation (iii) from Equation (ii) and noting $r_{ij} = r_{ji}$; we get

$$b_{31.2} = \frac{2}{3}\left[\frac{0.25 - 0.5 \times 0.4}{1 - (0.4)^2}\right] = \frac{2}{3}\left[\frac{0.25 - 0.20}{1 - 0.16}\right]$$

$$= \frac{2 \times 0.05}{3 \times 0.84} = 0.0397 = 0.04$$

$$\text{and } b_{312.1} = \frac{2}{4}\left[\frac{0.5 - 0.25 \times 0.4}{1 - (0.4)^2}\right]$$

$$= \frac{2}{4}\left[\frac{0.5 - 0.1}{1 - 0.16}\right] = \frac{1 \times 0.4}{2 \times 0.84} = 0.2381 = 0.24$$

▶ **Step III :** Substituting these values in Equation (i), the required equation of regression of X_3 on X_1 and X_2 becomes :

$$X_3 - 20 = 0.04\,(X_1 - 40) + 0.24\,(X_2 - 50)$$

$$\therefore \quad X_3 = 0.04\, X_1 + 0.24\, X_2 + (20 - 0.04 \times 40 - 0.24 \times 50)$$

$$\therefore \quad X_3 = 0.04\, X_1 + 0.24\, X_2 + 6.4$$

The estimated value of X_3 when $X_1 = 43$ inches and $X_2 = 54$ kg is given by

$$\hat{X}_3 = 0.04 \times 43 + 0.24 \times 54 + 6.4$$
$$= 17.2 + 12.9 + 6.40$$
$$= \mathbf{21.08 \text{ years}}$$
$$= \hat{X}_3 = 21.08 \text{ years}$$

Ex. 2.3.4 : In a three variate (X_1, X_2, X_3) multiple correlation analysis, the following results were found.

X_i = mean , S_i = s.d. of X_i respectively and

r_{ij} = Correlation coefficient between X_i and X_j :

(i, j) = 1, 2, 3 in a sample of size 20.

$r_{12} = 0.6$; $r_{23} = 0.4$, $r_{13} = 0.5$;

$\bar{X}_1 = 60$; $\bar{X}_2 = 70$; $\bar{X}_3 = 80$; $S_1 = 3$, $S_2 = 4$, $S_3 = 5$

Find the regression line of X_1 on X_2 and X_3. Also find X_1 when $X_2 = 74$ and $X_3 = 85$

☑ **Soln. :**

▶ **Step I :** Given data is :

$\bar{X}_1 = 60$; $\bar{X}_2 = 70$; $\bar{X}_3 = 80$; $S_1 = 3$, $S_2 = 4$, $S_3 = 5$

$$r_{12} = 0.6 ; \quad r_{23} = 0.4, \quad r_{13} = 0.5 ; \quad ...(i)$$

The line of regression of X_1 on X_2 and X_3 is,

$$X_1 - \bar{X}_1 = b_{12.3}\left(X_2 - \bar{X}_2\right) + b_{13.2}\left(X_3 - \bar{X}_3\right) \quad ...(ii)$$

▶ **Step II :**

We have,

$$b_{12.3} = \frac{S_1}{S_2}\left[\frac{r_{12} - r_{13}\, r_{23}}{1 - r_{23}^2}\right]$$
$$= \frac{3}{4}\left[\frac{0.6 - (0.5)\,(0.4)}{1 - (0.4)^2}\right]$$
$$= \frac{3\,(0.60 - 0.20)}{4\,(0.84)} = \frac{120}{336} = 0.3571 \quad ...(iii)$$

and $$b_{13.2} = \frac{S_1}{S_3}\left[\frac{r_{13} - r_{12} \cdot r_{32}}{1 - r_{32}^2}\right]$$
$$= \frac{3}{5}\left[\frac{0.5 - (0.6)\,(0.4)}{1 - (0.4)^2}\right] = \frac{3\,(0.50 - 0.24)}{5(1 - 0.16)}$$
$$= \frac{3 \times 0.26}{5 \times 0.84} = \frac{78}{420} = 0.1851 \quad ...(iv)$$

▶ **Step III :** Substituting the values from Equations (i), (iii), (iv) in Equation (ii), the line of regression of X_1 on X_2 and X_3 becomes :

$$X_1 - 60 = 0.3571\,(X_2 - 70) + 0.1857\,(X_3 - 80)$$
$$= 0.3571\,X_2 + 0.1857\,X_3 - 24.997 - 14.856$$
$$\therefore \quad X_1 = 0.3571 X_2 + 0.1857\,X_3 + 20.147 \quad ...(v)$$

Taking $X_2 = 74$ and $X_3 = 85$ in Equation (v), the estimated value of X_1 is given by :

$$\hat{X}_1 = 0.3571 \times 74 + 0.1857 \times 85 + 20.147$$
$$= 26.4254 + 19.7845 + 20.147$$
$$\therefore \quad \hat{X} = \mathbf{62.3569}$$

Ex. 2.3.5 : Prove that the necessary and sufficient condition for the three planes of regression to be coincident is that :

$$r_{12}^2 + r_{13}^2 + r_{23}^2 \pm 2\, r_{12}\, r_{13}\, r_{23} = 1$$

☑ **Soln. :**

▶ **Step I :** The equations of three planes of regression are

$$X_1 \text{ on } (X_2, X_3) = \frac{X_1}{\sigma_1} W_{11} + \frac{X_2}{\sigma_2} W_{12} + \frac{X_3}{\sigma_3} W_{13} = 0 \quad ...(i)$$

$$X_2 \text{ on } (X_3, X_1) = \frac{X_1}{\sigma_1} W_{21} + \frac{X_2}{\sigma_2} W_{22} + \frac{X_3}{\sigma_3} W_{23} = 0 \quad ...(ii)$$

$$X_3 \text{ on } (X_1, X_2) = \frac{X_1}{\sigma_1} W_{31} + \frac{X_2}{\sigma_2} W_{32} + \frac{X_3}{\sigma_3} W_{33} = 0 \quad ...(iii)$$

▶ **Step II :** The three planes of regression Equations (i) to (iii), will be coincident if the corresponding coefficients in them are proportional i.e. if

$$\frac{W_{11}}{W_{21}} = \frac{W_{12}}{W_{22}} = \frac{W_{13}}{W_{23}} \quad \text{From Equations (i) and (ii)} \quad ...(iv)$$

$$\text{and } \frac{W_{21}}{W_{31}} = \frac{W_{22}}{W_{32}} = \frac{W_{23}}{W_{33}} \quad \text{From Equations (ii) and (iii)} \quad ...(v)$$

Step III :

Now, $$W = \begin{vmatrix} 1 & r_{12} & r_{13} \\ r_{21} & 1 & r_{23} \\ r_{31} & r_{32} & 1 \end{vmatrix}$$

and W_{ij} =Cofactor of element in i^{th} row and j^{th} column of W : (i, j : 1, 2, 3)

Now,

$$W_{11} = (-1)^{1+1} \begin{vmatrix} 1 & r_{23} \\ r_{32} & 1 \end{vmatrix} = 1 - r_{23}^2$$

$$W_{22} = (-1)^{2+2} \begin{vmatrix} 1 & r_{13} \\ r_{31} & 1 \end{vmatrix} = 1 - r_{13}^2 \quad \ldots(vi)$$

$$W_{12} = W_{21} = (-1)^{1+2} \begin{vmatrix} r_{21} & r_{23} \\ r_{31} & 1 \end{vmatrix}$$

$$= -(r_{12} - r_{13} \cdot r_{23}) \quad \ldots(vii)$$

$$W_{33} = (-1)^{3+3} \begin{vmatrix} 1 & r_{12} \\ r_{21} & 1 \end{vmatrix}$$

$$= 1 - r_{12}^2 \quad \ldots(viii)$$

and $$W_{23} = W_{32} = (-1)^{2+3} \begin{vmatrix} 1 & r_{12} \\ r_{31} & r_{12} \end{vmatrix}$$

$$= -(r_{23} - r_{12} \cdot r_{13}) \quad \ldots(ix)$$

Step IV :

From Equation (iv), $W_{11} \cdot W_{22} = W_{12} \cdot W_{21}$

$$\therefore \left(1 - r_{23}^2\right) \cdot \left(1 - r_{13}^2\right) = \left(r_{12} - r_{13} \cdot r_{23}^2\right)^2$$

$$\therefore 1 - r_{13}^2 - r_{23}^2 + r_{13}^2 \cdot r_{23}^2 = r_{12}^2 + r_{13}^2 + r_{23}^2 - 2 \cdot r_{12} \cdot r_{13} \cdot r_{23}$$

$$\therefore r_{12}^2 + r_{13}^2 + r_{23}^2 - 2 r_{12} \cdot r_{13} \cdot r_{23} = 1 \quad \ldots(x)$$

Step V :

Note that last two equations of (iv) will give the same result

From Equation (v), last two equations become :

$W_{22} \cdot W_{33} = W_{23} \cdot W_{32}$

$$\therefore \left(1 - r_{13}^2\right) \cdot \left(1 - r_{12}^2\right) = \left(r_{23} - r_{12} \cdot r_{13}\right)^2$$

$$\therefore 1 - r_{13}^2 - r_{12}^2 + r_{13}^2 \cdot r_{12}^2 = r_{23}^2 + r_{12}^2 \cdot r_{13}^2 - 2 r_{23} \cdot r_{12} \cdot r_{13}$$

$$\therefore r_{12}^2 + r_{13}^2 + r_{23}^2 - 2 r_{12} \cdot r_{13} \cdot r_{23} = 1 \quad \ldots(xi)$$

Which is equivalent to Equation (x)

Similarly the last two Equations of (v), on simplification will give the same result.

2.3.13 Variance of Residual

Consider the plane of regression of X_1 on X_2 and X_3

i.e.

$$X_1 = b_{12.3} X_2 + b_{13.2} X_3 \quad \ldots(1)$$

Where X_i's are measured from means,

So that $E(X_i) = 0$, For $i = 1, 2, 3$ ...(2)

The residual or the error estimate of X_1 is given by,

$$X_{1.23} = X_1 - b_{12.3} X_2 - b_{13.2} X_3$$

Using Equation (2), we get $E[X_{1.23}] = 0$ ...(3)

The variance of residual is given by,

$$\sigma_{1.23}^2 = E[X_{1.23} - E(X_{1.23})]^2$$

$$= E[X_{1.23}]^2 \quad \text{[Using Equation (3)]}$$

$$= \frac{1}{N} \Sigma X_{1.23}^2$$

Where summation is taken over N values on each of the variable X_1, X_2, X_3.

Now,

$$\sigma_{1.23}^2 = \frac{1}{N} \Sigma X_{1.23}^2 = \frac{1}{N} \Sigma X_{1.23} \cdot X_{1.23}$$

$$= \frac{1}{N} \Sigma X_{1.23} [X_1 - b_{12.3} X_2 - b_{13.2} X_3]$$

$$= \frac{1}{N} [\Sigma X_1 \cdot X_{1.23} - b_{12.3} \Sigma X_2 X_{1.23} - b_{13.2} \Sigma X_3 X_{1.23}]$$

$$= \frac{1}{N} \Sigma X_1 \cdot X_{1.23}$$

$$= \frac{1}{N} \Sigma X_1 [(X_1 - b_{12.3} X_2 - b_{13.2} X_3)]$$

$$= \frac{1}{N} \Sigma [(X_1^2 - b_{12.3} X_1 X_2 - b_{13.2} X_1 X_3)]$$

$$= \frac{1}{N} [(\Sigma X_1^2 - b_{12.3} \Sigma X_1 X_2 - b_{13.2} \Sigma X_1 X_3))$$

$$= \sigma_1^2 - b_{12.3}\, r_{12}\, \sigma_1\, \sigma_2 - b_{13.2}\, r_{13}\, \sigma_1\, \sigma_2$$

$$= \sigma_1^2 - \frac{\sigma_1}{\sigma_2}\left[\frac{r_{12} - r_{13}\, r_{23}}{(1 - r_{23}^2)}\right] \cdot r_{12}\, \sigma_1\, \sigma_2 - \frac{\sigma_1}{\sigma_3}\left[\frac{r_{13} - r_{12}\, r_{23}}{(1 - r_{23}^2)}\right] r_{13}\, \sigma_1\, \sigma_3$$

$$\therefore \sigma_{1.23}^2 = \sigma_1^2 \left[1 - \left(\frac{r_{12}^2 + r_{13}^2 - 2\, r_{12}\, r_{13}\, r_{23}}{1 - r_{23}^2}\right)\right]$$

$$\therefore \sigma_{1.23}^2 = \sigma_1^2 \left[\frac{1 - r_{12}^2 - r_{13}^2 - r_{23}^2 + 2\, r_{12}\, r_{13}\, r_{23}}{1 - r_{23}^2}\right] \quad ...(4)$$

The result can also be written as :

$$\sigma_{1.23}^2 = \sigma_1^2 \frac{W}{W_{11}} \quad ...(5)$$

By symmetry, we have

$$\sigma_{2.13}^2 = \sigma_2^2 \left[\frac{1 - r_{12}^2 - r_{13}^2 - r_{23}^2 + 2\, r_{12} \cdot r_{13} \cdot r_{23}}{1 - r_{13}^2}\right]$$

$$= \sigma_2^2 \frac{W}{W_{22}} \quad ...(6)$$

$$\text{and } \sigma_{3.12}^2 = \sigma_3^2 \left[\frac{1 - r_{12}^2 - r_{13}^2 - r_{23}^2 + 2\, r_{12} \cdot r_{13} \cdot r_{23}}{1 - r_{12}^2}\right]$$

$$= \sigma_3^2 \frac{W}{W_{33}} \quad ...(7)$$

2.3.14 Standard Error of the Estimate

The Standard Error (S. E.) of estimate of X_1 is given by the plane of regression of X_1 on (X_2, X_3) is denoted by $\sigma_{1.23}$ and is given by

$$\sigma_{1.23} = \sqrt{\text{Var}\,(X_{1.23})}$$

$$= \sigma_1 \left[\frac{1 - r_{12}^2 - r_{13}^2 - r_{23}^2 + 2\, r_{12}\, r_{13}\, r_{23}}{1 - r_{23}^2}\right]^{1/2} \quad ...(i)$$

$$= \sigma_1 \left[1 \pm R_{1.23}^2\right]^{1/2} \quad ...(ii)$$

Where $R_{1.23}$ is the multiple correlation coefficient of X_1 on (X_2, X_3). This gives us a measure of reliability of the estimate (X_1).

The smaller values of $\sigma_{1.23}$ indicates that the given set of data values are closely scattered about the plane of regression and hence the estimate is quite reliable.

But large values of $\sigma_{1.23}$ indicate that the data values are widely scattered about the fitted plane of regression, indicating that the fitted plane is not a good fit and hence the estimate is not reliable.

2.4 LINEAR WEIGHTED LEAST SQUARES APPROXIMATION

- Data are generally not exact. They are subject to measurement errors. Modeling of data aims at summarising a given set of observations by fitting it to a model, a 'merit function' that depends on adjustable parameters.
- The parameters of the model are then adjusted to achieve a minimum in the merit function, yielding "best-fit" parameters.
- Least square fitting is a maximum likelihood estimation of the fitted parameters if the measurement errors are independent and normally distributed with constant standard deviation.
- The least square with constant standard deviation. The least square principle is to minimize the sum of the squares of the errors.
- For a given set of data, it gives a unique solution. For discrete data (x_0, y_0), (x_1, y_1), ..., (x_N, y_N) weights w_i are positive numbers prescribed according to the relative accuracy of the data points.
- For continuous (data) function, an integrable function w(x) is called a 'Weight function' on (a, b) of $w(x) \geq 0$ for $x \in [a, b]$.

General weighted least squares approximation

Suppose the function $y = f(x)$ is known only at $(N + 1)$ tabulated points $(x_0, y_0), \ldots, (x_N, y_N)$ in the form of a discrete data, with weights $w_0, w_1, w_2, \ldots, w_N$ respectively,

x	x_0	x_1	x_2	...	x_N
y	y_0	y_1	y_2	...	y_N
w	w_0	w_1	w_2	...	w_N

Then the function $f(x)$ can be approximated by a function of the form

$$P(x) = a_0 \phi_0(x) + a_1 \phi_1(x) + \ldots + a_m \phi_m(x) \quad \ldots(2.4.1)$$

Where the set of functions $\{\phi_0, \phi_1, \ldots \phi_N\}$ are linearly independent. These functions $\phi_i(x)$ are known as "basis" or "co-ordinate" functions.

The function $p(x)$ is said to be the weighted least squares approximation of $f(x)$ if the $(m + 1)$ unknown coefficients $a_0, a_1, \ldots a_m$ in Equation (2.4.1) are determined such that the error of approximation.

$$E(a_0, a_1, a_2, \ldots, a_m) = \sum_{k=0}^{N} w_k \left[f(x_k) - \sum_{i=0}^{m} a_i \phi_i(x_k) \right]^2 \quad \ldots(2.4.2)$$

is minimum. The necessary condition for the numbers $a_0, a_1, \ldots a_m$ to minimize E are,

$$\frac{\partial E}{\partial a_j} = 0 \text{ for } j = 0, 1, 2, \ldots, m$$

Differentiating Equation (2.4.2) partially with respect to j, we get $(m + 1)$ linear equations in $(m + 1)$ unknowns $a_0, a_1, \ldots, a_m$ known as normal equations given by,

$$\sum_{k=0}^{N} w_k \left[f(x_k) - \sum_{i=0}^{m} a_i \phi_i(x_k) \right] \phi_j(xk) = 0 \quad \ldots(2.4.3)$$

for $j = 0, 1, \ldots, m$

When the function $f(x)$ is known continuous function on [a, b] then the normal equations take the form

$$\int_a^b w(x) \left[f(x) - \sum_{i=0}^{m} a_i \phi_i(x) \right] \phi_j(x) \cdot dx = 0 \quad \ldots(2.4.4)$$

For $j = 0, 1, \ldots, m$.

Discrete (Data) case

Suppose $P_1(x) = a_0 + a_1 x \quad \ldots(2.4.5)$

be the linear weighted least squares straighted line fitted to the following discrete data

x :	x_0	x_1	x_2	...	x_N
y :	y_0	y_1	y_2	...	y_N
w :	w_0	w_1	w_2	...	w_N

In Equation (2.4.1), we consider $m = 1$ and $\phi_1(x) = x$, then normal Equation (2.4.3) reduce to,

$$a_0 \sum_{i=0}^{N} w_i + a_1 \sum_{i=0}^{N} w_i x_i = \sum_{i=0}^{N} w_i y_i \quad \ldots(2.4.6)$$

$$a_0 \sum w_i x_i + a_1 \sum w_i x_i^2 = \sum w_i x_i y_i \quad \ldots(2.4.7)$$

Solving Equation (2.4.6) and (2.4.7) we get a_0 and a_1 which when substituted in Equations (2.4.5) gives the required linear weighted least squares approximations.

Continuous function (Case)

Suppose $f(x)$ is a known continuous function defined on the interval [a, b], then the normal Equations (2.4.4) reduce to

$$a_0 \int_a^b w(x)\, dx + a_1 \int_a^b x \cdot w(x)\, dx = \int_a^b w(x) \cdot y(x)\, dx \quad \ldots(2.4.8)$$

$$\text{and } a_0 \int_a^b x\, w(x)\, dx + a_1 \int_a^b x^2 \cdot w(x)\, dx = \int_a^b x\, w(x)\, y(x)\, dx \quad \ldots(2.4.9)$$

Solving Equation (2.4.8) and (2.4.9) we get a_0 and a_1. Substituting these values in $y = a_0 + a_1 x$ we get the linear weighted least squares approximation in the continuous case.

2.4.1 Examples : Linear Weighted Least Squares Approximations

Ex. 2.4.1 (Discrete Data)

Fit a linear weighted least squares straight line to the following data :

x	−2	0	2	4	6
y	1	3	6	8	13
w	2	5	10	1	4

☑ **Soln. :**

Let $y = a_0 + a_1 x$ be the L.S. line. Then the normal equations are,

$$a_0 \sum_{i=1}^{5} w_i + a_1 \sum_{i=1}^{5} w_i x_i = \sum w_i y_i$$

and $$a_0 \sum w_i x_i + a_1 \sum w_i x_i^2 = \sum w_i x_i y_i$$

	x	y	w	wx	wx^2	wy	wxy
	−2	1	2	−4	8	2	−4
	0	3	5	0	0	15	0
	2	6	10	20	40	60	120
	4	8	1	4	16	8	32
	6	13	4	24	144	52	312
$\sum$			22	44	208	137	460

Thus $N = 5$, $\sum_{i=1}^{5} w_i = 22$, $\sum w_i x_i = 44$

$\sum w_i x_1^2 = 208$, $\sum w_i y_i = 137$, $\sum w_i x_i y_i = 460$

The two normal equations are :

$$22\, a_0 + 44\, a_1 = 137$$

$$44\, a_0 + 208\, a_1 = 460$$

$\therefore$ $a_1 = 1.55$, $a_0 = 3.127$

$\therefore$ Linear weighted least squares straight line fit is,

$y = 3.127 + 1.55\, x$

At $x = 1$, $y = 4.677$ ...Ans.

Ex. 2.4.2 : (Continuous function)

Fit a linear weighted least square straight line to the function $f(x) = \frac{1}{x}$ on [1, 3] with $w(x) = 1$.

☑ **Soln. :** Let $y = a_0 + a_1 x$ be the L.S. straight line.

The normal equations are

$$a_0 \int_1^3 dx + a_1 \int_1^3 x\, dx = \int_1^3 \frac{1}{x}\, dx$$

and $$a_0 \int_1^3 x\, dx + a_1 \int_1^3 x^2\, dx = \int_1^3 x \cdot \frac{1}{x}\, dx$$

$$\therefore \quad 2a_0 + 4\, a_1 = \log 3 - \log 1$$

$$= \log 3$$

$$4\, a_0 + \frac{26}{3}\, a_1 = 2$$

Solving $a_1 = -0.2959$ and $a_0 = 1.140$

$\therefore$ The required linear L.S. line is,

$$y = 1.140 - 0.2959\, x$$

At $x = 2$, $y(2) = 0.5484$

At $x = 2$, $f(x) = \frac{1}{x}$

$\therefore$ $f(2) = \frac{1}{2} = 0.50$ ...Ans.

2.4.2 Non-Linear Weighted Least Squares Approximation

Given a set of (N + 1) data points, we can fit a non-linear m^{th} degree polynomial of the form

$$y = a_0 + a_1 x + a_2 x^2 + \ldots + a_m x^m \qquad \ldots(2.4.10)$$

by minimizing the error function.

$$E(a_0, a_1, a_2 \ldots, a_m) = \sum w_i \left[y_i - \left(a_0 + a_1 x_i + \ldots + a_m x_i^m \right) \right]^2 \qquad \ldots(2.4.11)$$

The necessary conditions for minimisation of Equation (2.4.11) given the following (m + 1) normal equations :

$$a_0 \sum_{i=0}^{N} w_i + a_1 \sum_{i} w_i x_i + \ldots + a_m \sum x_i^m\, w_i$$

$$= \sum_i x_i^m \, y_i \cdot w_i$$

$$a_0 \sum_i x_i^m w_i + a_0 \sum x_i^{m+1} w_i + \ldots\ldots + a_m \sum x_i^{2m} w_i = \sum x_i^m \, y_i w_i$$

...(2.4.12)

when $m < N + 1$, then the above equations have unique solution.

Discrete case

Suppose $y = a_0 + a_1 x + a_2 x^2$ is the non-linear weighted least square approximation to a given set of (N + 1) data points. Then normal Equations (2.4.12) take the form

$$a_0 \sum_i w_i + a_1 \sum x_i w_i + a_2 \sum x_i^2 \, w_i = \sum y_i w_i$$

...(2.4.13)

$$a_0 \sum x_i w_i + a_1 \sum x_i^2 \, w_i + a_2 \sum x_i^3 \, w_i = \sum x_i y_i w_i$$

...(2.4.14)

$$a_0 \sum x_i^2 \, w_i + a_1 \sum x_i^3 \, w_i + a_2 \sum x_i^4 \, w_i = \sum x_i^2 \, y_i w_i$$

...(2.4.15)

Solving these equations, we obtain the values of a_0, a_1 and a_2.

Ex. 2.4.3 : Fit a non-linear weighted least squares (parabola) second degree polynomial

$y = a_0 + a_1 x + a_2 x^2$ to the following data

x	−3	−1	1	3
y	15	5	1	5
w	2	5	10	20

☑ **Soln. :** The normal equations are :

$$a_0 \sum_{i=1}^{4} w_i + a_1 \sum w_i x_i + a_2 \sum w_i x_i^2 = \sum w_i y_i$$

$$a_0 \sum_{i=1}^{4} w_i x_i + a_1 \sum w_i x_i^2 + a_2 \sum w_i x_i^3 = \sum w_i x_i y_i$$

$$a_0 \sum w_i x_i^2 + a_1 \sum w_i x_i^3 + a_2 \sum w_i x_i^4 = \sum w_i x_i^2 \, y_i$$

x	y	w	wx	wx^2	wx^3	wx^4	wy	wxy	wx^2y
−3	15	2	−6	18	−54	162	30	−90	270
−1	5	5	−5	5	−5	5	25	−25	25
1	1	10	10	10	10	10	10	10	10
3	5	20	60	45	540	1620	100	300	900
0	26	37	59	78	491	1797	165	195	1205

The data is,

$$N = 4, \sum w_i = 37, \sum w_i x_i = 54, \sum w_i x_i^2 = 78$$

$$\sum w_i x_i^3 = 491, \sum w_i x_i^4 = 1797, \sum w_i y_i = 165$$

$$\sum w_i x_i y_i = 195, \sum w_i x_i^2 \, y_i = 1205$$

Thus the three normal equations are,

$$37 a_0 + 59 a_1 + 78 a_2 = 165$$

$$59 a_0 + 78 a_1 + 491 a_2 = 195$$

$$78 a_0 + 491 a_1 + 1797 a_2 = 1205$$

Solving $a_0 = 0.38$, $a_1 = 2.65$, $a_2 = -0.07$

Thus the non-linear weighted least square quadratic fit is,

$$y = 0.38 + 2.65 x - 0.07 x^2$$

with $y(1) = 2.96$ ...Ans.

Continuous function

When f(x) is a continuous function defined on the interval [a, b] with a weighted function w(x) to fit a non-linear weighted least squares second degree polynomial $y = a_0 + a_1 x + a_2 x^2$ on the interval [a, b], then the normal equations are given by,

$$a_0 \int_a^b w(x)\,dx + a_1 \int_a^b x\, w(x)\,dx + a_2 \int_a^b x^2 w(x)\,dx$$

$$= \int_a^b w(x) \cdot y(x) \cdot dx$$ ...(2.4.16)

$$a_0 \int_a^b x\, w(x)\,dx + a_1 \int_a^b x^2 w(x)\,dx + a_2 \int_a^b x^3 w(x)\,dx$$

$$= \int_a^b x\, w(x)\, y(x) \cdot dx \qquad ...(2.4.17)$$

$$a_0 \int_a^b x^2\, w(x)\, dx + a_1 \int_a^b x^3\, w(x)\, dx + a_2 \int_a^b x^4 \cdot w(x) \cdot dx$$

$$= \int_a^b x^2\, w(x) \cdot y(x)\, dx \qquad ...(2.4.18)$$

Ex. 2.4.4 : Fit a non-linear weighted least squares second degree polynomial $y = a_0 + a_1 x + a_2 x^2$ to the function $y(x) = e^x$ on the interval [0, 1] with respect to the weight function $w(x) = x$.

☑ **Soln. :**

In the above equations we substitute

$a = 0, b = 1, w(x) = x, y(x) = e^x$

∴ The normal equations are :

$$a_0 \int_0^1 x\, dx + a_1 \int_0^1 x^2\, dx + a_2 \int_0^1 x^3\, dx = \int_0^1 x\, e^x \cdot dx$$

$$a_0 \int_0^1 x^2\, dx + a_1 \int_0^1 x^3\, dx + a_2 \int_0^1 x^4\, dx = \int_0^1 x^2\, e^x \cdot dx$$

and $$a_0 \int_0^1 x^3\, dx + a_1 \int_0^1 x^4\, dx + a_2 \int_0^1 x^5 \cdot dx = \int_0^1 x^3 \cdot e^x \cdot dx$$

After integration, we get

$$\frac{a_0}{2} + \frac{a_1}{3} + \frac{a_2}{4} = 1$$

$$\frac{a_0}{3} + \frac{a_1}{4} + \frac{a_2}{5} = e - 2$$

$$\frac{a_0}{4} + \frac{a_1}{5} + \frac{a_2}{6} = 6 - 2e$$

i.e. $6\, a_0 + 4\, a_1 + 3\, a_2 = 12$

$20\, a_0 + 15\, a_1 + 12\, a_2 = 60\,(e - 2)$

$15\, a_0 + 12\, a_1 + 10\, a_2 = 60\,(6 - 2e)$

Solving, $a_0 = 1632 - 600\, e$

$a_1 = 2340\, e - 6360$

$a_2 = 5220 - 1920\, e$

$\therefore y(x) = (1632 - 600e) + (2340e - 6360)x + (5220 - 1920e)\, x^2.$

At $x = \frac{1}{2}$, $y = 0.8226$ ...Ans.

2.5 MULTIPLE REGRESSION

It is observed in agriculture that, the crop yield (Y) not only depends on the amount of rainfall (X_1) but also on the amount of fertilizer (X_2) applied, pesticides (X_3) used, quality of seeds (X_4), quality of soil (X_5) etc.

Thus in multiple regression, the dependent variable Y is a function of more than one independent variables, i.e.

$$Y = f(X_1, X_2, ..., X_n)$$

In multiple nonlinear regression, f is non-linear.

In multiple linear regression f is linear

i.e., $$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_k X_k.$$

2.5.1 Linear Multiple Regression

Suppose Y depends on two independent variables X_1 and X_2;

i.e., $$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \qquad ...(2.5.1)$$

To estimate the coefficients $\beta_0, \beta_1, \beta_2$; we apply the least square method to minimise.

$$\sum_{i=1}^{N} [Y_i - (b_0 + b_1 X_{1i} + b_2 X_{2i})]^2$$

This results in three normal equations given by

$$\sum_{i=1}^{N} Y_i = Nb_0 + b_1 \sum_{i=1}^{N} X_{1i} + b_2 \sum_{i=1}^{N} X_{2i}$$

$$\sum_{i=1}^{N} X_{1i} Y_i = b_0 \sum X_{1i} + b_1 \sum_{i=1}^{N} X_{1i}^2 + b_2 \sum X_{1i} X_{2i}$$

$$\sum X_{2i} Y_i = b_0 \sum X_{2i} + b_1 \sum X_{1i} \cdot X_{2i} + b_2 \sum X_{2i}^2$$

Here b_0, b_1, b_2 are the least squares estimates of $\beta_0, \beta_1, \beta_2$.

2.5.2 Linear Multiple regression in k-Independent Variables

The above analysis can be generalised to fit N(k + 1) tuples (X_{1i}, X_{2i}, ..., X_{ki}) (i = 1 to N), to the equation.

$$Y = \beta_0 + b_1 X_1 + \beta_2 X_2 + ...+ \beta_k X_k.$$

The (k + 1) normal equations are :

$$\sum_{i=1}^{N} Y_i = Nb_0 + b_1 \sum_{i=1}^{N} X_{1i} + b_2 \sum X_{2i} + ...$$

$$+ b_k \sum_{i=1}^{N} X_{ki}$$

Ex. 2.5.1 : Fit a regression equation to estimate β_0, β_1, β_2 to the following data of a transport company on the weights of 6 shipments, the distances they were moved and the damage of the goods that was incurred. Estimate the damage when a shipment of 3700 kg. is moved to a distance of 260 km.

Weight X_1 (1000 kg)	4.0	3.0	1.6	1.2	3.4	4.8
Distance X_2 (100 km)	1.5	2.2	1.0	2.0	0.8	1.6
Damage y (Rs.)	160	112	69	90	123	186

☑ **Soln.** : Let weight X_1 and distance X_2 be independent variables and the damage y be the dependent variable.

Let the equation of regression be

$$y = b_0 + b_1 x_1 + b_2 x_2$$

Where b_0, b_1, b_2 are estimates of β_0, β_1, β_2. The three normal equations become.

$$\sum_{i=1}^{6} Y_i = nb_0 + b_1 \sum_{i=1}^{6} x_{1i} + b_2 \sum_{i=1}^{6} x_{2i}$$

$$\sum x_{1i}\, y_i = b_0 \sum x_{1i} + b_1 \sum x_{1i}^2 + b_2 \sum x_{1i} x_{2i}$$

$$\sum x_{2i}\, y_i = b_0 \sum x_{2i} + b_1 \sum x_{1i} x_{2i} + b_2 \sum x_{2i}^2$$

We prepare the table :

	x_1 (weight) (1000 kg)	x_2 distance 100 km	y damage Rs.	x_1^2	x_2^2	$x_1 x_2$	$x_1 y$	$x_2 y$
	4.0	1.5	160	16	2.25	6.0	640	240
	3.0	2.2	112	0	4.84	6.6	336	246.4
	1.6	1.0	69	2.56	1.0	1.6	110.4	69
	1.2	2.0	90	1.44	4.0	2.4	108	180
	3.4	0.8	123	11.56	0.64	2.72	418.2	98.4
	4.8	1.6	186	23.04	2.56	7.68	892.8	297.6
Total	18	9.1	740	63.6	15.29	27	250.54	1131.4

The data is : n = 6, $\sum x_{1i} = 18$, $\sum x_{2i} = 9.1$,

$\sum y_i = 740$, $\sum x_{1i}^2 = 63.6$, $\sum x_{2i}^2 = 15.29$, $\sum x_{1i} x_{2i} = 27$

$\sum x_i y_i = 250.54$, $\sum x_{2i} y_i = 1131.4$

∴ Normal equations become

$$740 = 6b_0 + 18 b_1 + 9.1 b_2$$

$$250.54 = 18b_0 + 63.6 b_1 + 27 b_2$$

$$1131.4 = 9.1 b_0 + 27 b_1 + 15.29 b_2$$

Solving, we get $b_0 = 14.56$, $b_1 = 30.109$,

$b_2 = 12.16$

Thus the required regression equation is

$$y = 14.56 + 30.109 x_1 + 12.16 x_2$$

Estimate

For a weight of 3700 kg. ($x_1 = 3.7$) and for a distance of 260 km. ($x_2 = 2.6$) the damage incurred in rupees is

$$y (x_1 = 3.7, x_2 = 2.6) = 14.56 + 30.109(3.7) + 16 (2.6)$$

$$= 714.58 = 715 \text{ Rs.}$$

2.6 CROSS-VALIDATION IN MACHINE LEARNING

- Cross-validation is a technique for validating the model efficiency by training it on the subset of input data and testing on previously unseen subset of the input data. **It is a technique how a statistical model generalises to an independent dataset.**
- In **machine learning**, we need to test the stability of the model. It means based only on the training dataset, we cannot fit our model on the training dataset.
- For this purpose, we test our model on the sample which is not part of the training dataset. And then we deploy the model on that sample.
- This complete process comes under cross-validation.
- The basic steps of cross-validation are :
  (i) Reserve a subset of the dataset as a validation set.
  (ii) Provide the training to the model using the training dataset.
  (iii) Evaluate model performance using the validation set.
- If the model performs well with the validation set, perform the further step, else check for the issues.

2.6.1 Methods used for Cross-Validation

The common methods used for cross-validation are :

(1) Validation set approach
(2) Leave-P-out cross-validation
(3) Leave one out cross- validation
(4) K-fold cross-validation
(5) Stratified K-fold cross-validation

Among these, K-fold cross-validation is easy to understand, and the output is less biased than other methods.

2.6.2 K-Fold Cross-Validation

- In each set (fold) training and the test would be performed precisely once during this entire process. It helps us to avoid overfitting. When a model is trained using all of the data in a single short and give the best performance accuracy.
- This K-fold cross-validation helps us to build the model which is a generalized one.
- To achieve this K-fold cross validation, we have to split the data set into three sets, training, testing and validation, with the challenge of the volume of the data.
- Here test and train data set will support building model and hyper parameter assessment.
- The model is validated multiple times based on the value assigned as a parameter and which is called K and it should be an **INTEGER**.
- The dataset X is divided randomly into K equal-sized parts, X_i, i = 1, 2, ..., K.
- To generate each pair, we keep one of the K parts out as validation set and combine the remaining (K – 1) parts to form the training set.
- Doing this K times, each time leaving out another one of the K parts out, we get K pairs :

$$V_1 = X_1, T_1 = X_2 \cup X_3 \cup \ldots \cup X_K$$
$$V_2 = X_2, T_2 = X_1 \cup X_3 \cup \ldots \cup X_K$$
$$\vdots$$
$$V_K = X_K, T_K = X_1 \cup X_2 \cup \ldots \cup X_{K-1}$$

- We come across two problems with this. First, to keep the training set large, we allow validation sets that are small.
- Second, the training sets overlap considerably namely, any two training sets share (K – 2) parts.
- K is typically 10 or 30. As K increases, the percentage of training instances increases and we get more robust estimators, but the validation set becomes smaller.

- Also, there is the cost of training the classifier K times, which increases as K is increased.
- As N increases, K can be smaller, if N is small, K should be large to allow large large enough training sets. (N is the number of instances).
- One extreme case of K-fold cross-validation is **leave-one-out** where given a dataset of N instances, only one instance is left out as the validation set (instance) and training uses (N – 1) instances.
- We then get N separate parts by leaving out a different instance at each iteration. This is typically used in applications such as medical diagnosis.

2.6.3 Life Cycle of K-fold Cross-Validation

- Let us have a generalised K-value. If K = 5, it means, we are splitting the given dataset into 5 folds and running the Train and Test.

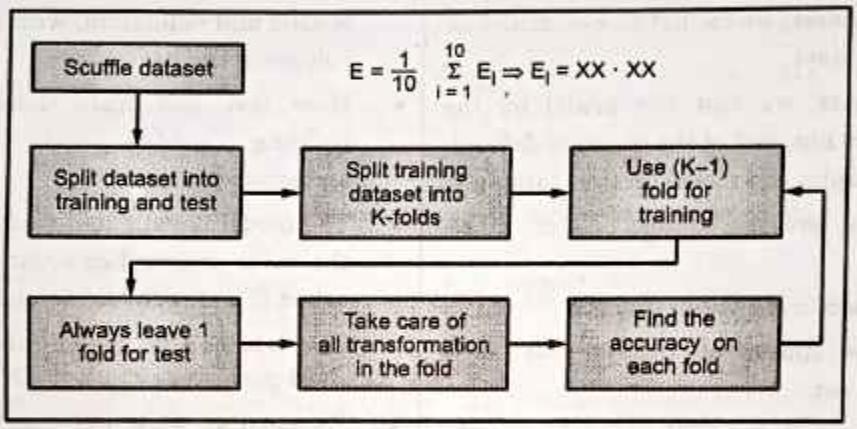


Fig. 2.6.1

- During each run, one fold is considered for testing and the rest will be for training and moving on with iterations, the below pictorial representation gives an idea of the flow of the fold-defined size

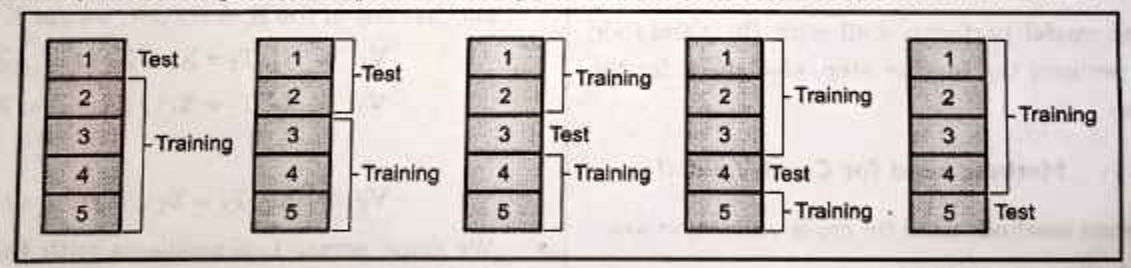


Fig. 2.6.2

- Here, each data-points is used, once in the hold-out set and K – 1 in Training. So during the full iteration at least once, one fold will be used for testing and the rest for training. In the above set, 5-Testing 20 Training.
- In each iteration, we will get an accuracy score and have to sum them and find the mean.
- Here we can understand how the data is spread in a way of consistency and will make a conclusion whether to go for the production with this model (or) NOT.

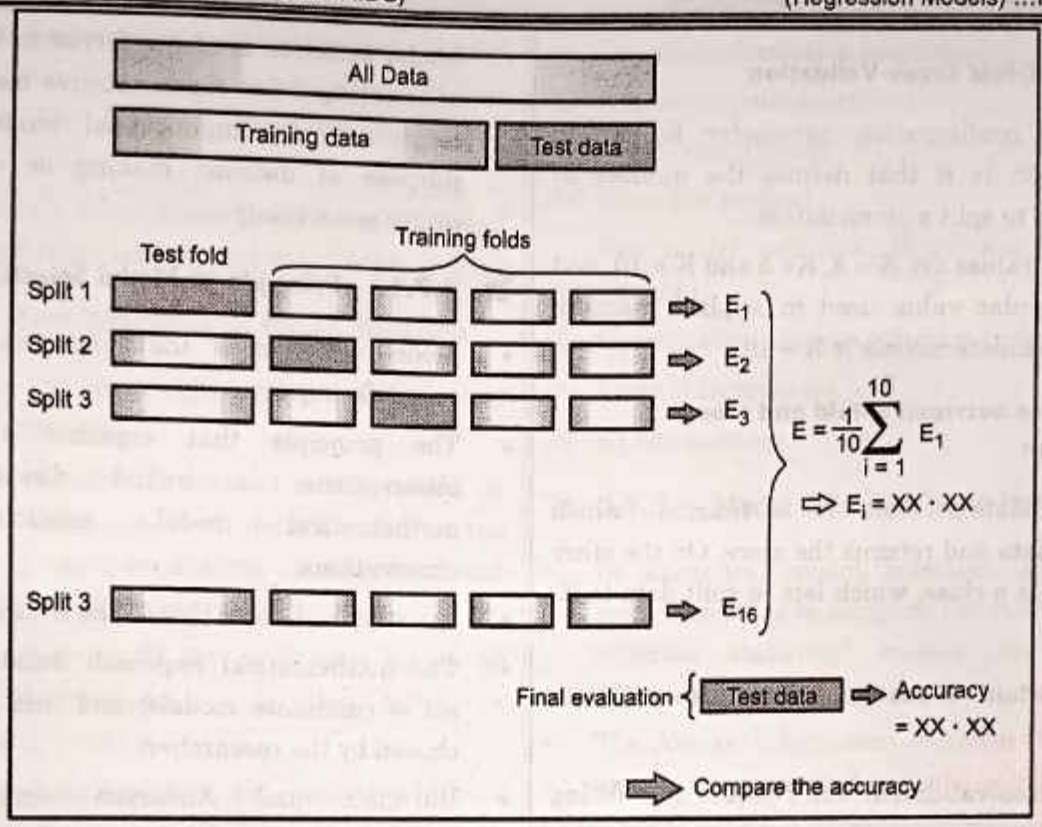


Fig. 2.6.3

2.6.4 Thumb Rules Associated with K-Fold

We discuss a few thumb-rules while dealing with K-fold.

(i) K should be always ≥ 2 and equal to number of records.

If 2, then just 2 iterations

If K = No of records in the dataset, then 1 for testing and n-for training.

(ii) The optimised value for the K is 10 and used with the data of good size, (i.e. commonly used).

(iii) If the K-value is large, then this will lead to less variance across the training set and limit the model currency, difference across the iterations.

(iv) The number of folds is generally inversely proportional to the size of the data set, which means, if the dataset size is too small, the number of folds can increase.

(v) Larger values of K eventually increase the running time of the cross-validation process.

2.6.5 Some Remarks

(1) Short note on K-cross Validation

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample.

The procedure has a single parameter called K that refers to the number of groups that a given data sample is to be split into. Hence, the procedure is often called K-fold cross-validation.

(2) Purpose of K-fold Cross-Validation

K-fold cross-validation is one method that attempts to maximise the use of the available data for training and then testing a model. It is particularly useful for assessing model performance, as it provides a range of accuracy scores across (somewhat) different data sets.

(3) Folds in K-fold Cross-Validation

The key configuration parameter for K-fold cross-validation is K that defines the number of folds in which to split a given dataset.

Common values are K = 3, K= 5 and K = 10, and the most popular value used in applied machine learning to evaluate models is K = 10.

(4) Difference between K-fold and cross-validation

Cross-validation score is a function which evaluates a data and returns the score. On the other hand, K-fold is a class, which lets to split data to K-folds.

(5) Does K-fold cross-validation prevent over fitting ?

K-fold cross-validation won't reduce over fitting on its own, but using it will generally gives a better insight on the model, which can help to avoid or reduce over fitting.

2.7 MODEL SELECTION

- Model selection is the task of selecting a model from among various candidates. It depends on the basis of performance criterion to choose the best one in the context of learning. It is the selection of a statistical model from a set of candidate models from given data.
- In the simplest cases, a pre-existing set of data is considered.
- However, the task consists also of involving the design of experiments, so that collected data is well-Suited to the problem of model selection.
- Konishi and Kitagawa state, "The majority of the problems in statistical inference can be considered to be problems related to statistical modelling". Also according to C_{ox}, "How the translation form subject-matter problem to statistical model is done is often the most critical part of an analysis."
- Model selection is also referred to the problem of selecting a few representative models from a large set of computational models for the purpose of decision making or optimisation under uncertainty

2.7.1 Principle of Model Selection

- Model selection is the fundamental task of scientific inquiry.
- The principle that explains a series of observations is linked directly to a mathematical model, predicting these observations.
- Question is, how to choose the best model?
- The mathematical approach decides among a set of candidate models; and this set must be chosen by the researcher.
- Burnham and Anderson emphasise the importance of choosing models on sound scientific principles, such as understanding of the phenomenological processes or mechanism underlying the data, e.g. chemical reactions.
- Once the set of candidate models has been chosen, the statistical analysis allows us to select the best of these models.
- Model selection technique is supposed to balance goodness of fit with simplicity. More complex models will be better able to adapt their shape to fit the data.
- Goodness of fit is generally determined using a 'likelihood ratio' approach and that leads to a 'chi-squared test.'
- The complexity is generally measured by counting the number of parameters in the model. Model selection techniques are actually estimators of some physical quantity, such as the probability of the model producing the given data. The variance and bias are both important measures of the quality of this estimator, efficiency is also an estimator.

- A standard example of model selection is that of curve fitting. We must select a curve that describes the function that generated the points.
- Selection of a curve depends on the given set of points and other background knowledge.

2.7.2 Two Directions of Model Selection

- There are two main objectives in inference and learning from data.
- One is scientific discovery. It is also called as statistical inference. It is understanding of the underlying data-generating mechanism and interpretation of the nature of the data.
- Another objective is for predicting future or unseen observations, it is also called as statistical prediction.
- Generally, data scientists are interested in both directions.
- Along with two different objectives, model selection can also have two directions: (i) model selection for inference and (ii) model selection for prediction.
- The first direction is to identify the best model for the data, which may provide a reliable characterisation of the sources of uncertainty for scientific interpretation.
- The second direction is to choose a model as machinery to offer excellent predictive performance.
- The model selection is fine for prediction goal, but the use of the selected model for insight and interpretation may be unreliable and misleading. Also, for very complex models selected this way, even predictions may be unreasonable for data.

2.7.3 Methods of Choosing the Set of Candidate Models

Assisting method in choosing the candidate models :

(i) Data transformation (statistics)

(ii) Exploratory data analysis

(iii) Model specification

(iv) Scientific method.

For model selection there are three main approaches :

(1) Optimisation of some selection criterion

(2) Tests of hypotheses, and

(3) ad hoc methods.

2.7.4 Model Selection Used For

- In statistics, model selection is a process researchers use to compare the relative value of different statistical models and determine which one is the best fit for the observed data.
- The Akaike information criterion is one of the most common method of model selection.

2.8 STEPWISE REGRESSION

- Stepwise Regression is the step by step iterative construction of a 'regression' model. It involves the selection of independent variables to be used in a final model.
- It involves adding or removing explanatory variables and testing for statistical significance after each iteration.
- The statistical software packages make stepwise regression possible.

2.8.1 Features of Stepwise Regression

(i) Stepwise regression is a method that iteratively examines the statistical significance of each independent variable in a linear regression model.

(ii) The forward selection approach starts with nothing and adds each new variable incrementally, testing for statistical significance.

(iii) The backward elimination method begins with a full model loaded with several variables and then removes one variable to test its importance relative to overall results.

(iv) Stepwise regression is an approach that fits data into a model to achieve the desired result.

2.8.2 Types of Stepwise Regression

- Stepwise regression is carried out through a series of tests, e.g. F-test, t-test, to find a set of independent variables that influence the dependent variable.
- This is done by computers through iteration. Conducting tests automatically with help from statistical software packages has the advantage of saving time and minimising the mistakes. Stepwise regression can be achieved by

(i) Trying out one independent variable at a time and including it in the regression model if it is 'statistically signification', or

(ii) By including all independent variables in the model and eliminating those that are not statistically significant.

Generally a combination of both method are used and therefore there are three approaches to stepwise regression :

(1) Forward Selection

- It begins with no variables in the model. It tests each variable as it is added to the model. And those that are statistically significant are kept.
- The process is repeated till the results are optimal.

(2) Backward elimination

- This begins with a set of independent variable. Then deletes one at a time, and testing to see if the removed variable is statistically significant.

(3) Bidirectional Elimination

- This is a combination of the first tow methods that test which variables should be included or excluded.

2.8.3 Example of Stepwise Regression

- Using backward elimination method, an example would be to understand energy usage at a factory using variables such as equipment run time, equipment age, staff size, temperature outside and time.
- In this, model includes all the variables-then one at a time is removed after checking which is least statistically significant.
- In this time and temperature are most significant, suggesting the peak energy consumption at the factory when conditioner usage is at its highest.

2.8.4 Limitations of Stepwise Regression

- Regressin analysis, both linear and multivariate, is widely used in the economics and investment world.
- A simple linear regression might look at the 'price-to-earnings ratios' and stock returns to determine if stocks with low P/E ratios ofter high returns.
- The problem with this approach is that market conditions often change, and the relationships in the past need not hold true in the present or the future.
- Statistician note several drawbacks to the approach, including incorrect results, an inherent bias in the process and necessity for significant computing power to develop complex regression models through iteration.

2.8.5 Stepwise Regression Formula

Let us standardise each dependent and independent variable, that is we subtract the mean and divide by standard deviation of a variable, we get the standardised regression coefficients.

We mention the formula :

$$b_{j\text{-}std} = b_j \left(\frac{S_{xj}}{S_y} \right)$$

- Where S_y and S_{xj} are the standard deviations for the dependent variable and the corresponding j^{th} independent variable.
- The percentage change in the square-root of mean square error, (this is true when the specified variable are added to, or deleted from the model) is called RMSE.
- The value is used by MinMSE method.
- This percentage change in Root Mean Square Error (RMSE) is calculate as :

$$\text{Percentage change} = \left[\frac{\text{RMSE}_{\text{previous}} - \text{RMSE}_{\text{current}}}{\text{RMSE}_{\text{current}}}\right] \times 100$$

2.8.6 Conclusion

- Thus we observe how stepwise regression is applied in the industry and what all it takes for the organisation to come to a conclusion.
- With the help of this regression, one is able to gather quality inputs form the feedback surveys and will be able to deliver outputs as per the organisation's needs.

2.9 PREDICATION USING REGRESSION

- The first step in regression is to find the criterion variable. The criterion variable should have acceptable measurement qualities (i.e. reliability and validity).
- Once the criterion has been selected, predictor variables should be identified (i.e. model selection).
- The aim of model selection is to minimise the number of predictors which results in maximum variance in the criterion.
- Thus, the most efficient model maximises the value of the coefficient of determination (R^2). This coefficient estimates the amount of variance in the criterion score by a linear combination of the predictor variables.
- The higher the value is for R^2, the less error or unexplained variance and the better prediction.
- R^2 depends on multiple correlation coefficient(R), which describes the relationship between the observed and predicted criterion scores.
- If there is no difference between the predicted and observed scores, R is equal to 1.00. This is a perfect prediction with no error, and no variance.
- When R is equal to 0.00, there exists no relation between the predictors and the criterion and no variance in scores has been explained ($R^2 = 0.00$).
- The chosen variables cannot predict the criterion. Thus the aim of model selection is to develop a model that results in the highest estimated value for R^2.
- Another method of determining the best model for prediction is to test the significance of adding one or more variables to the model using F-test. F-test is used in analysis of variance.
- It assesses the statistical significance of the difference between values for R^2 derived from 2 or more prediction models.

2.9.1 Assessing Accuracy of the Prediction

- Assessing accuracy of the model is achieved by analysing the Standard Error of Estimate (SEE) and the percentage that the SEE represents of the predicted mean (SEE %).
- The SEE represents the degree to which the predicted scores vary from the observed scores on the criterion measure.
- Lower values of the SEE indicate greater accuracy in prediction. Comparison of SEE for different models using the same sample allows for determination of the most accurate model to use for prediction.
- SEE % is calculated by dividing the SEE by the mean of the criterion (SEE/mean criterion) and it can be used to compare different models derived from different samples.

2.9.2 Assessing Stability of the Model for Prediction

- Once the efficient and accurate model for prediction is determined, it is necessary that the model be assessed for stability.
- A model is said to be 'stable' if it can be applied to different samples from the same population without losing the accuracy of the prediction.
- This is achieved through cross-validation of the model. Cross-validation checks how well the prediction model of one sample performs in another sample from the same population.
- Several method can be employed for cross-validation, including the use of 2 independent samples, split samples, and PRESS-related statistics developed from the same sample.
- Using 2 independent samples involve random selection of 2 groups form the same population.
- One group is "training" or "exploratory" group used for establishing the model of prediction.
- The second group, the "confirmatory" or "validatory" group is used to assess the model for stability.
- The researcher compares R^2 values from the 2 groups and the difference between the two values for R^2 is used as an indicator of model stability.
- Another technique of cross-validation uses split samples.
- Once the sample has been selected from the population, it is randomly divided into 2-subgroups.
- One subgroup becomes the "exploratory" group and the other is used as the "volidatory" group. Again, values for R^2 are compared and model stability is assessed.
- Holiday, Ballard, and McKeown advocate the use of PRESS- related statistics for cross-validation of regression model. This is as a means of dealing with the problem of data-splitting.
- The PRESS method is called as a jack-knife analysis. It is used to address the issue of estimate bias associated with the use of small sample sizes.
- A jack-knife analysis calculates the desired test static multiple times with individual cases omitted from the calculations. I case of the PRESS method, the differences between the actual values of the criterion for each individual and the predicted value are calculated.
- Once determined, the PRESS statistic can be used to calculate a modified form of R^2 and the SEE. R^2_{PRESS} is calculated using the formula :
- $R^2_{PRESS} = 1\left[\frac{PRESS}{SS_{Total}}\right]$, Where SS_{Total} is sum of squares for the original regression equation.
- The smaller the difference between the 2-values for R^2 and SEE, the more stable the model for prediction.

2.10 LOGISTIC REGRESSION (L.R)

Introduction

Logistic Regression is supervised learning classification algorithm used to predict the probability of an output variable. The nature of dependent variable is such that there would be only two possible classes.

2.10.1 L.R. Classification

In a classification problem output or target variable y, can take any discrete values for given set of features or inputs X. L.R. is a regression model. Steps of L-R are :

(1) Data Pre-processing step

(2) Fitting logistic regression to the training set.

(3) Predicting the test results,

(4) Test accuracy of the result

(5) Visualising the test set result.

2.10.2 Sigmoid Function

- It is a powerful machine learning algorithm that utilises a sigmoid function and works best an binary classification problems, although it can be used on multi-class classification problems through the 'one versus all' method.
- In spite of its name, logistic regression is not fit for 'regression tasks'.
- The idea of L.R is to find a relationship between features and probability of particular outcome, e.g. when we have to predict if a student passes or fails in an examination when the number of hours spent studying is given as a feature, the response variable has two values pass and fail.
- Logistic Regression is basically a statistical analysis method used to predict a data value based on prior observations of a data set, e.g. L.R. can be used to predict whether a student from a village will be admitted to a particular college.

2.10.3 Advantages of L.R.

(1) Logistic regression is better than linear regression. Linear regression is used to handle regression problems whereas Logistic Regression is used to handle the classification problems.

(2) Linear regression provides a continuous output but logistic regression provides discrete output.

(3) There are two reasons why linear regression is not suitable for classification. The first one is that linear regression deals with continuous values whereas classification problems require discrete values.

(4) When new data points are added, there is a shift in threshold value].

(5) Logistic reasoning is one of the most important supervised learning classification method.

(6) It is a fast, versatile extension of a generalised linear model.

(7) It is easier to implement and interpret. And it is very effective to train. If number of observations is lesser than the number of features, Logistic regression is not to be used, otherwise it may lead to overfitting.

(8) It makes no assumptions about distribution of classes in feature space.

2.10.4 Disadvantages of L.R.

(1) The limitation of logistic reasoning is the assumption of linearity between the dependent variable and the independent variable.

(2) It not only provides a measure of how appropriate a predictor is, but also its direction of association as (positive) or (negative)

2.10.5 Calculation of L.R.

- L.R. is calculated as the odds ratio denoted by OR. It is simply the odds of being a case for one group divided by the odds of being a case for another group. Logistic reasoning takes the natural logarithms of the odds (referred to as the 'logit' or log-odds) to create a continuous criterion.
- The log function has the effect of removing the floor restriction, i.e. 'logit' function transforms values in the range 0 to 1 to values over the entire real no. range $(-\infty, \infty)$.

2.10.6 Linear Classification with Logistic Regression

- The linear classifier announces a completely confident prediction of 1 or 0, even for examples that are very close to the boundary; in many situations, we need more gradated predictions. Such an issue can be resolved to a large extent by softening the threshold function-the logistic function

$$\text{Logistic}(z) = \frac{1}{1 + e^{-z}}$$

has more convenient mathematical properties.

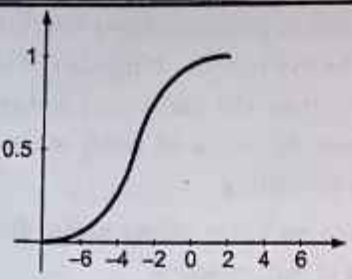


Fig. 2.10.1 : Logistic Function

Logistic (z) is also known as the sigmoid function.

Replacing the threshold function with the logistic function, we have

$$h_w(x) = \text{logistic}(w \cdot x) = \frac{1}{1 + e^{-w \cdot x}}$$

- We observe that the output, being a number between 0 and 1, can be interpreted as a **probability** of belonging to the class labeled 1.
- The hypothesis forms a soft boundary in the input space and gives a probability of 0.5 for any input at the centre of the boundary region, and approaches 0 or 1 as we move away from the boundary.
- The process of fitting the weights of this model to minimise loss on a data set is called **logistic regression.**
- There is no easy closed form solution to find the optimal value of w with this model, but the gradient descent computation is straight forward.

Since our hypothesis has output just 0 or 1, we use L_2 loss function.

For a single example (x, y), the derivation of the gradient is the same as for linear regression upto the point where the actual form of h is inserted.

We have, $\frac{\partial}{\partial w_i} \text{Loss}(w) = \frac{\partial}{\partial w_i} [y - h_w(x)]^2$

$= 2 [y - h_w(x)] \cdot \frac{\partial}{\partial w_i} [y - h_w(x)]$ (Using chain-rule)

$= -2 [y - h_w(x)] \cdot g'(w \cdot x) \cdot \frac{\partial}{\partial w_i} (w \cdot x)$

$= -2 [y - h_w(x)] \cdot g'(w \cdot x) \cdot x_i$

Here we use g to stand for the logistic function.

Since, $g(w \cdot x) = \frac{1}{1 + e^{-w \cdot x}}$

Using chain rule to differentiate,

$$g'(w \cdot x) = \left[\frac{-1 \cdot (-e^{-w \cdot x})}{(1 + e^{-wx})^2}\right] \frac{\partial}{\partial w_i} (wx)$$

$$= \frac{e^{-wx}}{(1 + e^{-wx})^2} \frac{\partial}{\partial w_i} (wx)$$

$$= \left[\frac{1 + e^{-wx} - 1}{(1 + e^{-wx})^2}\right] \cdot x_i \qquad \left[\because \frac{\partial}{\partial w_i} (wx) = x_i\right]$$

$$= \left[\frac{1}{(1 + e^{-wx})} - \frac{1}{(1 + e^{-wx})^2}\right] x_i$$

$$= \frac{1}{(1 + e^{-wx})} \left[1 - \frac{1}{(1 + e^{-wx})}\right] x_i = g(wx)[1 - g(wx)] x_i$$

$\therefore$ The weight update for minimizing the loss is,

$w_i \leftarrow w_i + \alpha [y - hw(x)] \times hw(x) [1 - hw(x)] \times x_i$

We note that logistic regression has become one of the most popular classification techniques for problems in medicine, marketing and survey analysis, credit scoring, public health and other applications.

2.10.7 Logistic Response Function and Logit

The Logit in logistic regression is a special case of link function in a generalised linear model : It is the link function for the Bemoulli distribution. The logit function is the negative of the derivative of the binary entropy function.

The logistic function is the inverse of the natural logit function, and so converts the logarithm of odds into a probability.

In statistics the logit function is the quantile function associated with the standard logistic distribution. It has many uses in data analysis and machine learning, especially in data transformations.

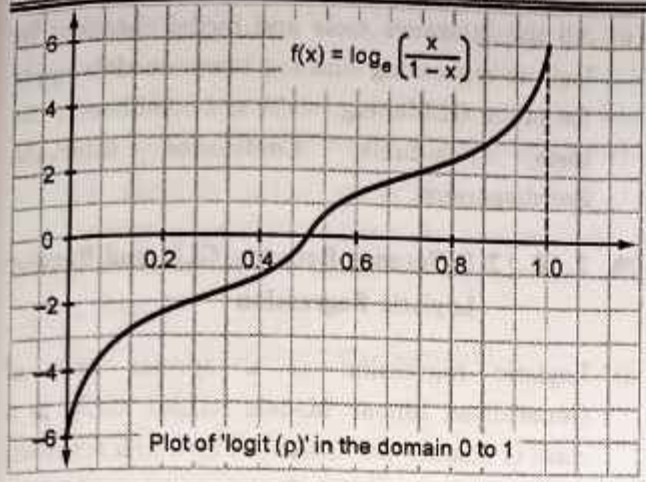


Fig. 2.10.2 : Plot of 'logit (ρ)' in the domain 0 to 1

- Mathematically, the logit is the inverse of the standard logistic function : $\sigma(x) = \frac{1}{1+e^{-x}}$,

  Hence the logit is defined as

  Logit $\rho = \sigma^{-1}(\rho) = \frac{\rho}{1-\rho}$ for $\rho \in (0, 1)$.

- The logit is also called as the log-odds since it is equal to the logarithm of the odds $\frac{\rho}{1-\rho}$. Where ρ is a probability.

2.10.8 Definition

- If ρ is a probability, then $\frac{\rho}{1-\rho}$ is the corresponding odds; the logit of the probability is the logarithm of the odds, i.e.

$$\begin{aligned} \text{logit}(\rho) &= \log_e\left(\frac{\rho}{1-\rho}\right) \\ &= \log_e(\rho) - \log_e(1-\rho) \\ &= -\log_e\left(\frac{1}{\rho} - 1\right) \end{aligned}$$

The 'logistic' function of any numberα is given by the inverse-logit:

$$\text{logit}^{-1}(\alpha) = \text{logistic}(\alpha) = \frac{1}{1-e^{-\alpha}} = \frac{e^{\alpha}}{e^{\alpha}+1} \quad \ldots(i)$$

Consider, $\frac{\tanh\left(\frac{\alpha}{2}\right)+1}{2} = \frac{\frac{e^{\alpha/2}-e^{-\alpha/2}}{e^{\alpha/2}+e^{-\alpha/2}}+1}{2}$

$$= \frac{e^{\alpha/2}-e^{-\alpha/2}+e^{\alpha/2}+e^{-\alpha/2}}{2\left(e^{\alpha/2}+e^{-\alpha/2}\right)}$$

$$= \frac{2e^{\alpha/2}}{2\left(e^{\alpha/2}+e^{-\alpha/2}\right)} = \frac{e^{\alpha/2}}{e^{\alpha/2}+\frac{1}{e^{\alpha/2}}}$$

$$= \frac{e^{\alpha/2}\cdot e^{\alpha/2}}{e^{\alpha/2}\cdot e^{\alpha/2}+1} = \frac{e^{\alpha}}{e^{\alpha}+1} \quad \ldots(ii)$$

From (i) and (ii),

$$\begin{aligned} \text{Logit–1}(\alpha) &= \text{logistic}(\alpha) \\ &= \frac{1}{1+e^{-\alpha}} = \frac{e^{\alpha}}{e^{\alpha}+1} \\ &= \frac{\tanh\left(\frac{\alpha}{2}\right)+1}{2} \end{aligned}$$

- The difference between the logits of two probabilities is the logarithm of the 'odds ratio(R)',

$$\begin{aligned} \text{i.e. } \log(R) &= \log\left[\frac{\rho_1/(1-\rho_1)}{\rho_2/(1-\rho_2)}\right] \\ &= \log\left(\frac{\rho_1}{1-\rho_1}\right) - \log\left(\frac{\rho_2}{1-\rho_2}\right) \\ &= \log(\rho_1) - \log(\rho_2) \end{aligned}$$

2.10.9 Uses and Properties

(i) The logit in logistic regression is a special case of a link function in a generalized linear model; it is the canonical link function for the Bernoulli distribution.

(ii) The logit function is the negative of the derivative of the binary entropy function.

(iii) The logit is also central to the probabilistic Rasch model for measurement, which has applications in psychological and educational assessment.

(iv) The inverse-logit function (i.e. logistic function) is also sometimes referred to as the expit function.

(v) In plant disease epidemology the logit is used to fit the data to a logistic model.

(vi) The log-odds function of probabilities is often used in state estimation algorithms because of its numerical advantages in the case of small probabilities.

2.10.10 Logistic Regression and GLM

- The logistic regression model is an example of a broad class of models known as Generalized Linear Models (GLM).
- For example, GLMs also include linear regression, ANOVA, Poisson regression, etc.

There are three components to a GLM.

(i) **Random component** : It refers to the probability distribution of the response variable (Y); e.g. binomial distribution for Y in the binary logistic regression.

(ii) **Systematic component** : It refers to the explanatory variables (X_1, X_2, ..., X_k) as a combination of linear predictors; e.g. $\beta_0 + \beta_1 X_1 + \beta_2 X_2$ as in case of logistic regression.

(iii) **Link Function** : η or $g(\mu)$: It specifies the link between random and systematic components. It indicates how the expected value of the response relates to the linear predictor of explanatory variables; e.g. η = logit (π) for logistic regression.

2.10.11 Advantage of GLM Over Traditional Regression

(i) We need not transform the response Y to have a normal distribution.

(ii) The choice of link is separate from the choice of random component and hence it has more flexibility in modeling.

(iii) If the link produces additive effects, then we do not need constant variance.

(iv) The models are fitted via maximum likelihood estimation; thus optimal properties of the estimators.

(v) All the inference tools and model checking for logistic regressing and log linear models apply for other GLMs; e.g. wald and likelihood ratio tests, Residuals. Confidence intervals, overdispersion.

2.10.12 Difference Between GLM and Regular Logistic Regression

(i) Logistic regression is a special case of Generalised Linear Models (GLM). GLM is a class of models, parametrised by a link function. If we choose logit link function, we get logistic regression.

(ii) The main benefit of GLM over logistic regression is over fitting avoidance.

- GLM usually tries to extract linearity between input variables and then avoid overfitting of the model.
- Overfitting implies very good performance on training data and poor performance on test data.

2.10.13 Purpose of GLM

- The generalized linear model (GLM) generalizes linear regression by allowing the linear model to be related to the response variable via a link function and allowing the magnitude of the variance of each measurement to be a function of its predicted value.
- Generalized, linear models of different kinds are used based on the probability distribution of the response variables.
- GLM is particularly useful when the response variable is not normally distributed or when the relationship between the predictor variables and the response variable is not linear.

2.11 GENERALISED LINEAR MODEL

- Generalized Linear Models (GLMs) explain that linear regression and Logistic regression are members of a much broader class of models.

- GLMs can be used ot construct the models for regression and classification problems. GLM uses the type of distribution which best describes the data given for training the model.
- We mention below some datasets and the corresponding distributions. And these help us in constructing the model for a particular type of data.

1. Binary classification data-Bernoulli distribution.
2. Real valued data – Gaussian distribution
3. Count-data- Poisson distribution

2.11.1 To Make the Idea of GLMs Clear, we First Define Exponential Families

- Exponential families are a class of distributions whose probability density function (PDF) can be moulded into the following form :

$P(y ; \eta) = b(y). \exp (n_T \cdot T(y) - a(\eta));$

where

η – Natural parameter (can be a scalar or a vector quantity)

y – Label for data

T(y) – Sufficient statistic (Here, it will be equaltoy)

$a(\eta)$ – Log – partition function (it should be purely a function of η).

b(Y) – It should be purely a function of y.

2.11.2 Linear Regression Model

- Linear Regression is a special case of the GLMs. It is considered that the output labels are continuous values and hence are Gaussian distribution.

So, we have

$y \mid x ; \theta \sim N (\mu, \sigma^2)$

$h_\theta (x) = E [y \mid x; \theta]$

$= \mu$

$= \eta$

$= \theta^T x$

- The first equation above states that the output labels (or target variables) should be the members of an exponential family.
- Second equation is the assumption that the hypothesis is equal to the expected value or mean of the distribution and the third equation refers to natural parameter and the input parameters which follow a linear relationship.

2.11.3 Logistic Regression Model

- Logistic Regression is a special case of the GLMs. Here the output labels are 'Binary valued' and hence it is Bernoulli distribution.

Thus, we have

$y | x; \theta \sim$ Bernoulli (ϕ)

$$h_\theta(x) = E[y | x; \theta]$$
$$= \mu$$
$$= 1 + e^{-\eta}$$

From the third assumption, it is proved that :

$$\eta = \theta^T x$$
$$h_\theta(x) = \frac{1}{1 + e^{-\theta_x^T}}$$

- The function that maps the natural parameter to the canonical parameter is known as the canonical response function (here, it is log-partition function) and the inverse of it is known as the 'canonical link function.'
- By using the three assumptions, it can be proved that the 'Logistic' and Linear Regression' belong to a much larger family of models known as GLMs.

2.12 PREDICTED VALUES FROM LOGISTIC REGRESSON

- Logistic regression analysis 'Predicts the odds of an outcome of a categorical variable' based on one or more predictor variables.
- A categorical variable is one that can take on a limited number of values, levels, or categories, such as "Valid" or "invalid".

- Logistic regression is carried out in cases where the response variable can take one of only two forms (i.e., it is binary). There are two general forms, the response variable can take :

(i) Presence/absence, that is, 0 or 1 (or some other binary form).

(ii) A success/failure matrix, where there are two frequencies for each observation (the "success" and the "failures").

2.12.1 Logistic Regression and Presence/Absence

- When the response is in binary form, the logistic regression converts the 1's and 0's to a likelihood (under the given various levels of predictor variable), so the result is in that form.
- When we carry out regression, we describe the data in terms of the variables that form the relationships. When we have regression model, we can describe the relationship using a mathematical model (which is actually the regression model).
- We can use the regression model to make predicted values. Here we predict the new values of the predictor (which is not present in the original dataset) and thereby we can predict the response variable.
- These predicted values are specially important in logistic regression, where the response is binary; i.e. it has only two possibilities.
- Thus the result that is obtained by 'predict' response values in a logistic regression is a probability, i.e., likelihood of getting a 'positive' result when the predictor variable is set to a particular value.

2.12.2 The predict ( ) Command

- The predict( ) command is used to compute predicted values from a regression model. The general form of the command is :

Predict (model, newdata, type)	
Model	A regression model, usually the result of Lm ( ) or gLm ( ).
newdata	A data frame giving the values of the predictor (S) to use in the prediction of the response variable.
type	The type of prediction, usually type = "response".

2.13 COEFFICIENTS AND ODDS RATIOS

- To calculate the odds ratio, we exponentiate the coefficient for a predictor.
- The result is the odds ratio for when the predictor is x + 1, compared to when the predictor is x.
- For example, if the odds ratio for mass in kilograms is 0.95, then for each additional kilogram, the probability of the event decreases by the amount 5%.
- The odds ratio tells us how much higher the odds of exposure are among given cases than among controls.
- An odds ratio = 1.0 (or close to 1.01) indicates that the odds of exposure among cases are the same as, or similar to, the odds of exposure among controls. The exposure is not associated with the given cases.
- When a logistic regression is calculated, the regression coefficient is the estimated increase in the log odds of the outcome per unit increase in the value of the exposure.
- Positive odds ratios indicate that the event is more likely to occur, while negative odd ratios indicate the event is less likely to occur. We observe that the coefficient is the "log odds ratio.'
- Odds ratio is a measure of the strength of association with an exposure and an outcome odds Ratio (OR) >1 means greater odds of association with the exposure,

OR =1 means there is no association between exposure and outcome.

OR < 1 means there is a lower odds of association between the exposure and outcome.

2.13.1 Function

$$\text{Odds Ratio} = \frac{\text{Odds of the event in the exposed group}}{\text{Odds of the event in the non-exposed group}}$$

- If the data is set up in a 2 × 2 table, then the odds ratio is $= \frac{a/b}{c/d} = \frac{ad}{bc}$
- **Examples :** We have a group of smokers (exposed) and non-smokers (not exposed), then the find the rate of lung cancer (event).
- If 17 smokers have lung-cancer, 83 smoker do not have lung cancer, one non-smoker has lung cancer, and 99 non-smokers do not have lung cancer, then the odds ratio is calculated as :

▶ **Step (I) :** First we calculate the odds in the exposed group.

$$\text{Odds in exposed group} = \frac{\text{Smokers with lung cancer}}{\text{somkers without lung cancer}}$$

$$= \frac{17}{83} = 0.205$$

▶ **Step (II) :** We calculate the odds for the non-exposed group.

$$\text{Odds in non-exposed group} = \frac{\text{Non-smokers with lung cancer}}{\text{Non-smokers without lung cancer}}$$

$$= \frac{1}{99} = 0.01$$

▶ **Step (III) :** Finally we calculate the odds ratio.

$$\text{Odds ratio} = \frac{\text{Odds in exposed group}}{\text{Odds in non-exposed group}}$$

$$= \frac{0.205}{0.01} = 20.5$$

- Thus using the odds ratio, this group of smokers has 20 times the odds of having lung cancer than non-smokers.
- Since odds ratio > 1; it is significant.

2.14 SIMILARITIES AND DIFFERENCES BETWEEN LINEAR AND LOGISTIC REGRESSION

Sr. No.	Linear Regression	Logistic Regression
1.	Linear regression is used to predict the continuous dependent variable using a given set of independent variables.	Logistic regression is used to predict the categorical dependent variable using a given set of independent variable.
2.	Linear regression used to solve regression problems	Logistic regression is used for solving classification problems.
3.	In linear regression we predict the values of continuous variables.	In logistic regression, we predict the values of categorical variables.
4.	In linear regression, we find, the best fit line, by which we can easily predict the output.	In logistic regression, we find the S-curve by which we can classify the sample.
5.	Least square estimation method is used for estimation of accuracy.	Maximum likelihood estimation method is used for estimation of accuracy.
6.	The output for linear regression must be a continuous value, such as price, age, etc.	The output of logistic regression must be a categorical value such as 0 or 1, Yes or Not etc.
7.	In linear regression, the relation between dependent and independent variable is linear.	In logistic regression, it is not required that the relation between dependent and independent variable be linear.
8.	In linear regression there may be collinearity between the independent variable.	In logistic regression, there need not be collinearity between the independent variable.

2.15 LINEAR AND LOGISTIC REGRESSION : ASSESSING THE MODELS

(I) (1) Evaluating Logistic Regression Model

(i) Akaike Information Criteria (AIC). One can look at AIC as counterpart of adjusted r square in multiple regression.

(ii) Null Deviance and Residual deviance

(iii) Confusion Matrix

(iv) Receiver Operator Characteristic (ROC)

(2) Model Evaluation in Regression

(i) R square/Adjusted R square

(ii) Mean Square Error (MSE)/Root mean square Error (RMSE)

(iii) Mean Absolute Error (MAE)

(II) (1) Assessing a Linear Relationship

(i) If the slope is positive, then there is a positive relationship, i.e., as one increase, the other also increases,

(ii) If the slope is negative, then there is a negative linear relationship, i.e., as one increases the other variable decreases.

(iii) If the slope is 0, then as one increases, the other remains constant.

(2) Accuracy in Logistic Regression

The most basic of logistic regression is predictive accuracy. For this we refer to prediction accuracy table (also known as classification table).

(III) Assessing the model

- The three main metrics used to evaluate a classification model are

  (i) accuracy, (ii) precision, (iii) recall.
- Accuracy is defined as the percentage of correct predictions for the test data.
- It can be calculated by dividing the number of correct predictions by the number of total predictions.

Chapter Ends...

□□□

Module 3

CHAPTER 3

Time Series

University Prescribed Syllabus

Overview of Time Series Analysis Box-Jenkins Methodology, ARIMA Model Autocorrelation Function (ACF), Autoregressive Models, Moving Average Models, ARMA and ARIMA Models, Building and Evaluating an ARIMA Model, Reasons to Choose and Cautions.

3.1 INTRODUCTION

Time series analysis is a specific way of analyzing a sequence of data point collected over an interval of time.

In time series analysis, analysts record data points at consistent intervals over a set period of time rather than just recording the data points randomly.

We should note that this type of analysis is not merely the act of collecting data over time. Here, the analysis show how variables change over time.

Time is a crucial variable because it indicates how the data adjusts over the course of data points as well as the final results. It provides an additional source of information and a set order of dependencies between the data.

Time series analysis requires a large number of data points to ensure consistency and reliability. It ensures that any trends or patterns discovered are not outliers and can account for seasonal variance.

Time series data can be used for forecasting predicting future data based on historical data. It can also be applied to 'real-valued', 'continuous data', 'discrete symbolic data' (i.e. sequence of characters, such as letters and words in English language).

3.1.1 Overview of Time Series Analysis

Modern time series analysis and related research methods represent a sophisticated leap in the ability to analyze longitudinal data gathered on single subjects or units. Early time series designs, especially as used within psychology, relied heavily on graphical analysis to describe and interpret results. While graphical methods are useful and still provide important ancillary information to the understanding of a time series process, the ability to bring a sophisticated statistical methodology to bear on this class of data has revolutionized the area of single subject research.

3.1.2 Time Series Analysis Examples

Time series analysis is also used for non-stationary data things that are constantly fluctuating over time or are affected by time.

Industries like finance, retail and economics frequently use time series analysis because currency and sales are always changing stock – market analysis is an excellent example of time – series analysis in action, especially with automated trading algorithms.

Time-series also analyses changes in weather – fore castes examples of time-series analysis in action include :

(i) Weather data,
(ii) Rainfall measurements,
(iii) Temperature readings,
(iv) Heart – rate monitoring (EKG),
(v) Brain – monitoring,
(vi) Quarterly sales,
(vii) Stock prices,
(viii) Automated stock trading,
(ix) Industry forecast,
(x) Internet rates.

3.2 OPERATIONS ON TIME SERIES ANALYSIS

GQ. Explain Time series analysis.

Since time-series analysis include many categories or variations of data, analysts must make complex models.

But analysts cannot account for all variances, and they cannot generalise a specific model to every sample.

Too complex models or models that try to do too many things can lead to a lack of fit.

Overfitting models cannot distinguish between random error and true relationships, and analysis may become skewed and may give incorrect forecast.

(I) Models of time series also include

(i) **Classification :** Identifies and assigns categories to the data

(ii) **Curve - fitting :** Data is being plotted along a curve to study the relationships of variables within the data.

(iii) **Explanative analysis :** Attempts to understand the data and the relationships within it, as well as cause and effect.

(iv) **Descriptive analysis :** It identifies patterns in time series data, like trends, cycles, or seasonal variation .

(v) **Explorative analysis :** It highlights the main characteristics of the time series data, usually in a visual format.

(vi) **Forecasting :** It predicts future data. This type is based on historical trends. It uses the historical data as a model for future data, predicting scenarios that could happen along future plot points.

(vii) **Intervention analysis :** It studies now an event can change the data.

(viii) **Segmentation :** It splits the data into segments to show the underlying properties of the source information.

(II) Data classification

Time series data can be classified into two main categories :

(i) **Stock time series data :** It implies measuring attributes at a certain point in time, like a static snapshot of the information.

(ii) **Flow time series data :** It means measuring the activity of the attributes over a certain period, which is generally part of the total whole and makes up a portion of the results.

(III) Data variations

In time-series data, variations occur throughout the data :

(i) **Functional analysis :** It can pick out the patterns and relationships within the data to identify notable events.

(ii) **Trend analysis :** It means determining consistent movement in a certain direction. There are two types of trends :

(a) deterministic, where one can find the underlying cause and

(b) stochastic which is random and unexplainable.

(iii) **Seasonal variations :** It describes events that occur at specific and regular intervals during the course of a year. Serial dependence occurs when data points close together in time tend to be related.

3.3 TIME-SERIES FOR AUTOCORRELATION

GQ. Discuss time-Series for Autocorrelation.

- Autocorrelation is a mathematical representation of the degree of similarity between a given **time series** and a lagged version of itself over successive time intervals.
- Conceptually it is similar to the correlation between two different time series. But autocorrelation uses the same time series twice :
  (i) Once in its original form and
  (ii) Once lagged one or more time periods.
- For example, if it is rainy today, the data suggests that it is more likely to rain tomorrow than if it is clear today.
- When it comes to investigating, a stock might have a strong positive autocorrelation of returns; it suggest that if it is 'up' today, it is more likely to be 'up' tomorrow. Thus autocorrelation can be a useful tool for traders to utilize ; particularly for technical analysts.

3.3.1 Dependency and Auto Correlation

- In time series analysis, dependence is assessed by calculating the values of the autocorrelations among the data points in the series.
- In contrast to a correlation coefficient, which is generally used to estimate the relationship between two different variables measured at the same time on multiple subjects, an autocorrelation estimates the relationships within one variable that is measured at regular intervals over time on only one subject.
- The degree of dependency in a time series is determined by the magnitude of the autocorrelations that can vary between -1.00 and 1.00, with a value of 0.00 indicating no relationship. These values can be interpreted as the strength of relationship between consecutive measurements.
- The accuracy of estimation improves as the number of observations increase. Generally, 50 or more observations provide reasonably accurate estimates (Glass, Willson, & Gottman, 1975; Ljung & Box, 1978; Box & Pierce, 1970).
- In practical terms, the degree of dependency indicates the extent to which an observation at any point in time is predictable from one or more preceding observations.
- The direction of dependency in a time series refers to whether an autocorrelation is positive or negative. The direction can be determined with a high degree of accuracy when there is strong dependency in the data.
- As the degree of dependency approaches zero the direction becomes less important. With strong dependency, the direction has clear implications. When the sign of the autocorrelation is negative, a high level for the series on one occasion predicts a lower level for the series on the next occasion. When the sign is positive, a high level of the series on one occasion predicts a higher level on the next occasion.
- In calculating an autocorrelation the data points of the series are paired off in a lagged manner against each other.
- Fig. 3.3.1 illustrates this process using the first 20 observations for Lag 1, Lag 2 and Lag 3. Note that for Lag 1 in this example, the second observation is paired with the first, the third observation is paired with the second, and so on, until the last observation is paired with the second from the last observation.
- If we now calculate the correlation between these paired observations, we will have calculated the lag one autocorrelation. If we were to pair the third observation with the first, the fourth observation with the second, and so on, we could then calculate the lag two autocorrelation.
- The lag of an autocorrelation refers to how far in the past the dependency among measurements is examined. In the behavioral sciences, the size of the autocorrelation generally decreases as the lag increases.
- An exception would be with seasonal or cyclic data, which are relatively common and will be discussed in more detail in a later section.
- The interpretation of the pattern of autocorrelations within a time series provides one diagnostic step of the model identification process.

Example. Lag 1			Example. Lag 2			Example. Lag 3		
Time	X	X - 1	Time	X	X - 2	Time	X	X - 3
1	6	--	1	6	--	1	6	--
2	10	6	2	10	--	2	10	--
3	4	10	3	4	6	3	4	--
4	13	4	4	13	10	4	13	6
5	4	13	5	4	4	5	4	10
6	11	4	6	11	13	6	11	4
7	4	11	7	4	4	7	4	13
8	6	4	8	6	11	8	6	4
9	4	6	9	4	4	9	4	11
10	15	4	10	15	6	10	15	4
11	5	15	11	5	4	11	5	6
12	14	5	12	14	15	12	14	4
13	5	14	13	5	5	13	5	15
14	13	5	14	13	14	14	13	5
15	5	13	15	5	5	15	5	14
16	10	5	16	10	13	16	10	5
17	3	10	17	3	5	17	3	13
18	14	3	18	14	10	18	14	5
19	3	14	19	3	3	19	3	10
20	16	3	20	16	14	20	16	3

Fig. 3.3.1 : Illustration of Arrangement of Data to Calculate Autocorrelations for First Three Lags Using First 20 Observations from ROD Example

The calculation and interpretation of the pattern of the related partial-autocorrelations calculated at each lag is employed as a second diagnostic step to aid in the identification of the specific ARIMA model which describes the process underlying the time series. Partial- autocorrelations are mathematically complex and will not be formally defined here. They are estimated from a solution of the Yule-Walker equation system and the interested reader should examine Box, Jenkins & Reinsel, (1994,), Glass Willson & Gottman (1975) or West & Hepworth (1991) for a detailed description. The interpretation of partial-autocorrelations is that of a measure of the correlation between specific lags of the time series values after the correlation at the intervening lags has been partialled out or controlled for. Fig. 3.3.2 illustrates the autocorrelations and partial-autocorrelations for the ROD data from Fig. 3.3.2.

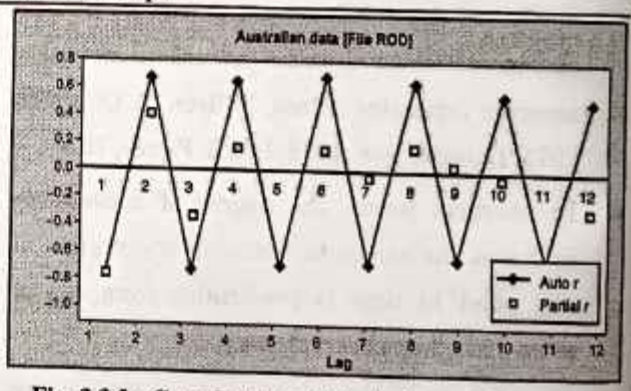


Fig. 3.3.2 : Correlogram of the Autocorrelations and Partial Autocorrelations for the ROD

3.3.2 Some Remarks

(1) Autocorrelation represents the degree of similarity between a given time series and a lagged version of itself over successive time intervals.

(2) Autocorrelation measures the relationship between a variable's current value and its past values.

(3) An autocorrelation of + 1 represents a perfect positive correlation, while an autocorrelation of degree – 1 represents a perfect negative correlation.

(4) Technical analysts can use autocorrelation to measure how much influence past prices for a security have on its future price.

Note : Different fields of study define autocorrelation differently, and not all of these definitions are equivalent. In some fields the term is used interchangeably with 'autocovariance'.

Unit root processes, trend-stationary processes, autoregressive processes, and moving average processes are specific forms of processes with autocorrelation.

3.3.3 Auto-correlation of Stochastic Processes

In statistics, the autocorrelation of a real or complex 'random process' is the 'Pearson Correlation' between values of the process at different times, as a function of the two times or of the time lag. Let $\{X_t\}$ be a random process, and t be any point in time (t may be an integer for a 'discrete time' process or a 'real number' for a 'continuous-time' process). Then X_t is the value (or realisation) produced by a given run of the process at time t.

Let the process possess mean μ_t and variance σ_t^2 at time t, for each t. Then the definition of the "auto-correlation function" between times t_1 and t_2 is

$$R_{xx}(t_1, t_2) = E\left[X_{t_1} \cdot \bar{X}_{t_2}\right] \quad ...(i)$$

where E is the expected value operator and the bar represents 'complex conjugation'.

Also, "auto-covariance function" between times t_1 and t_2 is defined as

$$K_{xx}(t_1, t_2) = E\left[(X_{t_1} - \mu_{t_1})\left(\overline{X_{t_2} - \mu_{t_2}}\right)\right]$$

$$= E\left[X_{t_1} \cdot \bar{X}_{t_2}\right] - \mu_{t_1} \cdot \bar{\mu}_{t_2} \quad ...(ii)$$

We note that this expression is not well-defined for all time series or processes, because the mean may not exist, or the variance may be zero (for a constant process) or infinite (for processes with distribution lacking well-behaved moments, such as certain types of 'power law').

3.3.4 Definition for Wide-Sense Stationary Stochastic Process

If $\{X_t\}$ is a 'wide-sense stationary process' then the mean μ and variance σ^2 are time-independent. In this case 'auto-covariance function' depends only on the lag between t_1 and t_2 : the auto covariance depends only on the time-distance between the pair of values but not on their position in time.

This implies that auto covariance and autocorrelation can be expressed as a function of time-lag, and this would be an even function of the lag $\tau = t_2 - t_1$.

And we get familiar forms for the 'auto-correlation function'.

$$R_{xx}(\tau) = E\left[X_{t+\tau} \cdot \bar{X}_t\right] \quad ...(iii)$$

and the auto-covariance function :

$$K_{xx}(\tau) = E\left[(X_{t+\tau} - \mu)\left(\overline{X_t - \mu}\right)\right]$$

$$= E\left[X_{t+\tau} \cdot \bar{X}_t\right] - \mu\,\bar{\mu} \quad ...(iv)$$

3.3.5 Normalization

We normalize the auto covariance function to get a time-dependent 'Pearson Correlation Coefficient'.

The definition of the auto-correlation coefficient of a stochastic process is

$$\rho_{xx}(t_1, t_2) = \frac{K_{xx}(t_1, t_2)}{\sigma_{t1} \cdot \sigma_{t2}}$$

$$= \frac{E\left[(X_{t_1} - \mu_{t_1})\left(\overline{X_{t_2} - \mu_{t_2}}\right)\right]}{\sigma_{t_1} \cdot \sigma_{t2}}$$

If the function ρ_{xx} is well-defined, its value must lie in the range [– 1, 1], with 1 indicating perfect correlation and – 1 indicating perfect 'anti-correlation'.

For a 'weak-sense stationary', 'wide-sense stationary' (wss), process, the definition is :

$$\rho_{xx}(\tau) = \frac{K_{xx}(\tau)}{\sigma^2} = \frac{E\left[(X_{t+\tau} - \mu)\ \overline{(X_t - \mu)}\right]}{\sigma^2}$$

where $K_{xx}(0) = \sigma^2$

Remarks : The normalisation is important because the interpretation of the autocorrelation as a correlation provides a scale-free measure of the strength of 'statistical dependence' and because the normalisation has an effect on the statistical properties of the estimated auto-correlations.

3.3.6 Properties

(1) Since the auto-correlation function R_{xx} is an even function, it can be stated as

$$R_{xx}(t_1, t_2) = \overline{R_{xx}(t_2, t_1)}$$

for a WSS process

$$R_{xx}(\tau) = \overline{R_{xx}(-\tau)}$$

(2) **Maximum at zero**

For a WSS process,

$$|R_{xx}(\tau)| \leq R_{xx}(0)$$

and we note that $R_{xx}(0)$ is always real.

(3) **Cauchy-Schwarz inequality**

The 'Cauchy-Schwarz inequality', inequality for stochastic processes :

$$|R_{xx}(t_1, t_2)|^2 \leq E\left[|X_{t_1}|^2\right] \cdot E\left[|X_{t_2}|^2\right]$$

(4) **Wiener-Khinchin theorem**

The 'Wiener-Khinchin theorem' relates the autocorrelation function R_{xx} to the 'power spectral density S_{xx}' via the Fourier transform.

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} S_{xx}(f) \cdot e^{i2\pi f\tau} \cdot df;$$

$$S_{xx}(f) = \int_{-\infty}^{\infty} R_{xx}(\tau) \cdot e^{-i2\pi f\tau} \cdot d\tau$$

For real-valued function, the symmetric autocorrelation function has a real-symmetric transform, so the Wiener-Khinchin theorem can be reexpressed in terms of real cosines only :

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} S_{xx}(f) \cdot \cos(2\pi f\tau)\, df$$

and $$S_{xx}(f) = \int_{-\infty}^{\infty} R_{xx}(\tau) \cdot \cos(2\pi f\tau)\, d\tau$$

3.3.7 Auto-correlation of Random Vectors

The (potentially time-dependent) 'auto-correlation matrix' (also called second moment) of a (potentially time-dependent) 'random vector'

$X = (X_1, X_2, \ldots, X_n)^T$ is a $n \times n$ matrix containing as elements the autocorrelations of all pairs of elements of the random vector X.

The autocorrelation matrix is used in various 'digital signal processing' algorithms.

For a 'random vector $X = (X_1, \ldots X_n)^T$ containing random elements ' whose expected value and variance exist, the 'auto-correlation matrix' is defined by

$$R_{xx} \doteq E[X X^T]$$

where 'T' denotes transposition and has dimension $n \times n$.

Written component-wise :

$$R_{xx} = \begin{bmatrix} E(X_1 X_1) & E(X_1 X_2) & \ldots & E(X_1 X_n) \\ E(X_2 X_1) & E(X_2 X_2) & \ldots & E(X_2 X_n) \\ \vdots & \vdots & \vdots & \vdots \\ E(X_n X_1) & E(X_n X_2) & \ldots & E(X_n X_n) \end{bmatrix}$$

If Z is a 'complex random vector', the autocorrelation matrix is defined by

$$R_{zz} \doteq E[Z Z^H],$$

where 'H' denotes Hermitian transposition.

For example, if $X = (X_1, X_2, X_3)^T$ is a random vector, then R_{xx} is a 3×3 matrix whose $(i, j)^{th}$ entry is $E[X_i X_j]$.

3.3.8 Properties of the Autocorrelation Matrix

(i) The autocorrelation matrix is a Hermitian matrix for complex random vectors and a symmetric matrix for real random vectors :

(ii) The autocorrelation matrix is a positive semidefinite matrix, i.e.

$a^T R_{xx}\ a \geq 0$ for all $a \in R^n$ for a real random vector, and

$a^H R_{zz}\ a \geq 0$ for all $a \in C^n$ in case of a complex random vector.

(iii) All eigen values of the autocorrelation matrix are real and non-negative.

(iv) The auto-covariance matrix is related to the auto-correlation matrix as follows :

$$K_{xx} = E[(X - E[X])(X - E[X])^T]$$
$$= R_{xx} - E[X]\,E[X]^T ; \text{ and}$$

for complex random vectors ;

$$K_{zz} = E[(Z - E[Z])(Z - E[Z])^H]$$
$$= R_{zz} - E[Z]\,E[Z]^H$$

3.3.9 Auto-correlation of Deterministic Signals

In 'signal processing', the above definition is often used without the normalisation, that is, without subtracting the mean and dividing by the variance. When the autocorrelation function is normalised by mean and variance, it is also called as 'autocorrelation coefficient' or 'auto covariance function'.

3.3.10 Auto-correlation of Continuous-Time Signal

Given a signal f(t), the continuous auto-correlation $R_{ff}(\tau)$ is most often defined as the continuous cross-correlation integral of f (t) with itself, at lag τ.

$$R_{ff}(\tau) = \int_{-\infty}^{\infty} f(t+\tau)\,\overline{f(t)}\,dt$$
$$= \int_{-\infty}^{\infty} f(t) \cdot f(t-\tau) \cdot dt \quad ...(v)$$

where $\overline{f(t)}$ represents the complex conjugate of f(t).

Note that the parameter t in the integral is a dummy variable.

3.3.11 Auto-correlation of Discrete-Time Signal

The discrete autocorrelation R at lag 'l' for a discrete-time signal y (n) is

$$R_{yy}(l) = \sum_{n \in Z} y(n)\,\overline{y(n-l)} \quad ...(vi)$$

The above definition works for signals that are square integrable, or square summable, that is, of **finite energy**.

For 'wide-sense-stationary random processes' the 'autocorrelations' are defined as

$$R_{ff}(\tau) = E\left[f(t)\,\overline{f(t-\tau)}\right]$$

and $$R_{yy}(l) = E\left[y(n)\,\overline{y(n-l)}\right]$$

For processes that are not stationary, these will also be functions of t or n.

For processes that are also 'ergodic', the expectation can be replaced by the limit of a time-average.

The autocorrelation of an 'ergodic process' is sometimes defined as :

$$R_{ff}(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_0^T f(t+\tau) \cdot \overline{f(t)} \cdot dt$$

and $$R_{yy}(l) = \lim_{N \to \infty} \frac{1}{N} \sum_{n=0}^{N-1} y(n)\,\overline{y(n-l)} \cdot$$

These definitions give well-defined single-parameter results for periodic functions, even when those functions are not the output of 'stationary ergodic processes'.

3.3.12 Definition for Periodic Signals

If f is a continuous periodic function of period T, the integration from $-\infty$ to ∞ is replaced by integration over any interval $[t_0, t_0 + T]$ of length T.

$$R_{ff}(\tau) \doteq \int_{t_0}^{t_0+T} f(t+\tau) \cdot \overline{f(t)} \cdot dt$$

and is equivalent to

$$R_{ff}(\tau) = \int_{t_0}^{t_0+T} f(t) \cdot \overline{f(t-\tau)} \cdot dt$$

3.3.13 Properties

We mention below properties of one-dimensional auto-correlations. These properties also hold for 'wide-sense stationary processes'.

(1) A fundamental property of the auto-correlation is symmetry,

$$R_{ff}(\tau) = R_{ff}(-\tau)$$

(i) When f is a real function, then

$$R_{ff}(-\tau) = R_{ff}(\tau)$$

(ii) When f is a complex function,

$R_{ff}(-\tau) = R_{ff}^{*}(\tau)$, the autocorrelation is a Hermitian function

(2) For any delay τ,

$$| R_{ff}(\tau) | \leq R_{ff}(0),$$

i.e. the continuous auto-correlation function reaches its peak at the origin, where it takes a real value.

The same result holds in the discrete case.

(3) The autocorrelation of a 'periodic function' is, itself, periodic with the same period.

(4) The autocorrelation of the sum of two completely uncorrelated functions is the sum of the autocorrelations of each function separately.

(5) Since autocorrelation is a specific type of 'cross-correlation', it maintains all the properties of cross-correlation.

3.3.14 Interrupted Time Series Analysis

- Often the goal of research with single subjects or units is to determine the efficacy of a specific intervention. This can be accomplished by employing various techniques that fall under the nomenclature of interrupted time series analysis.
- A simple example of an interrupted time series analysis is a design that involves repeated and equally spaced observations on a single subject or unit followed by an intervention.
- The intervention would then be followed by additional repeated and equally spaced observations of the subject or unit.
- The intervention could be an experimental manipulation such as a smoking cessation intervention for adolescents, or it could be a naturally occurring event such as a national change in the law regulating tobacco advertising.
- In order to determine if the intervention had an effect, an analysis of the data series would first necessitate some preprocessing of the data series to remove the effects of dependence.
- In addition to the traditional data transformation method, several alternative procedures for removing dependency in the data will also be described below.
- The actual statistical analysis used in an interrupted time series analysis employs a general linear model analysis using a generalized least squares or Aitken estimator.
- If the intervention effect is found to be statistically significant, an important and related question concerns an evaluation of the nature of the effect.

- One of the great advantages of time series analysis is the ability to assess the pattern of the change over time, which can involve both change in the mean level of a measured dependent variable and/or change in the slope over time of the dependent variable.
- We will present the most common variant forms of change over time and the methodology to evaluate these forms of change within this section.

3.3.14(A)Box-Jenkins Intervention Analysis

GQ. Explain Box-Jenkins Intervention Analysis.

- The most common methodology employed to examine the effects of a specific interrupted time series intervention is the Box-Jenkins procedure. This methodology is described in detail by Glass et al. (1975) and utilizes a two-step process.
- As described in the previous section, the autocorrelations and partial autocorrelations are calculated for various lags and this information is used for identification of the specific ARIMA (p, d, q) model parameter values.
- Accurate model identification is necessary to determine the specific transformation matrix to be used to remove the dependency from the data series so that it meets the assumptions of the general linear model.
- The remainder of this section, and parts of the next two sections, employ some matrix algebra to enhance the discussion of this and some other key aspects of time series analysis within the context of the general linear model.
- The general linear model is the general analytic procedure that includes the statistical techniques of multiple regression, analysis of variance, and analysis of covariance as special cases.
- After transforming the data series to remove the dependency in the data, the analysis follows standard estimation and testing procedures, and can be analyzed with a modified general linear model program in which the parameters of interest estimated and tested for significance.
- Several variations on the procedure of choosing a data transformation matrix have been proposed to eliminate the problematic model identification step, and will be described later in this section.
- A basic interrupted time series problem would be to determine if the average level of the series has changed as a result of the intervention. In such an analysis two parameters are estimated: L, the level af the series, and DL, the change in level after intervention. A test of significance would then examine the hypothesis of prime interest, H_o : DL = 0. In algebraic terms this can be expressed in terms of the general linear model as

(1) $Z = X b + a$

where Z is an N × I vector of observed variables, such that N is the total number of observations, with the first z_i observations occurring prior to the intervention, or

(2) $$Z = \begin{bmatrix} z_1 \\ z_2 \\ . \\ . \\ z_i \\ . \\ . \\ z_N \end{bmatrix}$$

and X is an N × p design matrix (see Table 3.3.1, described below, for examples), where p is the number of parameters estimated, b is the p × 1 vector of parameters, or

(3) $$b = \begin{bmatrix} L \\ D_L \end{bmatrix}$$

And a is the N × 1 vector of residual, or

(4) $$a = \begin{bmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ a_i \\ \cdot \\ \cdot \\ a_N \end{bmatrix}$$

The general linear model is an approach to data analysis that includes many familiar statistical procedures as special cases. In a multiple regression analysis, the X matrix contains the numeric observations for each of the p predictor variables for the N subjects, the Z vector contains the criterion scores for the N subjects, the b vector contains the regression weights, and the a vector contains the error of prediction and represents the difference between the actual score on the criterion and the predicted score-on the criterion. In an analysis of variance, the X matrix would consist of indicator variables, such as the numeric values "1"or "0" which indicate group membership, and the Z vector contains the dependent variable observations.

For this example, the vector of parameters contains two components, namely L, and DL. This design matrix is presented as (A) in Table 3.3.1.

The usual least squares solution, which minimizes the sum of the squared errors, is

(5) $b = (X'X)^{-1} X'Z$

and a test of significance for the null hypothesis $H_o : b_i = 0$ (i.e., $H_o : DL = 0$) is given by

(6) $t_{bi} = b_i / s_{bi}$

Where

(7) $s_{bi}^2 = s_a^2 C^{ii}$

and s_a^2 is the estimate of the error variance and C^{ii} is the ith diagonal element of $(X'X)^{-1}$. The test statistic would have a t distribution with degrees of freedom N – p. This is the same test of significance that is used for testing if the regression weight for a predictor is significant in multiple regression.

Table 3.3.1 : Examples of Common Design Matrices for Single Unit Analysis ($N_1 = N_2 = 6$)

(A) Immediate and constant changes in level

1	0
1	0
1	0
1	0
1	0
1	0
1	1
1	1
1	1
1	1
1	1
1	1

(B) Immediate and constant changes in level and slope

1	0	1	0
1	0	2	0
1	0	3	0
1	0	4	0
1	0	5	0
1	0	6	0
1	0	7	1
1	1	8	2
1	1	9	3
1	1	10	4
1	1	11	5
1	1	12	6

(C) Decaying change in level

1	0
1	0
1	0
1	0
1	0
1	0
1	1
1	.5
1	.25
1	.13
1	.07
1	.03

(D) delayed change in level

1	0
1	0
1	0
1	0
1	0
1	0
1	0
1	0
1	0
1	1
1	1
1	1

3.3.14(B)Transformation of Time Series Data

The general linear model cannot be directly applied to time series analysis because of the presence of dependency in the residuals. It is necessary to perform a transformation on the observed variable, Z_t, to remove dependency, prior to the statistical analysis. A transformation matrix T must be found, yielding

[8] Y=TZ,

and

[9] $X^* = TX$

The purpose of the model identification step is to determine the appropriate transformation of Z into Y. Table 3.3.2 presents mathematical descriptions and relevant comments

Table 3.3.2 : Common ARIMA Models

Label	(p, d, q)	Descriptive formula	Comment
White noise	(0, 0, 0)	$Z_t = L + a_t$	No dependency in the data
Autoregressive order one	(1, 0, 0)	$Z_t - L = \phi_1 (Z_{t-1} - L) + a_t$	Predicated from previous observations
Autoregressive order one	(2, 0, 0)	$Z_t - L = \phi_1 (Z_{t-1} - L) + \phi_2 (Z_{t-2} - L) + a_t$	Predicated from previous two observations
Moving averages order one	(0, 0, 1)	$Z_t - L = a_t - \theta_1 a_{t-1}$	Proportion of previous shock affect observations
Moving averages order two	(0, 0, 2)	$Z_t - L = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2}$	Proportion of two previous shocks affect observations
Integrated moving averages	(0, 1, 1)	$Z_t - Z_{t-1} = a_t - \theta_1 a_{t-1}$	Stochastic drift and proportion of previous shock affect observations

- On six commonly identified ARIMA models. After model identification, an estimation procedure is employed to determine the specific numeric values of ϕ and 0 which will be used in the appropriate transformation matrix.
- The particular ARIMA (p, d, q) model will determine the specific content of the transformation matrix T. Because the correction for dependency involves previous observations, all transformation matrices will have a similar form, a lower triangular matrix. For example, an ARIMA(1, 0, 0) model with five observations would have the following transformation matrix

$$(10)\ T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \phi_1 & 1 & 0 & 0 & 0 \\ 0 & \phi_1 & 1 & 0 & 0 \\ 0 & 0 & \phi_1 & 1 & 0 \\ 0 & 0 & 0 & \phi_1 & 1 \end{bmatrix}$$

that indicates that only the previous observation is necessary to explain the dependency in the data. For an ARIMA (2, 0, 0) model with five observations, the transformation matrix would be

$$(11)\ T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \phi_1 & 1 & 0 & 0 & 0 \\ \phi_2 & \phi_1 & 1 & 0 & 0 \\ 0 & \phi_2 & \phi_1 & 1 & 0 \\ 0 & 0 & \phi_2 & \phi_1 & 1 \end{bmatrix}$$

which indicates that the previous two observations are necessary to explain the dependency in the data. Glass, Willson, and Gottman (1975) present an inductive derivation of the necessary transformation for these two models and other common models.

Given T, the estimate of the parameters, b, may be expressed as a generalized least squares problem, i.e

(12) $b = (X'T'TX)^{-1}XT'T'Z = (X^{*'}X^{*})^{-1}X^{*'}Y$

3.3.14(C) Parameters of Interest

- For an interrupted time series analysis, there are typically four parameters of interest, the Level of the series (L), the Slope of the series (S), the Change in Level (DL), and the Change in Slope (DS). The slope parameters represent one of the other unique characteristics of a longitudinal design, the pattern of change over time.
- Investigating the pattern of change over time represents one of the real advantages of employing a longitudinal design.
- Fig. 3.3.3 Illustrates eight different outcomes for a simple one-intervention design. In a typical experimental design only one follow-up assessment occurs after treatment.
- By inspecting the different patterns of change over time, we can see that selecting different points in time for the single assessment would result in very different conclusions for four of the examples (C, F, G, and H). For example, ignoring the slope in C would lead the researcher to incorrectly conclude that the intervention was effective.
- The evolutionary effect (H) is a good example of where the intervention results in a temporary negative effect, perhaps while a response pattern is unlearned, followed by a positive effect.
- An early assessment would conclude that the treatment had a negative effect; a somewhat later assessment would find no treatment effect, while an even later assessment would find a positive treatment effect.
- Alternative specifications of the design matrix permit the investigation of different hypotheses concerning the nature of the intervention. Table 3.3.1 presents some illustrative examples for an N = 12 ($n_1 = n_2 = 6$) case. Only changes in level and slope parameters are presented in Table 3.3.1 because these are the most commonly examined effects in interrupted time series designs.
- It should also be noted that other representations for specific design matrices have been presented for investigating these parameters. Huitema and McKean (2000) present a detailed discussion of some of the issues related to design specification for the analysis of interventions in time series.
- As noted earlier, Table 3.3.1(A) is the design matrix for an immediate and constant treatment effect that tests for a change in the level of the data series. Table 3.3.1(B) is the design matrix for testing both a change in level and a change in slope.
- Table 3.3.1(C) is the design matrix for examining a decaying treatment effect. Table 3.3.1(D) is the design matrix for testing a delayed treatment effect. In addition to the designs presented in Table 3.3.1, alternative time series designs can provide an opportunity to examine additional change parameters that may be impacted by the intervention (e.g., changes in cycles, variance, and pattern or serial dependency).
- Although less common, such alternative applications can help to more fully elucidate the nature of the effects of an intervention.
- Although it is the most prevalent time series methodology, the Box-Jenkins approach to intervention analysis suffers from a number of difficulties. First gathering the number of data points required for accurate model identification is often prohibitive for research in applied settings. Second, even with the required number of points in hand, correct identification is problematic (Velicer & Harrop, 1983).
- Third, the method is complex, making applications by the mathematically unsophisticated researcher difficult. Three alternative approaches are described in the next section, all of which attempt to avoid the problematic model identification step.

3.3.14(D) Alternative Approaches

- Simonton (1977) proposed a procedure that avoids the problem of model identification by using an estimate of the variance-covariance matrix based on pooling the observations across all subjects observed. This approach also requires a basic assumption, namely that all series are assumed to be an ARIMA (1, 0, 0) model.
- While this assumption seems to be theoretically indefensible, empirical investigations indicate that this procedure works well in a wide variety of cases (Harrop & Velicer, 1985).

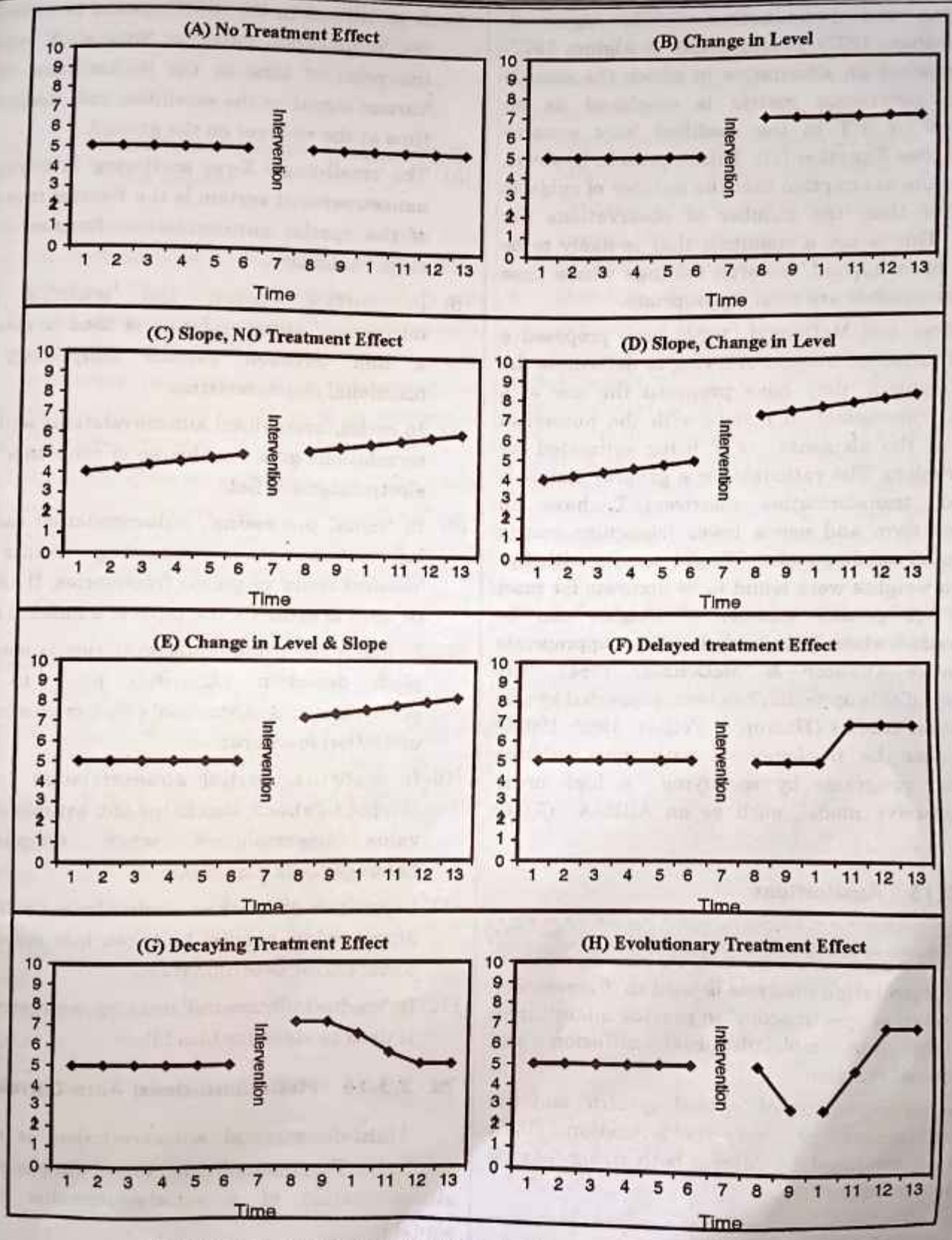


Fig. 3.3.3 : Examples of eight different patterns of intervention effects

Algina and Swaminathan (1979; Algina & Swaminathan, 1977; Swaminathan & Algina, 1977) have proposed an alternative in which the sample variance covariance matrix is employed as an estimator for T'T in the modified least squares solution (See Equation [7]). This approach, however, requires the assumption that the number of subjects is greater than the number of observations per subject. This is not a condition that is likely to be met in most applied research settings where time series approaches are most appropriate.

Velicer and McDonald (1984) have proposed a third alternative. Instead of trying to determine the specific matrix, they have proposed the use of a general transformation matrix with the numerical values of the elements of T being estimated for each problem. The rationale for a general matrix is that all transformation matrices, T, have an identical form and use a lower triangular matrix with equal subdiagonals. Weight vectors with five nonzero weights were found to be accurate for most cases. A greater number of weights can be employed where indicated by appropriate diagnostics (Velicer & McDonald, 1984). The accuracy of this approach has been supported by two simulation studies (Harrop & Velicer, 1985; 1990b) and it can be implemented with most existing computer programs by specifying a high order autoregressive model, such as an ARIMA (5,0,0) model.

3.3.15 Applications

GQ. Discuss application of autocorrelation.

(1) Autocorrelation analysis is used in 'fluorescence correlation spectroscopy' to provide quantitative insight into molecular-level diffusion and chemical reaction.

(2) The measurement of 'optical spectra' and the measurement of very-short-duration 'light pulses' produced by lasers, both using 'optical autocorrelators'.

(3) Autocorrelation is used to analyse 'dynamic light scattering' data, which determines the particle size distribution of nanometer-sized particles in a fluid.

(4) It is utilised in the 'GPS' system to correct for the 'propagation delay' or 'time shift' between the point of time at the transmission of the 'carrier signal' at the satellites, and the point of time at the receiver on the ground.

(5) The 'small-angle X-ray scattering' intensity of a nanostructured system is the Fourier transform of the spatial autocorrelation function of the electron density.

(6) In 'surface science' and 'scanning probe microscopy' autocorrelation is used to establish a link between surface morphology and functional characteristics.

(7) In optics, normalised autocorrelations and cross correlations give the 'degree of coherence' of an electromagnetic field.

(8) In 'signal processing', autocorrelation can give information about repeating events like 'musical beats' or pulsar frequencies. It can also be used to estimate the pitch of a musical note.

(9) In music recording, autocorrelation is used as a pitch detection algorithm prior to vocal processing as a 'distortion' effect or to eliminate undesired inaccuracies.

(10) In statistics, spatial autocorrelation between sample locations also helps one estimate 'mean value uncertainties' when sampling a heterogeneous population.

(11) In analysis of 'Markov Chain Monte Carlo' data, autocorrelation must be taken into account for correct error determination.

(12) In 'medical ultrasound' imaging, autocorrelation is used to visualize blood flow.

3.3.16 Multi-dimensional Auto-Correlation

'Multi-dimensional autocorrelation' is defined similarly. For example, in three dimensions the autocorrelation of a square-summable discrete signal is

$$R(j, k, l) = \sum_{n, q, r} x_{n, q, r} \overline{x}_{n-j, q-k, r-l}$$

When we subtract mean values from signals before computing autocorrelation function, the resulting function is usually called an auto-covariance function.

3.3.17 Univariate Versus Multivariate Time Series Models

Time series models may also be split into univariate time series models and multivariate time series models. Univariate time series models are models used when the dependent variable is a single time series. Trying to model an individual's heart rate per minute using only past observations of heart rate and exogenous variables is an example of a univariate time series model.

Multivariate time series models are used when there are multiple dependent variables. In addition to depending on their own past values, each series may depend on past and present values of the other series. Modeling U.S. gross domestic product, inflation, and unemployment together as endogenous variables is an example of a multivariate time series model.

Univariate Model Examples	Multivariate Model Examples
Univariate Generalized autoregressive conditional heteroscedasticity (GARCH).	Vector Autoregressive Models (VAR).
Seasonal Autoregressive Integrated Moving Average (SARIMA) Models.	Vector Error Correction Model (VECM).
Univariate unit root tests.	Multivariate unit root tests.

3.3.18 Linear Versus Nonlinear Time Series Models

When structural breaks are present in time series data they can diminish the reliability of time series models that assume the model is constant over time. For this reason, special models must be used to deal with the nonlinearities that structural breaks introduce.

Nonlinear time series analysis focuses on :

- Identifying the presence of structural breaks;
- Estimating the timing of structural breaks;
- Testing for unit roots in the presence of structural breaks;
- Modeling data behavior before, after, and between breaks.

There are different types of nonlinear time series models built around the different nature and characteristics of the nonlinearities. For example, the threshold autoregressive model assumes that jumps in the dependent data are triggered when a threshold variable reaches a specified level. Conversely, Markov-Switching models assume that an underlying stochastic Markov chain drives regime changes.

3.3.19 Real-world Examples of Structural Breaks in Time Series Data

Example	Description
Housing market prices	The concept of a housing bubble gained notoriety after the global financial crash of 2008. Since then much research and theoretical work is being done to identify and predict housing bubbles.
S&P 500 Unconditional Variance	Modeling stock price volatility is crucial to managing financial portfolios. Because of this, much attention is directed towards understanding the underlying behavior of market indicators like the S & P 500.
Global temperatures	Identifying structural breaks in global temperatures has provided support to proponents of global climate change.

3.3.20 Computation

To calculate the autocorrelation of the real signal sequence :

$x = (2, 3, -1)$; i.e. $x_0 = 2$, $x_1 = 3$, $x_2 = -1$ and $x_i = 0$ for other values of i.

We use 'brute-force' method based on signal processing definition :

$$R_{xx}(j) = \sum_n x_n \cdot \overline{x}_{n-j};$$

Note that the definition is just same as the 'usual' multiplication : (but with right shifts); and each vertical addition gives the autocorrelation for particular lag values :

$$\begin{array}{rrrrrr} & 2 & 3 & -1 & & \\ \times & 2 & 3 & -1 & & \\ \hline & -2 & -3 & 1 & & \\ + & & 6 & 9 & -3 & \\ & & & 4 & 6 & -2 \\ \hline & -2 & 3 & 14 & 3 & -2 \end{array}$$

Hence the required autocorrelation sequence is

$$R_{xx} = (-2\ 3\ 14\ 3\ -2); \text{ here}$$

$R_{xx}(0) = 14$, $R_{xx}(-1) = R_{xx}(1) = 3$ and

$R_{xx}(-2) = R_{xx}(2) = -2$

[$\because$ R_{xx} is even function]

Remark : If the signal is given as :

$$x = (..., 2, 3, -1, 2, 3, -1, ...),$$

the sequence is periodic, and we get circular autocorrelation as

$$R_{xx} = (..., 14, 1, 1, 14, 1, 1, ...)$$

and has the same period as the signal sequence of x.

3.3.21 Cross-lagged Correlations

- Time series analysis on a single dependent measure involves many of the procedures common to multivariate statistics because two vectors of unknowns must be estimated simultaneously: the vector of parameters and the vector of coefficients that represent the dependency in the data. However, when assessing a single unit or subject on multiple occasions, two or more variables can be observed on each occasion.
- The term multivariate time series will be used here to denote the observation of more than one variable at each point in time.
- The variables may be viewed conceptually as including both dependent and independent variables or just dependent variables.
- If some of the observed variables are appropriately viewed as independent variables, the appropriate analysis is the time series equivalent of an analysis of covariance.
- If the variables can be viewed as a set of dependent variables, i. e., multiple indicators of one or more constructs that form the outcome space of interest, the appropriate analysis would be the time series equivalent of a multivariate analysis, sometimes described as a dynamic factor analysis. The next two sections will discuss these two approaches in detail.
- One of the unique aspects of any time series analysis involving multiple variables observed on each occasion involves the extension of the correlation coefficient.
- The cross- lagged correlation coefficient for lag = 0 is calculated the same way as the pair wise correlation coefficient, using the number of observations over time in place of the number of subjects as the basis.
- The term lag refers to the time relationship between the two variables. Lag zero means that the observation at time t on Zi is matched with the observation at time t on Zj.
- However, the appropriate relationship between the variables may involve one variable at time t and the other variable at time t - 1; that is, there may be a delay between a change in one variable and the associated change in the other variable.

- If Zi lags Zj, the maximum correlation would occur between Zi at time at time t + 1 and Zj at time t. Alternatively, Zi could lead Zj, producing the maximum correlation between Zi at time t - 1 and Zj at time t.
- A critical decision for any multivariate time series analysis is determining the appropriate lag between the set of observed variables.
- There are generally three alternative methods. First, the lag could be determined on the basis of theory. In some areas, well-established theoretical models exist like the supply and demand models in economics. Second, the lag could be determined on the basis of previous empirical findings.
- If a set of variables has been extensively investigated, the accumulated empirical evidence could serve as a guide to the appropriate order of the lag.
- Third, the appropriate lag could be estimated as part of the model estimation procedure. This would involve calculating the cross-lagged correlations for a reasonable set of lags, e. g., from +5 to –5.
- The lag that produces the highest numeric value for the correlation would be assumed to be the appropriate lag.

3.4 PLOTTING THE PARTIAL AUTOCORRELATION FUNCTION

I cover both the autocorrelation function and partial autocorrelation function. We shall learn about the differences between these functions and what they can tell us about our data. In later posts, I'll show you how to incorporate this information in regression models of time series data and other time-series analyses.

3.4.1 Autocorrelation and Partial Autocorrelation Basics

- Autocorrelation is the correlation between two values in a time series. In other words, the time series data correlate with themselves - hence, the name. We talk about these correlations using the term "lags."
- Analysts record time-series data by measuring a characteristic at evenly spaced intervals - such as daily, monthly, or yearly.
- The number of intervals between the two observations is the lag. For example, the lag between the current and previous observation is one. If you go back one more interval, the lag is two, and so on.
- In mathematical terms, the observations at yt and yt–k are separated by k time units. K is the lag.
- This lag can be days, quarters, or years depending on the nature of the data. When k=1, we are assessing adjacent observations. For each lag, there is a correlation.

3.4.2 Autocorrelation Function (ACF)

- Use the autocorrelation function (ACF) to identify which lags have significant correlations, understand the patterns and properties of the time series, and then use that information to model the time series data.
- From the ACF, you can assess the randomness and stationarity of a time series. You can also determine whether trends and seasonal patterns are present.
- In an ACF plot, each bar represents the size and direction of the correlation. Bars that extend across the red line are statistically significant.
- In 'time series analysis', the 'partial autocorrelation function' (PACF) gives the 'partial correlation' of a 'stationary time series' with its own lagged values, and they are regressed by values of the time series at all shorter lags.
- It contrasts with the 'autocorrelation function' which does not control for other lags.

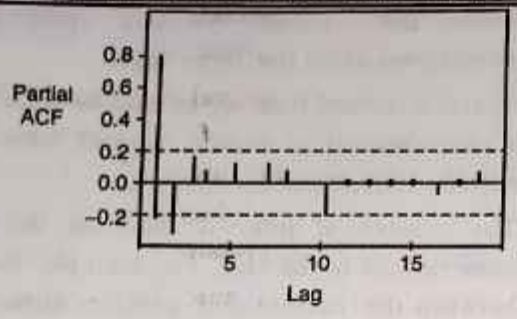
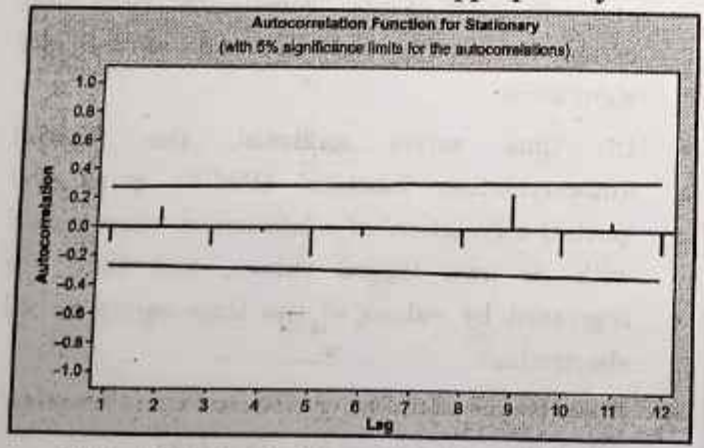


Fig. 3.4.1 : Lake Huron Level

- Partial autocorrelation function of Lake Huron's depth with confidence interval.
- This function plays an important role in data analysis. It is aimed at identifying the extent of the lag in an 'autoregressive (AR) model'.
- By plotting the autocorrelation functions one could determine the appropriate lags 'P' in an AR (P) model or in an extended ARIMA (p, d, q) model.

3.4.3 Randomness/White Noise

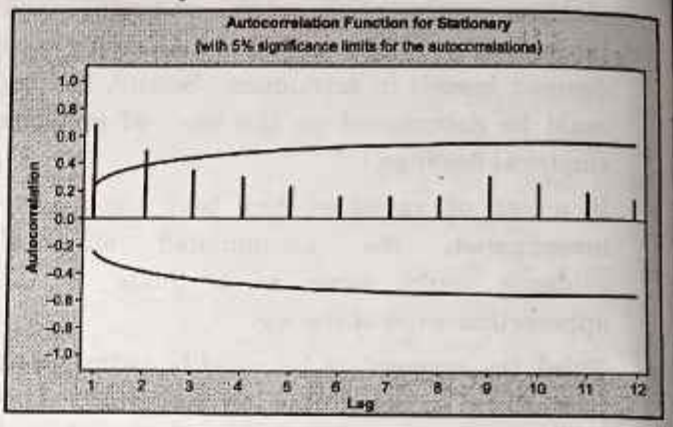
- For random data, autocorrelations should be near zero for all lags.
- Analysts also refer to this condition as white noise.
- Non-random data have at least one significant lag. When the data are not random, it's a good indication that you need to use a time series analysis or incorporate lags into a regression analysis to model the data appropriately.



This ACF plot indicates that these time series data are random.

3.4.4 Stationarity

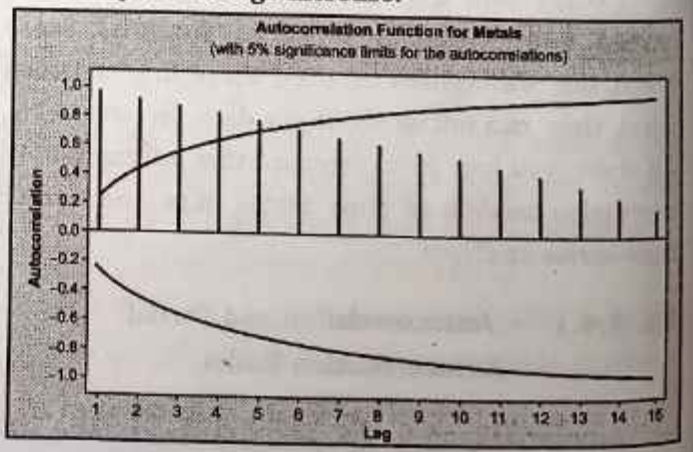
- Stationarity means that the time series does not have a trend, has a constant variance, a constant autocorrelation pattern, and no seasonal pattern.
- The autocorrelation function declines to near zero rapidly for a stationary time series. In contrast, the ACF drops slowly for a non-stationary time series.



In this chart for a stationary time series, notice how the autocorrelations decline to non- significant levels quickly.

3.4.5 Trends

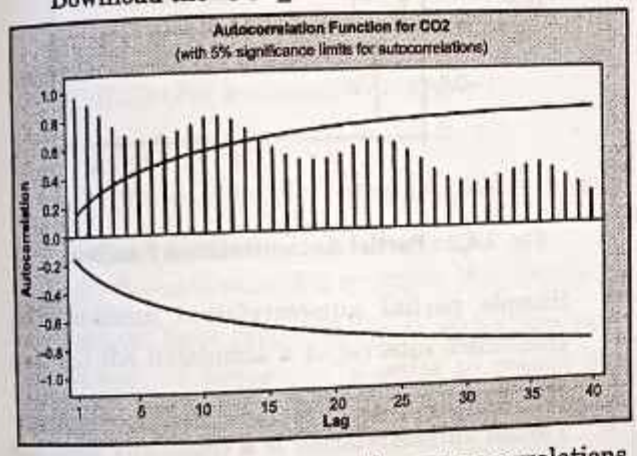
When trends are present in a time series, shorter lags typically have large positive correlations because observations closer in time tend to have similar values. The correlations taper off slowly as the lags increase.



In this ACF plot for metal sales, the autocorrelations decline slowly. The first five lags are significant.

3.4.6 Seasonality

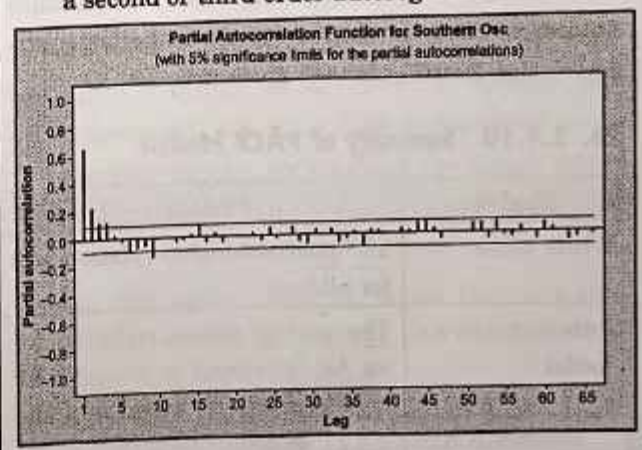
- When seasonal patterns are present, the autocorrelations are larger for lags at multiples of the seasonal frequency than for other lags.
- When a time series has both a trend and seasonality, the ACF plot displays a mixture of both effects.
- That's the case in the autocorrelation function plot for the carbon dioxide (CO2) dataset from NIST. This dataset contains monthly mean CO2 measurements at the Mauna Loa Observatory. Download the CO2_Data.



- Notice how you can see the wavy correlations for the seasonal pattern and the slowly diminishing lags of a trend.

3.4.7 Partial Autocorrelation Function (PACF)

- The partial autocorrelation function is similar to the ACF except that it displays only the correlation between two observations that the shorter lags between those observations do not explain.
- For example, the partial autocorrelation for lag 3 is only the correlation that lags 1 and 2 do not explain. In other words, the partial correlation for each lag is the unique correlation between those two observations after partialling out the intervening correlations.
- As you saw, the autocorrelation function helps assess the properties of a time series. In contrast, the partial autocorrelation function (PACF) is more useful during the specification process for an autoregressive model.
- Analysts use partial autocorrelation plots to specify regression models with time series data and Auto Regressive Integrated Moving Average (ARIMA) models. We shall focus on that aspect in posts about those methods.
- For this post, We shall show you a quick example of a PACF plot. Typically, you will use the ACF to determine whether an autoregressive model is appropriate. If it is, you then use the PACF to help you choose the model terms.
- This partial autocorrelation plot displays data from the southern oscillations dataset from NIST. The southern oscillations refer to changes in the barometric pressure near Tahiti that predicts El Niño.
- On the graph, the partial autocorrelations for lags 1 and 2 are statistically significant. The subsequent lags are nearly significant. Consequently, this PACF suggests fitting either a second or third-order autoregressive model.



3.4.8 Definition

Given a time series Z_t, the partial autocorrelation of lag k, denoted by $\phi_{k,\,k}$, is the autocorrelation between Z_t and Z_{t+k} with the linear dependence of Z_t on Z_{t+1} through Z_{t+k-1} removed.

Thus, $\phi_{1,\,1} = \text{corr}\,(Z_{t+1}, Z_t)$, for $k = 1$,

$$\phi_{k,\,k} = \text{corr}\left(Z_{t+k} - \hat{Z}_{t+k}, Z_t - \hat{Z}_t\right), \quad \text{for } k \geq 2,$$

where $\hat{Z}_{t+k}$ and $\hat{Z}_t$ are 'linear combinations' of $\{Z_{t+1}, Z_{t+2}, \ldots, Z_{t+k-1}\}$ and that minimize the 'mean squared error' of Z_{t+k} and Z_t respectively.

For 'stationary process', $\hat{Z}_{t+k}$ and $\hat{Z}_t$ are the same.

3.4.9 Calculation

The theoretical partial autocorrelation function of a stationary time series can be calculated by using Durbin-Levinson Algorithm.

$$\phi_{n,\,n} = \frac{\rho(n) - \sum_{k=1}^{n-1} \phi_{n-1,\,k} \cdot \rho(n-k)}{1 - \sum_{k=1}^{n-1} \phi_{n-1,\,k} \cdot \rho(k)}$$

where $\phi_{n,\,k} = \phi_{n-1,\,k} - \phi_{n,\,n}\ \phi_{n-1,\,n-k}$ for $1 \leq k \leq n-1$

and

$\rho(n)$ is the autocorrelation function.

The above formula can be used with sample autocorrelation, to find the sample partial auto-correlation function for any given time series.

3.4.10 Summary of PACF Models

Model	PACF
White noise	The partial auto-correlation is 0 for all lags.
Autoregressive model	The partial auto-correlation for an AR (p) model is non-zero for lags less than or equal to p and 0 for lags greater than p.
Moving average model	If $\phi_{1,1} < 0$, the partial autocorrelation geometrically decays to 0.
	If $\phi_{1,1} > 0$, the partial autocorrelation oscillates to 0.
Autoregressive moving average model	An ARMA (p, q) model's partial autocorrelation geometrically decays to 0 but only after lags greater than ρ.

The behavior of the partial auto-correlation function mirrors that of the auto-correlation function for autoregressive and moving-average models.

3.4.11 Autoregressive Model Identification

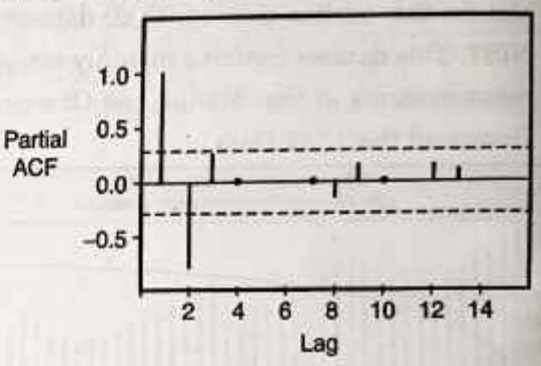


Fig. 3.4.2 : Partial Autocorrelation Function

- Sample partial autocorrelation function with confidence interval of a simulated AR (3) time series.
- Partial autocorrelation is a commonly used tool for identify the order of an autoregressive model.
- The partial auto-correlation of an AR (p) process is zero at lags greater than p. If an AR model is appropriate, then sample partial auto-correlation plot is examined to help identify the order.
- The partial autocorrelation of lags greater than p for an AR (p) time series are approximately independent and normal with mean O.
- Hence, a confidence interval can be constructed by dividing a selected Z-score by $\sqrt{n}$.
- Lags with partial autocorrelations outside of the confidence interval indicate that the AR model's order is likely greater than or equal to the lag.

3.5 FITTING AN ARIMA MODEL, RUNNING DIAGNOSTICS ON AN AIRMA MODEL

3.5.1 Time-Series Forecasting

GQ. Explain : Time-series forecasting.

- A sequence of recording a metric over the constant time intervals is known as **Time Series.**
- Based on the frequency, a Time series can be classified into the following categories :
  - (i) Yearly (For example, Annual Budget),
  - (ii) Quarterly (For example, expenses)
  - (iii) Monthly (For example, Air traffic)
  - (iv) Weekly (For example, Sale Quantity)
  - (v) Daily (For instance, Weather)
  - (vi) Hourly (For example, Stock Prices)
  - (vii) Minutes-wise (For example, inbound calls in a call center)
  - (viii) Seconds-wise (For example, Web Traffic)
- Once we have prepared Times-Series Analysis, we have to forecast it in order to predict the future values that the series will be going to take.
- There is a need for forecasting. Since forecasting a Time Series, such as Sales and Demand, is often of incredible commercial value, which increases the need for forecasting.
- Time-Series forecasting is generally used in many manufacturing companies as it drives the primary business planning, procurement and production activities.
- Any forecast's error will undulate throughout the chain of the supply or any business framework, for that stuff. Thus it is significant, in order to get accurate predictions saving the costs and is critical to success.
- The concepts and technologies behind Times Series forecasting can also be applied in any business, including manufacturing.

3.5.2 Classification of Time Series Forecasting

The Time Series Forecasting can be broadly Classified into two categories :

(1) Univariate time series forecasting

The univariate time series forecasting is a forecasting of time series where we utilise the former values of time series only in order to guess the forthcoming values.

(2) Multi-variate time series forecasting

The multi-variate time series forecasting is a forecasting of time series where we utilise the predictors other than the series, also known as exogenous variables, in order to forecast.

3.5.3 ARIMA Modeling

GQ. Explain in detail ARIMA modeling.

- Auto Regressive Integrated Moving Average, abbreviated as ARIMA, is an algorithm for forecasting that is centered on the concept that the data in the previous values of the time series can alone be utilized in order to predict the future values.
- Again, ARIMA is a class of models that 'demonstrates' a given time series based on its previous values : its lags and the lagged errors in forecasting, so that equation can be utilized in order to forecast future values.
- We can model any time series that are non-seasons exhibiting patterns and not a, random white noise with 'ARIMA models'.

3.5.4 'p', 'q' and 'd' in ARIMA Models

There are three terms that characterise an ARIMA model :

They are p, q and d;

where : p = the order of the AR term,
q = the order of the MA term,
d = the number of differences required to make the time series stationary.

If a time series has seasonal patterns, we have to insert seasonal periods, and it becomes 'SARIMA', short for 'Seasonal ARIMA'.

3.5.5 The Roles of 'p', 'q' and 'd' in the ARIMA Model

- The primary step is to 'make the times series stationary' in order to build an ARIMA model. This is because the term 'Auto Regressive' in ARIMA implies a Linear Regression Model using its lags as predictors.
- We note that Linear Regression Models work well for independent and non-correlated predictors.
- In order to make a series stationary, we utilize the most common approach that is to subtract the past value from the present value.
- Sometimes, depending upon the series complexity, multiple subtractions may be required.
- Hence, the value of d is the minimum number of subtractions required to make the series stationary. And if the time series is already stationary, then 'd' becomes zero.
- We now try to understand the terms 'p' and 'q'.
- The 'p' is the order of the 'AR' (Auto-Regressive) term, which means that the number of lags of Y to be utilised as predictors. And at the same time 'q' is the order of the 'MA' (Moving Average) term, which means that the number of lagged forecast errors should be used in the ARIMA model.

3.5.6 Auto-Regressive (AR) and Moving Average (MA) Models

We discuss the AR and MA models and the actual mathematical formulae for these models :

(I) A pure AR (Auto-regressive only) Model is a model which relies only on its own lags. Hence, we can also conclude that it is a function of the 'lags of Y_i'; i.e.,

$$Y_i = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots\ldots + \beta_p Y_{t-p} + \epsilon_t$$

where Y_{t-1} is the lag 1 of the series and β_1 is the coefficient of lag 1 and is the term of intercept that is calculated by the model.

(II) Similarly, a Pure MA (Moving Average only) model is a model where Y_t relies only on the lagged predicted errors.

$$Y_t = \alpha + \epsilon_1 + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + \ldots\ldots + \phi_q \epsilon_{t-q}$$

where the error terms are the AR models errors of the corresponding lags.

There errors ϵ_t and ϵ_{t-1} are the errors from the equations given below :

$$Y_t = \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots + \beta_0 Y_0 + \epsilon_t$$

$$Y_{t-1} = \beta_2 Y_{t-2} + \beta_3 Y_{t-3} + \ldots + \beta_0 Y_0 + \epsilon_{t-1}$$

Here we conclude Auto-Regressive (AR) and Moving Average (MA) models.

3.5.7 Equation of an ARIMA Model

An ARIMA model is a model where the series of time was subtracted at least once in order to make it stationary, and we combine the Auto-Regressive (AR) and the Moving Average (MA) terms. Hence, we get the following equation :

$$Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \ldots + \beta_p Y_{t-p} + \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + \ldots + \phi_q \epsilon_{t-q}$$

We forecast Y_t (i.e. ARIMA Model in words).

Forecasted Y_t = constant + Linear Combination Lags of Y (up to p lags) + Linear Combination of Lagged Predicted Errors (up to q lags).

Now, our aim to find the values of p, q and d.

First we begin with finding the 'd' in the ARIMA Model.

3.5.8 Finding the Order of Differencing 'd' in the ARIMA Model

The main purpose (or initial purpose) of differencing in the ARIMA model is to make the Time-Series stationary.

But we have to take care of not over-differencing the series as an over-differenced the series may also be stationary, and that will affect the parameter of the model. Now, we make clear the appropriate differencing order :

- The most appropriate differencing order is the minimum differencing needed in order to achieve an almost stationary series roaming around a defined mean and the ACF reaching zero relatively faster.
- In case the autocorrelations are positive for multiple lags (generally ten or more, the series requires further differencing. In contrast, if lag 1 auto correlated itself pretty negatively, then the series is positively over-differenced).
- In cases, where we cannot decide between two differencing orders, then we choose the order providing the minor standard deviation in the differenced series.
- We consider an example to check if the series is stationary. We use the Augmented Dickey-Fuller Test (ad fuller ( )) from the stats models package of 'Python Programming Language'.

Example

```
from statsmodels . tsa . stat tools import adfuller
from numpy import log
import pandas as pd
mydata = pd . read – SV ('mydata set . 'SV', names)
                =['value'], header = 0)
res = adfuller (mydata . value . dropna ())
print (Augmented Dickey-Fuller Statistic : % f' % res [0])
print ('P-value : % f' % res [1])
```

Output

```
Augmented Dickey-Fuller Statistic : – 2.464240
P-value : 0.124419
```

Explanation

- In the above example, we have imported the 'adfuller' module along with the 'numpy's' log module and 'pandas'.
- We have then used the 'pandas' library to read the CSV file. We have then used the 'adfuller' method and printed the values to the user.
- It is necessary to check whether the series is stationary or not. If not, we have to use difference, else, 'd' becomes 'zero'.
- The 'Augmented Dickey-Fuller (ADF)' test's null hypothesis is that the time series is not stationary.
- Thus, if the ADF test's p-value is less than the significance level (0.05), then we will reject the null hypothesis and infer that the time series is definitely stationary.
- We can observe that the p-value is more significant than the level of significance. Hence, we can difference the series and check the plot of autocorrelation as shown below :

Example

```
import numpy as np, pandas as pd
from statsmodels graphics.tsaplots import plot_acf, plot_pacf
import matplotlib.pyplot as plt
plt.rcParams.update ({'figure.figsize' : (9, 7),
                'figure.dpi'; 120})
#importing data
df = pd.read-CSV ('mydataset.CSV', names =
                ['value'], header = 0)
#The Genuine Series
  fig, axes = plf.subplots (3, 2 share x = True)
axes [0, 0] plot (df.value); axes [0, 0] - set_title
                ('The Genuine Series')
plot_acf (df. value, ax = axes [0, 1])
# order of differencing : First
axes [1, 0] - plot (df. value.diff ( )); axes [1, 0]
                set_title ('Order of Differencing : First')
plot_acf (df.value.diff ( ) . dropna( ), ax = axes [1, 1])
#order of Differencing: Second
axes [2, 0] - plot (df.value.diff( )) - diff( ));
    axes [2, 0] - set-title ('Order of Differencing : Second')
plot_acf (df-value - diff( ) - diff( ))- dropna ( ), ax = axes [2, ])
plt.show( )
```

☞ **Output**

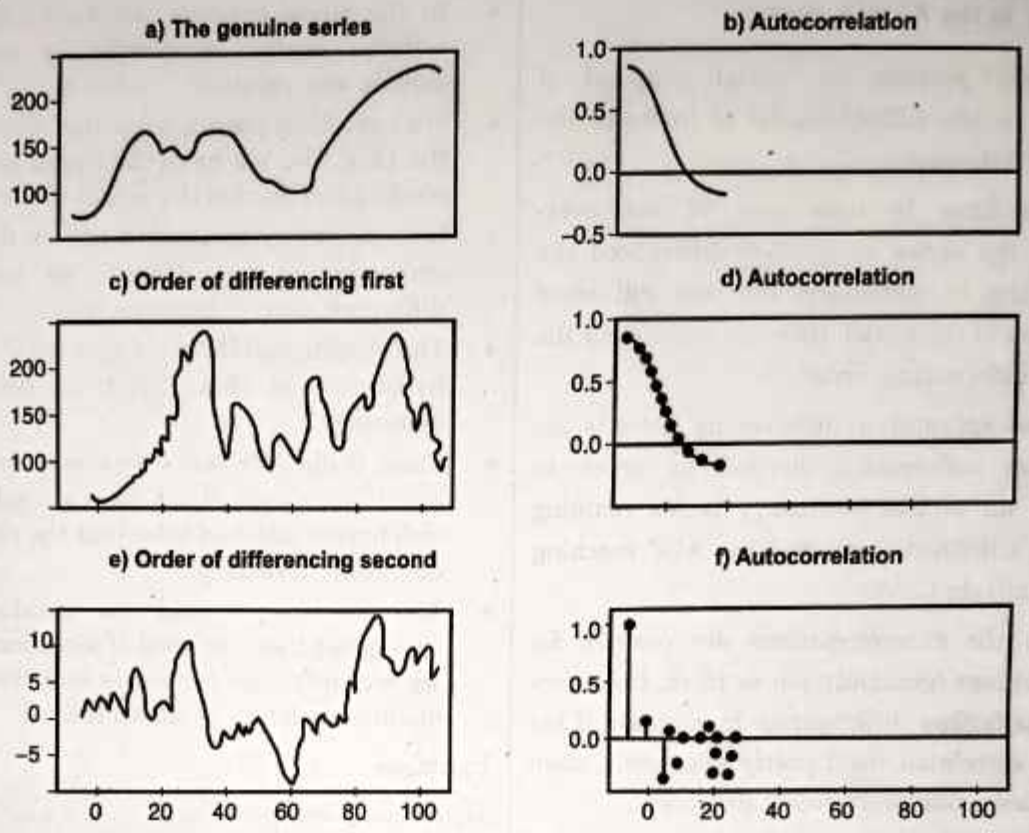


Fig. 3.5.1

Explanation

In this example, we have imported the required libraries and modules. Importing the data, we plotted different graphs. There are three graphs.

(i) Original series graph,

(ii) First order differencing and

(iii) Second order differencing and along with them autocorrelation graphs.

We note that time series has reached stationarity with two differencing orders. But at the plot of autocorrelation of 'second order of differencing' the lag goes far negative zone pretty faster, indicating the series might have been over differenced.

Thus we conclude that the series is not properly stationary, or the series has weak stationarity.

☞ **Example**

```
From pmdarima • arima • utils import ndiffs
import pandas as pd
df = pd • read – CSV
('my dataset • CSV', names = ['value'], header = 0)
X = df • value
# Augmented Dickey Fuller Test
adftest = ndiffs (X, test = 'adf')
# KPSS Test
KPSSTest = ndiffs (X, test = 'KPSS')
# PP Test
PP Test = ndiffs (X, test = 'PP')
Print ("ADF Test = ", adftest )
Print ("KPSS Test = ", KPSStest )
Print ("PP Test = ", PPftest )
```

☞ **Output**

```
ADF Test = 2
KPSS Test = 0
PP Test = 2
```

☞ **Explanation**

Here, we have imported the ndiffs method of the 'pmdarima' module. We have imported dataset 'X' as the object and it contains values from the dataset.

We used the 'ndiffs' method to perform ADF KPSS and PP Tests and printed their results to the users.

3.5.9 Finding the Order of the Auto-regressive (AR) Term (P)

- We use partial Autocorrelation (PACF) plot to check whether the model requires any 'Autoregressive (AR) terms'
- We consider 'Partial Autocorrelation' as the correlation between the series and its lag once we exclude the contribution from the intermediate lags. Thus, PACF tends to convey the pure correlation between the series and its lag.
- Hence, we can identity whether that lag is required in the Auto-regressive (AR) term or not.
- Partial Autocorrelation of lag(k) of a series is the coefficient of that lag in the Auto-regression Equation of Y.

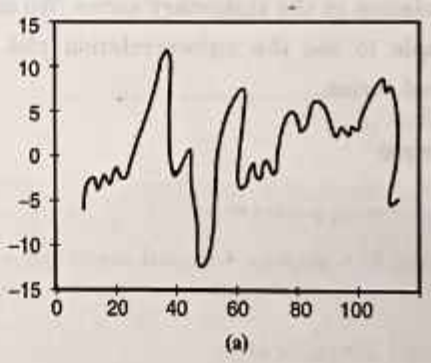
$$Y = a_0 + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \alpha_3 Y_{t-3}$$

- Now, we see the method of finding the number of AR terms. By inserting enough AR terms, any Autocorrelation in a stationary series can be rectified.
- So, we can take initially the order of the Auto – Regressive (AR) term equivalent to as many lags that cross the limit of significance in the PACF plot.

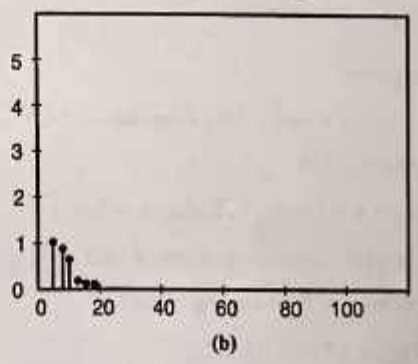
☞ **Example**

```
Import numpy as np, pandas as pd
from statsmodels • graphics • tsaplots
import plot-acf, plot-pacf
Import matplot lib• pyplot as plt
plt • rc Params • update ( {'Figure • figsize' : (9, 3), 'figure • dpi : 120'})
# importing data
df = pd • read – CSV ('my dataset • CSV', names = ['value'], header = 0)
fig, axes = plt • subplots (1, 2, sharex = True)
axes [0] • plot (df • value • diff () ) ; axes [0], set – title
                    ('Order of Differencing : first')
axes [1] • set (y | *m = (0, 5))
plot – pacf (df • value • diff ( ) • drop na( ), ax = axes [1] )
plt • show( )
```

☞ **Output**



(a) Order of Differencing First



(b) Partial Autocorrelation

Fig. 3.5.2

Explanation

Here, we have import the required libraries, modules and data sets. We plot the graphs to represent the First Order Differencing and its partial autocorrelation.

As a result, we see that the PACF lag 1 is pretty significant above the line of significance. Lag 2 also appears to be substantial, entirely maintaining to cross the limit of significance.

3.5.10 Finding the Order of the Moving Average (MA) Term (a)

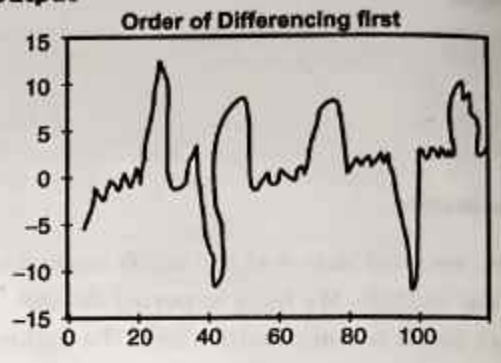
We use ACF Plot to find the number of Moving Average (MA) terms. A Moving Average term is the lagged fore cast's is error.

The ACF plot expresses the number of Moving Average (MA) terms needed to remove the autocorrelation in the stationary series. We consider an example to see the autocorrelation plot of the differenced series.

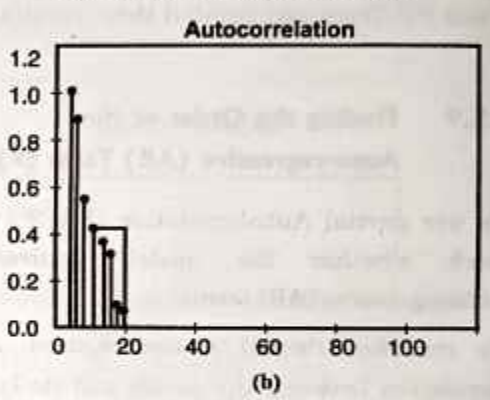
Example

```
Import numpy as np, pandas as pd
from statsmodels • graphics • tsaplots import plot-acf, plot-pacf
Import matplot lib• pyplot as plt
plt • rc Params • update ( {'Figure • figsize' : (9, 3), 'figure • dpi : 120'})
# importing data
mydata=df = pd • read – CSV ('my dataset • CSV', names = ['value'], header = 0)
fig, axes = plt • subplots (1, 2, sharex = True)
axes [0] • plot (mydata • value • diff () ) ; axes [0], set – title ('Order of Differencing : first')
axes [1] • set (y | *m = (0, 1, 2))
plot – acf (mydata • value • diff ( ) • drop na( ), ax = axes [1] )
plt • show( )
```

Output



(a) Order of Differencing First



(b) Autocorrelation

Fig. 3.5.3

Explanation

Here, we consider the required libraries, modules and data sets. We then plot the graphs to represent the First Order Differencing and its Autocorrelation.

We note that some lags are pretty above the line of significance. So, let us fix q as 2.

We can also use the simpler model in case of any doubt that adequately demonstrates the Y.

3.6 REASONS TO CHOOSE ARIMA MODEL AND CAUTION

An autoregressive integrated moving average, or ARIMA, is a statistical analysis model. It uses 'time-series' data to better understand the data set or to predict future trends.

(i) Autoregressive Integrated Moving Average (ARIMA) models predict future values based on past values.

(ii) ARIMA makes use of lagged moving averages to smooth time series data.

(iii) They are widely used in technical analysis to forecast future security prices.

(iv) Auto regressive models assume that the future will resemble the past.

(v) Hence, they prove inaccurate under certain market conditions, such as financial crises or periods of rapid technological changes.

3.6.1 Pros and Cons of ARIMA

ARIMA models have strong points and are good at forecasting. Forecasting is based on past circumstances.

ARIMA models assume that past values have some residual effect on current or future values and use data from the past to forecast future events. But there are strong disclaimers that state "past performance is not an indicator of future performance."

☞ **Pros**

(i) Good for short-term forecasting,

(ii) Only it needs historical data

(iii) Models non-stationary data.

(iv) It is based on the statistical concept of serial correlation, where past data points influence future data points.

☞ **Cons**

(i) Not built for long term forecasting.

(ii) Poor at predicting turning points.

(iii) Computationally expensive.

(iv) Parameters are subjective.

Chapter Ends...

❑❑❑

Module 4

CHAPTER 4

Text Analytics

University Prescribed Syllabus

History of text mining, Roots of text mining overview of seven practices of text analytic, Application and use cases for Text mining: extracting meaning from unstructured text, Summarizing Text.

Text Analysis Steps, A Text Analysis Example, Collecting Raw Text ,Representing Text ,Term Frequency - Inverse Document Frequency (TFIDF),Categorizing Documents by Topics, Determining Sentiments , Gaining Insights.

4.1 TEXT MINING

GQ. What is text mining? Explain text mining architecture and explain its need.

GQ. Write short note on text mining. (4 Marks)

4.1.1 Why Text Mining ?

- The age of Big Data is leading to an enormous increase of digital information.
- Today memory is not expensive and no longer a medium with limits. Until now, knowledge has been distributed to only some sources; however the lots of technical possibilities have lead to a quick increase in the number of text documents and memory locations.
- This may leads to a serious problem :
  - Given the collection of information, understanding text contents and associations – something which, until now has exclusively been performed by the human mind - can no longer be carried out reliably without technological help.
  - The data which can be located in the text document and cannot be found in the database are obtained with the help of highly professional Text Mining software solutions.

4.1.2 Definition of Text Mining

Definition : "Text Mining is defined as it is a process of obtaining information from unstructured texts".

- The word "*Mining*" in the term Text Mining is taken from an analogy to coal mining. It is used to get the possibly precious business insights from text-based content like emails, word documents and posts from the social media such as tweeter, facebook, Linkedin etc.
- To transform the words and phrases from unstructured data into the numerical values which may in turns connect to the structured data in the database and can be examined by the data mining techniques.

Text involved in text mining

Following things are involved in text mining :

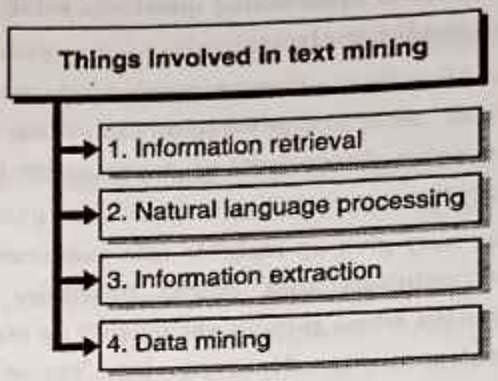


Fig. 4.1.1 : Text involved in text mining

1. Information retrieval

- In the information retrieval a user query is matched with the documents available in the database.
- To search the body of documents that is related to the research questions is the first stage in the text mining process.

2. Natural language processing

In natural language processing text can be examined in structures based on human speech. It permits the computer to perform a grammatical analysis of a sentence to read the text.

3. Information extraction

The information extraction is responsible for structuring the data which is generated by the NLP.

4. Data mining

Data mining is the process of identifying patterns in huge sets of data, to find that new knowledge.

4.1.3 Typical Applications for Text Mining

GQ. Explain applications of text mining. (4 Marks)

- Analyzing open-ended survey responses.
- In review research, it is not unusual to include a variety of open-ended questions related to the topic under analysis.
- The idea is to allow respondents to state their "views" or opinions without restricting them to particular dimensions or a particular response format.
- This may give up insights into customers' views and opinions that might otherwise not be revealed when relying exclusively on structured questionnaires designed by "experts." For example, you may find out a certain set of words or terms that are frequently used by respondents to express the advantages and disadvantages of a product or service, telling common misconceptions or confusion regarding the items in the study.
- Automatic processing of messages, emails, etc.
- One more familiar application is to assist in the automatic classification of texts. For example, it is possible to "filter" out automatically most unwanted "junk email" based on certain terms or words that are not likely to appear in valid messages, but instead recognize unwanted electronic mail.
- In this way, such messages can automatically be deleted. Such automatic systems for categorizing electronic messages can also be useful in applications where messages want to be routed to the most suitable department or agency.
- Analyzing warranty or insurance claims, diagnostic interviews, etc.
- In some business domains, the most of the information is gathered in open-ended, textual form. For example, when you take your automobile to a service station for repairs, normally, the employee will write some remarks about the problems that you report and what you believe desires to be fixed.
- Such remarks are collected electronically, hence those types of descriptions are available readymade for the purpose of input into text mining algorithms. This data can further be neatly brokened to, for example, identifying general group of issues and complaints on products, etc. In the same manner, in the medical field, complete descriptions by patients regarding the related symptoms may becomes useful clues for the purpose of actual medical diagnosis.
- Investigating competitors by crawling their web sites
- One more type of possibly very helpful application is to automatically process the contents of Web pages in a particular domain.
- For instance, you can go to a Web page, and start "crawling "the links you locate there to process all Web pages that are referenced.
- In this way, you can automatically obtain a list of terms and documents accessible at that site, and therefore rapidly determine the most important terms and features that are illustrated.

4.2 ROOTS OF TEXT MINING, OVERVIEW OF SEVEN PRACTICES OF TEXT ANALYTIC

- Text mining is a component of data mining that deals with unstructured text data.
- It involves the use of Natural Language Processing (NLP) techniques to extract useful information and insights from large amount of unstructured text data.
- Text mining can be used to extract structured information form unstructured text data such as :

(i) **Named Entity Recognition (NER) :** Identifying and classifying named entities such as people, organisations, and locations in text data.

(ii) **Sentiment Analysis :** Identifying and extracting the sentiment (e.g. positive, negative, neutral) of text data.

4.2.1 Procedure for Analysing Text Mining

(i) **Text summarisation :** To extract its partial content reflects its whole content automatically

(ii) **Text categorisation :** To assign a category to the text among categories predefined by users.

(iii) **Text clustering :** To segment text into several clusters, depending on their substantial relevance.

4.2.2 Issues in Text Mining

Numerous issues happen during the text mining process :

(i) The efficiency and effectiveness of decision-making.

(ii) The uncertain problem can come at an intermediate stage of text mining. To make the text mining process efficient, different guidelines are developed to normal the text, in the pre-processing stage.

(iii) Sometimes original message or meaning can be changed due to alteration.

(iv) Many algorithms and techniques support Multilanguage text, in text mining. And this creates ambiguity in text mining.

(v) The use of synonym and antonyms in the document text makes ambiguity in the text mining tools. Since text mining tools take both in a similar setting. Hence it is difficult to categorise such kinds of text/words.

4.2.3 Advantages of Text Mining

(i) **Large amounts of Data :** Text mining allows organisations to extract insights from large amounts of unstructured data. It includes social media posts, news articles and customer feedback.

(ii) **Variety of Applications :** Text mining has lot many applications, such as sentiment analysis, named entity recognition, and topic modelling.

Organisation can gain insights from unstructured text data using the above versatile tool.

(iii) **Improved Decision Making :** Text mining can be used to extracts insights from unstructured data and that can be used to make data-driven decisions.

(iv) **Cost-effective :** Text mining is a cost-effective way to extract insights from unstructured text data. It eliminates the need for manual data entry.

4.2.4 Disadvantages of Text Mining

(i) **Complexity :** Text mining requires advanced skills in natural language processing and machine learning. This makes it a complex process.

(ii) **Quality of Data :** The accuracy of the insights extracted from text-mining depends on quality of text-data. And hence quality of text data can vary.

(iii) **High computational cost :** Text mining needs high computational resources. And for small organisations it is not feasible, i.e., they may not afford it.

(iv) **Limited to text data :** Text mining is restricted to extracting insights from unstructured text data and hence cannot be used with other data types.

4.2.5 The seven Practices of Text Analytic

- Text mining can be divided into seven practice areas. They are based on unique characteristics of each area. These areas are basically distinct but they are interrelated.
- A text mining project required techniques from multiple areas. We discuss these area based on five resource and goal questions that text mining practitioners must know, and must be able to anser, when they face a new problem.

The seven practice area are as follows :

1. **Search and Information Retrieval (IR) :** Storage and retrieval of text documents, including search engines and also keyword search.
2. **Document Clustering :** Grouping and categorising terms, paragraphs or documents using data mining clustering methods.
3. **Document Classification :** Grouping and categorising paragraphs or documents using data mining classification methods, and this is based on models trained on labelled examples.
4. **Web Mining :** Data and text mining on the Internet, with a specific focus on the scale and interconnectedness of the web.
5. **Information Extraction (IE) :** Identification and extraction of relevant facts and relationships from unstructured text, the process of making structured data from unstructured and semi structured text.
6. **Natural Language Processing :** Low-level language processing and understanding tasks (e.g. tagging part of speech); used synonymously with computational linguistics.
7. **Concept Extraction :** Grouping of words and phrases into semantically similar groups.

- The Fig. 4.2.1 exhibits a Venn-diagram, where the overlap of the seven fields of text mining, data mining, statistics, artificial intelligence and machine learning, computational linguistics, library and information sciences, and databases.
- It also gives the seven practice areas at their key intersections.

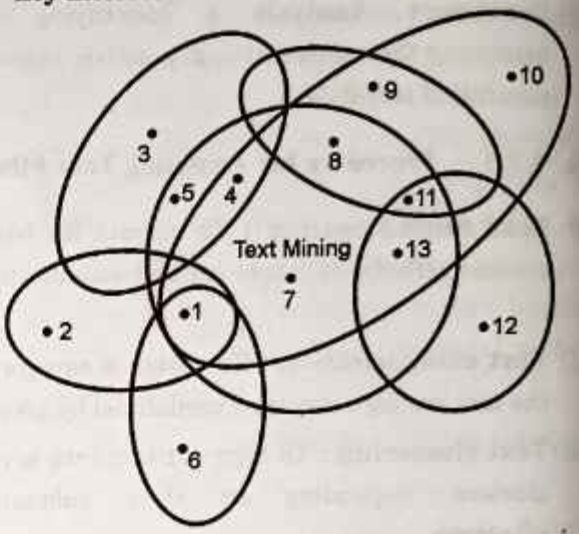


Fig. 4.2.1

- A Venn diagram of the intersection of text mining and six related fields, such as data mining, statistics and computational linguistics. The seven text mining practice areas exist at the major intersections of text mining with its six related fields.

1. Information retrieval,
2. Databases,
3. Data mining,
4. Document classification
5. Document clustering,
6. Library and Information Sciences
7. Web Mining
8. Information Extraction
9. AI and Machine Learning
10. Statistics
11. Natural Language Processing.
12. Computational Linguistics,
13. Concept Extraction.

4.2.6 Five Questions for Finding the Right Practice Area

- We draw a decision tree (Fig. 4.2.2) indicating how a few straightforward questions can direct us to the proper text mining solution.
- They split the major branches of text mining. The seven practice areas are depicted as leaf nodes of the tree.
- A text mining solution begins with raw input documents and moves towards fully encoded text.
- At each step, a group of questions must be answered to determine the appropriate processing task.

Five questions for finding the right practice area

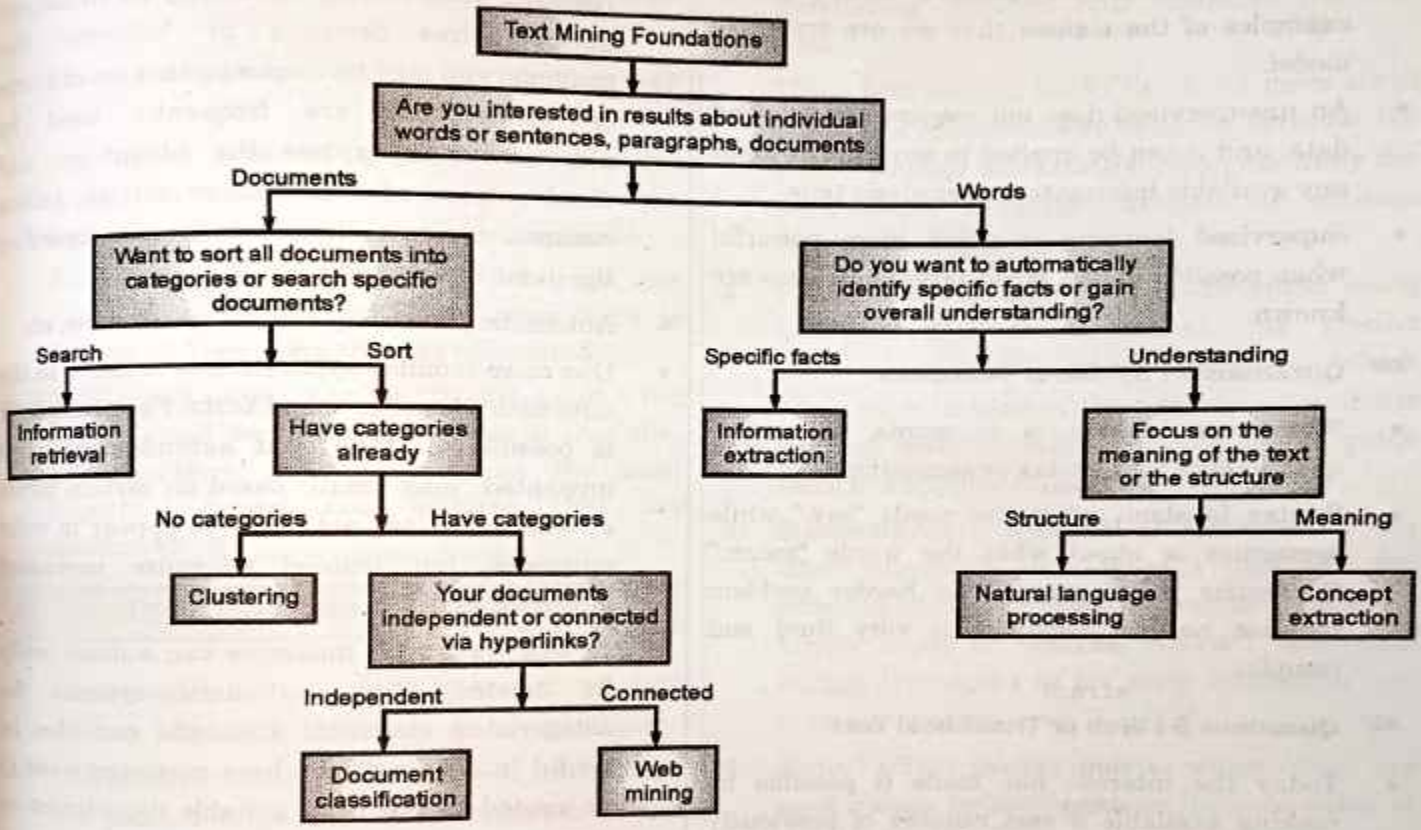


Fig. 4.2.2 : A decision tree for finding the right text mining practice area by answering 2 to 4 questions about the text resources and project goals

Question 1 : Granularity

- This questions finds level of detail of focus of the text mining task.
- While documents and words are both important to successful text mining, an algorithm virtually always emphasises one or the other.

Question 2 : Focus

- For documents or words, the next question is to the focus of the algorithm.
- To find specific words or documents or characteristics of the entire set is the question.
- This questions separates the two practice areas.

☞ **Questions 3 : Available Information**

- If we are interested in documents, the question is the available information of the time of analysis.
- This is equivalent to the supervised/unsupervised question from data mining.
- A supervised algorithm requires training data with an answer for positive and negative examples of the classes that we are trying to model.
- An unsupervised does not require any labelled data, and it can be applied to any data without any available information at analysis time.
- Supervised learning is much more powerful when possible to use, when target outcomes are known.

☞ **Questions 4 : Syntax or Semantics**

- When our interest is in words, the major question is about syntax or semantics.
- Syntax is about what the words "say," while semantics is about what the words "mean." Semantics is, of course, the harder problem because natural language is very fluid and complex.

☞ **Questions 5 : Web or Traditional Text**

- Today the internet has made it possible by making available a vast number of previously unreachable text documents.
- The structure and style of web documents provide unique opportunities and challenges when compared to nonweb documents.

⏭ 4.3 APPLICATIONS FOR TEXT MINING

GQ. Explain applications of text mining. **(4 Marks)**

- Analyzing open-ended survey responses.
- In review research, it is not unusual to include a variety of open-ended questions related to the topic under analysis.
- The idea is to allow respondents to state their "views" or opinions without restricting them to particular dimensions or a particular response format.
- This may give up insights into customers' views and opinions that might otherwise not be revealed when relying exclusively on structured questionnaires designed by "experts." For example, you may find out a certain set of words or terms that are frequently used by respondents to express the advantages and disadvantages of a product or service, telling common misconceptions or confusion regarding the items in the study.
- Automatic processing of messages, emails, etc.
- One more familiar application is to assist in the automatic classification of texts. For example, it is possible to "filter" out automatically most unwanted "junk email" based on certain terms or words that are not likely to appear in valid messages, but instead recognize unwanted electronic mail.
- In this way, such messages can automatically be deleted. Such automatic systems for categorizing electronic messages can also be useful in applications where messages want to be routed to the most suitable department or agency.
- Analyzing warranty or insurance claims, diagnostic interviews, etc.
- In some business domains, the most of the information is gathered in open-ended, textual form. For example, when you take your automobile to a service station for repairs, normally, the employee will write some remarks about the problems that you report and what you believe desires to be fixed.

- Such remarks are collected electronically, hence those types of descriptions are available readymade for the purpose of input into text mining algorithms. This data can further be neatly brokened to, for example, identifying general group of issues and complaints on products, etc. In the same manner, in the medical field, complete descriptions by patients regarding the related symptoms may becomes useful clues for the purpose of actual medical diagnosis.
- Investigating competitors by crawling their web sites
- One more type of possibly very helpful application is to automatically process the contents of Web pages in a particular domain.
- For instance, you can go to a Web page, and start "crawling "the links you locate there to process all Web pages that are referenced.
- In this way, you can automatically obtain a list of terms and documents accessible at that site, and therefore rapidly determine the most important terms and features that are illustrated.

4.4 USE CASES FOR TEXT MINING

- Text mining is a very rich branch of data science, filled with extremely useful techniques.
- Using text mining cases, one can understand the present to predict future. Nowadays, the predictive capacity of data mining has changed the design of business strategies.
- We mention below use cases and examples text (data) mining in current industry.

(1) **Marketing :** Data mining can be used to explore large databases to improve market segmentation.

- It can analyse the relationships between parameters such as customer age, tastes, gender etc., and thereby guess their behaviour to direct personalised campaigns.
- Text mining in marketing can predict which users are likely to unsubscribe from a service, what interests them based on their searches. The achieve a higher response rate, mailing list should be included.

2. **Banking :** To understand market risks, banks use data mining. It is applied to credit ratings, scrupulously done anti-fraud systems to analyse such transactions, card transactions, purchasing patterns and customer financial data.

   Using text mining, banks can learn more about our online preferences or habits to optimise the return on their marketing campaigns, study the performance of sales channels or manage regulatory obligations.

3 **Education :** Eductors can be benefited using data mining to access student data, predict achievement levels and find students which need extra attention. For example, students who are weak in maths subjects, it can guide them to a popular class.

4. **E-commerce :** Websites of E-commerce can offer cross-sells and up-sells through the websites of data mining. A popular and well-known name is Amazon. Amazon uses data mining techniques to get more customers into their e-commerce store.

5. **Retail :** Text mining detects which offers are most values by customers or increase sales at the checkout queue.

   Supermarkets use joint purchasing patterns to decide how to place products on the shelves.

6. **Service providers :** Service providers are generally mobile phones and utility industries. They use data mining to predict the reasons when a customer leaves their company.

   They analyse billing details, customer service interactions, complaints made to the company and assign each customer a probability score and offer incentives.

7. **Medicine :** Knowing all of the patient's information, such as medical records, physical examinations, and treatment patterns, data mining can effect efficient and cost-effective management of health resources. This is done by identifying risks, predicting illnesses in certain segments of populations.

   Data mining in medicine can detect fraud and irregularities, and strengthening ties with patients with knowledge of their needs are also advantages of using data mining.

8. **Insurance :** Data mining helps insurance companies to price their products profitable and promote new offers to the new customers.

9. **Manufacturing :** Wear and tear of production assets can be predicted with the help of data mining manufactures.

   They can anticipate maintenance which helps them to minimise downtime.

10. **Crime Investigation :** Data mining helps crime investigation agencies to deploy police work-force, where a crime is most likely to happen and when, and who to search at a border crossing etc.

11. **Television and Radio :** Some networks apply time-data mining to measure their online television and radio audiences.

    These systems collect and analyse information form channel views, broadcasts and programming. Data mining allows networks to make personalised recommendations to radio listeners and TV viewers. It also gets to know their interests and activities in real time and better understand their behaviour.

Networks also obtain valuable knowledge for their advertisers, who use this data to tap their potential customers accurately.

4.5 MEANING FROM UNSTRUCTURED TEXT

- The most common unstructured text analysis techniques used in extracting information from unstructured text. We mention them as follows :

1. Sentiment Analysis

- Sentiment analysis is also known as opinion mining. It is an NLP text extraction techniques to find out the pulse of the given data. It tries to find whether the data has a positive, negative, or neutral tone. This is used to analyse customer feedback to find out if they are satisfied or not.
- There are three different types of sentiment analysis; emotion detection, graded analysis, and multilingual analysis.

(i) **Emotion Detection :** This type of sentiment analysis detects emotions such as frustration, anger, happiness, sadness etc. This analysis goes for beyond good, bad or positive, negative sentiments.

- There are some lexicons that help algorithms to detect human emotions.
- But lexicons need not provide accurate information in extraction from text. It is because different people express emotions in different ways.
- For example, words like "Good" can be used in positive ways "this product is good' Also, in negative ways, "Good product, forget that completely."

(ii) **Graded Analysis :** Instead of mentioning three quantities; positive, negative and neutral, we can expand these sentiment analysis for higher precision. Like very positive, positive, neutral, negative and very negative.

This is a more fine-grained sentiment analysis.

(iii) **Multilingual Analysis :** Multilingual analysis is an information extraction NLP method. It can detect the language in text and then apply sentiment analysis to find a positive, negative, or neutral tone.

2. Named Entity Recognition

- Named Entity Recognition (NER) is a Machine Learning (ML) technique. It detects certain identifiers and how they are classified according to predefined categories.

For example, consider the text :

- Shekhar went to Pune university in 2020 and met madhuri there.
- Now, NER can recognise shekhar as a person, Pune University as a University, 2020 as a date and madhuri as another person.
- This needs intensive labelling to understand separate words and the categories to whom they belong. Apart from that, the model also must understand the context so that there is no ambiguity. Once the ambiguity is removed, it can be used for extracting information from unstructured text.
- NER can be used in variety of ways For Example, (i) it can be used to train the chatbots in banking applications to chat with customers.

(ii) It can be used in the medical industry to recognise vital terms used in medical reports.

(iii) It can also be used to read customer reviews to see how many times a specific term is repeated to understand the pain points.

It has also other uses. Customer concerns are automatically sent to the right department where they can be resolved, leading to better customer satisfaction.

3. Topic modelling

Topic modelling is a ML technique. It can scan documents, find phrases and words, and it can assign certain topics to them, accordingly.

It is an unsupervised and hence does not need preexisting tags. The topic modelling algorithm extracts some attributes from a large set of wards.

The most repeated ones are called topics. This way data is classified and hence no need to go through all the documents.

Using a topic model group, comparable feedback can be grouped. This is done by recognising certain patterns such as word/phrase frequency and the number of words between two words. These pieces of information helps the algorithm to infer the given data for extracting information from unstructured text using algorithms. There are various algorithms for this purpose :

- The most common method for topic modelling is Latent Dirichlet Allocations (LDA).
- According to this algorithm, documents are made of specific topics or tokens. LDA detects topics to which a particular document belongs.
- For example, the algorithm creates five topics based on the words in documents. It checks how many times those words recur in a particular document and then assigns documents to their topics.
- As an example, it is like assigning books to a particular shelf of the library. Depending on the contents of the book, the librarian decides which shelf it will belong to.

4. Text classifications

- Text classification is also called as text categorization or text tagging. It is used to analyse unstructured data. It assigns some tags to the text depend on its content.
- The three main methods for text classification are (i) rule-based, (ii) machine-based, (iii) hybrid.

(i) In the rule-based method. The model uses a set of linguistic rules to separate text into groups.

- These linguistic rules are defined and categorised by users.
- For example, words like Ratan Tata and Adar Poonawala are in Industrialists Category.

(ii) In the machine-based method, classifications are based on earlier learning's.

For example, if we apply the model to movie reviews. Based on the past reviews, it has a bag of data words. These words could be "funny," "sad", "tragic", "comedy", "boring", "action," "thrilling", and so on.

The model will find the occurrence and frequency of these words in a given review. And then it classifies the review.

(iii) The hybrid approach combines rule-based and machine-based approaches.

It uses a rule-based method to create tags. And machine learning to classify data based on those tags.

- It also uses humans to improve the list, and that makes it the best text classification method.

5. Dependency Graph

- A dependency graph is a data structure and that represents how one element of the system interacts with the other elements.
- It is made as a directed graph where every node directs to the one it depends on.
- A dependency graph reveals links between neighbouring words. This way the grammatical structure of a sentence can be analysed.
- Based on links between the words, it divides a sentence into several sections.
- Dependency parsing is based on the assumption that there is a relationship between every linguistic unit in a sentence.
- It is easier to extract information from unstructured text when text is arranged as a directed graph.

4.6 SUMMARISING TEXT

- A text summariser is an online tool that wraps up a text to a given short length. A long article is converted to main points. And because of time constraints, the need for text summarisers is increasing day by day.
- There is a need of shortcut methods to learn ideas in lesser time. At present, text summarisers are helping them to decide whether a book, a research paper, or an article is worth reading or not.

Oxford has defined 'summary' as :

"a short statement that gives only the main points of something, not the details."

4.6.1 Auto Summarisation

- Mainly two approaches have been developed over time for summarising a long text into a shorter one.

(I) Extraction Summarisation

- This approach gives the method to extract keywords and phrases from sentences and then joining them to produce a compact meaningful summary.

(II) Abstractive summarization

- In this summary generator, algorithms are developed in such a way to reproduce a long text into a shorter one by NLP.
- It changes the structure of sentences but retains its meaning.

4.6.2 Working of Text Summariser

- Text summariser uses the concept of abstractive summarisation to summarise a book, an article, or a research paper. It is trained by machine learning, 'Paraphraser · io'.
- This summariser tool uses NLP to create novel sentences. It generates summary by retaining the main idea.

- It is an advanced-level tool that uses AI for its work. Hence this summariser tool appears to be flawless and short.

4.6.3 Method of Using Text Summariser

The summarising tool is simple to use and also efficient. It is the best tool because of its flawlessness.

(i) Insert the text (article, research paper, book extract) into the text area.

(ii) Upload the content

(iii) Click the "Summarise" button.

(iv) If need be, toggle other features by selecting show bullets, best line, ranked base, and summary length.

4.6.4 Features of Text Summariser

AI Powered	Sums up text with advanced AI
Multilingual	8 supported languages
Price	100% Free unlimited words

The extra important features are as follows :

(I) Control Summarization

- This feature gives the freedom to choose the length of your summarised text. This is the best feature because of the freedom given.
- Sometimes a long summary is required and sometimes a short summary works. That depends upon the circumstances. This summary generator tool offers the choice to summarise the text according to the needs.

(II)Bullet points formation

- To create bullet points, text summariser can be used to analyse your text.
- This summariser tool can help in creating power point slides and presentations.

(III) Rating of the text

- This is a full-pack feature. It gives the whole ranking of the text.
- This summary maker gives the best line, best sentence and general ranking of the text according to its optimisation.

(IV) Free usage

- This text summariser has free usage and can be used whenever it is required. One can instantly use it without giving any prior intimations.

4.6.5 Users of Text Summariser

(I) Students : A text summarise can help the students to clear concepts by summarising them. The students can get the know-how of complex articles and books. They also use a text summariser to solve the assignments in lesser time.

(II) Journalists : The text summariser can also help journalists, as Journalists have to communicate an incident or an event. Giving quick headlines is more valuable than giving thorough news. This way they can use this summarising tool to inform people about daily happenings.

(III) Writers : The common difficulty faced by writers is of creating unique content either blogs or guest posts. They can produce exceptional content if they know the gist of the whole story.

- While getting ideas from different sources, they can use the text summariser to make out the necessary information. And this information is incorporated into what they are writing.

4.6.6 Key-Steps to Write a Summary

Thus, we mention five key-steps to write a summary :

(i) Read the text,

(ii) Break it down into sections,

(iii) Identify the key points in each section

(iv) Write the summary

(v) Check the summary against the article.

4.7 TEXT ANALYSIS STEPS

- The process of transforming unstructured text in documents into structured data which can be used for analysis, is text analysis.
- Text analysis finds patterns in the text and find meaning in them. It works by breaking phrases and sentences into components and then evaluate the meaning and roles using algorithms.
- To derive useful information and insights from huge amounts of raw data, such as social media comments, news articles and reviews, data analysts use text mining tools.

The steps involved in analysing an unstructured text documents are :

1. Language Identification,
2. Tokenisation,
3. Sentence breaking,
4. Part of speech tagging
5. Chunking,
6. Syntax parsing
7. Sentence chaining.

We discuss these in detail.

1. Language Identification

- The first step is to identify in which language text is written. Languiage identifications is a major and main process for every text analytics function because each language has its own rules of grammar and peculiar syntax.
- Thus it is noteworthy to know which language we are dealing with.

2. Tokenisation

- Tokenisation is the process of breaking down the sentence into small pieces. Thus tokens are the words, numbers or punctuations in the sentence.
- In text analystics, tokens are usually words. Thus a sentence of 15 words would have 15 tokens.
- Since tokenisation is language-specific, each language will have its own requirements. Generally alphabetic languages use whitespace and punctuation to indicate tokens in a sentence. Languages which are character-based like Chinese, Japanese use other systems.
- Lexalytics use rules-based algorithms to tokenise normal alphabetic languages, but languages like Chinese, Japanse require use of complex machine learning algorithms.

3. Sentence Breaking

- Once the token are identified, one can understand where the sentences are going to end.
- Small texts like statuses contain only single sentences many of the time.
- But huge, longer documents will require sentence breaking to separate each statement.
- In some documents, each sentence will be separated by a single punctuation mark. But some may contain punctuation marks that do not mean the end of the statement.
- For example, "Dr. Gandhi is an Indian rapper," contains two punctuation marks.
- That may indicate the end of a sentence. But, the punctuation after "Dr." does not indicate the end. Hence deeper text analytics must be done to tell where the boundaries are in a sentence.

4. Part of Speech Tagging

- Part of Speech tagging (or PoS tagging) is the process which determines the part of speech of every token in a document, and then tagging it.
- When a text document is shown, the tagger must be able to locate whether a given token indicates a common noun or a proper noun, or if it is adjective, a very, adverb or something else.

- Many languages follow some basic patterns and rules. They can be written into a basic part of speech tagger. Accuracy is very crucial in Pos tagging so it can give reliable sentiment analsis.

5. Chunking

- Chunking is parsing that refers to a range of sentence-breaking systems that fragment a sentence into its component phrases (verb phrases, noun phrases, and so on).
- Chunking is different than part of speech tagging in text analytics. PoS tagging assigns parts of speech to tokens where chunking assigns PoS-tagged tokens to phrases.

Let us consider the sentence:

- "The black cat is going to run fast across the bridge".
- PoS tagging will identify 'cat' and 'bridge' as nouns and 'run' as verb.

Chunking will identify :

'The black cat' as noun phrase, 'is going to run fast' as verb phrase, 'across the bridge,' as prepositional phrase.

6. Syntax Parsing

- Syntax parsing is a process which determines how a sentence is formed. It is very crucial step in sentiment analysis and other natural language processing features.
- One single sentence can have variety of meanings depending on how it is formed :

"Ramesh was fairing poorly in his exams until Mrs. Paranjapye started teaching him."

"Ramesh was fairing poorly in his exams because Mrs. Paranjapye started teaching him."

- We note that, in the first sentence, Ramesh is negative, whereas Mrs. Paranjapye is positive.
- In the second, Ramesh is still negative, but Mrs. Paranjapye is now negative.
- We consider one more case.
- Because Ramesh was fairing poorly in his exams, Mrs. Paranjapye started teaching him.
- Here Ramesh is still negative but Mrs. Paranjapye is neutral.
- Thus syntax-parsing is the most computationally-intensive step in text analytics.

7. Sentence Chaining

- Sentence chaining is the final step sentence chaining uses a technique to link individual sentences using each sentence's strength of association to an overall topic.

For example, we consider the following sentences :

I like Vespa.

Ramesh just bought a new SUV.

Tata launched a new Indigo

- These sentences are not next to each other in a body of text, but they are connected through the topics of Vespa > Suv > Indigo.

Sentence chaining makes these connections.

4.7.1 Benefits of Text Analytics

(i) It helps to understand emerging customer trends, product performance, and service quality.

(ii) It helps researchers to explore pre-existing literature and extracting what is relevant to their study.

(iii) Text analytic techniques help search engines to improve their performance. That way it provides fast user experiences.

(iv) It helps in making more data-driven decisions.

(v) It refines user content recommendation systems by categorizing related content.

(vi) It boosts efficiency of working with unstructured data.

4.8 A TEXT ANALYSIS EXAMPLE

- Text analysis is the process of distilling information and meaning form text. For example, this can be analyzing text written in reviews by customers on a retailer's website or analyzing documentation to understand its purpose.

4.8.1 Text Analytics used by Companies

- For many companies, Text Analysis is the first step in a data-driven approach towards management. Text sources get converted to data. This opens up opportunities to embed the results of the analysis in processes like strategic decision making, product development, marketing, competitor intelligence and more.

In a business context, analyzing text to get data from hem supports the broader tasks of :

(I) Content management,

(II) Semantic search

(III) Content recommendation,

(IV) Regulatory compliance.

In an organization with a large amount of unstructured data, uncategorised information that may be analysed; here the above applications are useful.

(I) Customer feedback monitoring

- One popular method to measure customer satisfaction is through Net Promoter Score (NPS).
- NPS surveys possess one powerful questions :
- How likely are you to recommend our product ?
- This simple question has made NPS surveys very popular to understand how customers perceive their product or service.
- Many of the organisations also have some structured methods for collecting feedback, such as follow-up phone calls or post-purchase emails. These methods can generate a huge amount of text.
- Also one has to take into account the responses to open-ended questions, because here lies the valuable insights. There are also social media, web forums, independent surveys, live chats, and more.
- This is an enormous quantity of feedback data, and these lie outside traditional NPS methodologies Hence Text analysis is the first step in understanding customers' thinking process.

(II) Social Media Monitoring

- Nearly 20% Indians have interacted with companies or institutions on at least one of their social media networks. And this number is increasing at a huge rate. This immediately tells us that there is a huge volume of interaction data before us.
- What are people complaining about? What are they praising ? How are they interacting with marketing messages or campaigns?
- To understand the difference between social media listening and Natural language processing we again refer to customer feedback monitoring and that has been listed.

(III) Brand monitoring

- Analysis of brand allows us to keep up-to-date with word-of-mouth credibility within the industry, to identify potential reputational risks and to quickly respond to them. This gives us indication over time to understand how the brand perception has changed in context with competitor news or social changes.
- In America, around 81% of buyers conduct online research before making a purchase.
- Consumers take note of what people are saying online about a brand.
- About 78% of consumers trust online reviews as much as personal recommendations.

- Part of this is because of social media and also specialist internet forums, local news etc.
- Knowing social opinion about the brand is important, but it is also vital to know who is talking about and in what content.
- Advance forms of analysis like sentiment analysis can understand the tone of language. This helps to define who and why influencers mention the brand.
- Sentiment analysis software can do this in real-time and across multiple channels.

(IV) Competitor and Market research

- With text analysis, you can easily find out how customers feel about the product and what they feel about the competitors and also what competitors feel about you. And what customers value about other industry players? Why would they choose a competitor over you? What do competitor offerings lack ? Through which channels do clients use to engage with competitors?
- These things help us to improve our communication and marketing strategies, and also develop our customer service operations and it becomes a more customer-centric company.
- Marketers can also find out consumer behaviour in real-time to assess and understand future trends and help management teams to make long-term decisions.
- The important point is, the data is already there- it does not require new market research, and it can be understood in real-time.

(V) Customer service prioritization

- Retailers, financial institutions of transport firms rely heavily on incoming customer contact.
- They use text-analysis to optimize their customer service-work.
- With text analysis, the business can automate the classification of inbound messages, topic, subject matter, and priority. Then queries can be easily sent to an appropriate specialist.
- For example, new messages from most angry customers may need to be processed first while questions about hardware faults may be sent to a specialized team.

(VI) Product development

- All successful companies are interested in knowing how product launches are working they collect early feedback so that they can optimize the products.
- Text analysis allows the business to sort comments by topic or sentiment, which product features are most or least critical or even how product messaging and packaging work. This way product launches can be optimized, and the business can confidently start the process of product optimization.

(VII) Workforce Analytics

- Text analysis can also be applied internally for HR-related processes. Most large companies measure employee satisfaction and try to isolate factors that may reduce company performance and employee satisfaction.
- Employee satisfaction surveys, and employee review data and employee surveys, can be analyzed to address problems and potential concerns.

(VIII) Prediction

- Naturally any kind of prediction exercise requires a large amount of data base to analyse and then test forecasts against text analytics can be the basis for this.
- For example, suppose we are interested in forecasting Indian economic performance.

- Text analysis will help to scan filings for key text, cluster related terms and determine which are causal.
- The same can be applied to news feeds, Reserve bank data, airline flights, State Bank data, gasoline prices, miscellanceous purchases, in fact everything.
- At a fine-tune level, the same principle can be applied to assessing the brand impact of product launches or also forecasting the effect of competitor activity.

4.9 COLLECTING RAW TEXT

- Data is the cornerstone of every analytics activity.
- Raw data refers to the primary data that is collected and will be processed to more understandable information.
- Data that is collected directly from the source and is not been processed, organised. Cleaned or visually presented is considered raw data.
- Once data is cleaned or organised, then this data will help to make valuable decisions.

4.9.1 Raw Data Example

Some examples, of raw data can be :

(i) Reviews of business or product,

(ii) A list of items purchased in the company

(iii) Survey responses

(iv) A list of prices for cars, hotels, real estate,

(v) Industry insights.

4.9.2 The Importance of Raw Data

(i) Before processing raw data appears to be confusing. But once the data is organised into something more useful, it can help make valuable investment decisions, help with machine learning, data science and data analysis.

(ii) Raw data is needed to create a conclusion or solution to a problem.

For example, suppose you have an E-commerce store and want to figure out what is best or worst selling product.

Here, you need to gather data like the cost of goods sold, profit, reviews and total revenue for each product. Once you gather the raw data, you shall be able to organise all your data to come up with an answer.

You may discontinue a product, improve it, spend more money advertising. Since you have firm understanding of what you need, you can create solutions.

4.9.3 Difference between Data and Raw Data

- In short, data comes from raw data.
- The collection of raw data is generally not ready for analysis, but once it is organised and cleaned up, it turns into data.
- The processed data is the type of data that is processed from raw data. Usually some kind of cleaning, transformation is performed to convert the raw data into a format that can be analysed, visualised.

4.9.4 Obtaining Raw Data

There are many ways you can gather raw data like :

(i) From the internet,

(ii) Data sets

(iii) Interviews

(iv) Data archives

- Raw data can be collected manually, but it is more efficient and effective to use a 'web scraping tool'. One can gather any type of data from any website.
- There are a handful of web scraping tools, but parse Hub has interesting features :

(i) IP rotation
(ii) Scheduling
(iii) Cloud-based scraping
(iv) Powerful
(v) Multiple export options
(vi) Dropbox integration
(vii) Many more !

4.9.5 Types of Raw Data

- Raw data refers to tables of data where each row contains an observation and each column represents a variable and that describes some property of each observation.
- Data in this format is sometimes referred to as (i) tiny data, (ii) flat data, (iii) primary data, (iv) atomic data, and (v) unit record data.

4.9.6 Processing Raw Data

Many sources produce raw data. Processing and storing of raw data depend on its source and intended use.

Examples of raw data can be financial transactions from a Point of Sale (POS) terminal, computer logs or even participant eye tracking data in a research project.

In many instances, users must clean raw data before it can be used.

- There are many ways to process raw data, and they range from simple to complex. A 'spreadsheet' such as Google sheets allows users to format, organise and graph data to reveal simple trends and help summarise data.
- More complicated systems such as Business Intelligence (BI) programs may use raw data for financial trending or forecasting purposes.

4.10 REPRESENTING TEXT

- All data on a computer system is represented using binary patterns, which are sequences of 1s and 0s. In order to represent text, each individual letter or character must be represented by a unique binary pattern.
- In virtually all computers, "alpha numeric data and special characters (i.e. letters, numbers, and symbols are each assigned a specific binary value, called a character code.
- ASCII (American Standard for Information Interchange) is a coding system for representing characters.
- All data inside a computer is transmitted as a series of electrical signals that are either 'on' or 'off'. Hence, in order for a computer to be able to process any kind of data, including text, images and sound, they must be converted into binary form.
- A code where each number represents a character can be used to convert text into binary.
- The ASCII code takes each character on the keyboard and assigns it a binary number.

For example :

(i) The letter 'a' has the binary number 01100001 (this is the denary number 97).

(ii) The letter 'b' has the binary number 01100010 (this is the denary number 98).

(iii) The letter 'c' has the binary number 01100011 (this is the denary number 99).

- Text characters start at denary number O in ASCII code, but this covers special characters including punctuation, the return key and control characters as well as the number keys, capital letters and lower case letters.
- ASCII code can only store 128 characters, which is enough for most words in English but not enough for other languages. In case of Russian alphabet and Chinese Mandarin, more characters are required.
- Therefore another code. Called 'Unicode' was created. Thus computers can be used by people using different languages.

4.10.1 Representing Images

- Images also need to be converted into 'binary', in order for a computer to process them so that they can be seen on our screen. Digital images are made up of 'pixels.' Each pixel in an image is made up of binary numbers.
- If we say that 1 is black (or on) and 0 is white (or off), then a simple black and white picture can be created using binary.
- To create the picture, a grid can be set out and the squares coloured (1 - black and 0-white). But before the grid can be created, the size of the grid is required to be known.
- This data is called 'metadata' and computers need metadata to know the size of an image.
- If the metadata for the image to be created is 10×10, this means the picture will be 10 pixels across and 10 pixels down.
- The system mentioned so far is fine for black and white images, but most images need to use colours as well. So, instead of using just 0 and 1, using four possible numbers will allow an image to use four colours. In binary this can be represented using two 'bits' per pixel :

  00 – white

  01 – blue

  10 – green

  11 – red
- Still this is not a very large range of colours, adding another binary digit will double the number of colours that are available :

  1 bit per pixel (0 or 1) : two possible colours

  2 bits per pixel (00 to 11) : four possible colours.

  3 bits per pixel (000 to 111) : eight possible colours

  4 bits per pixel (0000 to 1111) : 16 possible colour,

  16 bits per pixel

  (0000 0000 0000 0000 – 1111 1111 1111 1111);

  Over 65,000 possible colours.
- The number of bits used to store each pixel is called the 'colour depth'.
- Images with more colours need more pixels to store each available colour. This implies that images that use lots of colours are stored in larger files.

4.11 TERM FREQUENCY - INVERSE DOCUMENT FREQUENCY (TFIDF)

- TF-IDF stands for term frequency Inverse document frequency of records.
- It can be define as the calculation of how relevant a word in a series or corpus is to a text. The meaning increase proportionally to the number of times in the text a word appears but is compensated by the word frequency in the corpus (data-set).

4.11.1 Terminologies

(I) Term frequency

- In document d, the frequency represents the number of times of a given word t occurring.
- Since the ordering of the word is not significant, we can use the concept of a vector to describe the text in the bag of term models. For each specific term in the paper, there is an entry with the value being the term frequency.
- The weight of a terms that occurs in a document is simply proportional to the term frequency.

  t f(t, d) = count of t in d/number of words in d.

(II) Document frequency : (DF)

- This tests the meaning of the text and that is very similar to TF, in the whole corpus collection.

- The main difference is that in document d, TF is the frequency counter for a term t, while df is the number of occurrences in the document set N of the term t.
- That is, the number of papers in which the word is present is DF.

df(t) = occurrence of t in documents

(III) Inverse Document Frequency : (IDF)

- In particular, it tests how relevant the word is. The aim of the search is the locate the appropriate records that fit the demand.
- Since 'tf' considers all terms equally significant, therefore it uses the terms frequencies to measure the weight of the term in the paper.

First, find the document frequency of a term t by counting the number of documents containing the term :

df(t) = N(t),

where

df(t) = document frequency of a term t,

N (t) = Number of documents containing the term t.

- Term frequency is the number of instances of a term in a single document only. The frequency of the document is the number of separate documents in which the term appears, it depends on the entire corpus.
- The IDF of the word is the number of documents in the corpus separated by the frequency of the text.

$$\therefore \text{idf(t)} = \text{N/df(t)} = \frac{\text{N}}{\text{N(t)}}$$

We take the logarithm (with base 2) of the inverse frequency of the paper.

Hence the if of the term t becomes :

$$\text{idf(t)} = \log (\text{N/df(t)})$$

(IV) Computation

tf-idf metrics that determines how significant a term is to a text in a series or a corpus.

- tf-idf is a weighting system that assigns a weight to each word in a document based on its term frequency (ft) and the reciprocal document frequency (tf) (idf).
- The words with higher scores of weight are regarded as more significant.
- Generally, the tf-idf weight consists of two terms :

1. Normalised Term Frequency (tf)
2. Inverse document frequency (idf).

tf-idf(t, d) = tf (t, d) · idf (t).

4.11.2 Variants of Term Frequency (tf) Weight

Weighting scheme	tf weight
Binary	0, 1
Raw count	$f_{t, d}$
Term frequency	$f_{t, d} / \sum_{t' \in d} f_{t', d}$
Log normalisation	$\text{Log}\,(1 + f_{t, d})$
Double normalisation 0.5	$0.5 + 0.5 \frac{f_{t,d}}{\max_{t' \in d} f_{t', d}}$
Double normalisation K	$K + (1 - K) \frac{f_{t,d}}{\max_{t' \in d} f_{t', d}}$

Term frequency

- Term frequency, tf(t, d), is the relative frequency of term t within document d,

$$tf(t, d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t', d}},$$

where $f_{t,d}$ is the raw count of a term in a document, i.e., the number of times that term t occurs in document d.

4.11.3 Variants of Inverse Document Frequency (Idf) Weight

Weighting scheme	idf weight $n_t = (\lvert\{d \in D : t \in d\}\rvert)$
Unary	1
Inverse document	$\log \frac{N}{n_t} = -\log \frac{n_t}{N}$
Inverse document frequency smooth	$\log\left(\frac{N}{1+n_t}\right)+1$
Inverse document frequency max	$\log\left(\frac{\max_{t' \in d} n_{t'}}{1+n_t}\right)$
Probabilistic inverse document frequency	$\log \frac{N-n_t}{n_t}$

- The inverse document frequency measures the amount of information the word provides, i.e., if it is common or rare across all documents.
- It is the logarithmically scaled inverse traction of the documents that contain the word.

$$\text{idf}(t, D) = \log \frac{N}{\lvert\{d \in D ; t \in d\}\rvert}$$

Where,

N= total number of documents in the corpus $N = \lvert D \rvert$

$\lvert\{d \in D : t \in d\}\rvert$ = number of documents where the term t appears (i.e., $\text{tf}(t, d) \neq 0$).

- If the term is not in the corpus, it will lead to a division by zero. Hence it is common to adjust the denominator to

$1 + \lvert\{d \in D ; t \in d\}\rvert$

4.11.4 Variants of Term Frequency - Inverse Document Frequency (tf-idf) Weights

Weighting scheme	tf – idf
Count - idf	$f_{t,d} \cdot \log \frac{N}{n_t}$
double normalisation - idf	$\left(0.5 + 0.5 \frac{f_{t,q}}{\max_t f_{t,q}}\right) \cdot \log \frac{N}{n_t}$
log normalisation - idf	$(1 + \log f_{t,d}) \cdot \log \frac{N}{n_t}$

Then tf–idf is calculated as

$\text{tf idf}(t, d, D) = \text{tf}(t, d) \cdot \text{idf}(t, D)$

4.12 CATEGORISING DOCUMENTS BY TOPICS

- Document classification is the act of labeling document using categories, depending on their content.
- Document classification can also be manual, for example in library science, or automated within the field of computer science).
- It is used to sort and manage texts, videos or images.
- The advantage of classifying documents is that humans have full control over the process of classification, and they can also make decisions as to which categories to use.
- But when there is large handling of volumes of documents, then this process is very slow and monotonous.
- But it is much faster, more cost-efficient, and more accurate, to carry out automatic document classification, when it is carried out by machine learning.

4.12.1 Document Classification Vs Text Classification

- Text classification involes classifying text by performing text analysis techniques on text-based documents.
- Analysing text at different levels can be done with text classification :

(i) **Document-level :** Here one can get relevant information for a full document.

(ii) **Paragraph level :** Here one can get the most important categories of just one paragraph.

(iii) **Sentence level :** Relevant information of a single sentence can be obtained.

(iv) **Sub-sentence level :** Here relevant information of s ub-expressions within sentences can be had.

- When there are ambiguous sentences that mention varied topics, this method is useful.

Question arises, which is better ?

Of course, no straight answer is possible.

- The choice will depend on the data and the objectives.

4.12.2 Working of Automatic Document Classification

- Classifying large volumes of documents is essential so that we can obtain valuable insights humanly it is very difficult and tedious to manage large volumes of data.
- Here, automatic document classification is a great option. Using Natural Language Processing (NLP) and machine learning algorithms, we can assign one or more categories to huge amounts of text. Machine learning tools are faster, less biased and scalable than manual classification.
- We consider three different approaches to document classification :

(1) Supervised

- Here, we define a set of tags, e.g. Usability, pricing and customer service and manually tag a number of texts.
- This way machine learning models can start making predictions on their own.
- For example, a customer comment says "This Lap-Top is quite expensive,". This comment is to be tagged as 'pricing'.
- More and more texts to be classified to have better confidence of the model.

(2) Unsupervised

- In this method, similar words or similar sentences in the documents are grouped together by a classifier. This is done at random, i.e., without any prior training.
- For example, the words Mohan, Printer, Geeta would be considered as sharing similar qualities, and are grouped in the same cluster.

(3) Rule-based

- The method is based on linguistic rules and they give instructions to models.
- The rules and patterns based on morphology. Syntax, semantics and phonology, tag the texts.
- For example.

  (Update |OS| Bugs) → Software
- Here, the model will tag any text that mentions these terms as 'Software.'
- The main advantage of this method is that the performance of the model is constantly improving, that way it provides higher quality and more accurate insights.
- But the disadvantage is that creating this type of system is complex, hard to scale and time consuming. And to analyse a new type of text, will have to add new rules or change existing one every time.

4.12.3 Need to Use Document Classification

- Businesses are scaling in unstructured data and we have to use AI tools to make sense of it. Here automatic document classification can be helpful :

(i) **Triaging :** Here document classification can automatically sort articles or texts and route them to a relevant team.

- For example, if you are working in a software company and you use document classification to tag incoming support tickets.
- If you label a new ticket as 'Bug' it would be automatically routed to the technical team.

(ii) **Identification :** Automated classification can be used to identify the language, topic – for

example, texts that are suitable for different age groups, or interests.

(iii) **Analytics :** Automated classification can help for monitoring information, for example comments related to public health on social media or problems with your service or product.

4.12.4 Document Classification Through AI

- For automated document classification, there are two steps for preparing the dataset and training the algorithm.

We mention them :

(I) to gather dataset

- The dataset should contain enough documents or examples for each category so that the algorithm can learn to differentiate between them.
- For example, if you want to classify document into five categories, for training a classifier, there should be at least 100-300 documents per category to obtain predictive capabilities. So that the total number of documents within the dataset for training this classifier would be more than 500. Note that more the data we use, more accurate the classifier will be.
- Also the quality of the data is crucial when training a classifier with machine learning.
- If the examples that are fed to the classifier are incorrectly tagged, the model will commit similar errors whenever making predictions.

(II) Training the Algorithm

- Once we get the data to train the model, we use that data to train a classification algorithm. There are many complex algorithms we can use, for example, Naive Bayes and support vector machines.
- Knowing how to code, we can use open source tools such as scikit-learn or tensor flow to train these algorithms to classify the documents.

(III) To wrap Up

- It is better to begin with 'machine learning' for effective document classification. There are many classification tools that make it easy to begin with AI for document classification.

4.13 DETERMINING SENTIMENTS

- 'Sentiment analysis' is the process of classifying whether a block of text is positive, negative, or, neutral.
- Sentiment analysis is contextual mining of words. It indicates the social sentiment of a brand. It also helps the business to determine whether the product, that they are going to launch, will make the demand in the market or not.
- Sentiment analysis tries to analyse people's opinion to help business expand. It concentrates not only positive, negative and neutral, but also on emotions, like happy, sad, angry etc. It uses various 'natural language processing' algorithms like Rule-based, Automatic and Hybrid.
- We can use sentiment analysis to monitor product's reviews like whether the product is satisfying customer requirements, etc.
- When there is a large set of unstructured data, and we want to classify that data by tagging it, sentiment analysis monitors it will.

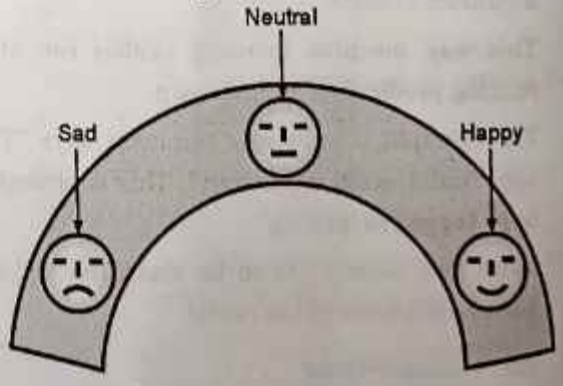


Fig. 4.13.1

4.13.1 Performing Sentiment Analysis

- Around 80% of the world's data is unstructured. The data needs to be analysed and be brought to structured manner. If may be in the form of emails, texts, documents, articles and so on.

(i) Sentiment analysis stores data in an efficient, cost-friendly manner.

(ii) Sentiment analysis solves real-time issues and can help you to solve all the real-time problems.

4.13.2 Types of Sentiment Analysis

1. **Fine-grained sentiment analysis :** This category can be designed as very positive, positive, neutral, negative, very negative. The rating is done on the scale 1 to 5. If the rating is five, then it is very positive, 2 then negative and 3 then neural.
2. **Emotion detection :** Under emotion detection comes the sentiment like happy, sad, anger, upset, jolly, pleasant, and so on. It is also known as a lexicon method of sentiment analysis.
3. **Aspect based sentiment analysis :** Here, it focuses on a particular or a special aspect. For example, if a person wants to check the feature of tablet then he checks the aspect such as battery, screen, camera quality etc.
4. **Multilingual sentiment analysis :** Multilingual, as the name suggests, consists of different languages. Here the classification is done as positive, negative and neutral. This is a challenging and comparatively difficult task.

4.13.3 Working of Sentiment Analysis

There are three approaches used :

1. Rule-based approach

- In this approach, it counts the number of positive and negative words in the given dataset.
- If the number of positive words is greater than the negative words, then the sentiment is positive else otherwise.
- Here the lexicon method, tokenization, parsing come n the rule-based.

2. Automatic Approach

- This approach is based on machine-learning technique.
- Firstly, the datasets are trained and predictive analysis is done. Then extraction of words from the text is done. This text extraction is done using techniques such as Naive Bayes, Linear Regression, Support Vector, Deep Learning etc.

3. Hybrid Approach

- This approach is the combination of both the above approaches That is rule-based and automatic approach.
- The main point here is the surplus is that accuracy is high compared to the other two approaches.

4.13.4 Applications

Sentiment analysis has a wide range of applications, such as :

(i) **Social media :** If the comments on social media side as instagram, here all the reviews are analysed and categorised as positive, negative, and neutral.

(ii) **Customer service :** All the comments in the form of 1 to 5 are done with the help of sentiment analysis approaches.

(iii) **Marketing sector :** In the marketing are where a particular product needs to be reviewed as good or bad.

(iv) **Reviewer side :** All the reviewers note the comments and check and give the overall review of the product.

4.13.5 Challenges of Semantic Analysis

Major challenges in sentiment analysis approach are :

(i) If the data is in the form of a 'tone', then it is difficult to note whether the comment is 'optimistic' or 'pessimistic'.

(ii) If the data is in the form of 'emoji' then one has to detect whether it is good or bad.

(iii) Sometimes ironic, sarcastic, comparing comments are difficult to understand.

(iv) Comparing a neutral statement is a big task.

4.14 GAINING IN SIGHTS

- Information comes in different forms and formats. Data can also be in any kind of structured or unstructured sources.
- The first step is to compile all the data including business data, scientific data, metrics and performance data, and data from social media platforms.

4.14.1 Pre-process Data

- After compiling the data, we need to reformat the data so that it becomes convenient for machine learning processing. For this purpose, we have to perform decomposing, filtering or normalisation on the data.

4.14.2 Cleaning Up Data : To Make Sense

- Even after pre-processing, there may remain some imperfections. It might be inconsistent, dirty or missing a few important values. One has to manually go through all the data values to find and rectify any inconsistencies.
- This step requires ample time and effort to clean it completely for the actual analysis phase.

4.14.3 Strategic Data Analysis

- When the data cleaned carefully and transformed, it is required to use data visualisation and statistical methods to uncover patterns in the data.
- Clustering is a common machine learning technique used for statistical data analysis.
- It separates all the data points into different groups based on their common properties and features.

4.14.4 Algorithms for Predictive Analysis

- Now we pick an appropriate machine learning model to make predictions about insights and future trends. The accurate model depends on the type of input given and the output that we require. Generally few selected models are implemented to see which one produces the most accurae results.

4.14.5 To Validate the Predictions

- Now we have to make ourselves sure that the predictions are accurate.
- It is important to evaluate and identify which model produces the best results for the given data set.
- To identity the right model for the most accurate data insights, we evaluate the performance of different machine learning models. And we get the ideal machine learning model and its accurate data predications.

4.14.6 Data-Driven Decisions

- We can transform the results into visual forecasting models or even decision trees so that the data can be easily understood by all the stakeholders involved and make better business decisions.

4.14.7 Insights of Data Analytics

The benefits of using data analytics :

(i) Personalise the customer experience,

(ii) Inform business-decision making,

(iii) Streamline operations.

(iv) Mitigate risk and handle setbacks.

(v) Enhance security.

4.14.8 Valuable Insights

To use data analytics to gain valuable insights :

(i) Let the objectives shape the Data analytics; not the other ways round.

(ii) Once the objective are identified, the data strategy can begin.

(iii) Start small and buid the data up.

(iv) Conclusion.

(v) Keep the eye on the prize. Determine measurable business results.

(vi) Know the source-start with the data that is available

(vii) Evaluate the users - find out who will be using the platfrom.

(viii) Maintain existing work flows.

...Chapter Ends

□□□

Module 5

CHAPTER 5

Data Analytics and Visualization with R

University Prescribed Syllabus

Introduction to R : Data Import and Export, Attribute and Data type, Descriptive statistics.

Exploratory Data Analysis : Visualization before analysis, DirtyData, visualizing single variable, examining Multiple variable, Data Exploration versus presentation.

5.1 INTRODUCTION TO R

GQ. What is R Programming explain in detail ?

- R is a programming language and software environment for statistical analysis, graphics representation and reporting. R was created by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand, and is currently developed by the R Development Core Team.
- R is freely available under the GNU General Public License, and pre-compiled binary versions are provided for various operating systems like Linux, Windows and Mac.
- This programming language was named R, based on the first letter of first name of the two R authors (Robert Gentleman and Ross Ihaka), and partly a play on the name of the Bell Labs Language S.
- R is the most popular data analytics tool as it is open-source, flexible, offers multiple packages and has a huge community.

GQ. Why Do We Need Analytics ?

Before an answer to above question, let us see some of the problems and their solutions in R in multiple domains.

1. **Banking :** Large amount of customer data is generated every day in Banks. While dealing with millions of customers on regular basis, it becomes hard to track their mortgages.
2. **Solution :** R builds a custom model that maintains the loans provided to every individual customer which helps us to decide the amount to be paid by the customer over time.
3. **Insurance :** Insurance extensively depends on forecasting. It is difficult to decide which policy to accept or reject.
   - By using the continuous credit report as input, we can create a model in R that will not only assess risk appetite but also make a predictive forecast as well.
4. **Healthcare :** Every year millions of people are admitted in hospital and billions are spent annually just in the admission process.

5.1.1 R Programming

Given the patient history and medical history, a predictive model can be built to identify who is at risk for hospitalization and to what extent the medical equipment should be scaled.

(I) What is Business Analytics ?

- Business analytics is a process of examining large sets of data and achieving hidden patterns, correlations and other insights. It basically helps you understand all the data that you have gathered, be it organizational data, market or product research data or any other kind of data.
- It becomes easy for you to make better decisions, better products, better marketing strategies etc.

Refer to the below image for better understanding :

- Now, if you want something specific such as a particular record in a database, it becomes cumbersome. To simplify this, you need analysis. With analysis, it becomes easy to strike a correlation between the data. Once you have established what to do, it becomes quite easy for you to make decisions such as, which path you want to follow or in terms of business analytics, which path will lead to the betterment of your organization.
- But you can't expect people in the chain above to always understand the raw data that you are providing them after analytics.

- So to overcome this gap, we have a concept of data visualization.

(II) Data visualization

- Data visualization is a visual access to huge amounts of data that you have generated after analytics. The human mind processes visual images and visual graphics are better than compare to raw data.
- It's always easy for us to understand a pie chart or a bar graph compare to raw numbers. Now you may be wondering how you can achieve this data visualization from the data you have already analyzed.

(III) Why R ?

- R is a programming and statistical language.
- R is used for data Analysis and Visualization.
- R is simple and easy to learn, read and write.
- R is an example of a FLOSS (Free Libre and Open Source Software) where one can freely distribute copies of this software, read its source code, modify it, etc.

(IV) Who uses R ?

- The Consumer Financial Protection Bureau uses R for data analysis
- Statisticians at John Deere use R for time series modeling and geospatial analysis in a reliable and reproducible way.
- Bank of America uses R for reporting.
- R is part of technology stack behind Four square's famed recommendation engine.
- ANZ, the fourth largest bank in Australia, using R for credit risk analysis.
- Google uses R to predict Economic Activity.
- Mozilla, the foundation responsible for the Firefox web browser, uses R to visualize Web activity.

5.1.2 Evolution of R

- R is an implementation of S programming language which was created by John Chambers at Bell Labs.
- R was initially written by Ross Ihaka and Robert Gentleman at the Department of Statistics of the University of Auckland in Auckland, New Zealand.
- R made its first public appearance in 1993.
- A large group of individuals has contributed to R by sending code and bug reports. Sincemid-1997 there has been a core group (the "R Core Team") who can modify the R source code archive.
- In the year 2000 R 1.0.0 released.
- R 3.0.0 was released in 2013.

5.1.3 Features of R

1. R supports procedural programming with functions and object-oriented programming with generic functions. Procedural programming includes procedure, records, modules, and procedure calls. While object-oriented programming language includes class, objects, and functions.
2. Packages are part of R programming. Hence, they are useful in collecting sets of R functions into a single unit.
3. R is a well-developed, simple and effective programming language which includes conditionals, loops, user defined recursive functions and input and output facilities.
4. R has an effective data handling and storage facility, R provides a suite of operators for calculations on arrays, lists, vectors and matrices.
5. R provides a large, coherent and integrated collection of tools for data analysis. It provides graphical facilities for data analysis and display either directly at the computer or printing at the papers.

6. Rs programming features include database input, exporting data, viewing data, variable labels, missing data, etc.
7. R is an interpreted language. So we can access it through command line interpreter.
8. R supports matrix arithmetic.
9. R, SAS, and SPSS are three statistical languages. Of these three statistical languages, R is the only an open source.
10. As a conclusion, R is world's most widely used statistics programming language. It is a good choice of data scientists and supported by a vibrant and talented community of contributors.

5.2 DATA IMPORT AND EXPORT

GQ. Explain Data Import and Export in R ?

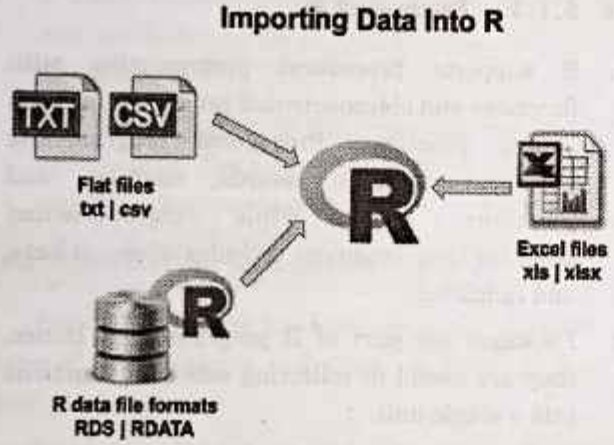


Fig. 5.2.1 : Importing Data into R

- The easiest form of data to import into R is a simple text file, and this will often be acceptable for problems of small or medium scale. The primary function to import from a text file is scan, and this underlies most of the more convenient functions discussed in Spreadsheet-like data.
- However, all statistical consultants are familiar with being presented by a client with a memory stick (formerly, a floppy disc or CD-R) of data in some proprietary binary format, for example 'an Excel spreadsheet' or 'an SPSS file'. Often the simplest thing to do is to use the originating application to export the data as a text file (and statistical consultants will have copies of the most common applications on their computers for that purpose). However, this is not always possible, and Importing from other statistical systems discusses what facilities are available to access such files directly from R. For Excel spreadsheets, the available methods are summarized in Reading Excel spreadsheets.
- In a few cases, data have been stored in a binary form for compactness and speed of access. One application of this that we have seen several times is imaging data, which is normally stored as a stream of bytes as represented in memory, possibly preceded by a header. Such data formats are discussed in Binary files and Binary connections.
- For much larger databases it is common to handle the data using a database management system (DBMS). There is once again the option of using the DBMS to extract a plain file, but for many such DBMSs the extraction operation can be done directly from an R package: See Relational databases. Importing data via network connections is discussed in Network interfaces.
- Unless the file to be imported from is entirely in ASCII, it is usually necessary to know how it was encoded. For text files, a good way to find out something about its structure is the file command-line tool (for Windows, included in Rtools). This reports something like

text.Rd: UTF-8 Unicode English text
text2.dat: ISO-8859 English text
text3.dat: Little-endian UTF-16 Unicode English character data,
with CRLF line terminators
intro.dat: UTF-8 Unicode text
intro.dat: UTF-8 Unicode (with BOM) text

- Modern Unix-alike systems, including macOS, are likely to produce UTF-8 files. Windows may produce what it calls 'Unicode' files (UCS-2LE or just possibly UTF-16LE1). Otherwise most files will be in a 8-bit encoding unless from a Chinese/Japanese/Korean locale (which have a wide range of encodings in common use). It is not possible to automatically detect with certainty which 8-bit encoding (although guesses may be possible and file may guess as it did in the example above), so you may simply have to ask the originator for some clues (e.g. 'Russian on Windows').

Exporting Data From R

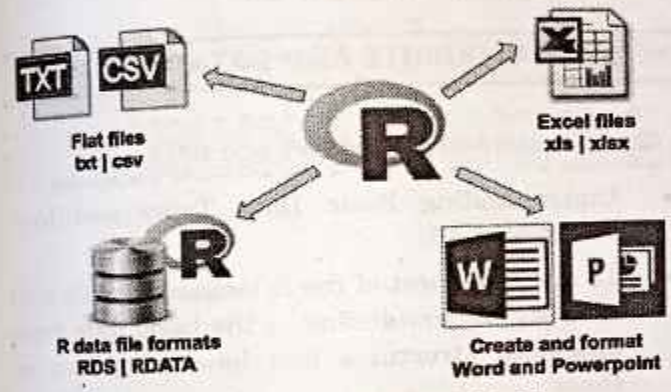


Fig. 5.2.2 : Exporting Data into R

- Exporting results from R is usually a less contentious task, but there are still a number of pitfalls. There will be a target application in mind, and often a text file will be the most convenient interchange vehicle. (If a binary file is required, see Binary files.)
- Function cat underlies the functions for exporting data. It takes a file argument, and the append argument allows a text file to be written via successive calls to cat. Better, especially if this is to be done many times, is to open a file connection for writing or appending, and cat to that connection, then close it.
- The most common task is to write a matrix or data frame to file as a rectangular grid of numbers, possibly with row and column labels. This can be done by the functions write.table and write. Function write just writes out a matrix or vector in a specified number of columns (and transposes a matrix). Function write.table is more convenient, and writes out a data frame (or an object that can be coerced to a data frame) with row and column labels.
- There are a number of issues that need to be considered in writing out a data frame to a text file.

(I) Precision

- Most of the conversions of real/complex numbers done by these functions is to full precision, but those by write are governed by the current setting of options(digits).
- For more control, use format on a data frame, possibly column-by-column.

(II) Header line

- R prefers the header line to have no entry for the row names, so the file looks like

  dist climb time

  Greenmantle 2.5 650 16.083
- Some other systems require a (possibly empty) entry for the row names, which is what write.table will provide if argument col. names = NA is specified. Excel is one such system.

(III) Separator

- A common field separator to use in the file is a comma, as that is unlikely to appear in any of the fields in English-speaking countries. Such files are known as CSV (comma separated values) files, and wrapper function write.csv provides appropriate defaults. In some locales the comma is used as the decimal point (set this in write.table by dec = ",") and there CSV files use the semicolon as the field separator: use write.csv2 for appropriate defaults.
- So far the operations using R program are done on a prompt/terminal which is not stored

anywhere. But in the software industry, most of the programs are written to store the information fetched from the program. One such way is to store the fetched information in a file. So the two most common operations that can be performed on a file are :

(IV)

- Importing Data to R scripts
- Exporting Data from R scripts
- Exporting Data from R Scripts
- When a program is terminated, the entire data is lost. Storing in a file will preserve one's data even if the program terminates. If one has to enter a large number of data, it will take a lot of time to enter them all. However, if one has a file containing all the data, he/she can easily access the contents of the file using a few commands in R.
- One can easily move his data from one computer to another without any changes. So those files can be stored in various formats. It may be stored in .txt(tab-separated value) file, or in a tabular format i.e. .csv(comma-separated value) file or it may be on internet or cloud. R provides very easier methods to export data to those files.

(A) Exporting data to a text file

- One of the important formats to store a file is in a text file. R provides various methods that one can export data to a text file.

  write.table(): The R base function write.table() can be used to export a data frame or a matrix to a text file.

Syntax

write.table(x, file, append = FALSE, sep = " ", dec = ".", row.names = TRUE, col.names = TRUE)

(B) Parameters

- x: a matrix or a data frame to be written.
- **file :** a character specifying the name of the result file.
- **sep :** the field separator string, e.g., sep = "\t" (for tab-separated value).
- **dec :** the string to be used as decimal separator. Default is "."
- **row.names :** either a logical value indicating whether the row names of x are to be written along with x, or a character vector of row names to be written.
- **col.names :** either a logical value indicating whether the column names of x are to be written along with x, or a character vector of column names to be written.

5.3 ATTRIBUTE AND DATA TYPE

GQ. Explain Attribute and Data type in R ?

- Understanding Basic Data Types and Data Structures in R
- To make the best of the R language, you'll need a strong understanding of the basic data types and data structures and how to operate on them.
- Data structures are very important to understand because these are the objects you will manipulate on a day-to-day basis in R. Dealing with object conversions is one of the most common sources of frustration for beginners.
- Everything in R is an object.
- R has 6 basic data types. (In addition to the five listed below, there is also raw which will not be discussed in this workshop.)

character
numeric (real or decimal)
integer
logical
complex

- Elements of these data types may be combined to form data structures, such as atomic vectors. When we call a vector atomic, we mean that the

vector only holds data of a single data type. Below are examples of atomic character vectors, numeric vectors, integer vectors, etc.

```
character: "a", "swc"
numeric: 2, 15.5
integer: 2L (the L tells R to store this as an integer)
logical: TRUE, FALSE
complex: 1+4i (complex numbers with real and imaginary parts)
```

- R provides many functions to examine features of vectors and other objects, for example

```
class() - what kind of object is it (high-level)?
typeof() - what is the object's data type (low-level)?
length() - how long is it? What about two dimensional objects?
attributes() - does it have any metadata?
# Example
x <- "dataset"
typeof(x)
[1] "character"
attributes(x)
NULL
y <- 1:10
y
[1] 1 2 3 4 5 6 7 8 9 10
typeof(y)
[1] "integer"
length(y)
[1] 10
z <- as.numeric(y)
z
[1] 1 2 3 4 5 6 7 8 9 10
typeof(z)
[1] "double"
```

- R has many data structures. These include

```
atomic vector
list
matrix
data frame
factor
```

☞ Vectors

A vector is the most common and basic data structure in R and is pretty much the workhorse of R.

Technically, vectors can be one of two types :

atomic vectors

lists

although the term "vector" most commonly refers to the atomic types not to lists.

5.3.1 The Different Vector Modes

- A vector is a collection of elements that are most commonly of mode character, logical, integer or numeric.
- You can create an empty vector with vector(). (By default the mode is logical. You can be more explicit as shown in the examples below.) It is more common to use direct constructors such as character(), numeric(), etc.
- Each variable in R has an associated data type. Each data type requires different amounts of memory and has some specific operations which can be performed over it. R Programming language has the following basic data types and the following table shows the data type and the values that each data type can take.

5.3.2 Data Types in R Programming Language

Basic Data Types	Values
Numeric	Set of all real numbers
Integer	Set of all integers, Z
Logical	TRUE and FALSE
Complex	Set of complex numbers
Character	"a", "b", "c", ..., "@", "#", "$", ..., "1", "2", ...etc

☞ Numeric Datatype

Decimal values are called numerics in R. It is the default data type for numbers in R. If you assign a decimal value to a variable x as follows, x will be of numeric type.

5.4 DESCRIPTIVE STATISTICS

GQ. Explain Descriptive statistics ?

In Descriptive analysis, we are describing our data with the help of various representative methods like using charts, graphs, tables, excel files, etc. In the descriptive analysis, we describe our data in some manner and present it in a meaningful way so that it can be easily understood. Most of the time it is performed on small data sets and this analysis helps us a lot to predict some future trends based on the current findings. Some measures that are used to describe a data set are measures of central tendency and measures of variability or dispersion.

(I) Process of Descriptive Analysis

(II) The measure of central tendency

(III) Measure of variability

(I) Measure of central tendency

- It represents the whole set of data by a single value. It gives us the location of central points. There are three main measures of central tendency :

  (i) Mean (ii) Mode (iii) Median

(II) Measure of variability

- Measure of variability is known as the spread of data or how well is our data is distributed. The most common variability measures are:

  (i) Range (ii) Variance

  (iii) Standard deviation

(III) Need of Descriptive Analysis

- Descriptive Analysis helps us to understand our data and is a very important part of Machine Learning.
- This is due to Machine Learning being all about making predictions. On the other hand, statistics is all about drawing conclusions from data, which is a necessary initial step for Machine Learning. Let's do this descriptive analysis in R.

☞ **Descriptive Analysis in R**

- Descriptive analyses consist of describing simply the data using some summary statistics and graphics.
- Here, we'll describe how to compute summary statistics using R software.

☞ **Import your data into R**

- Before doing any computation, first of all, we need to prepare our data, save our data in external .txt or .csv files and it's a best practice to save the file in the current directory. After that import, your data into R as follow :

```
# R program to illustrate
# Descriptive Analysis

# Import the data using read.csv()
myData = read.csv("CardioGoodFitness.csv",
stringsAsFactors = F)
# Print the first 6 rows
print(head(myData))
```

- It allows to check the quality of the data and it helps to "understand" the data by having a clear overview of it. If well presented, descriptive statistics is already a good starting point for further analyses.
- There exists many measures to summarize a dataset. They are divided into two types :

  Location measures and dispersion measures.
- Location measures give an understanding about the central tendency of the data, whereas dispersion measures give an understanding about the spread of the data.
- In this article, we focus only on the implementation in R of the most common descriptive statistics and their visualizations (when deemed appropriate).

5.5 EXPLORATORY DATA ANALYSIS

GQ. Explain Exploratory Data Analysis ?

- One of the first steps of any data analysis project is exploratory data analysis.

This involves exploring a dataset in three ways :

1. Summarizing a dataset using descriptive statistics.
2. Visualizing a dataset using charts.
3. Identifying missing values.

- By performing these three actions, you can gain an understanding of how the values in a dataset are distributed and detect any problematic values before proceeding to perform a hypothesis test or perform statistical modeling.
- The easiest way to perform exploratory data analysis in R is by using functions from the tidy verse packages.
- The following step-by-step example shows how to use functions from these packages to perform exploratory data analysis on the diamonds dataset that comes built-in with the tidy verse packages.

Step 1 : Load & View the Data

First, let's use the data() function to load the diamonds data

```
library(tidyverse)

#load diamonds dataset
data(diamonds)
```

We can take a look at the first six rows of the dataset by using the head() function :

```
#view first six rows of diamonds dataset
head(diamonds)

  carat cut       color clarity depth table price    x    y    z
1  0.23 Ideal     E     SI2     61.5    55   326 3.95 3.98 2.43
2  0.21 Premium   E     SI1     59.8    61   326 3.89 3.84 2.31
3  0.23 Good      E     VS1     56.9    65   327 4.05 4.07 2.31
4  0.290 Premium  I     VS2     62.4    58   334 4.2  4.23 2.63
5  0.31 Good      J     SI2     63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good J     VVS2    62.8    57   336 3.94 3.96 2.48
```

Step 2 : Summarize the Data

We can use the **summary()** function to quickly summarize each variable in the dataset :

```
#summarize diamonds dataset
summary(diamonds)
carat cut  color clarity depth
Min.:0.2000 Fair : 1610  D: 6775  SI1  :13065  Min.
:43.00
 1st Qu.:0.4000  Good  : 4906  E: 9797  VS2  :12258
1st Qu.:61.00
Median :0.7000  Very Good:12082  F: 9542  SI2  : 9194
Median :61.80
 Mean  :0.7979  Premium :13791  G:11292  VS1  : 8171
Mean  :61.75
 3rd Qu.:1.0400  Ideal  :21551  H: 8304  VVS2  : 5066
3rd Qu.:62.50
 Max.  :5.0100  I: 5422  VVS1  : 3655  Max.  :79.00
  J: 2808  (Other): 2531
table  price  x  y z
Min.:43.00  Min.  :  326  Min.  : 0.000  Min.  : 0.000
Min.  : 0.000
 1st Qu.:56.00  1st Qu.:  950  1st Qu.: 4.710  1st Qu.:
4.720  1st Qu.: 2.910
Median :57.00  Median : 2401  Median : 5.700  Median :
5.710  Median : 3.530
 Mean  :57.46  Mean  : 3933  Mean  : 5.731  Mean  :
5.735  Mean  : 3.539
 3rd Qu.:59.00  3rd Qu.: 5324  3rd Qu.: 6.540  3rd Qu.:
6.540  3rd Qu.: 4.040
 Max.  :95.00  Max.  :18823  Max.  :10.740  Max.
:58.900  Max.  :31.800
```

For each of the numeric variables we can see the following information :

- **Min :** The minimum value.
- **1st Qu :** The value of the first quartile (25th percentile).
- **Median :** The median value.
- **Mean** : The mean value.
- **3rd Qu :** The value of the third quartile (75th percentile).
- **Max :** The maximum value.

For the categorical variables in the dataset (cut, color, and clarity) we see a frequency count of each value.

For example, for the **cut** variable :

- **Fair :** This value occurs 1,610 times.
- **Good :** This value occurs 4,906 times.
- **Very Good :** This value occurs 12,082 times.
- **Premium :** This value occurs 13,791 times.
- **Ideal :** This value occurs 21,551 times.

We can use the **dim()** function to get the dimensions of the dataset in terms of number of rows and number of columns :

```
#display rows and columns
dim(diamonds)

[1] 53940 10
```

We can see that the dataset has 53,940 rows and **10** columns.

▶ **Step 3 :** Visualize the Data

We can also create charts to visualize the values in the dataset.

For example, we can use the **geom_histogram()** function to create a histogram of the values for a certain variable:

```
#create histogram of values for price
ggplot(data=diamonds, aes(x=price)) +
geom_histogram(fill="steelblue", color="black") +
ggtitle("Histogram of Price Values")
```

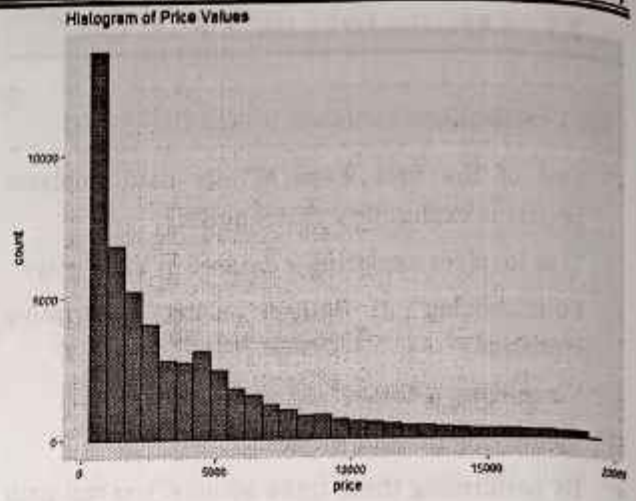


Fig. 5.5.1

We can also use the **geom_point()** function to create a scatterplot of any pairwise combination of variables :

```
#create scatterplot of carat vs. price, using cut as color
variable
ggplot(data=diamonds, aes(x=carat, y=price, color=cut)) +
geom_point()
```

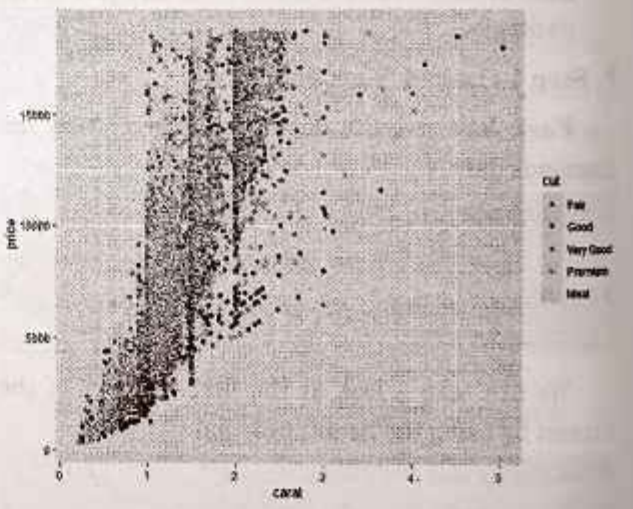


Fig. 5.5.2

We can also use the **geom_boxplot()** function to create a boxplot of one variable grouped by another variable :

```
#create scatterplot of price, grouped by cut
ggplot(data=diamonds, aes(x=cut, y=price)) +
geom_boxplot(fill="steelblue")
```

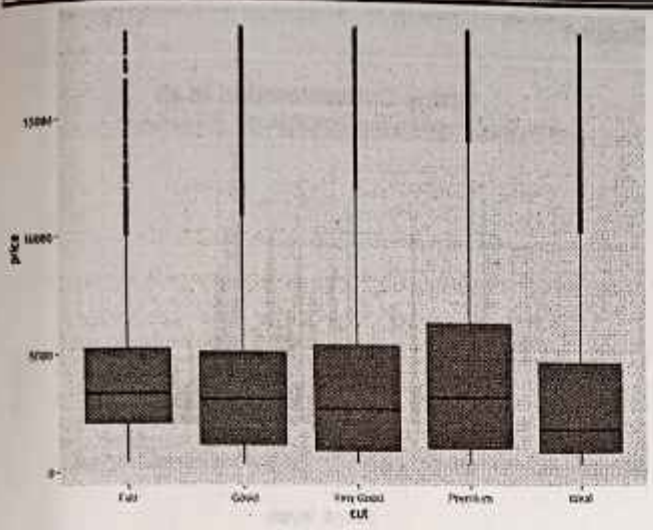


Fig. 5.5.3

We can also use the **cor()** function to create a correlation matrix to view the correlation coefficient between each pairwise combination of numeric variables in the dataset :

```
#create correlation matrix of (rounded to 2 decimal places)
round(cor(diamonds[c('carat', 'depth', 'table', 'price', 'x', 'y',
'z')]), 2)

        carat depth table price    x    y    z
carat    1.00  0.03  0.18  0.92 0.98 0.95 0.95
depth    0.03  1.00 -0.30 -0.01 -0.03 -0.03 0.09
table    0.18 -0.30  1.00  0.13  0.20  0.18 0.15
price    0.92 -0.01  0.13  1.00  0.88  0.87 0.86
x        0.98 -0.03  0.20  0.88  1.00  0.97 0.97
y        0.95 -0.03  0.18  0.87  0.97  1.00 0.95
z        0.95  0.09  0.15  0.86  0.97  0.95 1.00
```

Related : What is Considered to Be a "Strong" Correlation?

▶ **Step 4 :** Identify Missing Values

We can use the following code to count the total number of missing values in each column of the dataset :

```
#count total missing values in each column
sapply(diamonds, function(x) sum(is.na(x)))

carat   cut color clarity depth table price     x     y     z
    0     0     0       0     0     0     0     0     0     0
```

From the output we can see that there are zero missing values in each column.

5.6 VISUALIZATION BEFORE ANALYSIS

GQ. Explain Visualization before analysis ?

Data visualization is the technique used to deliver insights in data using visual cues such as graphs, charts, maps, and many others.

This is useful as it helps in intuitive and easy understanding of the large quantities of data and thereby make better decisions regarding it.

☞ Data Visualization in R Programming Language

- The popular data visualization tools that are available are Tableau, Plotly, R, Google Charts, Infogram, and Kibana. The various data visualization platforms have different capabilities, functionality, and use cases. They also require a different skill set. This article discusses the use of R for data visualization.
- R is a language that is designed for statistical computing, graphical data analysis, and scientific research. It is usually preferred for data visualization as it offers flexibility and minimum required coding through its packages.
- Consider the following air quality data set for visualization in R :

Ozone	Solar R.	Wind	Temp	Month	Day
41	190	7.4	67	5	1
36	118	8.0	72	5	2
12	149	12.6	74	5	3
18	313	11.5	62	5	4
NA	NA	14.3	56	5	5
28	NA	14.9	66	5	6

5.6.1 Types of Data Visualizations

Some of the various types of visualizations offered by Rare :

1. Bar Plot

There are two types of bar plots- horizontal and vertical which represent data points as horizontal or vertical bars of certain lengths proportional to the value of the data item. They are generally used for continuous and categorical variable plotting. By setting the **horiz** parameter to true and false, we can get horizontal and vertical bar plots respectively.

Example 1

```
R
# Horizontal Bar Plot for
# Ozone concentration in air
barplot(airquality$Ozone,
    main = 'Ozone Concenteration in air',
    xlab = 'ozone levels', horiz = TRUE)
```

Output

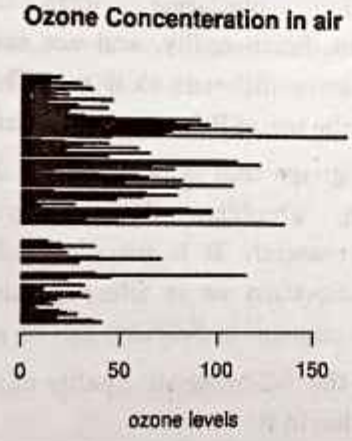


Fig. 5.6.1

Example 2

```
R
# Vertical Bar Plot for
# Ozone concentration in air
barplot(airquality$Ozone, main = 'Ozone Concenteration in
air',
    xlab = 'ozone levels', col ='blue', horiz = FALSE)
```

Output

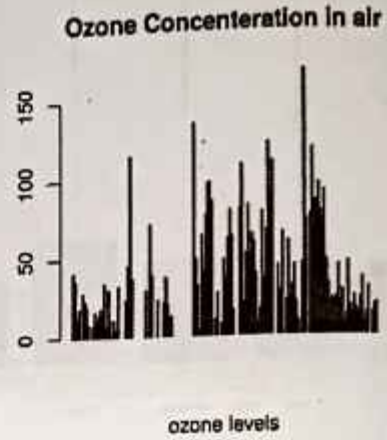


Fig. 5.6.2

Bar plots are used for the following scenarios :

- To perform a comparative study between the various data categories in the data set.
- To analyze the change of a variable over time in months or years.

2. Histogram

A histogram is like a bar chart as it uses bars of varying height to represent data distribution. However, in a histogram values are grouped into consecutive intervals called bins. In a Histogram, continuous values are grouped and displayed in these bins whose size can be varied.

Example

```
R
# Histogram for Maximum Daily Temperature
data(airquality)

hist(airquality$Temp, main ="La Guardia Airport's\
Maximum Temperature(Daily)",
  xlab ="Temperature(Fahrenheit)",
  xlim = c(50, 125), col ="yellow",
  freq = TRUE)
```

Output

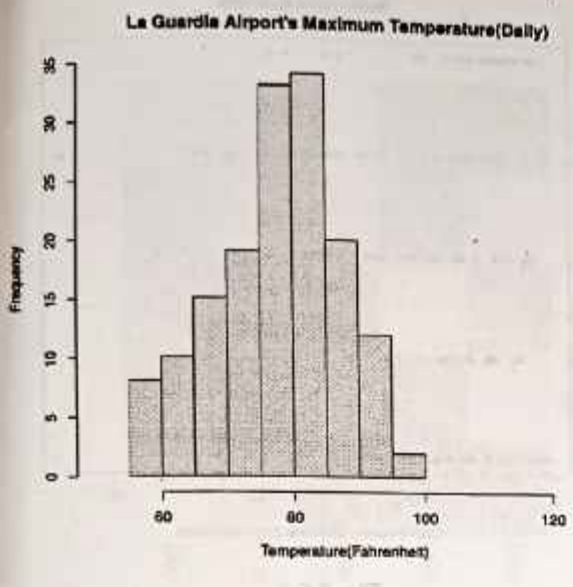


Fig. 5.6.3

- For a histogram, the parameter **xlim** can be used to specify the interval within which all values are to be displayed.
- Another parameter **freq** when set to TRUE denotes the frequency of the various values in the histogram and when set to FALSE, the probability densities are represented on the y-axis such that they are of the histogram adds up to one.

Histograms are used in the following scenarios :

- To verify an equal and symmetric distribution of the data.
- To identify deviations from expected values.

3. Box Plot

The statistical summary of the given data is presented graphically using a boxplot.

A boxplot depicts information like the minimum and maximum data point, the median value, first and third quartile, and interquartile range.

Example

```
R
# Box plot for average wind speed
data(airquality)
boxplot(airquality$Wind, main = "Average wind speed\
at La Guardia Airport",
    xlab = "Miles per hour", ylab = "Wind",
    col = "orange", border = "brown",
    horizontal = TRUE, notch = TRUE)
```

Output

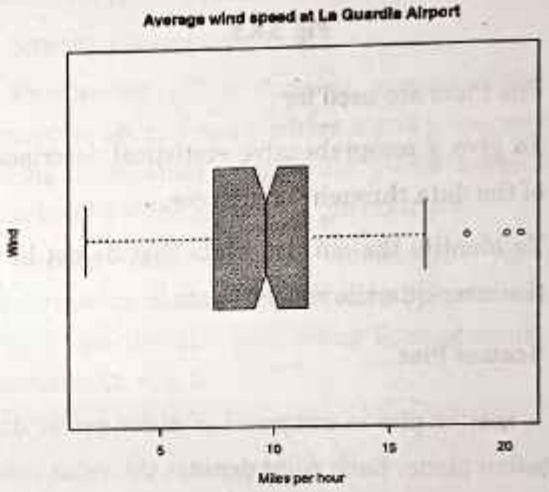


Fig. 5.6.4

Multiple box plots can also be generated at once through the following code :

Example

```
R
# Multiple Box plots, each representing
# an Air Quality Parameter
boxplot(airquality[, 0:4],
main ='Box Plots for Air Quality Parameters')
```

Output

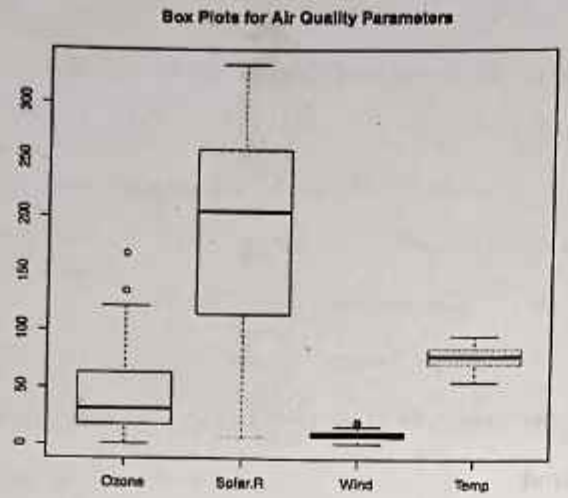


Fig. 5.6.5

Box Plots are used for :

- To give a comprehensive statistical description of the data through a visual cue.
- To identify the outlier points that do not lie in the inter-quartile range of data.

4. Scatter Plot

A scatter plot is composed of many points on a Cartesian plane. Each point denotes the value taken by two parameters and helps us easily identify the relationship between them.

Example

```
R
# Scatter plot for Ozone Concentration per month
data(airquality)

plot(airquality$Ozone, airquality$Month,
  main ="Scatterplot Example",
  xlab ="Ozone Concentration in parts per billion",
  ylab =" Month of observation ", pch = 19)
```

Output

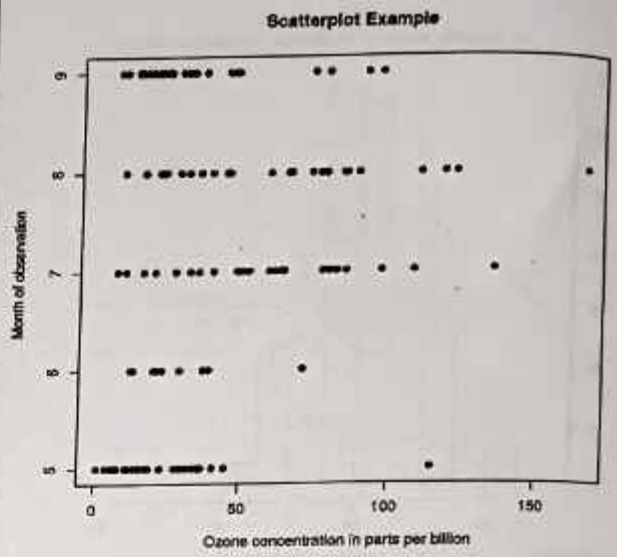


Fig. 5.6.6

Scatter Plots are used in the following scenarios :

- To show whether an association exists between bivariate data.
- To measure the strength and direction of such a relationship.

5. Heat Map

Heatmap is defined as a graphical representation of data using colors to visualize the value of the matrix. heatmap() function is used to plot heatmap.

Syntax : heatmap(data)

Parameters : data: It represent matrix data, such as values of rows and columns

Return : This function draws a heatmap.

```
R
# Set seed for reproducibility
# set.seed(110)

# Create example data
data <- matrix(rnorm(50, 0, 5), nrow = 5, ncol = 5)

# Column names
colnames(data) <- paste0("col", 1:5)
rownames(data) <- paste0("row", 1:5)

# Draw a heatmap
heatmap(data)
```

Output

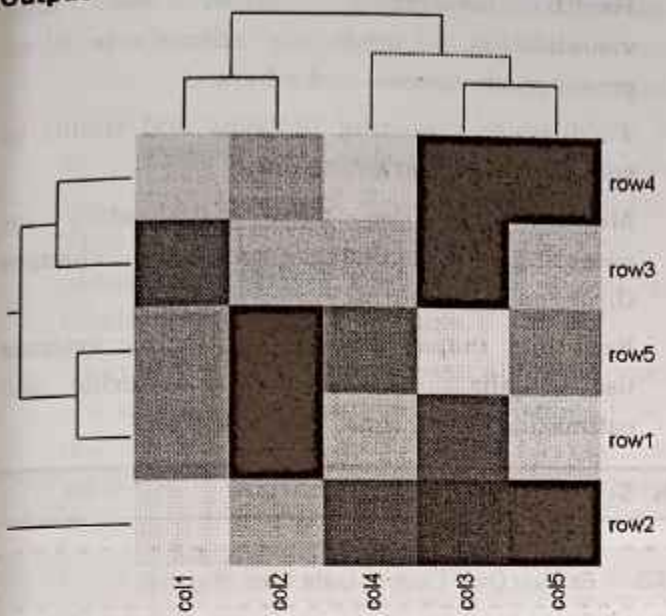


Fig. 5.6.7

☞ Map visualization in R

Here we are using maps package to visualize and display geographical maps using an R programming language.

install.packages("maps")

Link of the dataset: worldcities.csv

```
R
# Read dataset and convert it into
# Dataframe
data <- read.csv("worldcities.csv")
df<- data.frame(data)

# Load the required libraries
library(maps)
map(database = "world")

# marking points on map
points(x = df$lat[1:500], y = df$lng[1:500], col = "Red")
```

Output

Fig. 5.6.8 : 3D Graphs in R

Here we will use preps() function, This function is used to create 3D surfaces in perspective view. This function will draw perspective plots of a surface over the x–y plane.

Syntax : persp(x, y, z)

Parameter : This function accepts different parameters i.e. x, y and z where x and y are vectors defining the location along x- and y-axis. z-axis will be the height of the surface in the matrix z.

Return Value : persp() returns the viewing transformation matrix for projecting 3D coordinates (x, y, z) into the 2D plane using homogeneous 4D coordinates (x, y, z, t).

```
R
# Adding Titles and Labeling Axes to Plot
cone <- function(x, y){
sqrt(x ^ 2 + y ^ 2)
}
# prepare variables.
x <- y <- seq(-1, 1, length = 30)
z <- outer(x, y, cone)

# plot the 3D surface
# Adding Titles and Labeling Axes to Plot
persp(x, y, z,
main="Perspective Plot of a Cone",
zlab = "Height",
theta = 30, phi = 15,
col = "orange", shade = 0.4)
```

Output

Perspective Plot of a Cone

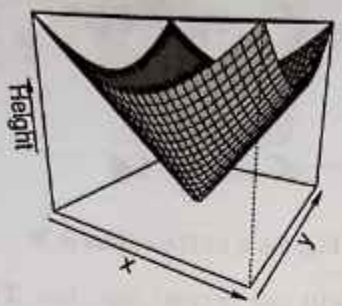


Fig. 5.6.9

5.6.2 Advantages of Data Visualization in R

R has the following advantages over other tools for data visualization :

- R offers a broad collection of visualization libraries along with extensive online guidance on their usage.
- R also offers data visualization in the form of 3D models and multipanel charts.
- Through R, we can easily customize our data visualization by changing axes, fonts, legends, annotations, and labels.

5.6.3 Disadvantages of Data Visualization in R

R also has the following disadvantages :

- R is only preferred for data visualization when done on an individual standalone server.
- Data visualization using R is slow for large amounts of data as compared to other counterparts.

5.6.4 Application Areas

- Presenting analytical conclusions of the data to the non-analysts departments of your company.
- Health monitoring devices use data visualization to track any anomaly in blood pressure, cholesterol and others.
- To discover repeating patterns and trends in consumer and marketing data.
- Meteorologists use data visualization for assessing prevalent weather changes throughout the world.
- Real-time maps and geo-positioning systems use visualization for traffic monitoring and estimating travel time.

5.7 DIRTY DATA

GQ. Explain Dirty Data or Data Cleaning in R ?

- Data Cleaning in R
- Data Cleaning is the process to transform raw data into consistent data that can be easily analyzed. It is aimed at filtering the content of statistical statements based on the data as well as their reliability.
- Moreover, it influences the statistical statements based on the data and improves your data quality and overall productivity.

5.7.1 Purpose of Data Cleaning

The following are the various purposes of data cleaning :

1. Eliminate Errors
2. Eliminate Redundancy
3. Increase Data Reliability
4. Delivery Accuracy
5. Ensure Consistency
6. Assure Completeness
7. Standardize your approach

- Overview of a typical data analysis chain
- This section represents an overview of a typical data analysis. Each rectangle in the figure represents data in a certain state while each arrow represents the activities needed to get from one state to the other.

- The first state (Raw data) is the data as it comes in. Raw data may lack headers, contain wrong data types, wrong category labels, unknown or unexpected character encoding, and so on. Once this pre-processing has taken place, data can be deemed Technically correct Data. That is, in this state data can be read into an R data.frame, with correct names, types, and labels, without further trouble.
- However, this does not mean that the values are error-free or complete. Consistent data is the stage where data is ready for statistical inference. It is the data that most statistical theories use as a starting point.
- How to clean data in R
- Here, this involves various steps, as from the initial raw data have to move toward the consistent and highly efficient data which is ready to me implement as per the requirements and produces the highly precise and accurate statistical results. The steps vary from data to data as in this case the user should be aware of the date he/she is using for the results. As there are many characteristics and common symptoms of messy data which totally depend on the data used by the user for analysis.

5.7.2 Characteristics of Clean Data include Data are

1. Free of duplicate rows/values
2. Error-free (misspellings free)
3. Relevant (special characters free)
4. The appropriate data type for analysis
5. Free of outliers (or only contain outliers that have been identified/understood)

Follows a "tidy data" structure :

Common symptoms of messy data

1. Special characters (e.g. commas in numeric values)
2. Numeric values stored as text/character data types
3. Duplicate rows
4. Misspellings
5. Inaccuracies
6. White space
7. Missing data
8. Zeros instead of null values vary.

5.7.3 Characteristics of Clean Data and Messy Data

- What exactly is clean data? Clean data is accurate, complete, and in a format that is ready to analyze. Characteristics of clean data include data that are:
- Free of duplicate rows/values
- Error-free (e.g. free of misspellings)
- Relevant (e.g. free of special characters)
- The appropriate data type for analysis
- Free of outliers (or only contain outliers have been identified/understood), and
- Follows a "tidy data" structure
- Common symptoms of messy data include data that contain :

Special characters (e.g. commas in numeric values)

- Numeric values stored as text/character data types :
  - Duplicate rows
  - Misspellings
  - Inaccuracies
  - White space
  - Missing data
  - Zeros instead of null values

5.7.4 Motivation

In this blog post, we will work with five property-sales datasets that are publicly available on the New York City Department of Finance Rolling Sales Data website.

We encourage you to download the datasets and follow along! Each file contains one year of real estate sales data for one of New York City's five boroughs. We will work with the following Microsoft Excel files :

- rollingsales_bronx.xls
- rollingsales_brooklyn.xls
- rollingsales_manhattan.xls
- rollingsales_queens.xls
- rollingsales_statenisland.xls
- As we work through this blog post, imagine that you are helping a friend launch their home-inspection business in New York City. You offer to help them by analyzing the data to better understand the real-estate market.
- But you realize that before you can analyze the data in R, you will need to diagnose and clean it first. And before you can diagnose the data, you will need to load it into R!

5.7.5 Load Data into R with readxl

- Benefits of using tidyverse tools are often evident in the data-loading process. In many cases, the tidyverse package readxl will clean some data for you as Microsoft Excel data is loaded into R. If you are working with CSV data, the tidyversereadr package function read_csv() is the function to use (we'll cover that later).
- How to clean the datasets in R?, Data cleansing is one of the important steps in data analysis. Multiple packages are available in r to clean the data sets, here we are going to explore the janitor package to examine and clean the data.
- Data cleaning is the process of transforming dirty data into reliable data that can be analyzed. Data cleansing improves your data quality and overall productivity.
- When you clean your data, all incorrect information is gone and leaving only reliable quality information.

The main functions of the Janitor package are :

1. Format ugly data frame column names
2. Isolate duplicate records in the data frame
3. Provide quick tabulations
4. Format tabulation results

5.8 VISUALIZING SINGLE VARIABLE

GQ. Explain Visualizing single variable ?

- Statistical computing is really helping us on creating a high-quality graphics. Selecting the right type of graph can help us to analyze our data better.
- It will explain how you can use R to get the best visual from a single variable data.

There are 4 types of plots that we can use to observe a single variable data : Histograms Index plots Time-series plots Pie Charts

Histograms

How to create a histogram in R? And what information that we can get from histogram? Histogram shows a frequency distribution. It is a great graph for showing the mode, the spread, and the symmetry (skewness) of your data. Here is a histogram of 1,000 random points drawn from a normal distribution with a mean of 2.5.

```
# How to create Histogram in R
# byMichaelinoMervisiano
datavar<-rnorm(1000,2.5)
hist(datavar,main="Awesome Histogram",
col="Blue",prob=TRUE,
xlab="Random Numbers from a Normal Distribution with
Mean 2.5")
```

Fig. 5.8.1 Histogram result in R

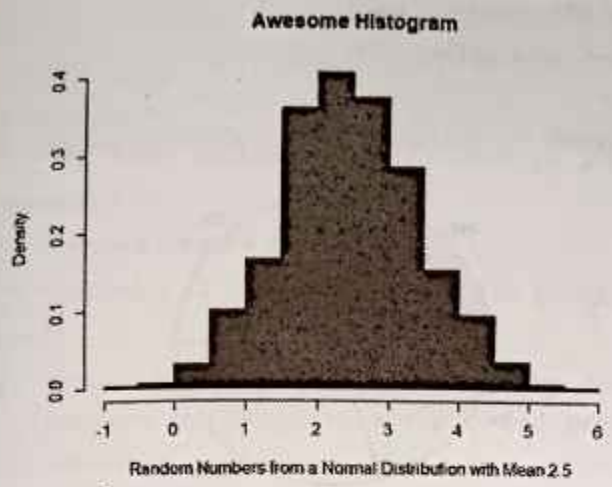


Fig. 5.8.1

Fig. 5.8.1 shows us the distribution of the data. We can see that the data is spread evenly between the left and right tail. Also, the frequency is showing us the mode of the data is around 2 and 3. Next, you can add a line below to get a density curve along your histogram

```
hist(datavar,main="Awesome Histogram",
col="Blue",prob=TRUE,
xlab="Random Numbers from a Normal Distribution with
Mean 2.5")
lines(density(datavar), col = "red")
```

Fig. 5.8.2 Histogram + Density Line

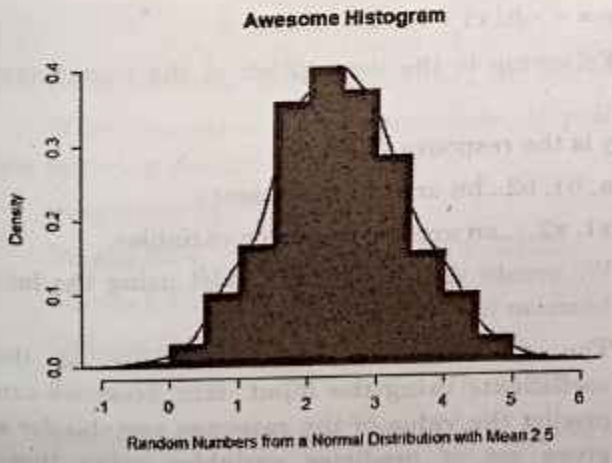


Fig. 5.8.2

Index Plots

- The other plot that is effective to analyze a single variable data is index plot. This type of plot displays a single continuous variable and plots the values on the vertical axis, while plot the order of the number in vector on the horizontal axis.
- I personally like to use this plot for error checking. For this example, I will use our favorite sample data, Iris. There are 150 observations in this data set and we will take the petal length variable as our single variable to analyze.

```
datavar<-iris$Petal.Length
plot(datavar,col="orange")
```

- Fig. 5.8.3 exhibits all observations from our single variable data. If there's an outlier in our data, then it will stand out like a sore thumb. Then, we can check if this might be related to data entry error or need to be analyzed separately.

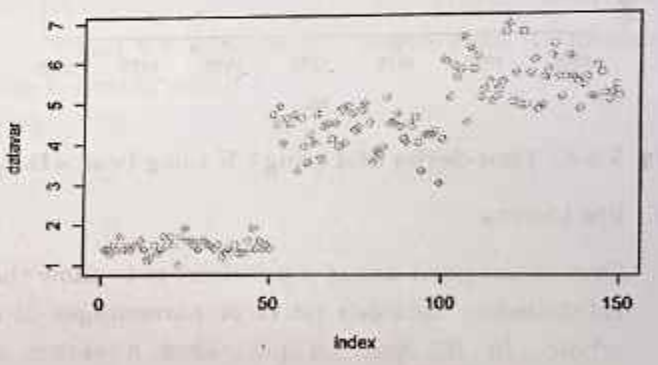


Fig. 5.8.3 : Index Plot Result using Iris Petal Length data

Time-Series Plots

If you have a complete data for time series, then it will be very straightforward to plot it. You can joining each observation in an ordered set of y values. However, the problem will occur if you have missing values in the time series.

You can use a simple interpolation or forecasting model to cope with the missing values issue. For illustration, we will use UK Lung Deaths from 1974–1980.

```
data(UKLungDeaths)
ts.plot(ldeaths, mdeaths, fdeaths,
xlab="Year", ylab="Deaths",col="purple",lty=c(1:3))
```

Fig. 5.8.4 shows three different lines: the upper, solid line shows total deaths, the heavier dashed line shows male deaths and the faint dotted line shows female deaths. We can clearly see the different number of deaths between sexes. Additionally, there is a strong seasonality effect in the data as you can easily observed number of deaths are peaking in midwinter.

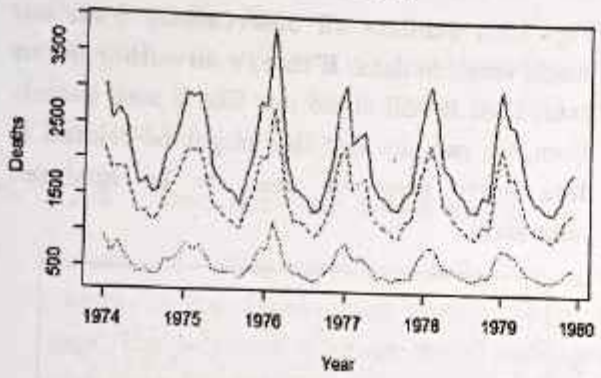


Fig. 5.8.4 : Time-Series Plot using UK Lung Deaths Data

☞ Pie Charts

- One of the good use of a pie chart is to show the relationship between parts or percentages of a whole. In R, function pie takes a vector of numbers change them into proportions and divides up the circle based on total proportion.
- For the next example, we will use Titanic (it's also my favorite movie!) sample data and see the proportion passenger class ticket. We can see easily the proportion of passengers in Fig. 5.8.5. More than one-third of the passengers are the Crew. The proportion between first and second class passengers are very close.

```
df<-data.frame(Titanic)
df<-df[df$Survived=='Yes',]
datavar<-xtabs(df$Freq~df$Class)
datavar
pie(datavar)
```

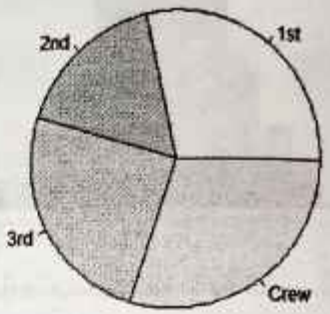


Fig. 5.8.5 : Pie Chart using Titanic Passengers Data

5.9 EXAMINING MULTIPLE VARIABLE

GQ. Explain Examining Multiple variable?

- Multiple regression is an extension of linear regression into relationship between more than two variables.
- In simple linear relation we have one predictor and one response variable, but in multiple regression we have more than one predictor variable and one response variable.
- The general mathematical equation for multiple regression is :

$$y = a + b1x1 + b2x2 + ...bnxn$$

Following is the description of the parameters used :

- y is the response variable.
- a, b1, b2...bn are the coefficients.
- x1, x2, ...xn are the predictor variables.
- We create the regression model using the lm() function in R.
- The model determines the value of the coefficients using the input data. Next we can predict the value of the response variable for a given set of predictor variables using these coefficients.

☞ lm() Function

This function creates the relationship model between the predictor and the response variable.

Syntax

The basic syntax for lm() function in multiple regression is :

lm(y ~ x1 + x2 + x3...,data)

Following is the description of the parameters used :

- formula is a symbol presenting the relation between the response variable and predictor variables.
- data is the vector on which the formula will be applied.

Example

☞ Input Data

- Consider the data set "mtcars" available in the R environment. It gives a comparison between different car models in terms of mileage per gallon (mpg), cylinder displacement("disp"), horse power("hp"), weight of the car("wt") and some more parameters.
- The goal of the model is to establish the relationship between "mpg" as a response variable with "disp","hp" and "wt" as predictor variables. We create a subset of these variables from the mtcars data set for this purpose.

```
input<- mtcars[,c("mpg","disp","hp","wt")]
print(head(input))
```

When we execute the above code, it produces the following result :

```
                   mpgdisphpwt
Mazda RX4          21.0 160  110  2.620
Mazda RX4 Wag      21.0 160  110  2.875
Datsun 710         22.8 108  93   2.320
Hornet 4 Drive     21.4 258  110  3.215
Hornet Sportabout  18.7 360  175  3.440
Valiant            18.1 225  105  3.460
```

Create Relationship Model & get the Coefficients

```
input<- mtcars[,c("mpg","disp","hp","wt")]
# Create the relationship model.
model<- lm(mpg~disp+hp+wt, data = input)
# Show the model.
print(model)
# Get the Intercept and coefficients as vector elements.
cat("# # # # The Coefficient Values # # # ","\n")

a <- coef(model)[1]
print(a)
Xdisp<- coef(model)[2]
Xhp<- coef(model)[3]
Xwt<- coef(model)[4]
print(Xdisp)
print(Xhp)
print(Xwt)
```

When we execute the above code, it produces the following result :

```
Call:
lm(formula = mpg ~ disp + hp + wt, data = input)
Coefficients:
(Intercept)        disphpwt
 37.105505     -0.000937      -0.031157   -3.800891
# # # # The Coefficient Values # # #
(Intercept)
  37.10551
disp
-0.0009370091
hp
-0.03115655
wt
-3.800891
```

Create Equation for Regression Model

Based on the above intercept and coefficient values, we create the mathematical equation.

$Y = a + Xdisp.x1 + Xhp.x2 + Xwt.x3$

or

$Y = 37.15 + (-0.000937) * x1 + (-0.0311) * x2 + (-3.8008) * x3$

Apply Equation for predicting New Values

We can use the regression equation created above to predict the mileage when a new set of values for displacement, horse power and weight is provided.

For a car with disp = 221, hp = 102 and wt = 2.91 the predicted mileage is :

$Y = 37.15 + (-0.000937)*221 + (-0.0311) * 102 + (-3.8008) * 2.91 = 22.7104$

5.10 DATA EXPLORATION VERSUS PRESENTATION

GQ. Explain Data Exploration versus presentation.

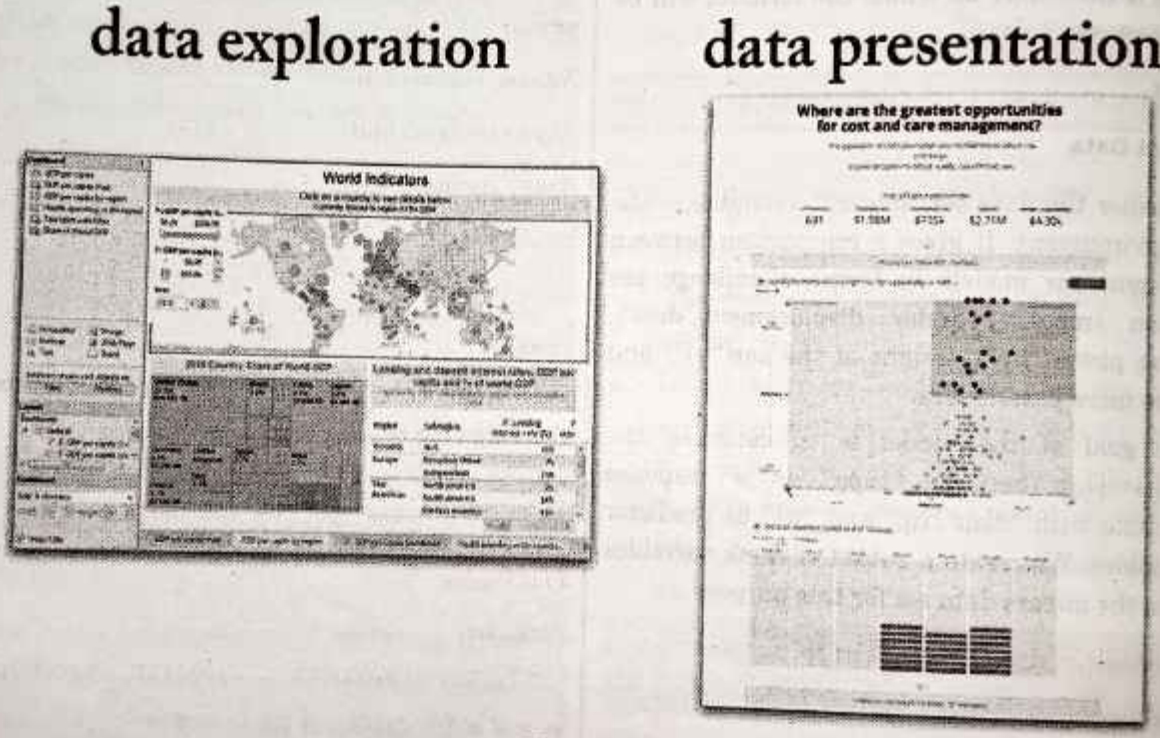


Fig. 5.10.1

- Data exploration means the deep-dive analysis of data in search of new insights.
- Data presentation means the delivery of data insights to an audience in a form that makes clear the implications.
- Your toolbox for data exploration tools is flush with technology solutions such as Tableau, PowerBI, Looker, and Qlik. "Visual analytics" tools give analysts a super-powered version of Excel for dicing data needs to handle lots of data sources and doesn't know in which direction she will go with the analysis.

- Data presentation is a different class of problem with distinct use cases, goals, and audience needs. Think about the incredible data stories delivered by the The Upshot, Fivethirtyeight, and Bloomberg. These data journalists often demonstrate data presentation at its finest, complete with guided storytelling, compelling visuals, and thoughtful text descriptions. When compared to these examples, it becomes obvious that the best efforts by a data exploration tool cannot deliver high-quality data presentation.

1. Audience - Who is the data for?

- For data exploration, the primary audience is the data analyst herself. She is the person who is both manipulating the data and seeing the results. She needs to work with tight feedback cycles of defining hypotheses, analyzing data, and visualizing results.
- For data presentation, the audience is a separate group of end-users, not the author of the analysis. These end-users are often non-analytical, they are on the front-lines of business decision-making, and may difficulty connecting the dots between an analysis and the implications for their job.

2. Message - What do you want to say?

- Data exploration is about the journey to find a message in your data. The analyst is trying to put together the pieces of a puzzle.
- Data presentation is about sharing the solved puzzle with people who can take action on the insights. Authors of data presentations need to guide an audience through the content with a purpose and point of view.

3. Explanation - What does the data mean?

- For the analysts using data exploration tools, the meaning of their analysis can be self-evident. A 1% jump in your conversion metric may represent a big change that changes your marketing tactics. The important challenge for the analysts is to answer why is this happening.
- Data presentations carry a heavier burden in explaining the results of analysis. When the audience isn't as familiar with the data, the data presentation author needs to start with more basic descriptions and context. How do we measure the conversion metric? Is a 1% change a big deal or not? What is the business impact of this change?

4. Visualizations - How do I show the data?

- The visualizations for data exploration need to be easy to create and may often show multiple dimensions to unearth complex patterns.
- For data presentation, it is important that visualizations be simple and intuitive. The audience doesn't have the patience to decipher the meaning of a chart. I used to love presenting data in treemaps but found that as a visualization it could seldom stand-alone without a two-minute tutorial to teach new users how to read the content.

5. Interactions - How are data insights created and shared ?

- Data exploration can be a lonely endeavor: Analysts work on their own to gather data, connect data across silos, and dig into the data to find insights. Data exploration is often a solitary activity that only connects with other people when insights are found and need to be shared. That is, when...
- Data presentation is a collaborative, social activity. The value emerges when insights found in data are shared with people who understand the context of the business. The dialogue that emerges is the point, not a failure of the analysis.
- There is something between the extreme ends of data exploration and data presentation. We believe data storytelling lies in this intersection. Data stories aren't entirely about "telling", nor are they in the wilderness of "finding".

- It is the opportunity to explain the data in a guided, narrative way where message meets exploration.

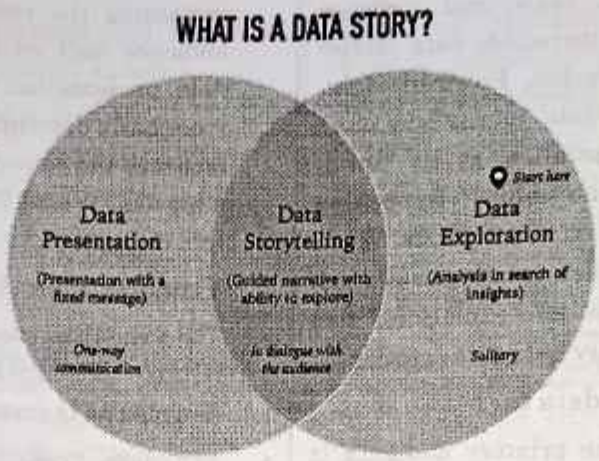


Fig. 5.10.2 : Data Exploration versus presentation

...Chapter Ends

❑❑❑

Module 6

CHAPTER 6

Data Analytics and Visualization with Python

University Prescribed Syllabus

Essential Data Libraries for data analytics : Pandas, NumPy, SciPy. Plotting and visualization with python: Introduction to Matplotlib, Basic Plotting with Matplotlib, Create Histogram, BarChart, Pie chart, Box Plot, violin plot using Matplotlib.

Introduction to seaborn Library, MultiplePlots, Regression plot, regplot.

6.1 ESSENTIAL DATA LIBRARIES FOR DATA ANALYTICS : PANDAS

GQ. Explain Pandas in detail.

- Python is a great language for doing data analysis, primarily because of the fantastic ecosystem of data-centric Python packages. Pandas are one of those packages, and makes importing and analyzing data much easier.
- All of us can do data analysis using pen and paper on small data sets. We require specialized tools and techniques to analyze and derive meaningful information from massive datasets. Pandas Python is one of those libraries for data analysis. that contains high-level data structures and tools to manipulate data in a simple way. Providing an effortless yet effective way to analyze data requires the ability to index, retrieve, split, join, restructure, and various other analyses on both multi and single-dimensional data.

Key Features of Pandas

Pandas data analysis library has some unique features that provide these capabilities-

(i) **The Series and Data Frame Objects :** These two are high-performance array and table structures for representing the heterogeneous and homogeneous data sets in Pandas Python.

(ii) **Restructuring of Data Sets :** Pandas python provides the flexibility for reshaping the data structures to be inserted in both rows and columns of tabular data.

(iii) **Labelling :** To allow automatic data alignment and indexing, pandas provide labeling on series and tabular data.

(iv) **Multiple Labels for a Data Item :** Heterogeneous indexing of data spread across multiple axes, which helps in creating more than one label on each data item.

(v) **Grouping :** The functionality to perform split-apply-combine on series as well on tabular data.

(vi) **Identify and Fix Missing Data :** Programmers can quickly identify and mix missing data floating and non-floating pointing numbers using pandas.

(vii) Powerful capabilities to load and save data from various formats such as JSON, CSV, HDF5, etc.

(viii) Conversion from NumPy and Python data structures to pandas objects.

(ix) Slicing and sub-setting of datasets, including merging and joining data sets with SQL- like constructs.

- Although pandas provide many statistical methods, it is not enough to do data science in Python. Pandas depend upon other python libraries for data science like NumPy, SciPy, Sci-Kit Learn, Matplotlib, ggvis in the Python ecosystem to conclude from large data sets. Thus, making it possible for Pandas applications to take advantage of the robust and extensive Python framework.

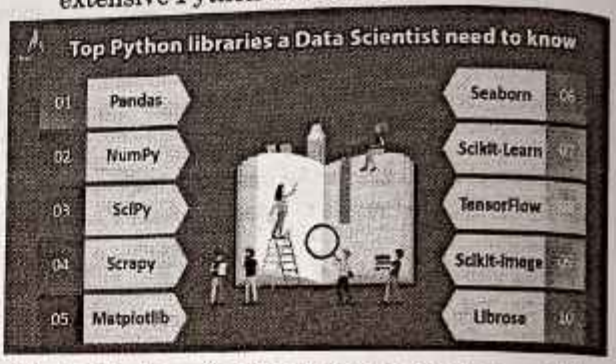


Fig. 6.1.1 : Essential Data Libraries for data analytics: Pandas

Pros of using Pandas

- Pandas allow you to represent data effortlessly and in a simpler manner, improving data analysis and comprehension. For data science projects, such a simple data representation helps glean better insights.
- Pandas is highly efficient as it enables you to perform any task by writing only a few lines of code.
- Pandas provide users with a broad range of commands to analyze data quickly.

Cons of using Pandas

- The learning curve for Pandas may appear to be simple at first, but as you start working with it, you may find it challenging to grasp.
- One of the most evident flaws of Pandas is that it isn't suitable for working with 3D matrices.

6.2 NUMPY

GQ. Explain Numpy?

- Numerical Python code name: - NumPy is a Python library for numerical calculations and scientific computations. NumPy provides numerous features which Python enthusiasts and programmers can use to work with high-performing arrays and matrices. NumPy arrays provide vectorization of mathematical operations, which gives it a performance boost over Python's looping constructs.
- Pandas Series and DataFrame objects rely primarily on NumPy arrays for all the mathematical calculations like slicing elements and performing vector operations.

Key Features of NumPy

Below are some of the features provided by NumPy-

(I) Integration with legacy languages

(1) **Mathematical Operations :** It provides all the standard functions required to perform operations on large data sets swiftly and efficiently, which otherwise have to be achieved through looping constructs.

(2) **ndarray:** It is a fast and efficient multidimensional array that can perform vector-based arithmetic operations and has powerful broadcasting capabilities.

(3) **I/O Operations:** It provides various tools which can be used to write/read huge data sets from disk. It also supports I/O operations on memory-based file mappings.

(4) Fourier transform capabilities, Linear Algebra, and Random Number Generation.

(II) Pros of using NumPy

(1) NumPy provides efficient and scalable data storage and better data management for mathematical calculations.

(2) The Numpy array contains a variety of functions, methods, and variables that make computing matrices simpler.

(III) Cons of using NumPy

(1) "Nan" is an acronym for "not a number" intended to deal with the issue of missing values. Although NumPy supports "nan," Python's lack of cross-platform compatibility makes it challenging for users. As a result, we may run into issues while comparing values within the Python interpreter.

(2) When data is stored in contiguous memory addresses, insertion and deletion processes become expensive since shifting.

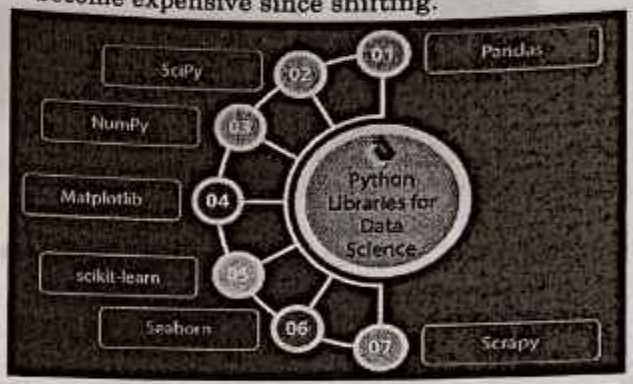


Fig. 6.2.1 : NumPy

6.3 SCIPY

GQ. Explain SciPy?

- Scientific Python code name, SciPy-It is an assortment of mathematical functions and algorithms built on Python's extension NumPy. SciPy provides various high-level commands and classes for manipulating and visualizing data. SciPy is useful for data-processing and prototyping systems.
- Apart from this, SciPy provides other advantages for building scientific applications and many specialized, sophisticated applications backed by a robust and fast-growing Python community.

Pros of using SciPy

(1) Visualizing and manipulating data with high-level commands and classes.

(2) Python sessions that are both robust and interactive.

(3) For parallel programming, there are classes and web and database procedures.

Con of using SciPy

- SciPy does not provide any plotting function because its focus is on numerical objects and algorithms.

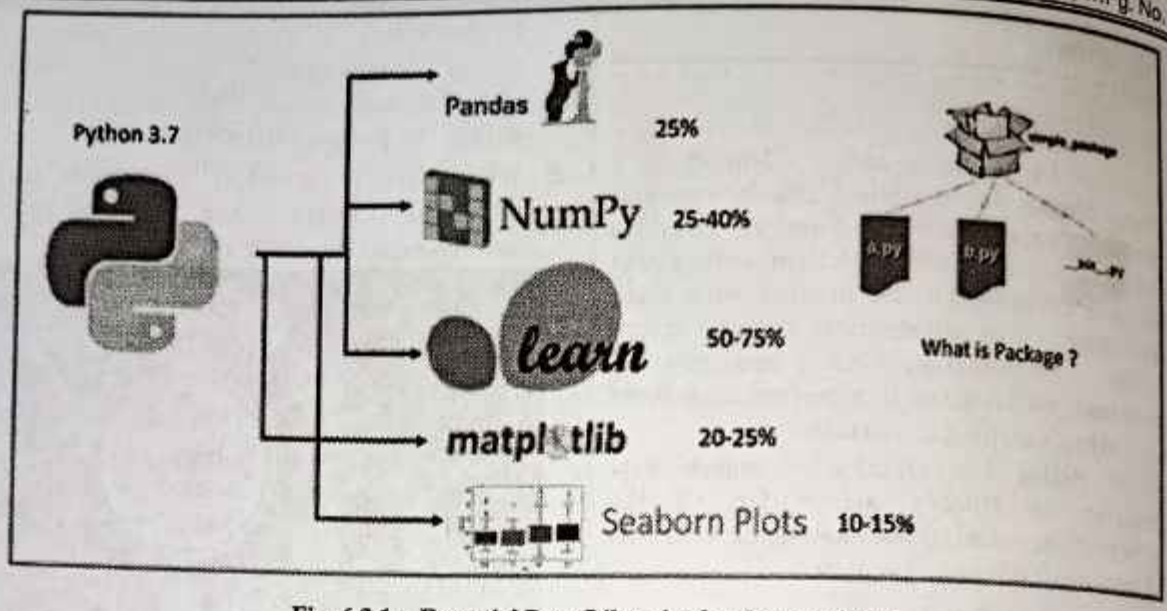


Fig. 6.3.1 : Essential Data Libraries for data analytics

6.4 PLOTTING AND VISUALIZATION WITH PYTHON: INTRODUCTION TO MATPLOTLIO

GQ. Explain Matplotlio ?

- Data Visualization is the process of presenting data in the form of graphs or charts. It helps to understand large and complex amounts of data very easily. It allows the decision-makers to make decisions very efficiently and also allows them in identifying new trends and patterns very easily.
- It is also used in high-level data analysis for Machine Learning and Exploratory Data Analysis (EDA). Data visualization can be done with various tools like Tableau, Power BI, Python.

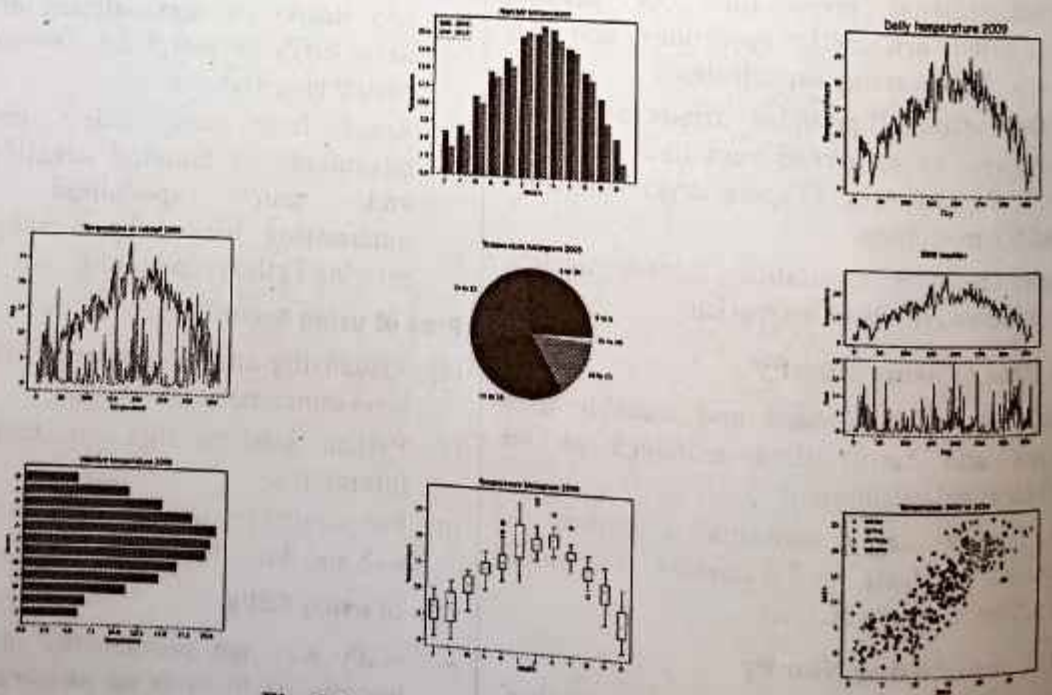


Fig. 6.4.1 : Introduction to Matplotlio

Matplotlib

- Matplotlib is a low-level library of Python which is used for data visualization. It is easy to use and emulates MATLAB like graphs and visualization. This library is built on the top of NumPy arrays and consist of several plots like line chart, bar chart, histogram, etc. It provides a lot of flexibility but at the cost of writing more code.
- Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays. Matplotlib is written in Python and makes use of NumPy, the numerical mathematics extension of Python. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, WxPythonotTkinter. It can be used in Python and IPython shells, Jupyter notebook and web application servers also.
- Matplotlib has a procedural interface named the Pylab, which is designed to resemble MATLAB, a proprietary programming language developed by MathWorks. Matplotlib along with NumPy can be considered as the open source equivalent of MATLAB.
- Matplotlib is an open-source drawing library that supports various drawing types
- You can generate plots, histograms, bar charts, and other types of charts with just a few lines of code .It's often used in web application servers, shells, and Python scripts

Basic Plotting with Matplotlib

```
In [5]: import matplotlib.pyplot as plt
        fig = plt.figure()
        ax1 = fig.add_subplot(1,1,1)
        ax1.plot([1,2,3])
        ax2 = fig.add_subplot(2,2,1, facecolor='w')
        ax2.plot([1,2,0.5])

Out[5]: [<matplotlib.lines.Line2D at 0x9db937ab48>]
```

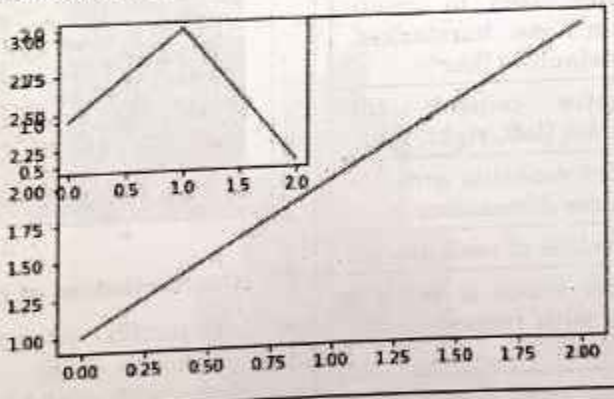


Fig. 6.4.2

6.5 CREATE HISTOGRAM

GQ. Explain process of creation of Histogram using Python ?

- A histogram, with which you may be well-acquainted, is a kind of bar plot that gives a discretized display of value frequency. The data points are split into discrete, evenly spaced bins, and the number of data points in each bin is plotted.
- Using the tipping data from before, we can make a histogram of tip percentages of the total bill using the hist method on the Series.
- A histogram is basically used to represent data provided in a form of some groups. It is accurate

method for the graphical representation of numerical data distribution.

- It is a type of bar plot where X-axis represents the bin ranges while Y-axis gives information about frequency.
- To create a histogram the first step is to create bin of the ranges, then distribute the whole range of the values into a series of intervals, and count the values which fall into each of the intervals.
- Bins are clearly identified as consecutive, non-overlapping intervals of variables.
- The matplotlib.pyplot.hist() function is used to compute and create histogram of x.

The following table shows the parameters accepted by matplotlib.pyplot.hist() function :

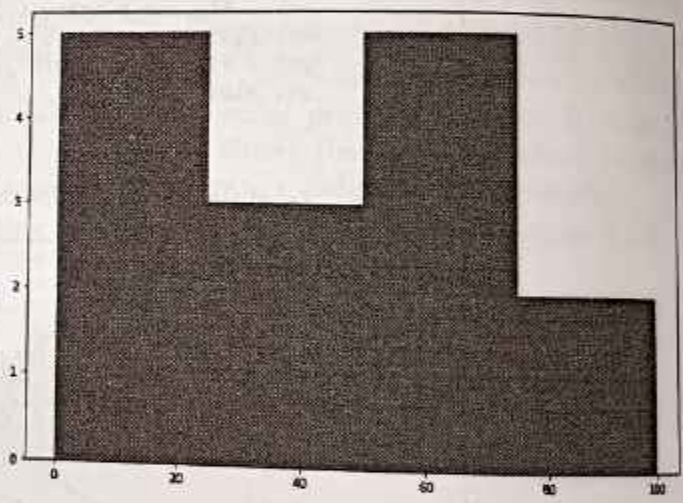
Attribute	parameter
x	array or sequence of array
bins	optional parameter contains integer or sequence or strings
density	optional parameter contains boolean values
range	optional parameter represents upper and lower range of bins
histtype	optional parameter used to create type of histogram [bar, barstacked, step, stepfilled], default is "bar"
align	optional parameter controls the plotting of histogram [left, right, mid]
weights	optional parameter contains array of weights having same dimensions as x
bottom	location of the baseline of each bin
rwidth	optional parameter which is relative width of the bars with respect to bin width
color	optional parameter used to set color or sequence of color specs
label	optional parameter string or sequence of string to match with multiple datasets
log	optional parameter used to set histogram axis on log scale

Let's create a basic histogram of some random values. Below code creates a simple histogram of some random values:

Python3

```
from matplotlib import pyplot as plt
import numpy as np

# Creating dataset
a = np.array([22, 87, 5, 43, 56,
              73, 55, 54, 11,
              20, 51, 5, 79, 31,
              27])

# Creating histogram
fig, ax = plt.subplots(figsize =(10, 7))
ax.hist(a, bins = [0, 25, 50, 75, 100])

# Show plot
plt.show()
```

Output



Customization of Histogram

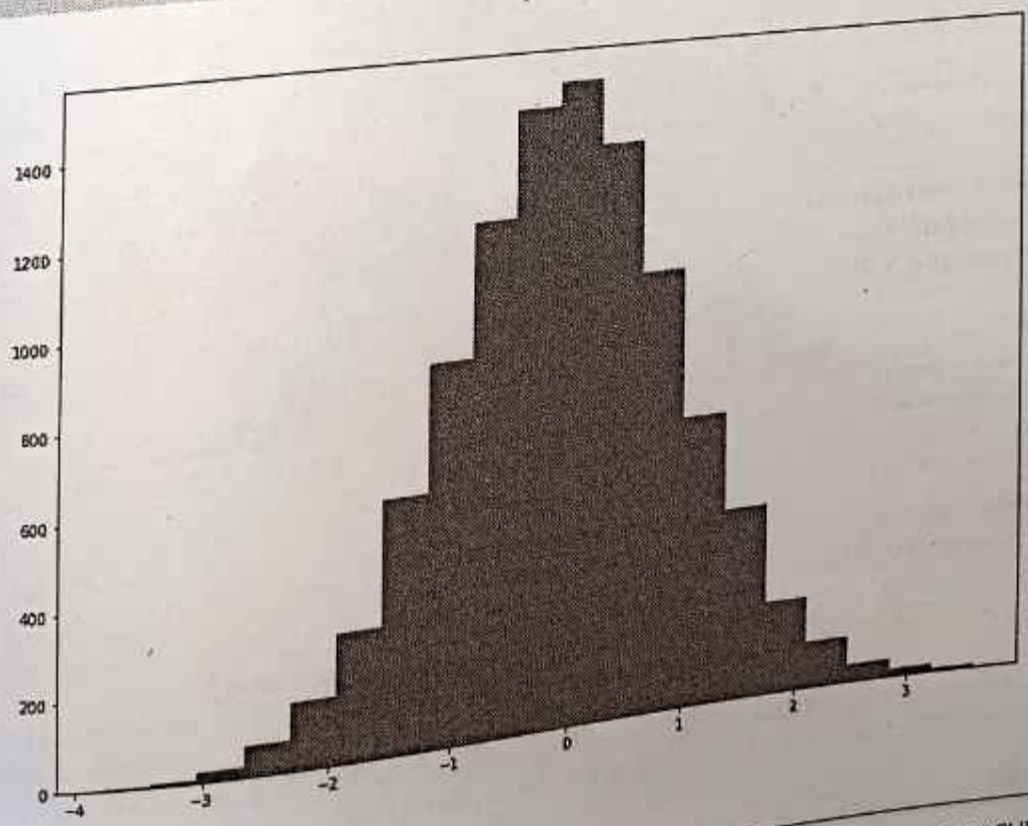
- Matplotlib provides a range of different methods to customize histogram. matplotlib.pyplot.hist() function itself provides many attributes with the help of which we can modify a histogram.The hist() function provide a patches object which gives access to the properties of the created objects, using this we can modify the plot according to our will.

➢ **Example 6.5.1**

Python3

```
import matplotlib.pyplot as plt
import numpy as np
from matplotlib import colors
from matplotlib.ticker import PercentFormatter

# Creating dataset
np.random.seed(23685752)
N_points = 10000
n_bins = 20

# Creating distribution
x = np.random.randn(N_points)
y = .8 ** x + np.random.randn(10000) + 25

# Creating histogram
fig, axs = plt.subplots(1, 1,
                        figsize =(10, 7),
                        tight_layout = True)

axs.hist(x, bins = n_bins)

# Show plot
plt.show()
```

Output



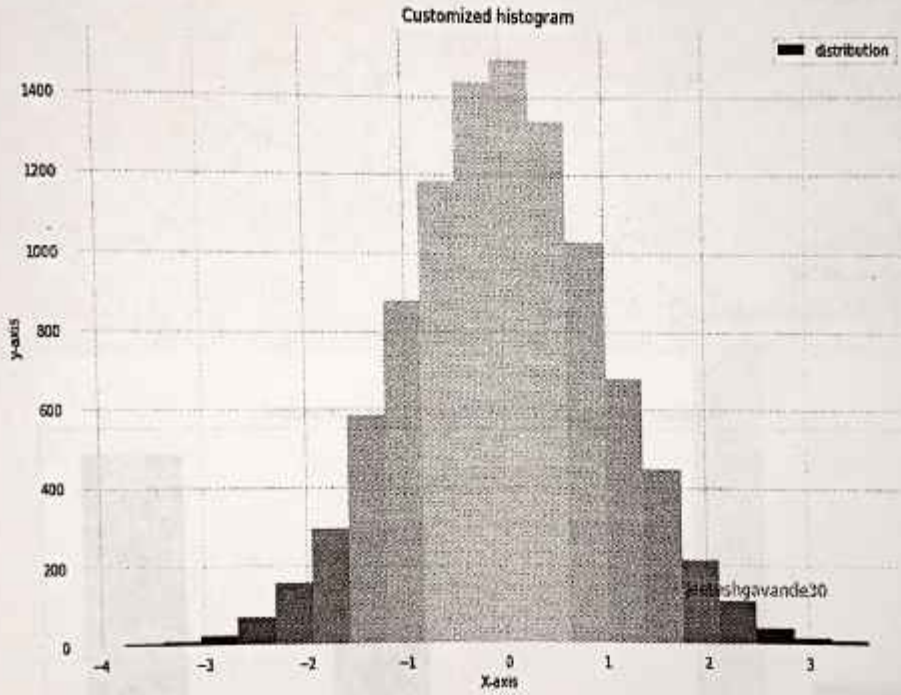
➢ **Example 6.5.2 :** The code below modifies the above histogram for a better view and accurate readings.

Python3

```
import matplotlib.pyplot as plt
import numpy as np
from matplotlib import colors
from matplotlib.ticker import PercentFormatter


# Creating dataset
np.random.seed(23685752)
N_points = 10000
n_bins = 20

# Creating distribution
x = np.random.randn(N_points)
y = .8 ** x + np.random.randn(10000) + 25
legend = ['distribution']

# Creating histogram
fig, axs = plt.subplots(1, 1,
                        figsize =(10, 7),
                        tight_layout = True)


# Remove axes splines
for s in ['top', 'bottom', 'left', 'right']:
    axs.spines[s].set_visible(False)

# Remove x, y ticks
axs.xaxis.set_ticks_position('none')
axs.yaxis.set_ticks_position('none')

# Add padding between axes and labels
axs.xaxis.set_tick_params(pad = 5)
axs.yaxis.set_tick_params(pad = 10)

# Add x, y gridlines
axs.grid(b = True, color ='grey',
         linestyle ='-.', linewidth = 0.5,
         alpha = 0.6)

# Add Text watermark
fig.text(0.9, 0.15, 'Jeeteshgavande30',
         fontsize = 12,
         color ='red',
         ha ='right',
         va ='bottom',
         alpha = 0.7)

# Creating histogram
```

```
N, bins, patches = axs.hist(x, bins = n_bins)

# Setting color
fracs = ((N**(1 / 5)) / N.max())
norm = colors.Normalize(fracs.min(), fracs.max())

for thisfrac, thispatch in zip(fracs, patches):
    color = plt.cm.viridis(norm(thisfrac))
    thispatch.set_facecolor(color)

# Adding extra features
plt.xlabel("X-axis")
plt.ylabel("y-axis")
plt.legend(legend)
plt.title('Customized histogram')

# Show plot
plt.show()
```

Output



6.5.1 Bar Chart

GQ Explain Bar Chart in detail ?

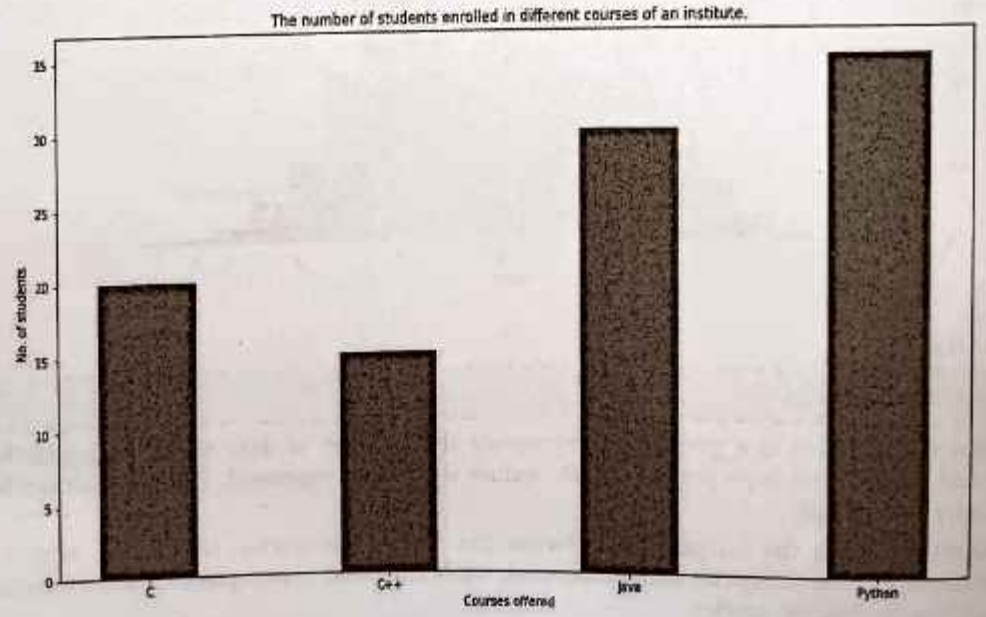
- A bar plot or bar chart is a graph that represents the category of data with rectangular bars with lengths and heights that is proportional to the values which they represent. The bar plots can be plotted horizontally or vertically.
- A bar chart describes the comparisons between the discrete categories. One of the axis of the plot represents the specific categories being compared, while the other axis represents the measured values corresponding to those categories.

- The **matplotlib** API in Python provides the bar() function which can be used in MATLAB style use or as an object-oriented API.
  The syntax of the bar() function to be used with the axes is as follows:-
- plt.bar(x, height, width, bottom, align)
- The function creates a bar plot bounded with a rectangle depending on the given parameters.
- Following is a simple example of the bar plot, which represents the number of students enrolled in different courses of an institute.

☞ **Python 3**

```
import numpy as np
import matplotlib.pyplot as plt

# creating the dataset
data = {'C':20, 'C++':15, 'Java':30,
        'Python':35}
courses = list(data.keys())
values = list(data.values())


fig = plt.figure(figsize = (10, 5))

# creating the bar plot
plt.bar(courses, values, color ='maroon',
        width = 0.4)


plt.xlabel("Courses offered")
plt.ylabel("No. of students enrolled")
plt.title("Students enrolled in different courses")
plt.show()
```

Output

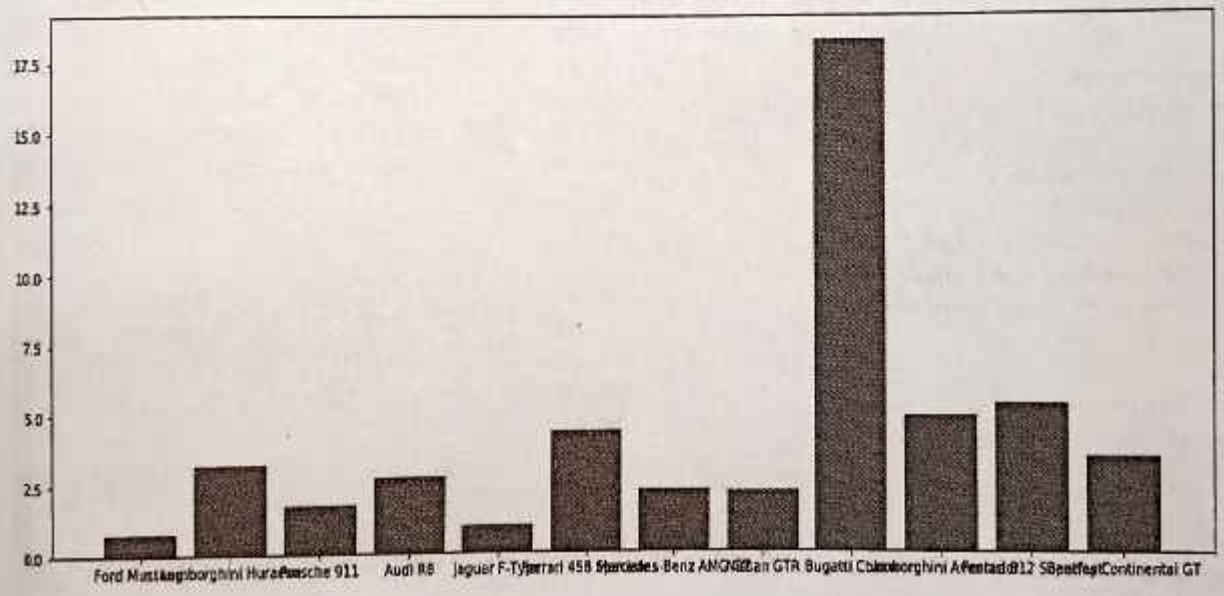


- Here plt.bar(courses, values, color='maroon') is used to specify that the bar chart is to be plotted by using the courses column as the X-axis, and the values as the Y-axis.
- The color attribute is used to set the color of the bars(maroon in this case).plt.xlabel("Courses offered") and plt.ylabel("students enrolled") are used to label the corresponding axes.plt.title() is used to make a title for the graph.plt.show() is used to show the graph as output using the previous commands.Customizing the bar plot

Python3

```
import pandas as pd
from matplotlib import pyplot as plt

# Read CSV into pandas
data = pd.read_csv(r"cars.csv")
data.head()
df = pd.DataFrame(data)

name = df['car'].head(12)
price = df['price'].head(12)

# Figure Size
fig = plt.figure(figsize =(10, 7))

# Horizontal Bar Plot
plt.bar(name[0:10], price[0:10])

# Show Plot
plt.show()
```

Output

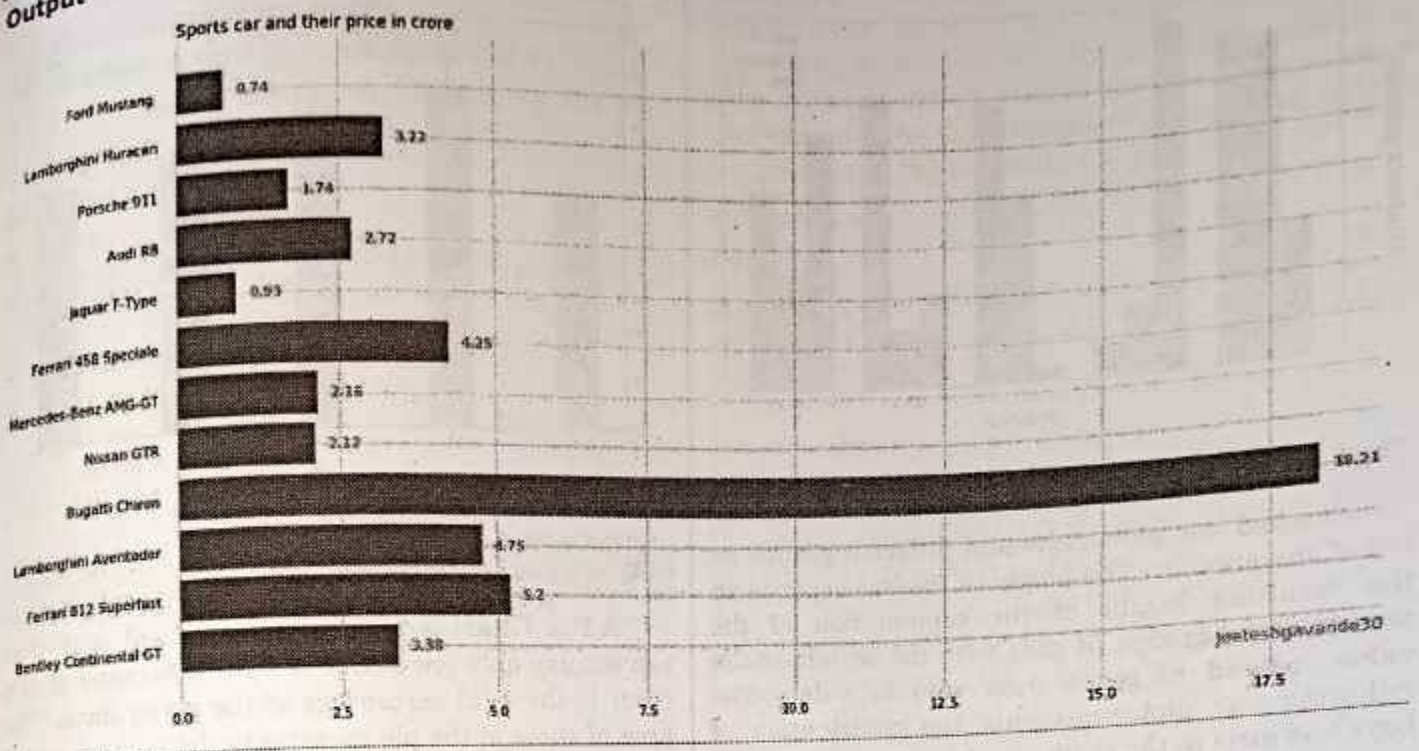


It is observed in the above bar graph that the X-axis ticks are overlapping each other thus it cannot be seen properly. Thus by rotating the X-axis ticks, it can be visible clearly. That is why customization in bar graphs is required.

Python 3

```
import pandas as pd
from matplotlib import pyplot as plt

# Read CSV into pandas
data = pd.read_csv(r"cars.csv")
data.head()
df = pd.DataFrame(data)

name = df['car'].head(12)
price = df['price'].head(12)

# Figure Size
fig, ax = plt.subplots(figsize =(16, 9))

# Horizontal Bar Plot
ax.barh(name, price)

# Remove axes splines
for s in ['top', 'bottom', 'left', 'right']:
    ax.spines[s].set_visible(False)

# Remove x, y Ticks
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

# Add padding between axes and labels
ax.xaxis.set_tick_params(pad = 5)
ax.yaxis.set_tick_params(pad = 10)

# Add x, y gridlines
ax.grid(b = True, color ='grey',
        linestyle ='-.', linewidth = 0.5,
        alpha = 0.2)

# Show top values
ax.invert_yaxis()

# Add annotation to bars
for i in ax.patches:
    plt.text(i.get_width()+0.2, i.get_y()+0.5,
             str(round((i.get_width()), 2)),
             fontsize = 10, fontweight ='bold',
             color ='grey')

# Add Plot Title
ax.set_title('Sports car and their price in crore',
             loc ='left', )

# Add Text watermark
fig.text(0.9, 0.15, 'Jeeteshgavande30', fontsize = 12,
         color ='grey', ha ='right', va ='bottom',
         alpha = 0.7)

# Show Plot
plt.show()
```

Output



There are many more Customizations available for bar plots.

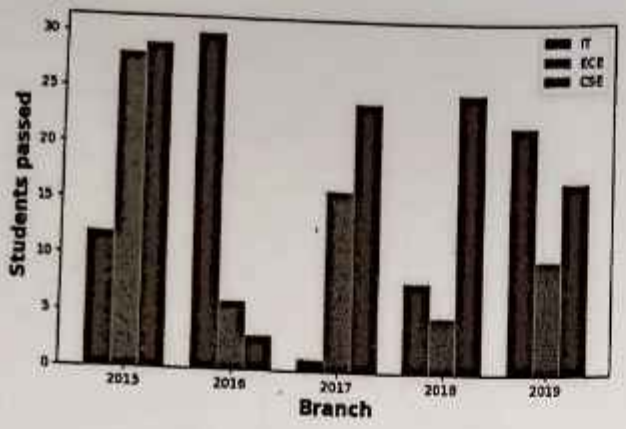
6.5.2 Multiple Bar Plots

Multiple bar plots are used when comparison among the data set is to be done when one variable is changing. We can easily convert it as a stacked area bar chart, where each subgroup is displayed by one on top of the others. It can be plotted by varying the thickness and position of the bars. Following bar plot shows the number of students passed in the engineering branch:

Python3

```
import numpy as np
import matplotlib.pyplot as plt

# set width of bar
barWidth = 0.25
fig = plt.subplots(figsize =(12, 8))

# set height of bar
IT = [12, 30, 1, 8, 22]
ECE = [28, 6, 16, 5, 10]
CSE = [29, 3, 24, 25, 17]

# Set position of bar on X axis
br1 = np.arange(len(IT))
br2 = [x + barWidth for x in br1]
br3 = [x + barWidth for x in br2]

# Make the plot
plt.bar(br1, IT, color ='r', width = barWidth,
        edgecolor ='grey', label ='IT')
plt.bar(br2, ECE, color ='g', width = barWidth,
        edgecolor ='grey', label ='ECE')
plt.bar(br3, CSE, color ='b', width = barWidth,
        edgecolor ='grey', label ='CSE')

# Adding Xticks
plt.xlabel('Branch', fontweight ='bold', fontsize = 15)
plt.ylabel('Students passed', fontweight ='bold', fontsize = 15)
plt.xticks([r + barWidth for r in range(len(IT))],
        ['2015', '2016', '2017', '2018', '2019'])

plt.legend()
plt.show()
```

Output



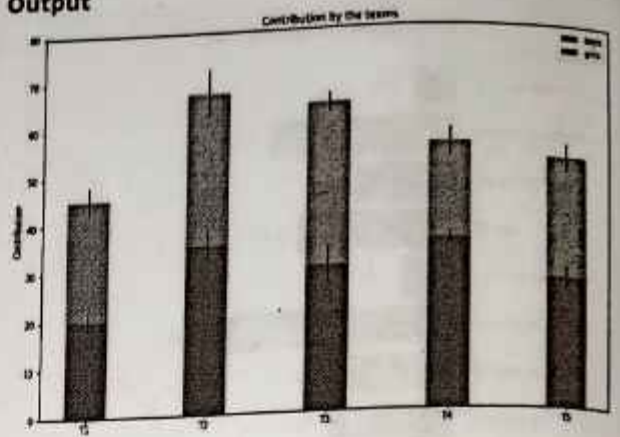
6.5.3 Stacked Bar Plot

Stacked bar plots represent different groups on top of one another. The height of the bar depends on the resulting height of the combination of the results of the groups. It goes from the bottom to the value instead of going from zero to value. The following bar plot represents the contribution of boys and girls in the team.

Python3

```
import numpy as np
import matplotlib.pyplot as plt

N = 5

boys = (20, 35, 30, 35, 27)
girls = (25, 32, 34, 20, 25)
boyStd = (2, 3, 4, 1, 2)
girlStd = (3, 5, 2, 3, 3)
ind = np.arange(N)
width = 0.35

fig = plt.subplots(figsize =(10, 7))
p1 = plt.bar(ind, boys, width, yerr = boyStd)
p2 = plt.bar(ind, girls, width,
        bottom = boys, yerr = girlStd)

plt.ylabel('Contribution')
plt.title('Contribution by the teams')
plt.xticks(ind, ('T1', 'T2', 'T3', 'T4', 'T5'))
plt.yticks(np.arange(0, 81, 10))
plt.legend((p1[0], p2[0]), ('boys', 'girls'))

plt.show()
```

Output



6.5.4 Pie Chart

GQ Explain Pie chart in detail?

A **Pie Chart** is a circular statistical plot that can display only one series of data. The area of the chart is the total percentage of the given data. The area of slices of the pie represents the percentage of the parts of the data. The slices of pie are called wedges. The area of the wedge is determined by the length of the arc of the wedge. The area of a wedge represents the relative percentage of that part with respect to whole data. Pie charts are commonly used in business presentations like sales, operations, survey results, resources, etc as they provide a quick summary.

☞ Creating Pie Chart

Matplotlib API has pie() function in its pyplot module which create a pie chart representing the data in an array.

- **Syntax:** matplotlib.pyplot.pie(data, explode=None, labels=None, colors=None, autopct=None, shadow=False)

Parameters

- **data** represents the array of data values to be plotted, the fractional area of each slice is represented by **data/sum(data)**. If um(data)<1, then the data values returns the fractional area directly, thus resulting pie will have empty wedge of size 1-sum(data).
- **labels** is a list of sequence of strings which sets the label of each wedge.
- **color** attribute is used to provide color to the wedges.

- **autopct** is a string used to label the wedge with their numerical value.
- **shadow** is used to create shadow of wedge.

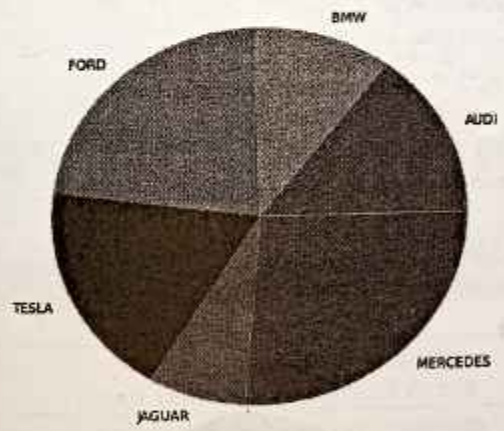
Let's create a simple pie chart using the pie() function:

Example

Python3

```
# Import libraries
from matplotlib import pyplot as plt
import numpy as np


# Creating dataset
cars = ['AUDI', 'BMW', 'FORD',
        'TESLA', 'JAGUAR', 'MERCEDES']

data = [23, 17, 35, 29, 12, 41]

# Creating plot
fig = plt.figure(figsize =(10, 7))
plt.pie(data, labels = cars)

# show plot
plt.show()
```

Output



6.5.5 Customizing Pie Chart

A pie chart can be customized on the basis several aspects. The startangle attribute rotates the plot by the specified degrees in counter clockwise direction performed on x-axis of pie chart. shadow attribute accepts boolean value, if its true then shadow will appear below the rim of pie. Wedges of the pie can be customized using wedgeprop which takes Python dictionary as parameter with name values pairs denoting the wedge properties like linewidth, edgecolor, etc. By setting frame=True axes frame is drawn around the pie chart.autopct controls how the percentages are displayed on the wedges. Let us try to modify the above plot:

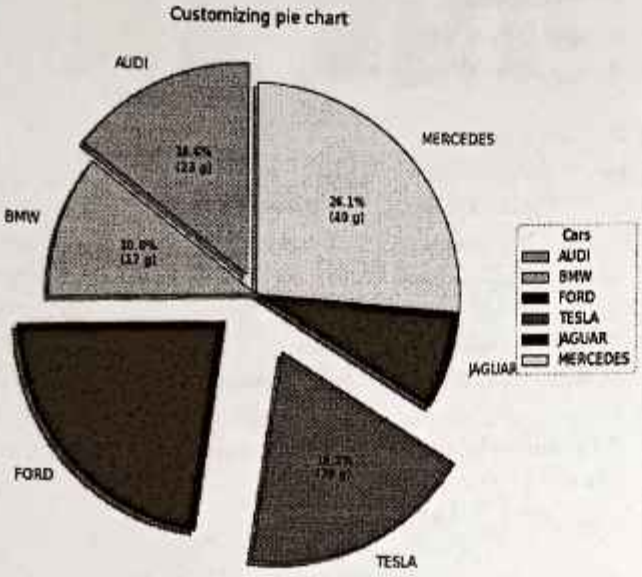
Example 1

Python3

```
# Import libraries
import numpy as np
import matplotlib.pyplot as plt

# Creating dataset
cars = ['AUDI', 'BMW', 'FORD',
        'TESLA', 'JAGUAR', 'MERCEDES']

data = [23, 17, 35, 29, 12, 41]

# Creating explode data
explode = (0.1, 0.0, 0.2, 0.3, 0.0, 0.0)

# Creating color parameters
colors = ( "orange", "cyan", "brown",
          "grey", "indigo", "beige")

# Wedge properties
wp = { 'linewidth' : 1, 'edgecolor' : "green" }

# Creating autocpt arguments
def func(pct, allvalues):
    absolute = int(pct / 100.*np.sum(allvalues))
    return "{:.1f}%\n({:d} g)".format(pct, absolute)


# Creating plot
fig, ax = plt.subplots(figsize =(10, 7))
wedges, texts, autotexts = ax.pie(data,
                                  autopct = lambda pct: func(pct, data),
                                  explode = explode,
                                  labels = cars,
                                  shadow = True,
                                  colors = colors,
                                  startangle = 90,
                                  wedgeprops = wp,
                                  textprops = dict(color ="magenta"))

# Adding legend
ax.legend(wedges, cars,
          title ="Cars",
```

```
        loc ="center left",
        bbox_to_anchor =(1, 0, 0.5, 1))

plt.setp(autotexts, size = 8, weight ="bold")
ax.set_title("Customizing pie chart")

# show plot
plt.show()
```

Output



➤ **Example 6.5.3 :** Creating a Nested Pie Chart

Python3

```
# Import libraries
from matplotlib import pyplot as plt
import numpy as np

# Creating dataset
size = 6
cars = ['AUDI', 'BMW', 'FORD',
        'TESLA', 'JAGUAR', 'MERCEDES']

data = np.array([[23, 16], [17, 23],
                 [35, 11], [29, 33],
                 [12, 27], [41, 42]])

# normalizing data to 2 pi
norm = data / np.sum(data)*2 * np.pi

# obtaining ordinates of bar edges
left = np.cumsum(np.append(0,
                           norm.flatten()[:-1])).reshape(data.shape)

# Creating color scale
cmap = plt.get_cmap("tab20c")
outer_colors = cmap(np.arange(6)*4)
inner_colors = cmap(np.array([1, 2, 5, 6, 9,
                              10, 12, 13, 15,
                              17, 18, 20 ]))

# Creating plot
fig, ax = plt.subplots(figsize =(10, 7),
                       subplot_kw = dict(polar = True))

ax.bar(x = left[:, 0],
       width = norm.sum(axis = 1),
       bottom = 1-size,
       height = size,
       color = outer_colors,
       edgecolor ='w',
       linewidth = 1,
       align ="edge")

ax.bar(x = left.flatten(),
       width = norm.flatten(),
       bottom = 1-2 * size,
       height = size,
       color = inner_colors,
       edgecolor ='w',
       linewidth = 1,
       align ="edge")

ax.set(title ="Nested pie chart")
ax.set_axis_off()

# show plot
plt.show()
```

Output

6.5.6 Box Plot

GQ. Explain Box Plot?

A **Box Plot** is also known as **Whisker plot** is created to display the summary of the set of data values having properties like minimum, first quartile, median, third quartile and maximum. In the box plot, a box is created from the first quartile to the third quartile, a vertical line is also there which goes through the box at the median. Here x-axis denotes the data to be plotted while the y-axis shows the frequency distribution.

Creating Box Plot

The matplotlib.pyplot module of matplotlib library provides boxplot() function with the help of which we can create box plots.

Syntax

matplotlib.pyplot.boxplot(data, notch=None, vert=None, patch_artist=None, widths=None)

Parameters

Attribute	Value
data	array or sequence of array to be plotted
notch	optional parameter accepts boolean values
vert	optional parameter accepts boolean values false and true for horizontal and vertical plot respectively
bootstrap	optional parameter accepts int specifies intervals around notched boxplots
usermedians	optional parameter accepts array or sequence of array dimension compatible with data
positions	optional parameter accepts array and sets the position of boxes
widths	optional parameter accepts array and sets the width of boxes
patch_artist	optional parameter having boolean values
labels	sequence of strings sets label for each dataset
meanline	optional having boolean value try to render meanline as full width of box
order	optional parameter sets the order of the boxplot

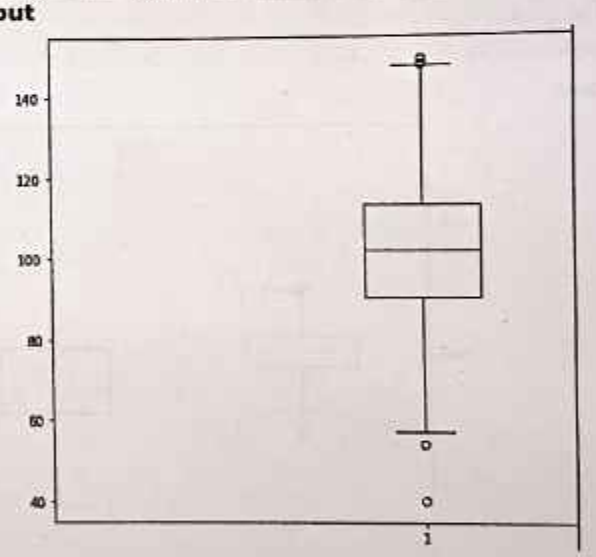
The data values given to the ax.boxplot() method can be a Numpy array or Python list or Tuple of arrays. Let us create the box plot by using numpy.random.normal() to create some random data, it takes mean, standard deviation, and the desired number of values as arguments.

Example

Python3

```
# Import libraries
import matplotlib.pyplot as plt
import numpy as np

# Creating dataset
np.random.seed(10)
data = np.random.normal(100, 20, 200)

fig = plt.figure(figsize =(10, 7))

# Creating plot
plt.boxplot(data)

# show plot
plt.show()
```

Output



Customizing Box Plot

The matplotlib.pyplot.boxplot() provides endless customization possibilities to the box plot. The notch = True attribute creates the notch format to the box plot, patch_artist = True fills the boxplot with colors, we can set different colors to different boxes.

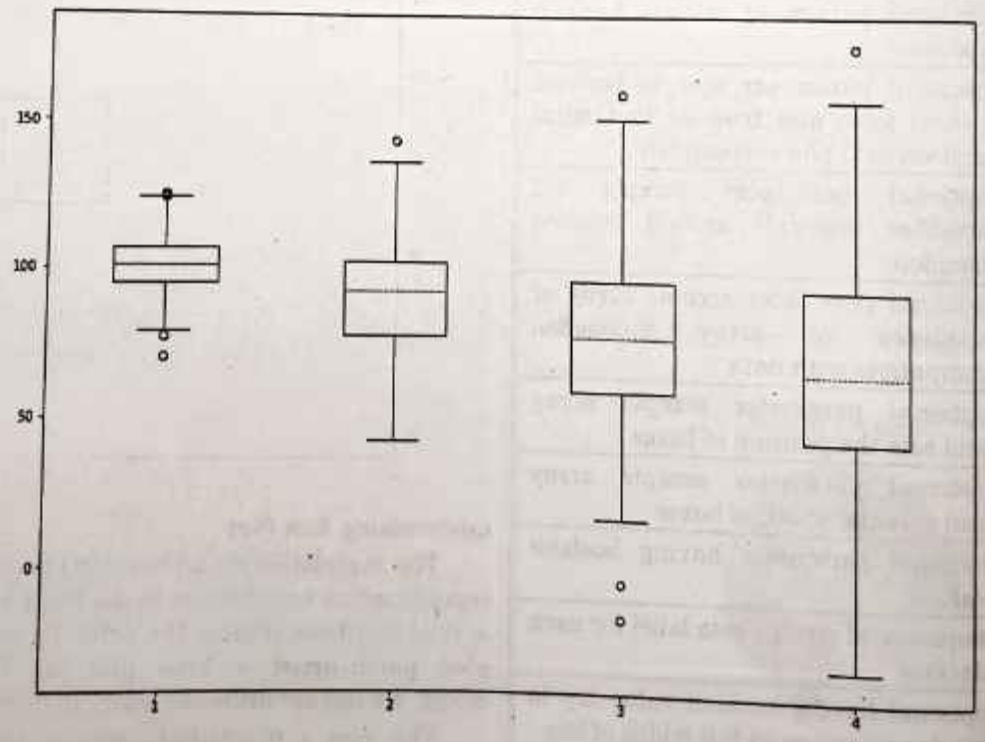
The vert = 0 attribute creates horizontal box plot. labels takes same dimensions as the number data sets.

➢ **Example 6.5.4 :**

Python3

```
# Import libraries
import matplotlib.pyplot as plt
import numpy as np

# Creating dataset
np.random.seed(10)

data_1 = np.random.normal(100, 10, 200)
data_2 = np.random.normal(90, 20, 200)
data_3 = np.random.normal(80, 30, 200)
data_4 = np.random.normal(70, 40, 200)
data = [data_1, data_2, data_3, data_4]

fig = plt.figure(figsize =(10, 7))

# Creating axes instance
ax = fig.add_axes([0, 0, 1, 1])

# Creating plot
bp = ax.boxplot(data)

# show plot
plt.show()
```

Output



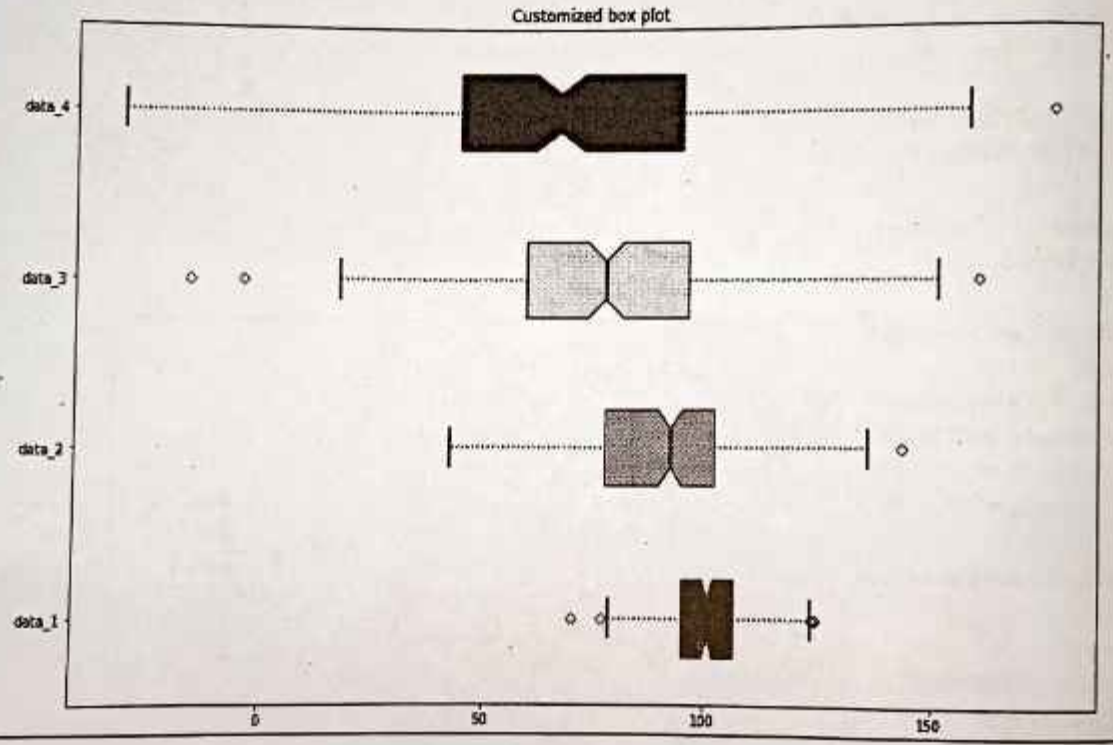
Example 6.5.5 : Let's try to modify the above plot with some of the customizations

Python3

```
# Import libraries
import matplotlib.pyplot as plt
import numpy as np

# Creating dataset
np.random.seed(10)
data_1 = np.random.normal(100, 10, 200)
data_2 = np.random.normal(90, 20, 200)
data_3 = np.random.normal(80, 30, 200)
data_4 = np.random.normal(70, 40, 200)
data = [data_1, data_2, data_3, data_4]

fig = plt.figure(figsize =(10, 7))
ax = fig.add_subplot(111)

# Creating axes instance
bp = ax.boxplot(data, patch_artist = True,
          notch ='True', vert = 0)

colors = ['#0000FF', '#00FF00',
      '#FFFF00', '#FF00FF']

for patch, color in zip(bp['boxes'], colors):
  patch.set_facecolor(color)

# changing color and linewidth of
# whiskers
for whisker in bp['whiskers']:
  whisker.set(color ='#8B008B',
          linewidth = 1.5,
          linestyle =":")

# changing color and linewidth of
# caps
for cap in bp['caps']:
  cap.set(color ='#8B008B',
        linewidth = 2)

# changing color and linewidth of
# medians
for median in bp['medians']:
  median.set(color ='red',
          linewidth = 3)

# changing style of fliers
for flier in bp['fliers']:
  flier.set(marker ='D',
```

```
        color ='#e7298a',
        alpha = 0.5)

# x-axis labels
ax.set_yticklabels(['data_1', 'data_2',
                    'data_3', 'data_4'])

# Adding title
plt.title("Customized box plot")

# Removing top axes and right axes
# ticks
ax.get_xaxis().tick_bottom()
ax.get_yaxis().tick_left()

# show plot
plt.show()
```

Output



6.6 VIOLIN PLOT USING MATPLOTLIB

GQ. Explain Violin plot using Matplotlib?

Matplotlib is a plotting library for creating static, animated, and interactive visualizations in Python. Matplotlib can be used in Python scripts, the Python and IPython shell, web application servers, and various graphical user interface toolkits like Tkinter, awxPython, etc.

What does a violin plot signify?

Violin plots are a combination of box plot and histograms. It portrays the distribution, median, interquartile range of data. So we see that iqr and median are the statistical information provided by

box plot whereas distribution is being provided by the histogram.

Violin Plot

- The white dot refers to the median.
- The end points of the bold line represent the iqr1 and iqr3.
- The end points of the thin line represent the min and max similar to the box plot.
- The distribution above 1.5x interquartile(min, max end points of the thin line) denotes the presence of outliers.
- **Syntax:** violinplot(dataset, positions=None, vert=True, widths=0.5, showmeans=False, showextrema=True, showmedians=False, quantiles=None, points=100, bw_method=None, *, data=None)

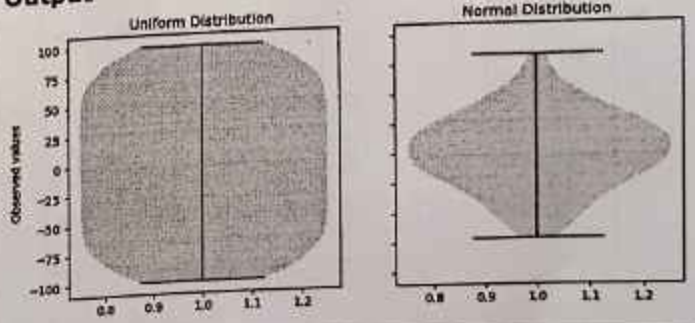
Parameters

- **dataset:** Array or a sequence of vectors.
  The input data.
- **positions:** array-like, default = [1, 2, ..., n].
  Sets the positions of the violins. The ticks and limits are automatically set to match the positions.
- **vert:** bool, default = True.
  If true, creates a vertical violin plot. Otherwise, creates a horizontal violin plot.
- **widths:** array-like, default = 0.5
  Either a scalar or a vector that sets the maximal width of each violin. The default is 0.5, which uses about half of the available horizontal space.
- **showmeans:** bool, default = False
  If True, will toggle rendering of the means.
- **showextrema:** bool, default = True
  If True, will toggle rendering of the extrema.
- **showmedians:** bool, default = False
  If True, will toggle rendering of the medians.
- **quantiles:** array-like, default = None
  If not None, set a list of floats in interval [0, 1] for each violin, which stands for the quantiles that will be rendered for that violin.
- **points:** scalar, default = 100
  Defines the number of points to evaluate each of the gaussian kernel density estimations at.
- **bw_method:** str, scalar or callable, optional
  The method used to calculate the estimator bandwidth. This can be 'scott', 'silverman', a scalar constant or a callable. If a scalar, this will be used directly as kde.factor. If a callable, it should take a GaussianKDE instance as its only parameter and return a scalar. If None (default), 'scott' is used.

Example 6.6.1

```
import numpy as np
import matplotlib.pyplot as plt

# creating a list of
# uniformly distributed values
uniform = np.arange(-100, 100)

# creating a list of normally
# distributed values
normal = np.random.normal(size = 100)*30

# creating figure and axes to
# plot the image
fig, (ax1, ax2) = plt.subplots(nrows = 1,
                               ncols = 2,
                               figsize =(9, 4),
                               sharey = True)

# plotting violin plot for
# uniform distribution
ax1.set_title('Uniform Distribution')
ax1.set_ylabel('Observed values')
ax1.violinplot(uniform)


# plotting violin plot for
# normal distribution
ax2.set_title('Normal Distribution')
ax2.violinplot(normal)

# Function to show the plot
plt.show()
```

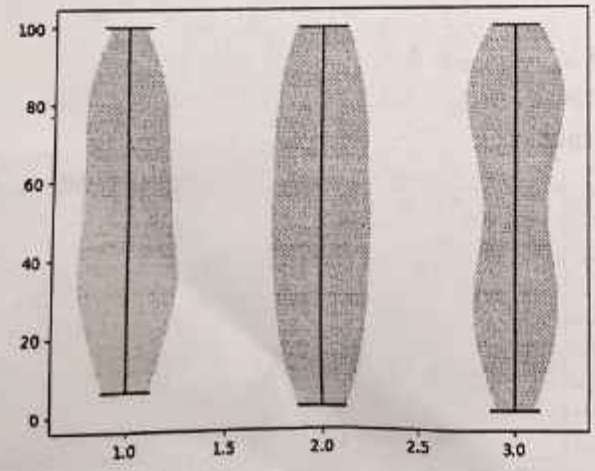
Output



➢ **Example 6.6.2 :** Multiple Violin plots

```
import numpy as np
import matplotlib.pyplot as plt
from random import randint

# Creating 3 empty lists
l1 = []
l2 =[]
l3 =[]

# Filling the lists with random value
for i in range(100):
    n = randint(1, 100)
    l1.append(n)

for i in range(100):
    n = randint(1, 100)
    l2.append(n)

for i in range(100):
    n = randint(1, 100)
    l3.append(n)

random_collection = [l1, l2, l3]

# Create a figure instance
fig = plt.figure()

# Create an axes instance
ax = fig.gca()

# Create the violinplot
violinplot = ax.violinplot(random_collection)
plt.show()
```

Output



6.7 INTRODUCTION TO SEABORN LIBRARY

GQ. Explain seaborn Library?

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and also closely integrated to the data structures from pandas. Seaborn aims to make visualization the central part of exploring and understanding data. It provides dataset-oriented APIs, so that we can switch between different visual representations for same variables for better understanding of dataset.

6.7.1 Different Categories of Plot in Seaborn

Plots are basically used for visualizing the relationship between variables. Those variables can be either be completely numerical or a category like a group, class or division. Seaborn divides plot into the below categories –

- **Relational plots:** This plot is used to understand the relation between two variables.
- **Categorical plots:** This plot deals with categorical variables and how they can be visualized.
- **Distribution plots:** This plot is used for examining univariate and bivariate distributions
- **Regression plots:** The regression plots in seaborn are primarily intended to add a visual guide that helps to emphasize patterns in a dataset during exploratory data analyses.
- **Matrix plots:** A matrix plot is an array of scatterplots.
- **Multi-plot grids:** It is an useful approach is to draw multiple instances of the same plot on different subsets of the dataset.
- Installation
- For python environment :
- pip install seaborn
- For conda environment :
- conda install seaborn

Dependencies

- Python 3.6+
- numpy (>= 1.13.3)
- scipy (>= 1.0.1)
- pandas (>= 0.22.0)
- matplotlib (>= 2.1.2)
- statsmodel (>= 0.8.0)

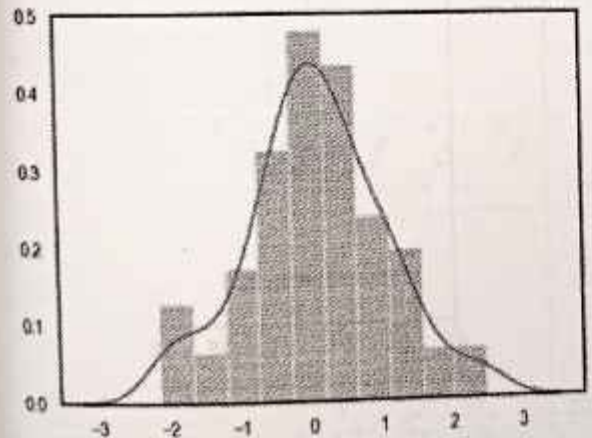
Some basic plots using seaborn

Dist plot : Seaborn dist plot is used to plot a histogram, with some other variations like kdeplot and rugplot.

Python3

```
# Importing libraries
import numpy as np
import seaborn as sns


# Selecting style as white,
# dark, whitegrid, darkgrid
# or ticks
sns.set(style="white")

# Generate a random univariate
# dataset
rs = np.random.RandomState(10)
d = rs.normal(size=100)

# Plot a simple histogram and kde
# with binsize determined automatically
sns.distplot(d, kde=True, color="m")
```

Output



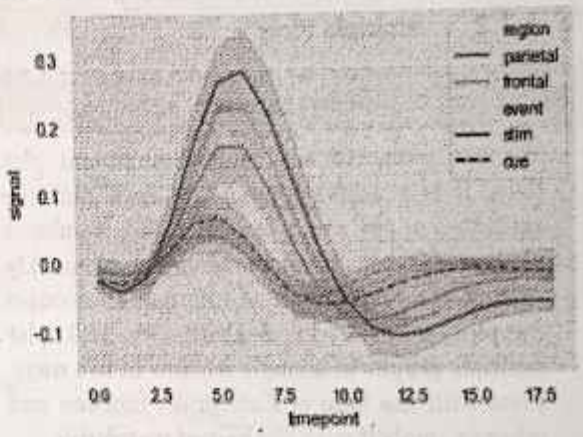
Line plot

The line plot is one of the most basic plot in seaborn library. This plot is mainly used to visualize the data in form of some time series, i.e. in continuous manner.

Python 3

```
import seaborn as sns

sns.set(style="dark")
fmri = sns.load_dataset("fmri")

# Plot the responses for different\
# events and regions
sns.lineplot(x="timepoint",
             y="signal",
             hue="region",
             style="event",
             data=fmri)
```

Output



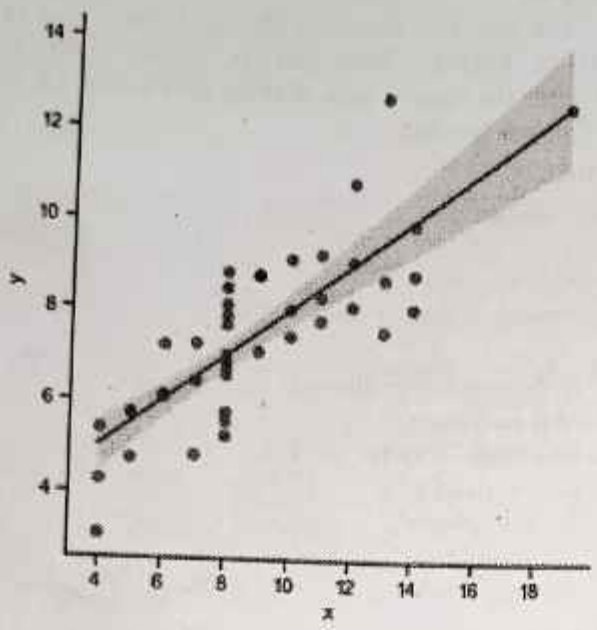
Lmplot

The lmplot is another most basic plot. It shows a line representing a linear regression model along with data points on the 2D-space and x and y can be set as the horizontal and vertical labels respectively.

Python3

```
import seaborn as sns

sns.set(style="ticks")

# Loading the dataset
df = sns.load_dataset("anscombe")

# Show the results of a linear regression
sns.lmplot(x="x", y="y", data=df)
```

Output



6.7.2 Multiple Plots

GQ Explain Multiple Plots in detail?

- We are going to see multi-dimensional plot data, It is a useful approach to draw multiple instances of the same plot on different subsets of your dataset. It allows a viewer to quickly extract a large amount of information about a complex dataset. In Seaborn, we will plot multiple graphs in a single window in two ways. First with the help of Facetgrid() function and other by implicit with the help of matplotlib.
- FacetGrid: FacetGrid is a general way of plotting grids based on a function. It helps in visualizing distribution of one variable as well as the relationship between multiple variables. Its object uses the dataframe as Input and the names of the variables that shape the column, row, or color dimensions of the grid, the syntax is given below:
- **Syntax:** seaborn.FacetGrid(data, **kwargs)
- data: Tidy dataframe where each column is a variable and each row is an observation.
- **kwargs: It uses many arguments as input such as, i.e. row, col, hue, palette etc.

Below is the implementation of above method:

Import all Python libraries needed

Python3

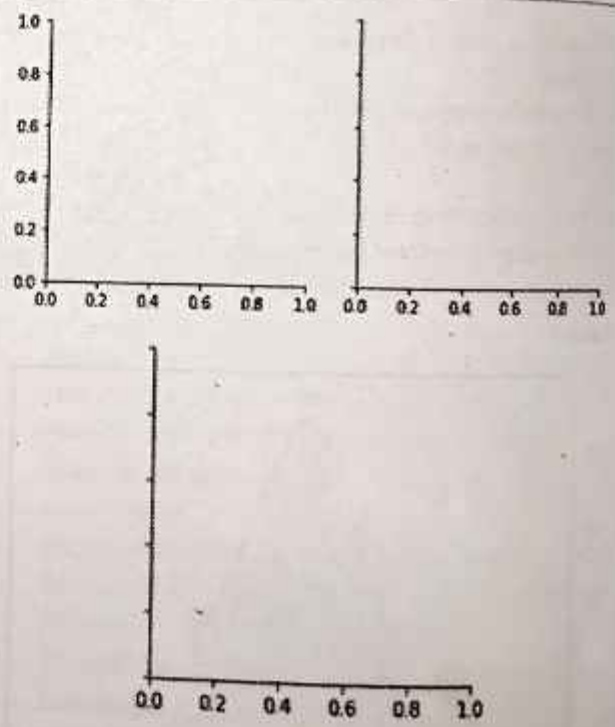
```
import seaborn as sns
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

➢ **Example 6.7.1 :** Here, we are Initializing the grid like this sets up the matplotlib figure and axes, but doesn't draw anything on them, we are using the Exercise dataset which is well known dataset available as an inbuilt dataset in seaborn. The basic usage of the class is very similar to FacetGrid. First you initialize the grid, then you pass plotting function to a map method and it will be called on each subplot.

Python3

```
# loading of a dataframe from seaborn
exercise = sns.load_dataset("exercise")

# Form a facetgrid using columns
sea = sns.FacetGrid(exercise, col = "time")
```

Output

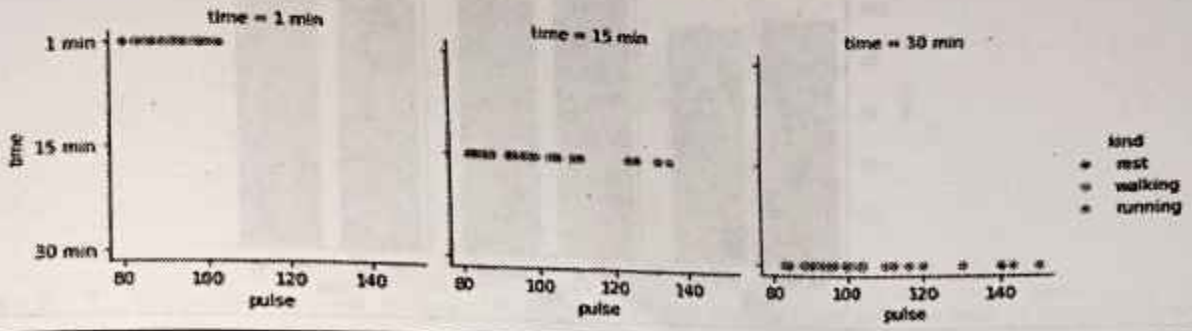


➢ **Example 6.7.2 :** This function will draw the figure and annotate the axes. To make a relational plot, First, you initialize the grid, then you pass the plotting function to a map method and it will be called on each subplot.

Python3

```
# Form a facetgrid using columns with a hue
sea = sns.FacetGrid(exercise, col = "time", hue = "kind")

# map the above form facetgrid with some attributes
sea.map(sns.scatterplot, "pulse", "time", alpha = .8)

# adding legend
sea.add_legend()
```

Output

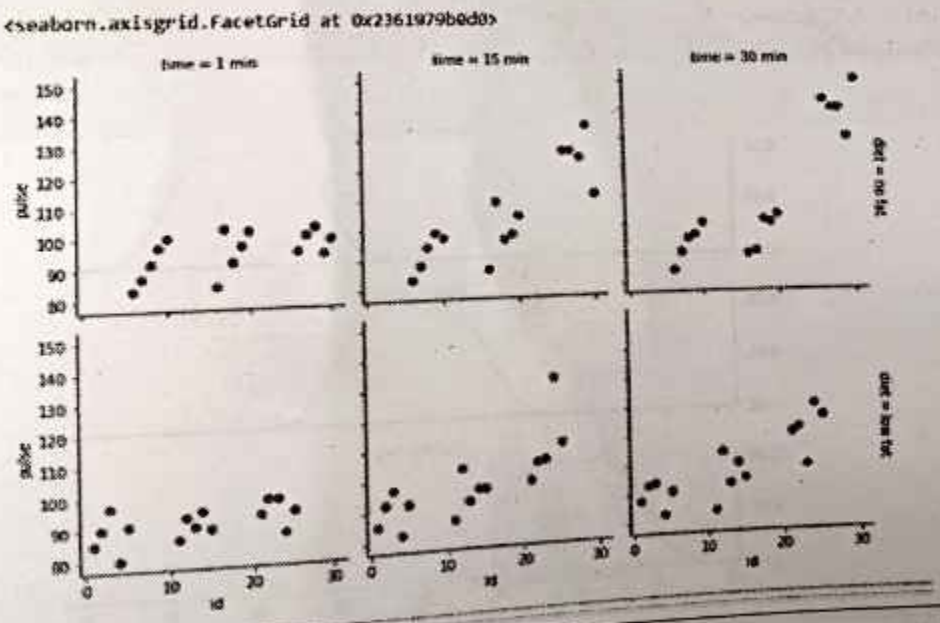


➢ **Example 6.7.3 :** There are several options for controlling the look of the grid that can be passed to the class constructor.

Python3

```
sea = sns.FacetGrid(exercise, row = "diet",
                    col = "time", margin_titles = True)
sea.map(sns.regplot, "id", "pulse", color = ".3",
        fit_reg = False, x_jitter = .1)
```

Output

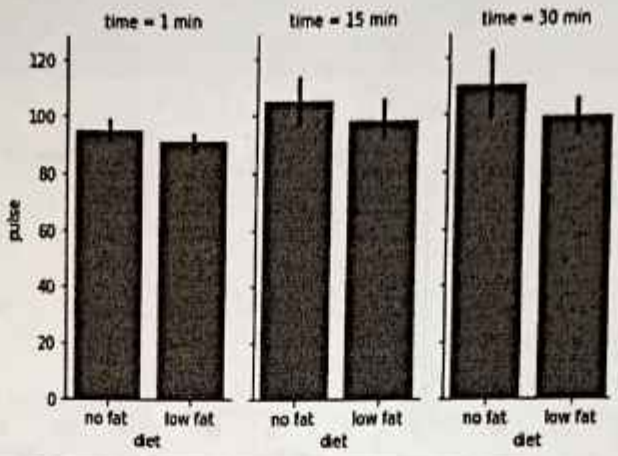


➤ **Example 6.7.4 :** The size of the figure is set by providing the height of each facet, along with the aspect ratio:

Python3

```
sea = sns.FacetGrid(exercise, col = "time",
                    height = 4, aspect =.5)

sea.map(sns.barplot, "diet", "pulse",
        order = ["no fat", "low fat"])
```

Output

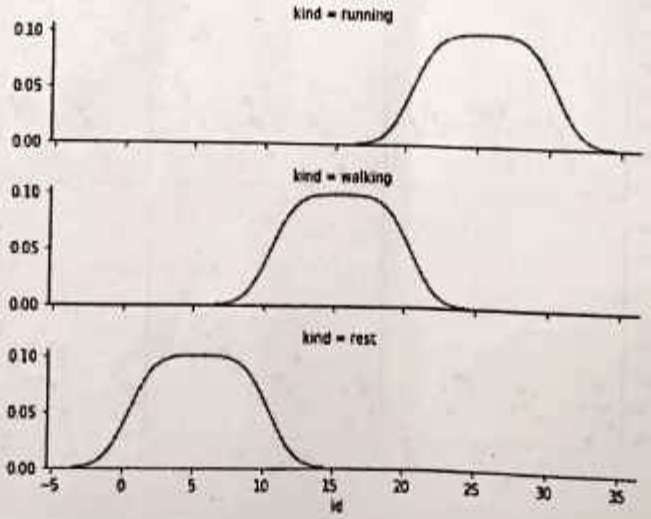


➤ **Example 6.7.5 :** The default ordering of the facets is derived from the information in the DataFrame. If the variable used to define facets has a categorical type, then the order of the categories is used. Otherwise, the facets will be in the order of appearance of the category levels. It is possible, however, to specify an ordering of any facet dimension with the appropriate *_order parameter:

Python3

```
exercise_kind = exercise.kind.value_counts().index
sea = sns.FacetGrid(exercise, row = "kind",
                    row_order = exercise_kind,
                    height = 1.7, aspect = 4)
sea.map(sns.kdeplot, "id")
```

Output

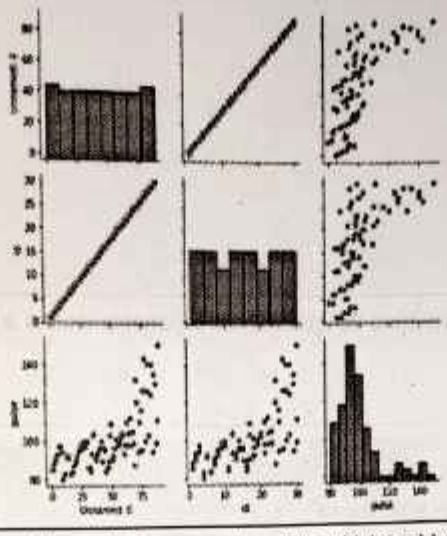


➢ **Example 6.7.6 :** If you have many levels of one variable, you can plot it along the columns but "wrap" them so that they span multiple rows. When doing this, you cannot use a row variable.

Python3

```
g = sns.PairGrid(exercise)
g.map_diag(sns.histplot)
g.map_offdiag(sns.scatterplot)
```

Output

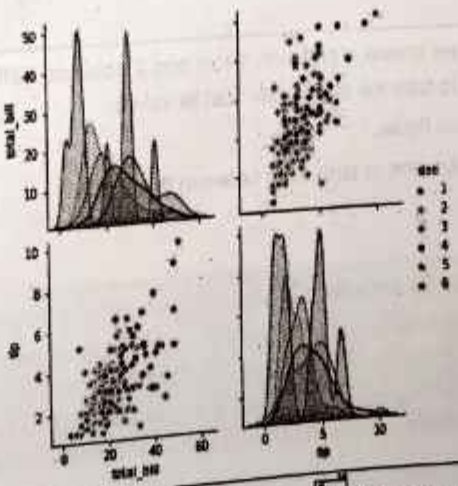


➢ **Example 6.7.7 :** In this example, we will see that we can also plot multiplot grid with the help of pairplot() function. This shows the relationship for (n, 2) combination of variable in a DataFrame as a matrix of plots and the diagonal plots are the univariate plots.

Python3

```
# importing packages
import seaborn
import matplotlib.pyplot as plt

# loading dataset using seaborn
df = seaborn.load_dataset('tips')

# pairplot with hue sex
seaborn.pairplot(df, hue ='size')
plt.show()
```

Output



Method 2 : Implicit with the help of matplotlib.

In this we will learn how to create subplots using matplotlib and seaborn.

Import all Python libraries needed

Python3

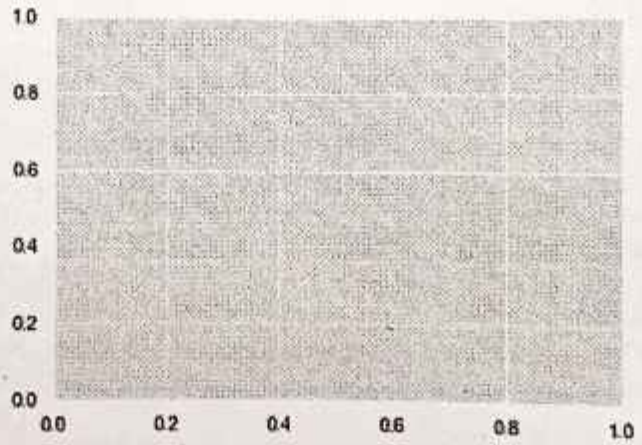
```
import seaborn as sns
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# Setting seaborn as default style even
# if use only matplotlib
sns.set()
```

➢ **Example 6.7.8 :** Here, we are Initializing the grid without arguments returns a Figure and a single Axes, which we can unpack using the syntax bellow.

Python3

```
figure, axes = plt.subplots()
figure.suptitle('Geeksforgeeks - one axes with no data')
```

Output :



➢ **Example 6.7.9 :** In this example we create a plot with 1 row and 2 columns, still no data passed i.e. nrows and ncols. If given in this order, we don't need to type the arg names, just its values.

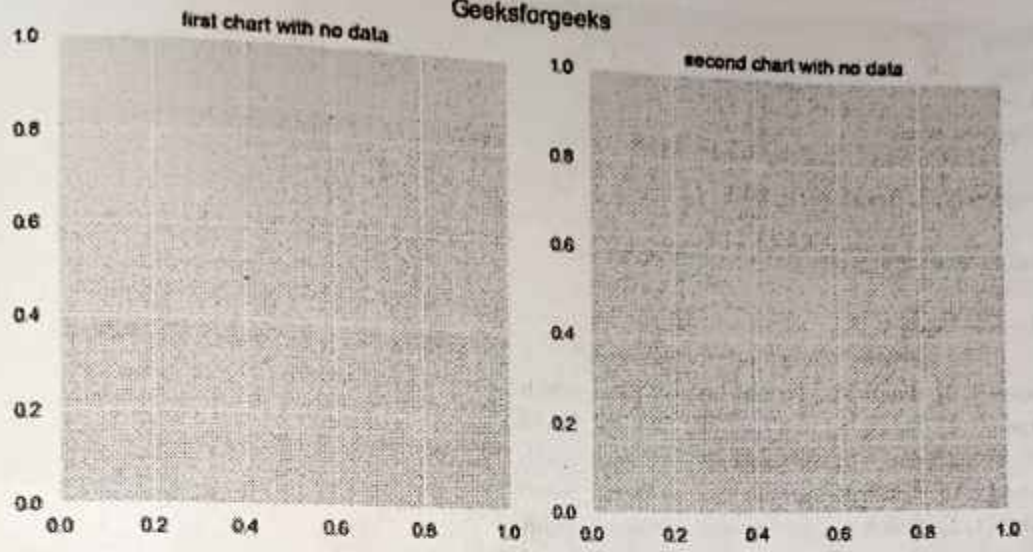
figsize set the total dimension of our figure.

sharex and sharey are used to share one or both axes between the charts.

Python3

```
figure, axes = plt.subplots(1, 2, sharex=True, figsize=(10,5))
figure.suptitle('Geeksforgeeks')
axes[0].set_title('first chart with no data')
axes[1].set_title('second chart with no data')
```

Output

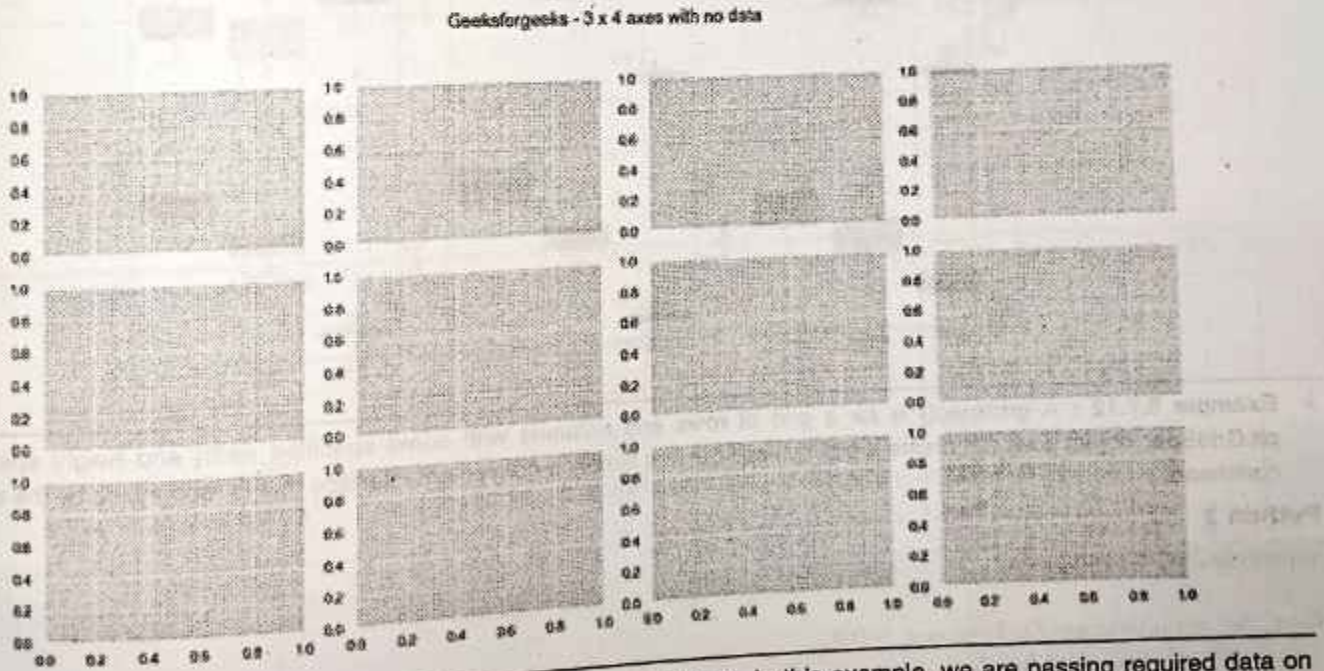


➢ **Example 6.7.10 :** If you have many levels

Python3

```
figure, axes = plt.subplots(3, 4, sharex=True, figsize=(16,8))
figure.suptitle('Geeksforgeeks - 3 x 4 axes with no data')
```

Output

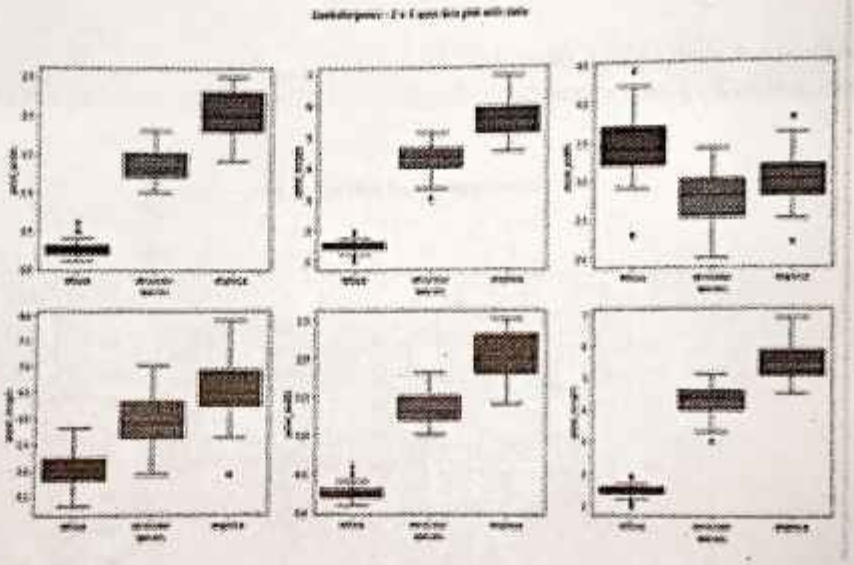


➢ **Example 6.7.11 :** Here, we are Initializing matplotlib figure and axes, In this example, we are passing required data on them with the help of the Exercise dataset which is a well-known dataset available as an inbuilt dataset in seaborn. By using this method you can plot any number of the multi-plot grid and any style of the graph by implicit rows and columns with the help of matplotlib in seaborn. We are using sns.boxplot here, where we need to set the argument with the correspondent element from the axes variable.

Python3

```
import seaborn as sns
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

fig, axes = plt.subplots(2, 3, figsize=(18, 10))

fig.suptitle('Geeksforgeeks - 2 x 3 axes Box plot with data')

iris = sns.load_dataset("iris")

sns.boxplot(ax=axes[0, 0], data=iris, x='species', y='petal_width')
sns.boxplot(ax=axes[0, 1], data=iris, x='species', y='petal_length')
sns.boxplot(ax=axes[0, 2], data=iris, x='species', y='sepal_width')
sns.boxplot(ax=axes[1, 0], data=iris, x='species', y='sepal_length')
sns.boxplot(ax=axes[1, 1], data=iris, x='species', y='petal_width')
sns.boxplot(ax=axes[1, 2], data=iris, x='species', y='petal_length')
```

Output

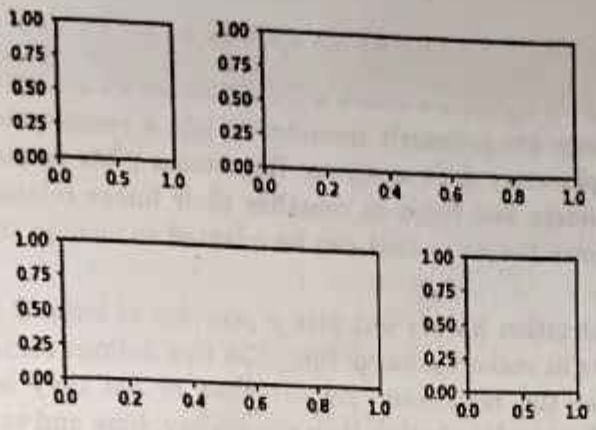


➢ **Example 6.7.12 :** A gridspec() is for a grid of rows and columns with some specified width and height space. The plt.GridSpec object does not create a plot by itself but it is simply a convenient interface that is recognized by the subplot() command.

Python 3

```
import matplotlib.pyplot as plt

Grid_plot = plt.GridSpec(2, 3, wspace = 0.8,
                         hspace = 0.6)

plt.subplot(Grid_plot[0, 0])
plt.subplot(Grid_plot[0, 1:])
plt.subplot(Grid_plot[1, :2])
plt.subplot(Grid_plot[1, 2])
```

Output

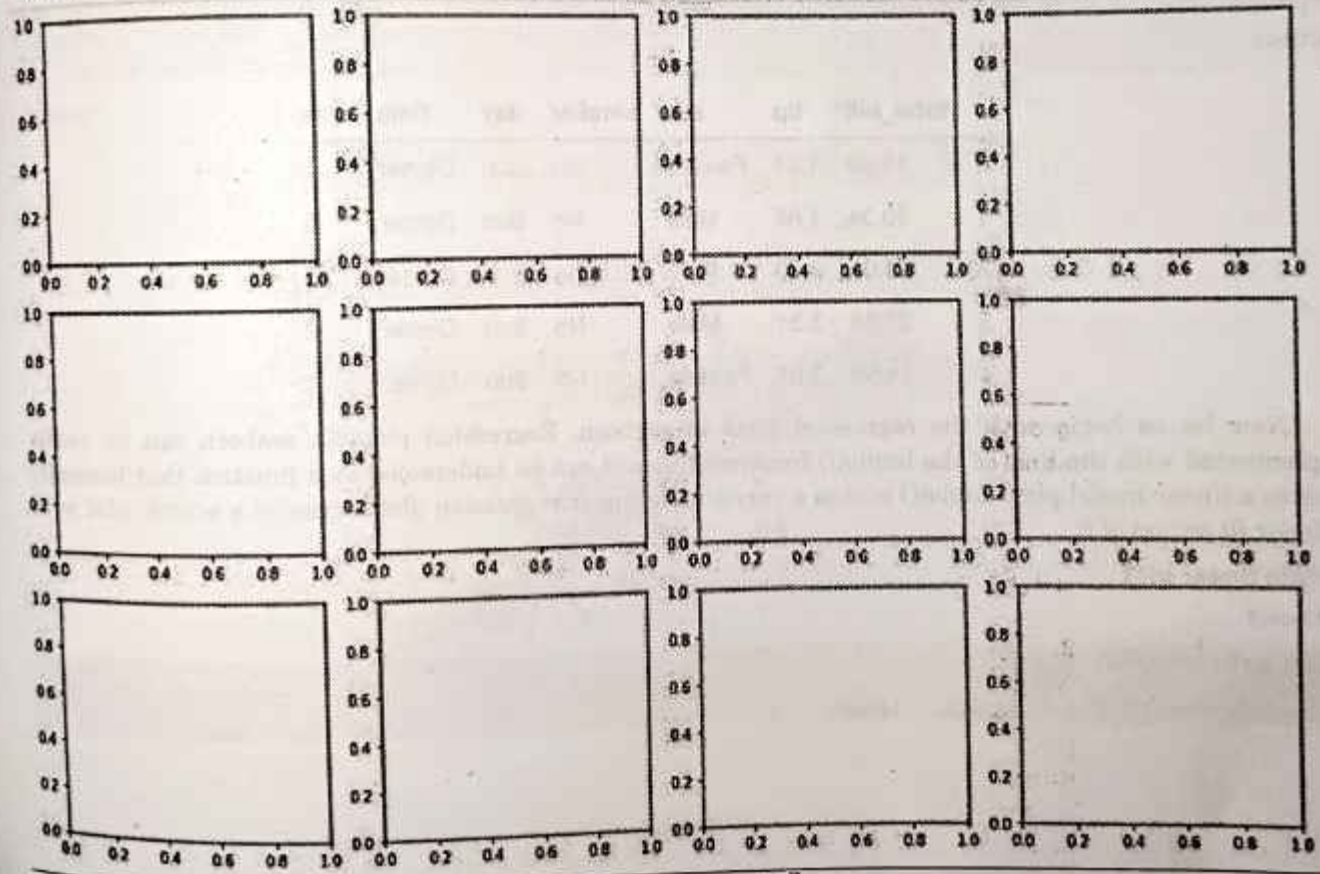


Example 6.7.13 : Here we'll create a 3x4 grid of subplot using subplots(), where all axes in the same row share their y-axis scale, and all axes in the same column share their x-axis scale.

Python3

```
import matplotlib.pyplot as plt

figure, axes = plt.subplots(3, 4,
                            figsize = (15, 10))

figure.suptitle('Geeksforgeeks - 2 x 3 axes grid plot using subplots')
```

Output



6.7.3 Regression Plot

GQ. Explain Regression plot

The regression plots in seaborn are primarily intended to add a visual guide that helps to emphasize patterns in a dataset during exploratory data analyses. Regression plots as the name suggests creates a regression line between 2 parameters and helps to visualize their linear relationships. We consider those kinds of plots in seaborn and shows the ways that can be adapted to change the size, aspect, ratio etc. of such plots.

Seaborn is not only a visualization library but also a provider of built-in datasets. Here, we will be working with one of such datasets in seaborn named 'tips'. The tips dataset contains information about the people who probably had food at the restaurant and whether or not they left a tip. It also provides information about the gender of the people, whether they smoke, day, time and so on.

Let us have a look at the dataset first before we start with the regression plots.

Load the dataset

Python3

```
# import the library
import seaborn as sns

# load the dataset
dataset = sns.load_dataset('tips')

# the first five entries of the dataset
dataset.head()
```

Output

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

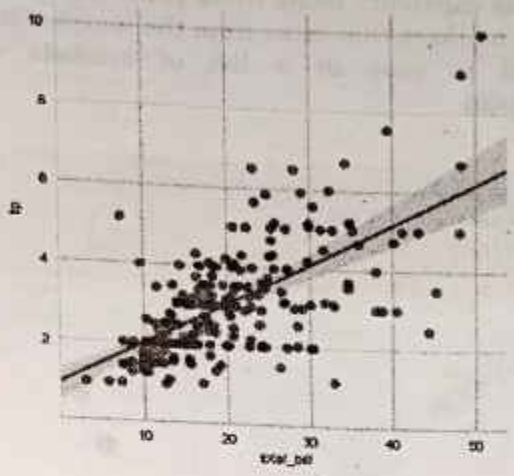
]Now let us begin with the regression plots in seaborn. Regression plots in seaborn can be easily implemented with the help of the lmplot() function. lmplot() can be understood as a function that basically creates a linear model plot. lmplot() makes a very simple linear regression plot.It creates a scatter plot with a linear fit on top of it.

Simple linear plot

Python3

```
sns.set_style('whitegrid')
sns.lmplot(x ='total_bill', y ='tip', data = dataset)
```

Output



Explanation

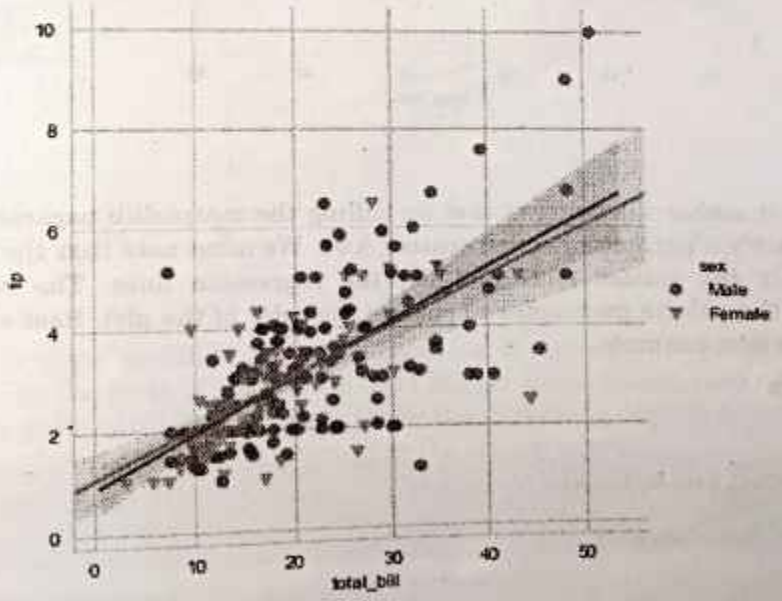
x and y parameters are specified to provide values for the x and y axes. sns.set_style() is used to have a grid in the background instead of a default white background. The data parameter is used to specify the source of information for drawing the plots.

Linear plot with additional parameters

Python3

```
sns.set_style('whitegrid')
sns.lmplot(x ='total_bill', y ='tip', data = dataset,
       hue ='sex', markers =['o', 'v'])
```

Output



Explanation

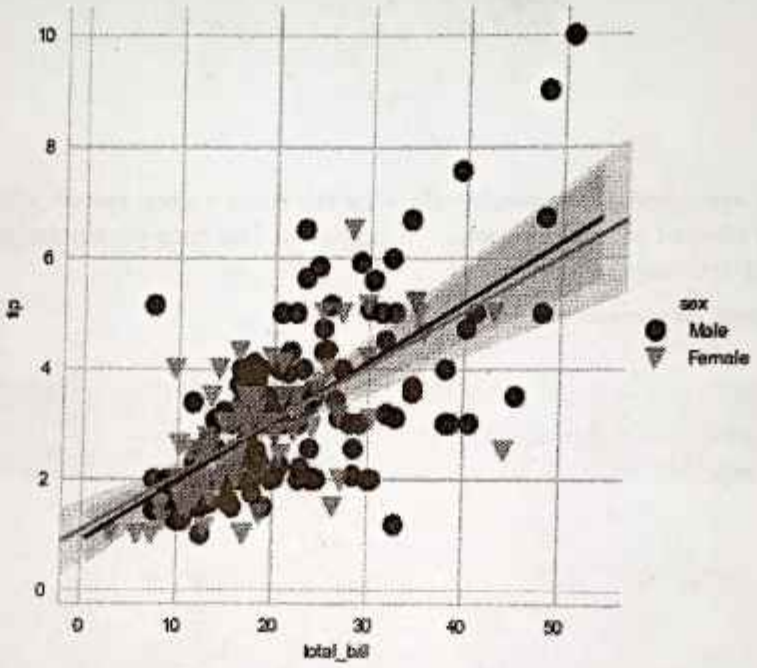
In order to have a better analysis capability using these plots, we can specify hue to have a categorical separation in our plot as well as use markers that come from the matplotlib marker symbols. Since we have two separate categories we need to pass in a list of symbols while specifying the marker.

Setting the size and color of the plot

Python3

```
sns.set_style('whitegrid')
sns.lmplot(x ='total_bill', y ='tip', data = dataset, hue ='sex',
        markers =['o', 'v'], scatter_kws ={'s':100},
        palette ='plasma')
```

Output



Explanation

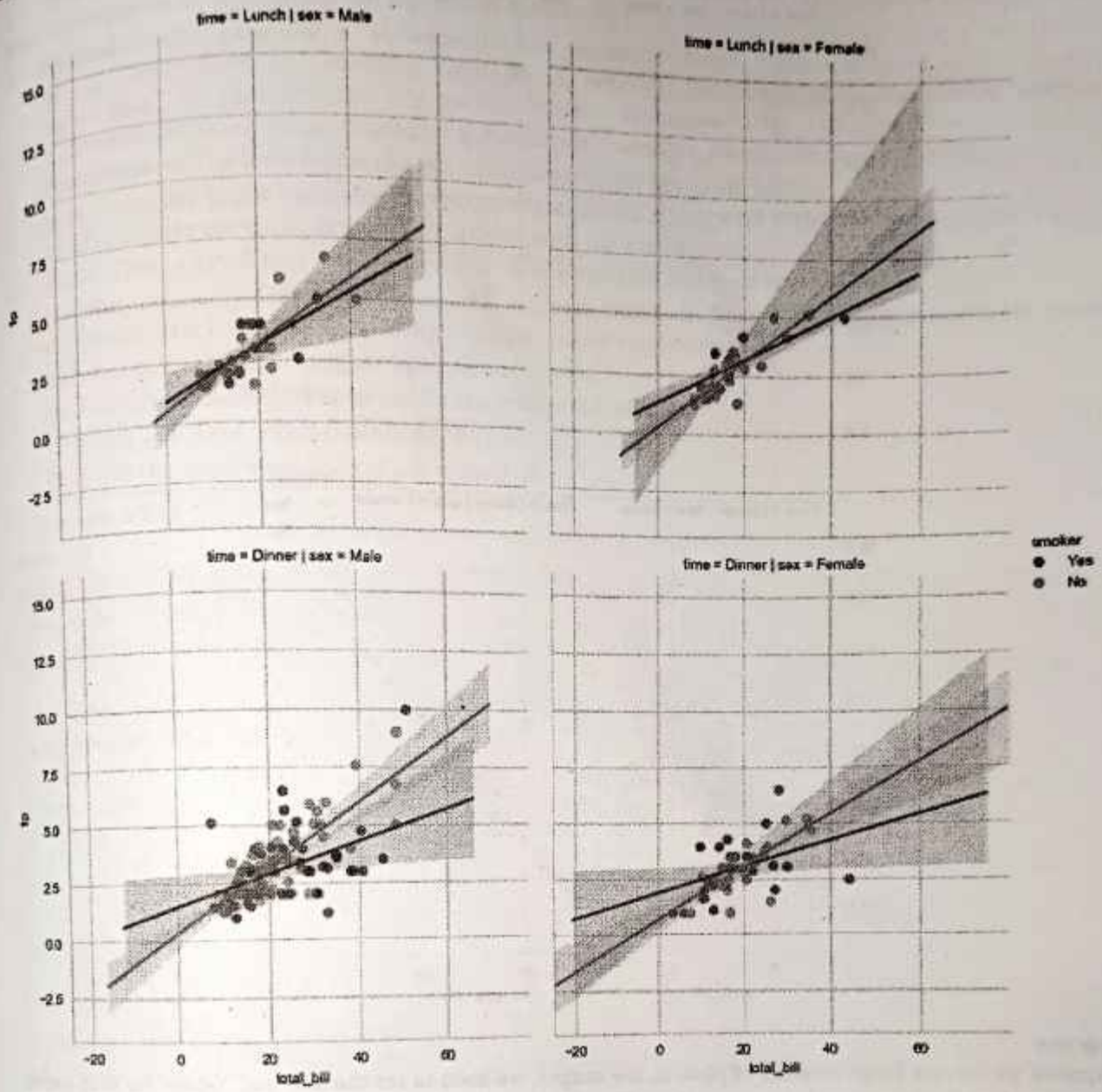
In this example what seaborn is doing is that its calling the matplotlib parameters indirectly to affect the scatter plots. We specify a parameter called scatter_kws. We must note that the scatter_kws parameter changes the size of only the scatter plots and not the regression lines. The regression lines remain untouched. We also use the palette parameter to change the color of the plot. Rest of the things remain the same as explained in the first example.

Displaying multiple plots

Python3

```
sns.lmplot(x ='total_bill', y ='tip', data = dataset,
        col ='sex', row ='time', hue ='smoker')
```

Output



Explanation

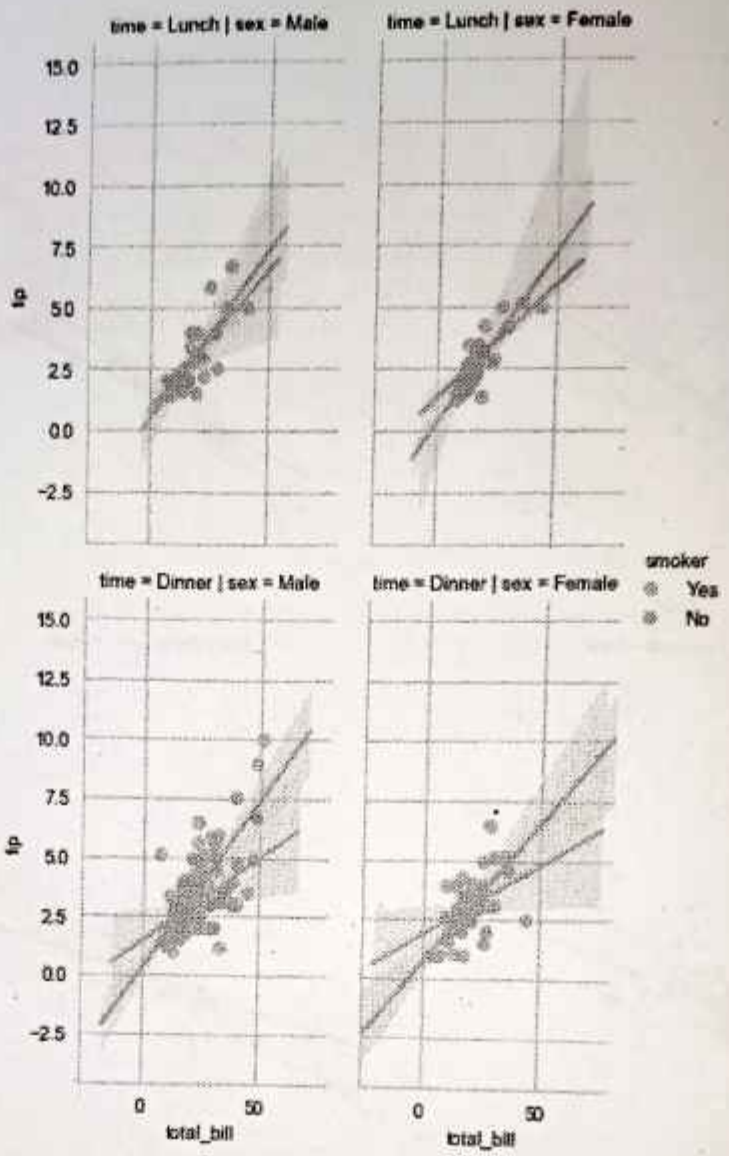
In the above code, we draw multiple plots by specifying a separation with the help of the rows and columns. Each row contains the plots of tips vs the total bill for the different times specified in the dataset. Each column contains the plots of tips vs the total bill for the different genders. A further separation is done by specifying the hue parameter on the basis of whether the person smokes.

Size and aspect ratio of the plots

Python3

```
sns.lmplot(x ='total_bill', y ='tip', data = dataset, col ='sex',
        row ='time', hue ='smoker', aspect = 0.6,
        size = 4, palette ='coolwarm')
```

Output



Explanation

Suppose we have a large number of plots in the output, we need to set the size and aspect for it in order to better visualize it. aspect: scalar, optional specifies the aspect ratio of each facet, so that "aspect * height" gives the width of each facet in inches.

6.7.4 Regplot

GQ. Explain Regplot?

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn helps resolve the two major problems faced by Matplotlib; the problems are ?

- Default Matplotlib parameters
- Working with data frames

As Seaborn compliments and extends Matplotlib, the learning curve is quite gradual. If you know Matplotlib, you are already half-way through Seaborn.

seaborn.regplot() :

This method is used to plot data and a linear regression model fit. There are a number of mutually exclusive options for estimating the regression model.

- **Syntax :** seaborn.regplot(x, y, data=None, x_estimator=None, x_bins=None, x_ci='ci', scatter=True, fit_reg=True, ci=95, n_boot=1000, units=None, order=1, logistic=False, lowess=False, robust=False, logx=False, x_partial=None, y_partial=None, truncate=False, dropna=True, x_jitter=None, y_jitter=None, label=None, color=None, marker='o', scatter_kws=None, line_kws=None, ax=None)
- **Parameters:** The description of some main parameters are given below:
- **x, y:** These are Input variables. If strings, these should correspond with column names in "data". When pandas objects are used, axes will be labeled with the series name.
- **data:** This is dataframe where each column is a variable and each row is an observation.
- **lowess:** (optional) This parameter take boolean value. If "True", use "statsmodels" to estimate a nonparametric lowess model (locally weighted linear regression).
- **color:** (optional) Color to apply to all plot elements.
- **marker:** (optional) Marker to use for the scatterplot glyphs.
- **Return:** The Axes object containing the plot.

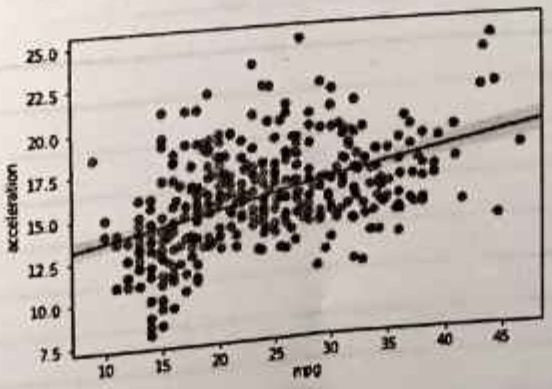
Below is the implementation of above method:

Example 6.7.14

Python3

```
# importing required packages
import seaborn as sns
import matplotlib.pyplot as plt

# loading dataset
data = sns.load_dataset("mpg")

# draw regplot
sns.regplot(x = "mpg",
        y = "acceleration",
        data = data)

# show the plot
plt.show()

# This code is contributed
# by Deepanshu Rustagi.
```

Output

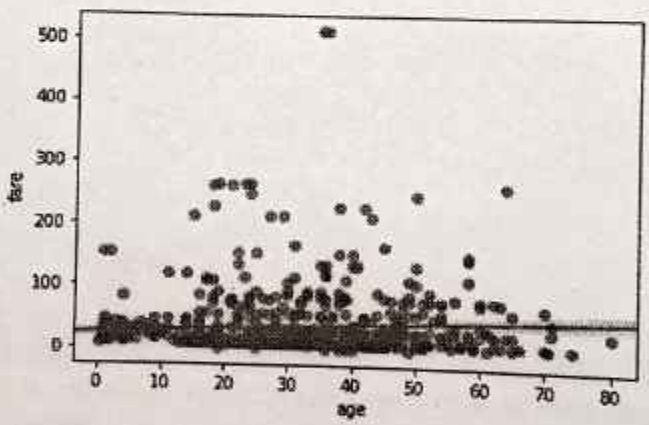


➢ **Example 6.7.15**

Python3

```
# importing required packages
import seaborn as sns
import matplotlib.pyplot as plt

# loading dataset
data = sns.load_dataset("titanic")

# draw regplot
sns.regplot(x = "age",
            y = "fare",
            data = data,
            dropna = True)
# show the plot
plt.show()

# This code is contributed
# by Deepanshu Rustagi.
```

Output



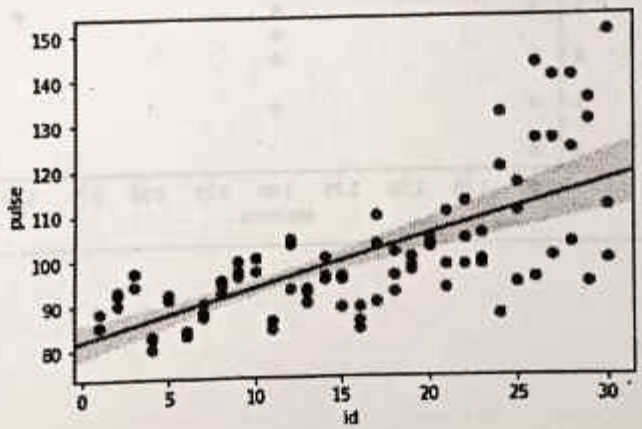
NOTES

➢ **Example 6.7.16**

Python3

```
# importing required packages
import seaborn as sns
import matplotlib.pyplot as plt

# loading dataset
data = sns.load_dataset("exercise")

# draw regplot
sns.regplot(x = "id",
        y = "pulse",
        data = data)

# show the plot
plt.show()

# This code is contributed
# by Deepanshu Rustagi.
```

Output



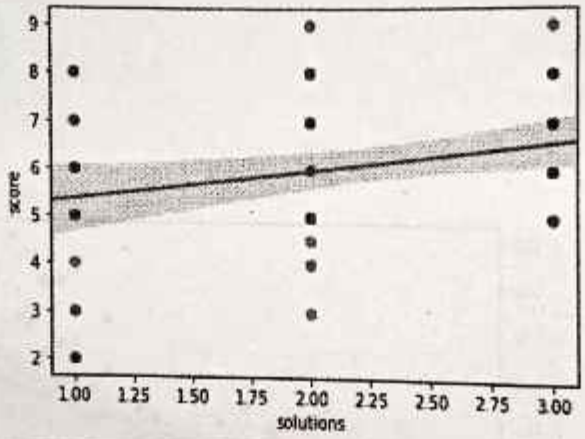
NOTES

➢ **Example 6.7.17**

Python3

```
# importing required packages
import seaborn as sns
import matplotlib.pyplot as plt

# loading dataset
data = sns.load_dataset("attention")

# draw regplot
sns.regplot(x = "solutions",
            y = "score",
            data = data)

# show there plot
plt.show()

# This code is contributed
# by Deepanshu Rustagi.
```

Output



...Chapter Ends

□□□