

Course Pack For Statistical Programming using R

Course: BCA Course

Code: Data Science 504-3A

Semester: V

Year: 2024-25

Course Leader: Dr. Rakhee Chhibber

Course Instructor: Dr. Rakhee Chhibber

Mr. Nripesh Kumar Nrip
Program Coordinator
Director

Dr. Daljeet Singh Bawa
Forwarded by: HOD

Dr. Yamini Agarwal
Approved By:



**Bharati Vidyapeeth (Deemed to be University) Institute of
Management & Research, New Delhi**
An ISO 9001: 2008 & 14001:2004 Certified Institute
A-4, Paschim Vihar, New Delhi-110063
(Ph: 011-25284396, 25285808, Fax No. 011-25286442)

Note: "Strictly for Internal academic use only"

BVIMR SNAPSHOT

Established in 1992, Bharati Vidyapeeth (Deemed to be University) Institute of Management and Research (BVIMR), New Delhi focuses on imbibing the said values across various stakeholders through adequate creation, inclusion, and dissemination of knowledge in management education. The institute has over the past few years emerged in the lead with a vision of Leadership in professional education through innovation and excellence. This excellence is sustained by consistent value enhancement and initiation of value-added academic processes in institute's academic systems.

Based on the fabulous architecture and layout on the lines of Nalanda Vishwa Vidyalaya, the institute is a scenic marvel of lush green landscape with modern interiors. The Institute which is ISO 9001:2015 certified is under the ambit of Bharati Vidyapeeth University (BVU), Pune as approved by Govt. of India on the recommendation of UGC under Section 3 of UGC Act vide its letter notification No. F. 9 – 16 / 2004 – U3 dated 25th February 2005.

Strategically located in West Delhi on the main Rohtak Road, BVIMR, New Delhi has splendid layout on sprawling four acres of plot with 'state-of-art' facilities with all classrooms, Library, Labs, Auditorium etc. that are fully air-conditioned. The Institute that has an adjacent Metro station "Paschim Vihar (East)", connects the entire Delhi and NCR.

We nurture our learners to be job providers rather than job seekers. This is resorted to by fostering the skill and enhancement of knowledge base of our students through various extracurricular, co-curricular and curricular activities by our faculty, who keep themselves abreast by various research and FDPs and attending Seminars/Conferences. The Alumni has a key role here by inception of SAARTHI Mentorship program who update and create professional environment for learners' centric academic ambiance and bridging industry-academia gap.

Our faculty make distinctive contribution not only to students but to Academia through publications, seminars, conferences apart from quality education. We also believe in enhancing corporate level interaction including industrial projects, undertaken by our students under continuous guidance of our faculty. These form the core of our efforts which has resulted in being one of the premier institutes of management.

At BVIMR, we are imparting quality education in management at Doctorate, Postgraduate and Undergraduate levels.

Dr. Rakhee Chhibber (Guest Faculty, BVIMR)

Dr. Rakhee Chhibber

Mobile: 9868080676

E-Mail: rakheechhibber1971@gmail.com

She is Doctorate in Computer science form Mewar University, Rajasthan, MTech (IT) from Karnataka State University, MCA and MBA. She is currently working in IT industry as an Application Developer (Data Science) - Technical Lead and previously worked as Assistant Professor in RDIAS (IPU college) for 10 years. Also Worked as visiting Faculty in BVIMR, Delhi for 5 years. Total Experience is more than 20+yeas in Academics and Industry. Her specialized areas are Data Science, Machine Learning, Artificial Intelligence, R programming, Power BI , Tableau, Python , Java C, C++, teaching subjects are C#, Java, C, C++ programming, Database Management System , Oracle, SQL, PLSQL, ETL tools – informatica, Talend, Data Warehousing and Data Mining , Operating System , Digital Electronics, Computer Organization and Architecture, Software Engineering.

Index

SN	CONTENTS	PAGE NO
1	Course outline	I - XVII
2	Study Notes	1-366
3	Unit-wise MCQ	367-391
4	Practice Questions	392-396
5	Case Studies	397-406
6	Internal Question Paper Format	407-408

Programme: BCA CBCS –Revised Syllabus w.e.f. - Year 2022 – 2023											
Semester	Course Code	Course Title									
V	Data Science 504-3-A	Statistical Programming using R									
	Prepared by	Dr. M.K.Patil									
Type	Credits	Evaluation		Marks							
DSE	3	IA		100							
Course Objectives:											
<ul style="list-style-type: none"> To teach the Beginners of R Programming of a master level. A variety of topics will be covered that are important for Data science to prepare the students for real life prediction of data engineering. To impart knowledge of the concepts related to Probability and Application on data sets. It also gives the idea how data is managed in various environments with emphasis on Predictions measures as implemented in data sets. 											
Course Outcomes:											
CO1: Remember the definitions of concepts and their Implementation in R.											
CO2: Understand the concept of data and statistical techniques for its Implementation.											
CO3: Design different data behaviors and their Predictions.											
CO4: Analyzing Data set & Studying Historical Data.											
CO5: Convert the historical Data into Prediction Model using R											
Unit No.	Unit	Session (Hrs.)	COs Number	Teaching Methodology	Cognition Level	Evaluation Tools					
1	Introduction of Probability Concept, Types of Probability, Permutation and Combination concept, Addition and Multiplication Theorem, Condition Probability, Bayes's Theorem	8	CO 1 CO 2	Lecture with PPTs	Understand	Problems and its Solution					

2	Random Variable Concept, Discrete and Continuous Random Variable, Probability density function, Mathematical Expectation and their Theorem	5	CO 1 CO 2	Problem Illustration	Apply (Analyze)	Problems and its Solution
3	Data Distribution Distribution, Types of Data distribution, Exponential distribution, Binomial distribution, Normal distribution, Poisson distribution, Random number generation, Monte Carlo Simulation.	7	CO 3	Concept Explanation, Mathematical Problems, and its Solution	Analyze	Problems and its Solution
4	Testing of Hypothesis Procedure of Testing Hypothesis, Standard Error and Sampling distribution, Estimation, Student's t-distribution, Chi-Square test and goodness of fit, F-test and analysis of variance. Factor analysis.	5	CO4	Concept Explanation, Mathematical Problems, and its Solution	Evaluate	Problems and its Solution
5	Introduction to R programming language Getting R, Managing R, Arithmetic and Matrix Operations, Introduction to Functions, Control Structures. Working with Objects and Data: Introduction to Objects, Manipulating Objects, Constructing Data Objects, types of Data items, Structure of Data items, Reading and Getting Data, Manipulating Data, Storing Data.	5	CO 5	Software Demonstration and use of R Language Software Demonstration	Create	Problems and its Solution

6	Graphical Analysis using R Basic Plotting, Manipulating the plotting window, Box Whisker Plots, Scatter Plots, Pair Plots, Pie Charts, Bar Charts.	5	CO 5	Software Demonstration and use of R Language	Evaluate	Problems and its Solution
7	Advanced R Statistical models in R, Correlation and regression analysis, Analysis of Variance (ANOVA), creating data for complex analysis, Summarizing data, and case studies.	10	CO 5	Software Demonstration and use of R Language	Evaluate	Problems and its Solution

Text Books	"Fundamentals of Statistics" Seven Edition By S.C.Gupta
References	<p>1."Fundamentals of Statistics" Seven Edition By S.C.Gupta</p> <p>2.“R Programming Fundamentals by KaelenMedeiras</p> <p>3.“ Reinforcement Learning e-book.</p> <p>4. Learning R Programming Guide on line</p> <p>Suggested MOOC :Please refer these websites for MOOCS: NPTEL / Swayam www. edx.com, www.coursera.com</p>

1. CO-PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1 0	PO1 1	PO1 2
CO1	3	3	-	-	1	-	2	-	-	-	-	-
CO2	2	3	2	-	1	-	2	-	-	-	-	-
CO3	3	3	3	3	2	1	2	-	-	-	-	-
CO4	1	3	3	-	2	-	-	-	-	-	-	-
CO5	1	3	3	3	3	1	-	-	-	-	-	-
CO.	1.67	2.5	1.83	1.16	1.67	0.5	1	-	-	-	-	0.16
CO	2	3	2	1	2	1	1					0

1- Low , 2- Medium, 3- High, If no correlation put ‘-’

2. Evaluation

Internals: 40%

Externals: 60%

Total : 100%

3. Assessment Mapping

Parameter	Marks	CO1	CO2	CO3	CO4	CO5
Class Participation/ Attendance	20	4	5	4	4	2
Assignments/Projects	10	1	2	3	3	1
Test	10	2	2	2	3	1
Internal	40	7	9	9	10	4
End Term(Univ)	60					

4. Rationale for Mapping Program Outcomes and Course Outcomes:

CO1 & PO1 Mapped at 3	These objectives and outcomes provide students with a comprehensive understanding of programming fundamentals and their practical applications.
CO1 & PO 2 Mapped at 3	These objectives and outcomes prepare students to excel in both programming fundamentals and problem-solving skills in the field of computer science.
CO1 & PO5 Mapped at 1	These objectives and outcomes work together to prepare students to excel in both the foundational aspects of programming and the application of cutting-edge technology in software development.
CO1 & PO7 Mapped at 2	These objectives and outcomes work together to prepare students to excel in programming fundamentals and to maintain relevance and competitiveness as computing professionals.
CO2 & PO1 Mapped at 2	These focus on the importance of applying mathematical and computational knowledge to conceptualize and address problems in various domains, ensuring that students can apply what they've learned effectively.
CO2 & PO2 Mapped at 3	These objectives and outcomes prepare students for success in computer science and problem-solving.
CO2 & PO3 Mapped at 2	These objectives and outcomes prepare students for success in the field of computer science, problem-solving, and technology integration.
CO2 & PO5 Mapped at 1	These objectives and outcomes improve students' understanding and retention of core computer science concepts, dynamic programming, and more.

Mapped at 1	driven problem-solving ensuring that students make responsible and compliant decisions in their computing practices.
CO2 & PO7 Mapped at 2	These objectives and outcomes improve students' understanding and emphasize the need for ongoing learning and adaptation within the ever-evolving computing industry, preparing students to excel as computing professionals.
CO3 & PO1 Mapped at 3	These objectives and outcomes prepare students for proficiency in data structures and computational problem-solving.
CO3 & PO2 Mapped at 3	These objectives and outcomes prepare students to excel in data structures and problem-solving in the field of computer science.
CO3 & PO3 Mapped at 3	These objectives and outcomes prepare students for success in data structures, problem-solving, and technology integration in practical contexts.
CO3 & PO4 Mapped at 3	These objectives and outcomes provide students with a strong foundation in data structures and emphasizes the ability to use scientific methods to experiment, collect data, and draw meaningful conclusions, ensuring a comprehensive skill set in the field of computer science.
CO3 & PO5 Mapped at 2	These objectives and outcomes equip students with a strong foundation in software development in today's rapidly evolving technological landscape.
CO3 & PO7 Mapped at 2	These objectives and outcomes ensure that students are well-prepared for success as computing professionals.
CO4 & PO1 Mapped at 1	These objectives and outcomes prepare students for proficiency in handling data in this common format.
CO4 & PO2 Mapped at 3	These objectives and outcomes prepare students for proficiency in working with CSV data and addressing related challenges.
CO4 & PO3 Mapped at 3	These objectives and outcomes prepare students to excel in data analysis and problem-solving in the context of emerging technologies

	and business scenarios.
CO4 & PO5 Mapped at 2	These objectives and outcomes focus on teaching students how to analyze problems based on CSV files, and techniques for developing innovative software solutions, ensuring students are well-prepared for success in data analysis.
CO5 & PO1 Mapped at 1	These objectives and outcomes ensure that students can make well-informed decisions about data structures in their problem-solving processes.
CO5 & PO2 Mapped at 3	These objectives and outcomes prepare students for proficiency in data structure selection and problem-solving in the field of computerscience.
CO5 & PO3 Mapped at 3	These objectives and outcomes prepare students to excel in data structure selection and problem-solving using emerging technologies in practical contexts.
CO5 & PO4 Mapped at 3	These objectives and outcomes ensure students are well-prepared for effective data-driven problem-solving.
CO5 & PO 5 Mapped at 3	These objectives and outcomes prepare students for effective data- driven problem-solving and software development in a rapidly evolving technological landscape.

5. Session plan

Session	Topic	Pedagogy	Learning Outcome
Module I: Introduction of Probability			
1	Introduction to Probability	Lecture with PPT and Discussion	CO1, CO2
2	Probability concept	Lecture with PPT and Discussion	CO1, CO2
3	Types of Probability	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
4	Permutation and Combination concept	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
5	Addition Theorem	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
6	Multiplication Theorem	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
7	Condition Probability	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
8	Bayes's Theorem	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2

Module II: Random Variable			
9	Concept of Random Variables	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
10	Discrete and Continuous Random Variable	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
11	Probability density function	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
12-13	Mathematical Expectation and their Theorem	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2
Module III: Data Distribution			
14	Data Distribution concepts	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3
15	Types of Data distribution	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3
16	Exponential distribution	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3
17	Binomial distribution	Lecture with PPT, sample examples, Discussion and	CO1, CO2, CO3

		practice questions	
18	Normal distribution,	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3
19	Poisson distribution	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3
20	Random number generation, Monte Carlo Simulation.	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3
Module IV: Testing of Hypothesis			
21	CES 1		
22	Introduction to Hypothesis in Data Analysis	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3, CO4
23	Procedure of Testing Hypothesis,	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3, CO4
24	Standard Error and Sampling distribution, Estimation,	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3, CO4

25	Student's t-distribution, Chi-Square test and goodness of fit, F-test and analysis of variance. Factor analysis.	Lecture with PPT, sample examples, Discussion and practice questions	CO1, CO2, CO3, CO4
----	---	--	--------------------

Module V: Introduction to R programming language

26	Introduction to R programming language, Getting R, Managing R, Arithmetic and Matrix Operations	Lecture with PPT and practical on R-studio with coding	CO1, CO2, CO5
27	Introduction to Functions, Control Structures.	Lecture with PPT and practical on R-studio with coding	CO1, CO2, CO5
28	Working with Objects and Data: Introduction to Objects, Manipulating Objects, Constructing Data Objects	Lecture, Demonstration and Practical Exercise	CO1, CO2, CO5
29	types of Data items, Structure of Data items,	Demonstration Practical Exercise	CO1, CO2, CO5
30	Reading and Getting Data, Manipulating Data, Storing Data.	Demonstration Practical Exercise	CO1, CO2, CO5

Module VI: Graphical Analysis using R

31	Basic Plotting,	Lecture, Demonstration and Practical Exercise	CO1, CO2, CO3, CO5
----	-----------------	--	--------------------

32	Manipulating the plotting window	Lecture, Demonstration and Practical Exercise	CO1, CO2, CO3, CO5
33	Box Whisker Plots,	Demonstration Practical Exercise	CO1, CO2, CO3, CO5
34	Scatter Plots, Pair Plots	Demonstration Practical Exercise	CO1, CO2, CO3, CO5
35	Pie Charts, Bar Charts.	Lecture, Demonstration and Practical Exercise	CO1, CO2, CO3, CO5
Module VI: Advanced R			
36	CES 2		
37-38	Statistical models in R	Lecture with PPT, Demonstration and PracticalExercise	CO1, CO2, CO3, CO5
39	Correlation Analysis using R	Lecture with PPT, Demonstration and PracticalExercise	CO1, CO2, CO3, CO5
40	Regression analysis using R	Lecture with PPT, Demonstration and PracticalExercise	CO1, CO2, CO3, CO5
41-42	Analysis of Variance (ANOVA),	Lecture with PPT, Demonstration and PracticalExercise	CO1, CO2, CO3, CO5
43	Creating data for complex analysis	Lecture with PPT, Demonstration and PracticalExercise	CO1, CO2, CO3, CO5
44	Summarizing data	Lecture with PPT, Demonstration	CO1, CO2, CO3, CO5

		and PracticalExercise	
45	Case studies	Lecture with PPT, Demonstration and PracticalExercise	CO1, CO2, CO3, CO5

6. Textbook:

1. Artificial Intelligence by Elaine Rich and Kevin Knight, Tata McGraw Hill
2. Understanding Machine Learning. Shai Shalev-Shwartz and Shai Ben-David. Cambridge University Press.
3. Artificial Neural Network, B. Yegnanarayana, PHI, 2005 Tom Mitchell, “Machine Learning”, McGraw Hill, 1997
2. E. Alpaydin, “Introduction to Machine Learning”, PHI, 2005.

8. Reference Book:

1. Christopher M. Bishop. Pattern Recognition and Machine Learning (Springer)
2. Introduction to Artificial Intelligence and Expert Systems by Dan W. Patterson, Prentice Hall of India
3. Andrew Ng, Machine learning yearning, <https://www.deeplearning.ai/machine-learning-yearning/>
4. Aurolien Geron,” Hands-On Machine Learning with Scikit-Learn and TensorFlow, Shroff/O'Reilly”,2017
5. Andreas Muller and Sarah Guido,” Introduction to Machine Learning with Python: A Guide for Data Scientists”, Shroff/O'Reilly, 2016

9. MOOC

- a) Swayam : https://onlinecourses.swayam2.ac.in/cec22_cs20/preview
- b) NEPTEL https://onlinecourses.nptel.ac.in/noc19_cs41/preview
- c) EDX : <https://www.edx.org/learn/python>

Unit 1

Introduction to

Probability

Topics

- Introduction of Probability Concept
 - Types of Probability
 - Permutation and Combination concept
 - Addition and Multiplication Theorem
 - Condition Probability
 - Bayes's Theorem
-

Introduction of Probability Concept

History of Probability

Probability theory, as a formal branch of mathematics, traces its roots back to the 16th and 17th centuries. However, the concepts underpinning probability have existed much earlier in the form of gambling and games of chance. Ancient civilizations like the Egyptians, Greeks, and Romans used early forms of probability in divination and decision-making processes. Despite these early instances, it wasn't until the Renaissance that probability began to be studied systematically.

The Beginnings: The Renaissance and the Birth of Probability Theory

The formal study of probability is often attributed to the correspondence between French mathematicians Blaise Pascal and Pierre de Fermat in the 1650s. Their discussions were centered around problems in gambling, such as the “problem of points,” which concerned dividing stakes in a game of chance that was interrupted before its conclusion. This collaboration laid the groundwork for the mathematical theory of probability.

During the same period, the Italian mathematician Gerolamo Cardano, in his work **Liber de Ludo Aleae** (The Book on Games of Chance), was one of the first to formalize the calculation of odds and outcomes in gambling. Although Cardano's work was published posthumously, it demonstrated a clear understanding of the principles of probability and laid a foundation for future developments.

The Development of Classical Probability: 17th to 18th Centuries

The 17th century saw the formalization of probability theory, which continued into the 18th century. One of the key figures during this period was the Dutch mathematician Christiaan Huygens, who, in 1657, published the first book on probability theory titled **De Ratiociniis in Ludo Aleae** (On Reasoning in Games of Chance). Huygens' work built upon the ideas of Pascal and Fermat, further formalizing the concepts of expected value and fair games.

In the 18th century, probability theory was further developed by figures such as Jakob Bernoulli

and Pierre-Simon Laplace. Bernoulli's work, **Ars Conjectandi** (The Art of Conjecture), published posthumously in 1713, introduced the law of large numbers, a fundamental theorem in probability theory. This theorem states that as the number of trials of a random event increases, the average of the results will converge to the expected value.

Laplace, in his seminal work **Théorie Analytique des Probabilités** (Analytical Theory of Probability) published in 1812, provided a comprehensive framework for probability theory and applied it to various fields, including astronomy, statistics, and social sciences. Laplace's definition of probability as the ratio of favorable outcomes to the total number of equally likely outcomes became the foundation of classical probability theory.

The Expansion of Probability: 19th to Early 20th Centuries

The 19th century saw probability theory expanding beyond gambling and games of chance into broader applications. The development of statistics and the theory of errors in measurement contributed significantly to the evolution of probability. Karl Friedrich Gauss, in the early 19th century, introduced the concept of the normal distribution, also known as the Gaussian distribution, which became central to probability theory and statistics.

Another major development during this period was the concept of the random walk, introduced by Karl Pearson in 1905, and the notion of Brownian motion, studied by Albert Einstein in 1905. These concepts laid the foundation for the theory of stochastic processes, which would become a significant area of research in the 20th century.

The Formalization of Probability Theory: 20th Century

The 20th century marked a significant shift in the formalization and abstraction of probability theory. The Russian mathematician Andrey Kolmogorov played a crucial role in this process. In 1933, Kolmogorov published **Grundbegriffe der Wahrscheinlichkeitsrechnung** (Foundations of the Theory of Probability), which established a rigorous axiomatic foundation for probability theory. Kolmogorov's axioms provided a formal mathematical structure for probability, defining it as a measure on a sigma-algebra of events.

The mid-20th century also saw the application of probability theory in various scientific fields, including quantum mechanics, genetics, economics, and computer science. The development of Bayesian probability, named after the Reverend Thomas Bayes, who introduced Bayes' Theorem in the 18th century, gained significant attention. Bayesian probability provided a framework for updating probabilities based on new evidence, becoming widely used in statistics, decision theory, and machine learning.

Modern Applications and Ongoing Developments

Today, probability theory is a cornerstone of modern mathematics and is applied in a wide range of disciplines. From predicting stock market trends to understanding the behavior of subatomic particles, probability theory continues to evolve and find new applications. The development of computational methods and algorithms has further expanded the scope of probability theory, allowing for the analysis of complex systems and large datasets.

The history of probability is a testament to the power of mathematical abstraction and its ability to provide insights into the uncertain and unpredictable aspects of the world. As probability theory continues to evolve, it will undoubtedly play an increasingly important role in shaping our understanding of the world and the decisions we make.

Probability Concept : What is Probability?

Probability is a mathematical concept that measures the likelihood or chance of an event occurring. It is a fundamental tool for dealing with uncertainty and is widely used in various fields such as mathematics, statistics, finance, science, engineering, and everyday life. The concept of probability helps us quantify the uncertainty surrounding the outcomes of random processes and make informed predictions about future events.

Basic Definition

In its simplest form, probability is defined as the ratio of the number of favorable outcomes to the total number of possible outcomes in a given experiment. It is typically expressed as a fraction, decimal, or percentage, and always falls within the range of 0 to 1, where:

- 0 indicates that the event will not occur (impossible event).
- 1 indicates that the event will certainly occur (certain event).
- Any value between 0 and 1 represents the likelihood of the event occurring, with higher values indicating greater likelihood.

Life is full of uncertainties. We don't know the outcomes of a particular situation until it happens. Will it rain today? Will I pass the next math test? Will my favorite team win the toss? Will I get a promotion in next 6 months? All these questions are examples of uncertain situations we live in. Let us map them to few common terminology which we will use going forward.

- Experiment – are the uncertain situations, which could have multiple outcomes. Whether it rains on a daily basis is an experiment.
- Outcome is the result of a single trial. So, if it rains today, the outcome of today's trial from the experiment is “It rained”
- Event is one or more outcome from an experiment. “It rained” is one of the possible event for this experiment.
- Probability is a measure of how likely an event is. So, if it is 60% chance that it will rain tomorrow, the probability of Outcome “it rained” for tomorrow is 0.6.
- **Sample Space** - The *set* of all possible outcomes, e.g. we can roll a one, two, three, four, five

or six.

- **Mutually Exclusive** - Two events are *mutually exclusive* if both cannot occur at the same time, for example, we cannot roll a six and an odd number at the same time.
- **Independent** - Two events are *independent* if the occurrence of one does not affect the probability of the other occurring, e.g. rolling a 6 the first time does not affect the probability of rolling a 6 the next time.

Why do we need probability?

In an uncertain world, it can be of immense help to know and understand chances of various events. You can plan things accordingly. If it's likely to rain, I would carry my umbrella. If I am likely to have diabetes on the basis of my food habits, I would get myself tested. If my customer is unlikely to pay me a renewal premium without a reminder, I would remind him about it.

The mathematical expression for probability is:

$$\text{Probability (P)} = \frac{\text{Number of favorable outcomes}}{\text{Total number of possible outcomes}}$$

Example

Consider a simple example of rolling a six-sided die. The die has six faces, numbered from 1 to 6. If we want to calculate the probability of rolling a 4, we can determine the following:

- Total number of possible outcomes: There are 6 possible outcomes (1, 2, 3, 4, 5, and 6).
- Number of favorable outcomes: There is 1 favorable outcome (rolling a 4).

Thus, the probability of rolling a 4 is:

$$P(\text{rolling a 4}) = \frac{1}{6} \approx 0.167$$

This means there is approximately a 16.7% chance of rolling a 4.

Types of Probability

Probability can be interpreted and calculated in several ways depending on the context:

1. Classical Probability: This is the traditional approach where all outcomes are assumed to be equally likely. It is often used in situations involving games of chance, such as flipping a coin or rolling a die.

2. Empirical Probability: This type of probability is based on experimental data or observed frequencies. It is calculated by dividing the number of times an event occurs by the total number of trials. For example, if you flip a coin 100 times and it lands on heads 60 times, the empirical probability of getting heads is $60/100 = 0.6$.

3. Subjective Probability: Subjective probability is based on personal judgment, experience, or intuition rather than precise calculations. It is often used when there is no historical data or when the probability is difficult to measure. For instance, a doctor may estimate the probability of a patient recovering from an illness based on their experience with similar cases.

4. Bayesian Probability: Bayesian probability involves updating the probability of an event as new evidence or information becomes available. It is grounded in Bayes' Theorem, which provides a mathematical framework for revising probabilities considering new data.

Applications of Probability

Probability is a versatile tool with a wide range of applications:

- **Statistics :** Probability forms the basis for many statistical methods, including hypothesis testing, confidence intervals, and regression analysis. It helps statisticians make inferences about populations based on sample data.

- **Finance :** In finance, probability is used to model and assess risks, price financial instruments, and develop investment strategies. Techniques like Monte Carlo simulations rely on probabilistic models to predict market behavior.

- Science and Engineering : Probability is essential in scientific research for analyzing experimental data and modeling natural phenomena. Engineers use probability to assess system reliability, manage risks, and optimize processes.
- Everyday Life : People use probability, often unconsciously, in everyday decision-making. For example, when deciding whether to carry an umbrella based on a weather forecast, you are considering the probability of rain.

Types of Probability – Detailed Explanation with Practical Examples

Probability is a versatile concept in mathematics that can be interpreted and applied in various ways depending on the context. Understanding the different types of probability is crucial for effectively applying probabilistic reasoning in diverse fields such as statistics, finance, engineering, and everyday decision-making. This section delves into the primary types of probability, providing detailed explanations and practical examples for each.

1. Classical Probability

Definition: Classical probability, also known as "a priori" or "theoretical probability," is based on the assumption that all possible outcomes of an experiment are equally likely. It is calculated using the ratio of the number of favorable outcomes to the total number of possible outcomes.

Formula:

$$P(\text{Event}) = \frac{\text{Number of favorable outcomes}}{\text{Total number of possible outcomes}}$$

Key Characteristics:

- Relies on known and finite sample spaces.
- Assumes perfect randomness and equal likelihood of all outcomes.
- Often used in games of chance and combinatorial problems.

Practical Examples:

1. Rolling a Fair Die:

- **Experiment:** Rolling a standard six-sided die.
- **Sample Space:** {1, 2, 3, 4, 5, 6}
- **Event :** Rolling an even number (2, 4, 6).
- **Probability Calculation:**

$$P(\text{Even}) = \frac{3}{6} = 0.5$$

- **Interpretation:** There is a 50% chance of rolling an even number.

2. Flipping a Fair Coin:

- **Experiment:** Flipping a coin.
- **Sample Space:** {Heads, Tails}
- **Event:** Getting Heads.
- **Probability Calculation:**

$$P(\text{Heads}) = \frac{1}{2} = 0.5$$

- **Interpretation:** There is a 50% chance of the coin landing on Heads.

3. Drawing a Card from a Standard Deck:

- **Experiment:** Drawing one card from a standard 52-card deck.
- **Sample Space:** 52 unique cards.
- **Event:** Drawing an Ace.
- **Probability Calculation:**

$$P(\text{Ace}) = \frac{4}{52} = \frac{1}{13} \approx 0.077$$

- **Interpretation:** There is approximately a 7.7% chance of drawing an Ace.

4. Spinning a Fair Spinner:

- **Experiment:** Spinning a spinner divided into 8 equal sectors numbered 1 through 8.
- **Sample Space:** {1, 2, 3, 4, 5, 6, 7, 8}

- **Event:** Spinner lands on a number greater than 5 (i.e., 6, 7, 8).

- **Probability Calculation:**

$$P(>5) = \frac{3}{8} = 0.375$$

- **Interpretation:** There is a 37.5% chance of the spinner landing on a number greater than 5.

2. Empirical Probability

Definition:

Empirical probability, also known as "experimental" or "a posteriori" probability, is based on observed data or experiments rather than theoretical calculations. It is determined by conducting experiments or collecting data and calculating the relative frequency of the event occurring.

$$P(\text{Event}) = \frac{\text{Number of times the event occurs}}{\text{Total number of trials}}$$

Key Characteristics:

- Relies on actual experiments or historical data.
- Can accommodate complex and non-uniform sample spaces.
- Useful when theoretical probabilities are difficult to determine.

Practical Examples:

1. Weather Forecasting:

- **Experiment:** Recording daily occurrences of rain over a year.

- **Data:** Suppose it rained 120 days out of 365.

- **Probability Calculation:**

$$P(\text{Rain}) = \frac{120}{365} \approx 0.329$$

- **Interpretation:** Based on past data, there is approximately a 32.9% chance of rain on any given day.

2. Quality Control in Manufacturing:

- **Experiment:** Inspecting 1,000 units produced by a factory.
- **Data:** Found 50 defective units.
- **Probability Calculation:**

$$P(\text{Defective}) = \frac{50}{1000} = 0.05$$

- **Interpretation:** There is a 5% probability that a randomly selected unit is defective.

3. Sports Performance:

- **Experiment:** Tracking a basketball player's free-throw success rate over 200 attempts.
- **Data:** Successfully made 150 free throws.
- **Probability Calculation:**

$$P(\text{Success}) = \frac{150}{200} = 0.75$$

- **Interpretation:** The player has a 75% probability of making a free throw based on past performance.

4. Epidemiology Studies:

- **Experiment:** Observing the occurrence of a particular disease in a population over a decade.
- **Data:** 300 out of 10,000 individuals developed the disease.
- **Probability Calculation:**

$$P(\text{Disease}) = \frac{300}{10000} = 0.03$$

- **Interpretation:** There is a 3% probability of an individual in the population developing the disease based on historical data.

3. Subjective Probability

Definition:

Subjective probability is based on personal judgment, intuition, or experience rather than on formal calculations or empirical data. It reflects an individual's degree of belief in the occurrence of an event.

Key Characteristics:

- Influenced by personal opinions, biases, and experiences.
- Not necessarily quantifiable or consistent across different individuals.
- Useful in scenarios where objective data is unavailable or incomplete.

Practical Examples:

1. Investment Decisions:

- **Scenario:** An investor assesses the likelihood of a stock's price increasing based on their intuition and market experience.
- **Subjective Probability:** The investor believes there is a 70% chance the stock will rise, based on their analysis of market trends and company performance.
- **Interpretation:** The probability is a personal estimate and may differ from objective measures.

2. Medical Diagnoses:

- **Scenario:** A doctor estimates the probability that a patient has a specific disease based on symptoms and medical history.
- **Subjective Probability:** The doctor believes there is an 80% chance the patient has the disease, informed by their clinical experience.
- **Interpretation:** The probability reflects the doctor's judgment and may be adjusted with further tests.

3. Project Management:

- **Scenario:** A project manager assesses the likelihood of a project being completed on time.
- **Subjective Probability:** Based on team performance and project complexity, the manager

estimates a 60% probability of on-time completion.

- **Interpretation:** The estimate relies on the manager's experience and perception of project dynamics.

4. Personal Decision-Making:

- **Scenario:** Deciding whether to carry an umbrella based on the forecast and personal judgment.
- **Subjective Probability:** Believing there is a 40% chance of rain based on the weather forecast and personal observation of cloud patterns.
- **Interpretation:** The decision is influenced by both objective data and personal intuition.

4. Bayesian Probability

Definition:

Bayesian probability is an interpretation of probability that incorporates prior knowledge or beliefs and updates them as new evidence becomes available. It is grounded in Bayes' Theorem, which provides a mathematical framework for revising probabilities.

Formula (Bayes' Theorem):

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Where:

- $P(A|B)$ is the posterior probability of event A given event B.
- $P(B|A)$ is the likelihood of event B given event A.
- $P(A)$ is the prior probability of event A.
- $P(B)$ is the marginal probability of event B.

Key Characteristics:

- Combines prior beliefs with new evidence.
- Provides a dynamic approach to probability updating.
- Widely used in fields like statistics, machine learning, and decision theory.

Practical Examples:

1. Medical Testing:

- **Scenario:** Determining the probability that a patient has a disease given a positive test result.
- **Prior Probability ($P(\text{Disease})$):** 1% (prevalence of the disease).
- **Likelihood ($P(\text{Positive}|\text{Disease})$):** 99% (test accuracy).
- **Marginal Probability ($P(\text{Positive})$):** Calculated based on overall prevalence and test accuracy.

- **Bayesian Calculation:**

$$P(\text{Disease}|\text{Positive}) = \frac{0.99 \times 0.01}{P(\text{Positive})}$$

- **Interpretation:** Even with a positive test, the posterior probability may remain low due to the disease's low prevalence.

2. Spam Email Filtering:

- **Scenario:** Updating the probability that an email is spam based on the presence of certain keywords.
- **Prior Probability ($P(\text{Spam})$): 20%.**
- **Likelihood ($P(\text{Keyword}|\text{Spam})$): 80%.**
- **Likelihood ($P(\text{Keyword}|\text{Not Spam})$): 10%.**
- **Bayesian Calculation:**

$$P(\text{Spam}|\text{Keyword}) = \frac{0.8 \times 0.2}{P(\text{Keyword})}$$

- **Interpretation:** The posterior probability of an email being spam increases if it contains specific keywords.

3. Quality Assurance:

- **Scenario:** Estimating the probability that a product is defective after an initial test result.
- **Prior Probability ($P(\text{Defective})$): 5%.**
- **Likelihood ($P(\text{Passed Test}|\text{Defective})$): 20%.**
- **Likelihood ($P(\text{Passed Test}|\text{Not Defective})$): 95%.**
- **Bayesian Calculation:**

$$P(\text{Defective}|\text{Passed Test}) = \frac{0.2 \times 0.05}{P(\text{Passed Test})}$$

- **Interpretation:** Even if a product passes the test, there remains a non-zero probability that it is defective, adjusted based on test characteristics.

4. Legal Proceedings:

- **Scenario:** Assessing the probability of a defendant's guilt based on new evidence.
- **Prior Probability ($P(\text{Guilt})$):** 10% (based on initial evidence).
- **Likelihood ($P(\text{New Evidence}|\text{Guilt})$):** 90%.
- **Likelihood ($P(\text{New Evidence}|\text{Innocent})$):** 30%.
- **Bayesian Calculation:**

$$P(\text{Guilt}|\text{New Evidence}) = \frac{0.9 \times 0.1}{P(\text{New Evidence})}$$

- **Interpretation:** The new evidence significantly increases the probability of guilt compared to the prior probability.

5. Frequentist Probability (Additional Type)

Definition:

Frequentist probability defines the probability of an event as the limit of its relative frequency in many trials. It interprets probability strictly in terms of long-run frequencies of events.

Key Characteristics:

- Objective interpretation based on long-term frequencies.
- Does not incorporate prior beliefs or subjective opinions.
- Commonly used in classical statistical inference.

Practical Examples:

1. Coin Tossing Experiments:

- **Scenario:** Tossing a fair coin 10,000 times.
- **Frequency Calculation:** If heads appear 5,002 times, the frequentist probability of heads is:

$$P(\text{Heads}) = \frac{5002}{10000} = 0.5002$$

- **Interpretation:** The probability of getting heads is approximately 50%, aligning with the theoretical probability.

2. Manufacturing Defects:

- **Scenario:** Monitoring the defect rate in a production line over time.
- **Frequency Calculation:** Out of 50,000 units produced, 250 are defective.

$$P(\text{Defective}) = \frac{250}{50000} = 0.005$$

- **Interpretation:** The frequentist probability of producing a defective unit is 0.5%.

3. Elections Polling:

- **Scenario:** Conducting a poll with 1,000 respondents to estimate voter preference.

- **Frequency Calculation:** If 600 respondents favor Candidate A, the frequentist probability is:

$$P(\text{Candidate A}) = \frac{600}{1000} = 0.6$$

- **Interpretation:** Based on the poll, there is a 60% probability that a randomly selected voter favors Candidate A.

4. Quality Control Testing:

- **Scenario:** Testing batches of products to determine the failure rate.

- **Frequency Calculation:** In 200 tested units, 4 fail.

$$P(\text{Failure}) = \frac{4}{200} = 0.02$$

- **Interpretation:** The frequentist probability of a unit failing is 2%.

Understanding the different types of probability—Classical, Empirical, Subjective, Bayesian, and Frequentist—is essential for accurately modeling and interpreting uncertainty in various contexts. Each type offers unique perspectives and tools for assessing likelihoods, making informed decisions, and conducting rigorous analyses. By applying the appropriate type of probability based on the nature of the problem and the available information, one can effectively navigate and quantify uncertainty in both theoretical and practical scenarios.

Example

In Newcastle, 70% of small businesses use the internet to advertise new products; 50% of small businesses use flyers to advertise new products and a quarter of small businesses use *both* flyers *and* the internet.

(A) What is the probability that a randomly chosen small business in Newcastle uses *either* flyers *or* the internet to advertise new products?

(B) What is the proportion of small businesses in Newcastle that use neither the internet *nor* flyers to advertise new products?

Solution (A)

- Let F denote the event that a business advertises new products using flyers and I denote the event that a business uses the internet to advertise new products.

- We wish to find $P(F \text{ or } I)$. Using the Addition Law, we have:

$$\begin{aligned}
 P(F \text{ or } I) &= P(F) + P(I) - P(F \text{ and } I) \\
 &= \frac{7}{10} + \frac{1}{2} - \frac{1}{4} \\
 &= \frac{19}{20} = 0.95.
 \end{aligned}$$

- There is a 95% probability that a randomly chosen business in Newcastle uses either flyers or the internet to advertise new products

Tree Diagrams

Tree Diagrams can be used to help us to visualize and calculate complex probabilities. When drawing a tree diagram we begin with a dot. From this dot lines (“branches”) are then drawn, extending from the right of the first dot, to represent all possible outcomes for the given situation. The probabilities of each of these outcomes is written just above the corresponding line.

To calculate the probability that two events *both* happen, we draw another “branch” extending from the “branch” corresponding to one of these events to represent the second event occurring after the first. Above this line we write the probability (or conditional probability for events which are not independent) of the second event occurring after the first. Multiplying these probabilities “along the branches” gives the required probability.

To calculate the probability that one or both of two *independent* events occurs we add the probabilities of the two events “down the columns”.

Example

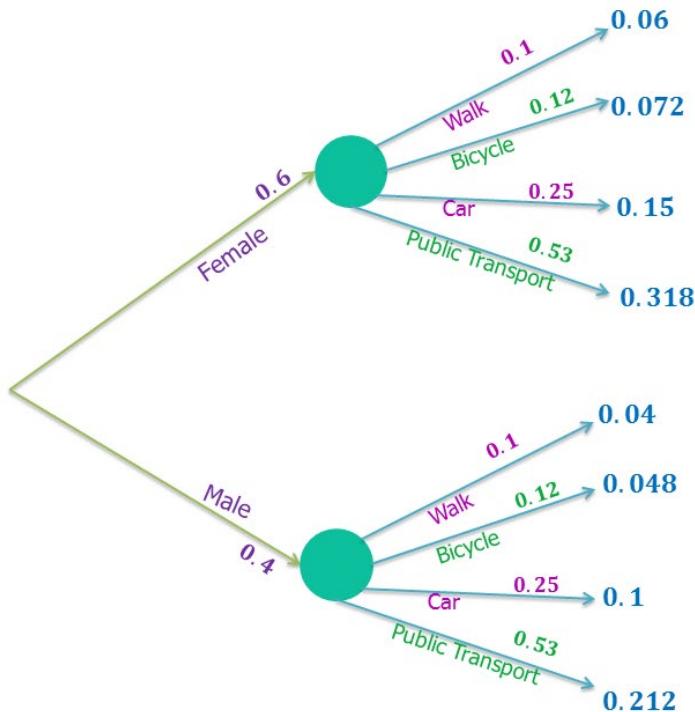
60% of employees at a department store in Newcastle are women. Government research into methods of commuting to city jobs in the North East has shown on average that:

- **12% of people cycle into work.**
- **A quarter of the people drive.**
- **10% of people walk.**
- **And the rest use public transport.**

What is the probability that a randomly selected employee of the department store in Newcastle commutes using public transport and is male? Now calculate the probability

that a randomly selected employee is female and drives into work.

Solution We can use a tree diagram to present all of the information given to us and calculate the required probability.



The probabilities in blue are calculated using the multiplication law. So, the probability the employee is female and drives is $0.6 \times 0.25 = 0.15$

Tip: To make sure all your calculations are correct, you can check to see that your final probabilities (the blues ones) add up to 11. This must be the case because at least one of all of the possible events **must** (is certain to) occur.

Decision Trees - *Decision trees* are very similar to the probability tree diagrams but are used specifically to calculate expected monetary values.

Example 4

The manager of a small business has the opportunity to buy a fixed quantity of a new product and offer it for sale for a limited time.

There will be a fixed cost of £100,000 to buy the product and offer it for sale. The amount of the product that the manager would be able to sell is not certain but market research has suggested that:

- The probability that sales would be “poor” is 0.25. Selling this quantity would raise an income of £75,000.
- The probability that sales would be “medium” is 0.6. Selling this quantity would raise an income of £110,000.
- The probability that sales would be “good” is 0.15. Selling this quantity would raise an income of £145,000.

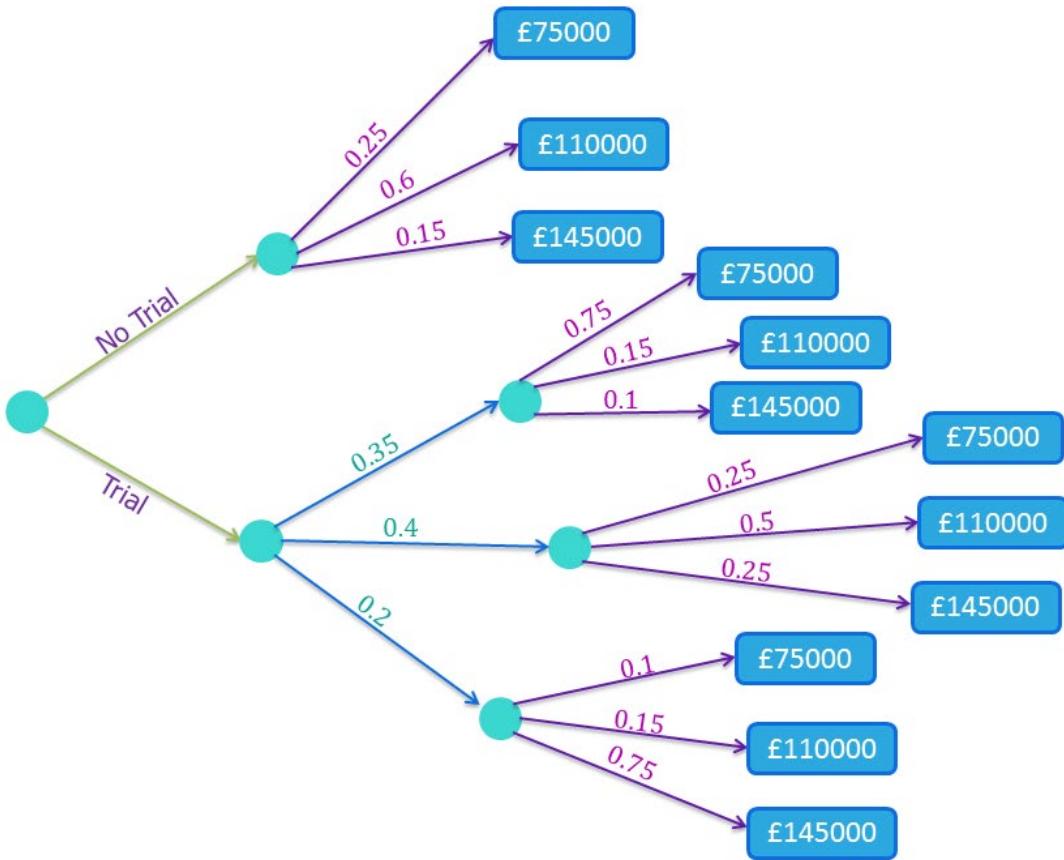
The product can be sold for a trial period before a final decision is made and it costs £18,000 to run the trial. The results of the trial will be “poor” with probability 0.35, “medium” with probability 0.4 or “good” with probability 0.25. Knowing the outcome of the trial changes the probabilities for the main sales project:

Trial Outcome	Main sales probabilities		
	Poor	Medium	Good
Poor	0.75	0.15	0.1
Medium	0.25	0.5	0.25
Good	0.1	0.15	0.75

The manager will make decisions based on expected monetary value.

- (A) Draw a decision tree for this problem.**
- (B) What is the EMV of a decision to go ahead with the product without a trial?**
- (C) Complete the solution of the decision problem and determine the optimal course of action for the company.**

Solution (A)



The values in the blue boxes are the final incomes from buying the new product when sales are “poor”, “medium” or “good” (top to bottom).

Solution (B)

To calculate the expected monetary value, we need to utilize the formula in the pink box above.

- For No Trial:

$$EMV = 0.25 \times £75,000 + 0.6 \times £110,000 + 0.15 \times £145,000 = £106,500.$$

The manager has an expected income of £106,500 from selling the new product without a trial.

Solution (C)

To solve the decision problem it is best to first calculate the separate EMVs for when a trial is run and when a trial is not run. We must then compare the EMVs for each option (trial or no trial) and choose the option with the highest EMV. This is the optimal course of action for the company.

- When a trial is carried out and has a poor result:

$$EMV = 0.75 \times £75,000 + 0.15 \times £110,000 + 0.1 \times £145,000 = £87,250.$$

- When a trial is carried out and has a medium result:

$$EMV = 0.25 \times £75,000 + 0.5 \times £110,000 + 0.25 \times £145,000 = £110,000.$$

- When a trial is carried out and has a good result:

$$EMV = 0.1 \times £75,000 + 0.15 \times £110,000 + 0.75 \times £145,000 = £132,750.$$

Now to calculate the overall EMV we multiply each of these by their associated probabilities:

$$\begin{aligned} EMV &= P(\text{Poor result}) \times £87,250 + P(\text{Medium result}) \times £110,000 + P(\text{Good result}) \times £132,750 \\ &= £107,725. \end{aligned}$$

We now need to calculate the expected profit (or loss) the business would make from each option (trial or no trial).

- No Trial:

$$\begin{aligned} \{\text{Expected Profit}\} &= \{\text{EMV}\} \{ \text{for no trial} \} - \{\text{Cost of new product}\} \\ &= £106,500 - £100,000 \\ &= £6,500. \end{aligned}$$

So if the manager goes ahead with the product without a trial, the expected profit is £6,500.

- Trial:

$$\begin{aligned} \{\text{Expected Profit}\} &= \{\text{EMV}\} \{ \text{for trial} \} - \{\text{Cost of new product}\} - \{\text{Cost of trial}\} \\ &= £107,725 - £100,000 - £18,000 \\ &= -£10,275. \end{aligned}$$

With the trial, there will be an expected loss of £10,275.

From these results we can see that optimal course of action for the company is to sell the new product but without the trial period as this yields a higher EMV. It is important to note that although the *expected* monetary value is higher when the manager chooses not to run the trial, the realized profit or loss may be or may not be better than it would have been if a trial had been carried out.

Permutation
and
Combination
Concept

What is a Permutation?

A permutation is a mathematical technique that determines the number of possible arrangements in a set when the order of the arrangements matters. Common mathematical problems involve choosing only several items from a set of items in a certain order. Permutations in probability theory and other branches of mathematics refer to sequences of outcomes where the order matters. For example, 9-6-8-4 is a permutation of a four-digit PIN because the order of numbers is crucial. When calculating probabilities, it's frequently necessary to calculate the number of possible permutations to determine an event's probability. A permutation is an arrangement of all or part of a set of objects, with consideration of the order of arrangement. The concept of permutation is used when the arrangement order matters. For example, the permutation of letters "ABC" is different from "CAB" because the order of letters differs.



Permutations are frequently confused with another mathematical technique called combinations. However, in combinations, the order of the chosen items does not influence the selection. In other words, the arrangements ab and be in permutations are considered different arrangements, while in combinations, these arrangements are equal.

Formula for Calculating Permutations

The general permutation formula is expressed in the following way:

$$P(n, k) = \frac{n!}{(n - k)!}$$

Where:

- **n** – the total number of elements in a set
- **k** – the number of selected elements arranged in a specific order
- **!** – factorial

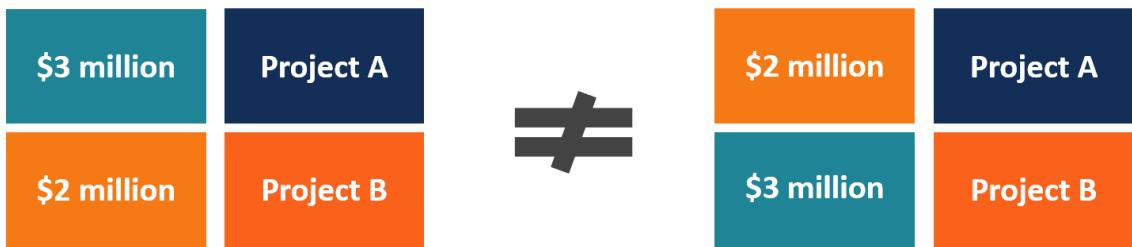
Factorial (noted as “!”) is the product of all positive integers less than or equal to the number preceding the factorial sign. For example, $3! = 1 \times 2 \times 3 = 6$.

The formula above is used in situations when we want to select only several elements from a set of elements and arrange the selected elements in a special order.

Example of a Permutation

You are a partner in a private equity firm. You want to invest \$5 million in two projects. Instead of equal allocation, you decided to invest \$3 million in the most promising project and \$2 million in the less promising project. Your analysts shortlisted six projects for potential investment. How many possible arrangements are available for your investment decision?

The example above is a permutation problem. Since the allocation of the money for the two projects is not equal, the selection order matters in this problem. For example, consider the following arrangement: invest \$3 million in Project A and \$2 million in Project B vs. invest \$2 million in Project A and \$3 million in Project B. The options are not equal to each other. Therefore, we must use the formula above to determine the number of available arrangements:



Using the formula above, we can determine the number of available arrangements:

$$C(6,2) = \frac{6!}{(6-2)!} = \frac{6!}{4!} = 30$$

Therefore, you can get 30 possible investment arrangements based on the six projects shortlisted by your analysts.

Permutations of Distinct Objects

Permutations of distinct objects refer to the various ways in which a set of distinct (unique) objects can be arranged or ordered. In combinatorial mathematics, a permutation is an arrangement of all the elements in a specific sequence or order. When the objects are distinct, each arrangement is considered unique because the order of objects matters.

Formula for Permutations of Distinct Objects

For n distinct objects, the number of possible permutations is given by $n!$ (n factorial). The factorial of a number n is the product of all positive integers from 1 to n .

Mathematically, $n!$ is expressed as: $n! = n \times (n-1) \times (n-2) \times \dots \times 2 \times 1$

When all the objects in the set are distinct, the permutation formula is straightforward.

Example 1: Consider the set {A, B, C}. The number of ways to arrange all three letters is:

$$P(3,3) = \frac{3!}{(3-3)!} = \frac{3!}{0!} = 6$$

The possible permutations are: ABC, ACB, BAC, BCA, CAB, and CBA.

Example 2: For a set of 5 different books, the number of ways to arrange 2 books on a shelf is:

$$P(5,2) = \frac{5!}{(5-2)!} = \frac{120}{6} = 20$$

Permutations with Repetition

Permutations with repetition involve arranging items where some items may be repeated, and we're interested in the number of different sequences that can be formed considering the repetitions.

For example, on a pizza, you might have a combination of three toppings: pepperoni, ham, and mushroom. The order doesn't matter. For example, using letters for the toppings, you can have PHM, PMH, HPM, and so on. It doesn't matter for the person who eats the pizza because you have the same combination of three toppings. In other words, the order of these three letters does not matter and they form one combination.

However, imagine we're using those letters for a weak password. In this case, the order is crucial, making them permutations. PHM, PMH, HPM, etc., are distinct permutations. If the password is PHM, entering HPM will not work. When you have at least two permutations, the number of permutations is greater than the number of combinations. Learn more about the [differences between permutation vs combination](#).



This type of lock should be known as a permutation lock because the order of digits matters!

Concept of Permutations with Repetition

When we allow for repetitions, the problem often involves arranging items where the order is still important, but items are not necessarily unique. The key difference from permutations of distinct objects is that here, we may have multiple arrangements involving the same items.

When the outcomes in a permutation can repeat, [statisticians](#) refer to it as permutations with

repetition. For example, in a four-digit PIN, you can repeat values, such as 1-1-1-1. Analysts also call this permutations with replacement.

To calculate the number of permutations, take the number of possibilities for each event and then multiply that number by itself X times, where X equals the number of events in the sequence. For example, with four-digit PINs, each digit can range from 0 to 9, giving us 10 possibilities for each digit. We have four digits. Consequently, the number of permutations with repetition for these PINs = $10 * 10 * 10 * 10 = 10,000$.

Imagine that a class with 15 children can choose one cookie from five types of cookies: Gingerbread, Sugar, Chocolate Chip, Mint, and Peanut Butter. There are enough cookies that they are free to choose one of any type. How many possible permutations of cookies are there?



In this example,

- o $n = 5$ because there are five possible cookie choices.
- o $r = 15$ because there are 15 students in the class, making it the size of the permutation.

Consequently, there are $5^{15} = 30,517,578,125$ permutations with repetition. That's over 30 billion permutations! If you were to make random guesses for the cookie choice of all 15 children, you'd have a probability of $1/30,517,578,125$ of correctly guessing the selections for the entire class! That assumes you don't have insider knowledge about each child's cookie preference! I think you'd have better luck in a lottery!

Formula for Permutations with Repetition

If we have n types of items, and we are arranging r items (where repetition of items is allowed), the number of possible permutations is given by:

$$n^r$$

where:

- n is the number of distinct types of items.
- r is the number of positions to fill.

Examples

Example 1: 3 Types of Items, 2 Positions

Suppose we have 3 types of items (say, A, B, and C) and want to arrange 2 items.

To find the number of permutations:

Apply the formula: $n^r = 3^2 = 9$

List the permutations:

- AA
- AB
- AC
- BA
- BB
- BC
- CA
- CB
- CC

There are 9 unique ways to arrange 2 items where each position can be filled with any of the 3 types of items.

Example 2: 4 Types of Items, 3 Positions

Consider 4 distinct items (say, 1, 2, 3, and 4), and we want to arrange 3 items.

Apply the formula: $n^r = 4^3 = 64$

List a few permutations (for illustration):

- 111
- 112
- 113
- 121
- 122
- 123
- (and so on...)

There are 64 possible arrangements of 3 items with 4 possible choices for each position.

Permutations with repetition involve:

- **Permutations with Repetition (Formula):** n^r , where n is the number of distinct items, and r is the number of positions to be filled.
- **Order Matters:** The arrangement of items is significant, so permutations consider the sequence of items.
- **Repetition Allowed:** The same item can appear in multiple positions.

This concept is widely applicable in scenarios like password generation, where each position can be filled by any of the allowed characters, and in scenarios where choices are repeated multiple times.

Permutations without Repetition

When the outcomes cannot repeat, statisticians call them permutations without repetition. This situation frequently occurs when you're working with unique physical objects that can occur only once in a permutation. Imagine you have 10 different books and want to calculate how many possible ways you can arrange them on a bookshelf. After you place the first book, the second book must be a different book. Consequently, this is an example of permutations without repetition. Analysts also call these permutations without replacement.



For the first book, you have 10 books from which to choose. For the second book, you have nine. There are eight options for the third book, and so on. Like before, this process involves multiplying the number of possible outcomes together. However, we must reduce the number of outcomes for each subsequent event.

Mathematically, we'd calculate the permutations for the book example using the following method:

$$10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1 = 3,628,800$$

There are 3,628,800 permutations for ordering 10 books on a shelf without repeating books.

Whew! I bet you didn't realize there were so many possibilities with 10 books. I'll stick to alphabetical order!

Using Factorials for Permutations

When you multiply all numbers from 1 to n, it's a factorial. In the book example, we multiplied all numbers from 1 to 10. Instead of using the long string of multiplication, you can write it as $10!$ and read it as 10 factorial.

In general, $n!$ equals the product of all numbers up to n. For example, $3! = 3 * 2 * 1 = 6$. The exception is $0! = 1$, which simplifies equations.

Factorials are crucial concepts for permutations without replication. The number of permutations for n unique objects is $n!$. This number snowballs as the number of items increases, as the table below shows.

Factorial	Permutations without repetition
1	1
2	2
3	6
4	24
5	120
6	720
7	5,040
8	40,320
9	362,880
10	3,628,800

Partial Permutations without Repetition

In some cases, you want to consider only a portion of the possible permutations. In the bookshelf example, we wanted to know the total number for 10 books. But what if we could fit only five of the 10 books on the shelf? How many permutations of five books are possible using our 10 books?

Use the following formula to calculate the number of arrangements of r items from n objects. There are several standard methods that statisticians use to notate permutations without repetition, which I show below with the formula.

$$P(n,r) = {}^n P_r = {}^n P_r = \frac{n!}{(n-r)!}$$

Where:

- n = the number of unique items. For instance, $n = 10$ for the book example because there are 10 books.
- r = the size of the permutation. For example, $r = 5$ for the five books we want to place on the shelf.

This equation works both for the complete and partial sets of permutations without repetitions, depending on the values you enter in the equation. For complete sets, $n = r$. Additionally, r cannot be greater than n because there are no repetitions.

For the book example, we have 10 books, but we can put only five on the shelf. The first book still has 10 options. However, for placing the second book, we have only nine options because we already placed one. We have eight options for the third book and so on until we place the fifth book. Mathematically, we'd write this as the following for the five books:

$$10 * 9 * 8 * 7 * 6 = 30,240$$

There are 30,240 permutations for placing five books out of our 10 books on a shelf.

Using the equation to calculate the number of permutations

Now, we'll use the formula to calculate this example. Again, we'll use $n=10$ and $r=5$.

$$\frac{10!}{5!} = \frac{10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1}{5 * 4 * 3 * 2 * 1} = 30,240$$

Notice how the $5!$ cancels itself out in the fraction? That leaves us with the $10 * 9 * 8 * 7 * 6$ that we had before.

Here's how the equation works. The numerator calculates the complete number of permutations for all the unique items. The denominator cancels out the permutations in which we're not interested. For the book example, the denominator cancels out permutations with more than five books.

Using one form of the notation, we'd write this problem as $P(10, 5) = 30,240$.

Worked Example of Using Permutations to Calculate Probabilities

When you're given a probability problem that uses permutations, you need to follow these steps to solve the problem.

1. Set up a ratio to determine the probability.
2. Determine whether the numerator and denominator require combinations, permutations, or a mix? For this post, we'll stick with permutations.
3. Are these permutations with repetitions, without, or a mix?
4. Both types of repetition require you to identify the n and r to enter into the equations.

Problem: What is the probability that a four-digit PIN does not have repeated digits?

This question builds on several of the examples in this post.

Let's set up our ratio for the probability. In this example, we can use the following ratio for the events of interests and the total number of events.

$$\text{Probability} = \frac{\text{Permutations of interest}}{\text{Total number of permutations}}$$

Numerator

Let's tackle the numerator. We need to find the number of four-digit PINs that do not have repeating digits. That's a permutation because order matters, and it's without replication because we can't have repeats. Let's identify the n and r. We'll use n=10 because 10 digits are available for the first item and r=4 because we're discussing four-digit PINs.

Let's enter that into the equation for permutations without repetition to calculate the numerator:

$${}_{10}P_4 = \frac{n!}{(n-r)!} = \frac{10!}{6!} = \frac{10 * 9 * 8 * 7 * 6 * 5 * 4 * 3 * 2 * 1}{6 * 5 * 4 * 3 * 2 * 1} = 5,040$$

Denominator

For the denominator, we need to calculate all possible permutations for four-digit PINs with repeats. We need to enter our n and r into the equation for permutations with repeats.

$$n^r = 10^4 = 10,000$$

Consequently, the probability of a four-digit PIN with no repeating digits equals the following:

$$\frac{{}_{10}P_4}{10^4} = \frac{5,040}{10,000} = 0.504$$

Just over half of all four-digit PINs have repeated values.

The birthday problem is a classic probability problem. What's the smallest size group that has a greater than 50% chance of people sharing a birthday? Solving this problem uses similar methods. Read my post about [Solving the Birthday Problem](#) to find out!

Circular Permutation

Circular permutations refer to arrangements of objects in a circle where the order of the objects is important, but rotations of the arrangement are considered identical. This concept is useful in problems where the arrangement is cyclic and you want to count distinct configurations that are rotationally unique.

Formula for Circular Permutations

For n distinct objects arranged in a circle, the number of distinct circular permutations is given by:

$$\frac{(n-1)!}{1} = (n-1)!$$

Here's why: In a linear permutation, n objects can be arranged in $n!$ ways. However, in a circular arrangement, each unique arrangement can be rotated n different ways (all considered the same circular permutation). Therefore, to find the number of distinct circular permutations, you divide the total number of linear permutations by n .

Examples

Example 1: 3 Distinct Objects

Consider 3 distinct objects: A, B, and C.

1. **Calculate the number of circular permutations:** $(n-1)! = (3-1)! = 2! = 2$
2. **List the permutations:**

- ABC
- BCA
- CAB

Here, rotations of ABC (like BCA and CAB) are considered the same circular permutation. Thus, there are 2 unique circular permutations.

Example 2: 4 Distinct Objects

Consider 4 distinct objects: 1, 2, 3, and 4.

1. **Calculate the number of circular permutations:** $(n-1)! = (4-1)! = 3! = 6$
2. **List the permutations:**

- 1234
- 2341
- 3412
- 4123
- 3412
- 4123

These are the unique circular permutations, where each arrangement is considered identical under rotation.

Example 3: 5 Distinct Objects

Consider 5 distinct objects: A, B, C, D, and E.

1. **Calculate the number of circular permutations:** $(n-1)! = (5-1)! = 4! = 24$
2. **List the permutations:**

- ABCDE

- BCDEA
- CDEAB
- DEABC
- EABCD
- (and so forth)

Each permutation is distinct in a circular arrangement, and there are 24 unique ways to arrange 5 objects in a circle.

Example 4: 6 Distinct Objects

Consider 6 distinct objects: 1, 2, 3, 4, 5, and 6.

1. **Calculate the number of circular permutations:** $(n-1)! = (6-1)! = 5! = 120$
2. **List the permutations:**

- 123456
- 234561
- 345612
- 456123
- 561234
- 612345
- (and so on)

There are 120 unique circular permutations for 6 distinct objects.

Example 5: 7 Distinct Objects

Consider 7 distinct objects: A, B, C, D, E, F, and G.

1. **Calculate the number of circular permutations:** $(n-1)! = (7-1)! = 6! = 720$
2. **List the permutations:**

- ABCDEFG
- BCDEFGH
- CDEFGAB
- DEFGABC
- EFGABCD
- FGABCDE
- GABCDEF
- (and so forth)

With 7 distinct objects, there are 720 unique circular permutations.

- **Circular Permutations Formula:** For n distinct objects, the number of unique circular permutations is $(n-1)!$.
- **Consider Rotations as Identical:** Each permutation can be rotated n ways, so you only count one arrangement per unique rotation.

Circular permutations are particularly useful in scenarios where the arrangement is cyclic, such as in circular tables, clock arrangements, or certain scheduling problems. If you have more questions or need further examples, feel free to ask!

Basics of Combinations

Definition of Combination

A **combination** is a selection of all or part of a set of objects without regard to the order of arrangement. The concept of combination is used when the order of selection does not matter.

Formula for Combination

The number of combinations of n distinct objects taken r at a time is given by:

$$C(n, r) = \frac{n!}{r!(n - r)!}$$

where:

- $N!$ is the factorial of n
- $r!$ is the factorial of r .

Combinations of Distinct Objects

When all the objects in the set are distinct, and the order does not matter, the combination formula is used.

Example 1: Consider the set {A, B, C, D}. The number of ways to select 2 letters out of 4 is:

$$C(4, 2) = \frac{4!}{2!(4 - 2)!} = \frac{24}{4 \times 2} = 6$$

The possible combinations are: AB, AC, AD, BC, BD, and CD.

Example 2: In a lottery where 6 numbers are chosen out of 49, the number of possible combinations is:

$$C(49, 6) = \frac{49!}{6!(49 - 6)!} = 13,983,816$$

Combinations with Repetition

When repetition is allowed in combinations, the formula changes slightly. The number of combinations of n objects taken r at a time with repetition is given by:

$$C(n + r - 1, r) = \frac{(n + r - 1)!}{r!(n - 1)!}$$

Example 3: If you have three types of fruits (apple, banana, cherry), and you want to select 2 fruits with repetition, the number of possible combinations is:

$$C(3 + 2 - 1, 2) = \frac{4!}{2!2!} = 6$$

The possible combinations are AA, AB, AC, BB, BC, and CC.

Practical Applications of Permutations and Combinations

Permutations in Real-Life Scenarios

Example 4: Password Generation Consider a scenario where you need to create a password using 4 letters (where repetition is not allowed) from the alphabet set of 26 distinct letters. The total number of permutations possible is:

$$P(26, 4) = \frac{26!}{(26 - 4)!} = 26 \times 25 \times 24 \times 23 = 358,800$$

This indicates there are 358,800 possible unique passwords.

Example 5: Arranging People Suppose 5 people need to be arranged in a row for a group photo. The total number of permutations possible is:

$$P(5, 5) = 5! = 120$$

Thus, there are 120 different ways to arrange 5 people in a line.

Combinations in Real-Life Scenarios

Example 6: Forming Committees Imagine you need to form a committee of 3 members from a group of 10 people. The number of possible combinations is:

$$C(10, 3) = \frac{10!}{3!(10 - 3)!} = \frac{10 \times 9 \times 8}{3 \times 2 \times 1} = 120$$

This means there are 120 different ways to form a committee of 3 members from 10 people.

Example 7: Selecting Ingredients In a recipe, you can choose 4 ingredients out of 8 available. The number of combinations is:

$$C(8, 4) = \frac{8!}{4!4!} = 70$$

So, there are 70 possible ways to choose 4 ingredients from 8.

Advanced Permutation and Combination Concepts

Permutations of Non-Distinct Objects

When some objects in a set are identical, the formula for permutations needs to be adjusted. The number of distinct permutations of n objects where there are n_1, n_2, \dots, n_k objects of the same type is given by:

$$P = \frac{n!}{n_1! \times n_2! \times \dots \times n_k!}$$

Example 8: Arranging Letters Consider the word "BALLOON". The total number of distinct permutations of these letters is:

$$P = \frac{7!}{1! \times 2! \times 2! \times 1! \times 1!} = \frac{5040}{4} = 1260$$

So, there are 1,260 unique ways to arrange the letters in "BALLOON".

Practical Examples of Permutations

Example 1: Arranging Books on a Shelf Suppose you have 5 different books and you want to arrange them on a shelf. The total number of permutations is:

$$P(5, 5) = 5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$$

So, there are 120 different ways to arrange these 5 books.

Example 2: Forming a 3-Digit Number Consider forming a 3-digit number using the digits 1, 2, 3, 4, and 5, without repetition. The number of possible permutations is:

$$P(5, 3) = \frac{5!}{(5 - 3)!} = \frac{120}{2} = 60$$

This means you can form 60 different 3-digit numbers.

Example 3: Seating Arrangements If you have 4 people and want to seat them in a row, the number of permutations is:

$$P(4, 4) = 4! = 24$$

Thus, there are 24 possible ways to arrange these 4 people in a row.

Example 4: Selecting and Arranging Employees Suppose you need to select 3 employees from a group of 6 and assign them different positions. The number of permutations is:

$$P(6, 3) = \frac{6!}{(6 - 3)!} = \frac{720}{6} = 120$$

There are 120 different ways to select and assign these 3 employees to the positions.

Example 5: Creating a Password If you need to create a 4-character password using the letters A, B, C, and D, with repetition not allowed, the total number of permutations is:

$$P(4, 4) = 4! = 24$$

There are 24 possible unique passwords.

Example 6: Lottery Number Arrangement Imagine a lottery where you must choose 3 numbers from a set of 5 (1, 2, 3, 4, 5), and the order matters. The number of possible permutations is:

$$P(5, 3) = 60$$

So, there are 60 different possible outcomes in this lottery.

Example 7: Arranging Letters in a Word Consider the word "TRAIN". The number of ways to arrange the letters is:

$$P(5, 5) = 5! = 120$$

So, there are 120 different ways to arrange the letters in "TRAIN".

Example 8: Organizing a Race Suppose 7 runners are competing in a race, and you want to know how many different ways the first 3 places can be awarded. The number of permutations is:

$$P(7, 3) = \frac{7!}{(7 - 3)!} = 210$$

This indicates there are 210 possible ways to award the top 3 positions.

Example 9: Forming Committees If you need to select a president, vice-president, and treasurer from a group of 8 members, the number of permutations is:

$$P(8, 3) = \frac{8!}{(8 - 3)!} = 336$$

There are 336 different ways to assign these positions.

Example 10: Deck of Cards If you want to arrange 5 cards from a standard deck of 52, without repetition, the number of permutations is:

$$P(52, 5) = \frac{52!}{(52 - 5)!} = 311,875,200$$

There are 311,875,200 different ways to arrange 5 cards.

Example 11: Permutations with Repetition If you want to create a 2-letter code from the letters A, B, C, D, allowing repetition, the number of permutations is:

$$P(4, 2) = 4^2 = 16$$

There are 16 possible codes.

Addition Theorem in Probability

The **Addition Theorem** is used to find the probability of the occurrence of at least one of two events. There are two cases to consider:

1. **Mutually Exclusive Events (Disjoint Events):** Events that cannot happen simultaneously. For example, rolling a die and getting either a 3 or a 5.
2. **Non-Mutually Exclusive Events:** Events that can occur together. For example, drawing a card from a deck that is both a heart and an ace.

1. Addition Theorem for Mutually Exclusive Events

For mutually exclusive events, the probability of either event A or event B occurring is the sum of their individual probabilities. Formula

$$P(A \text{ or } B) = P(A) + P(B)$$

Example: Consider rolling a fair six-sided die. What is the probability of rolling a 2 or a 4?

- Probability of rolling a 2, $P(A) = \frac{1}{6}$
- Probability of rolling a 4, $P(B) = \frac{1}{6}$

Since these two events are mutually exclusive:

$$P(A \text{ or } B) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6} = \frac{1}{3}$$

2. Addition Theorem for Non-Mutually Exclusive Events

For non-mutually exclusive events, the probability of either event A or event B occurring is the sum of their individual probabilities, minus the probability of both events occurring together (to avoid double-counting).

$$P(A \text{ or } B) = P(A) + P(B) - P(A \text{ and } B)$$

Example: Consider drawing a card from a standard deck of 52 cards. What is the probability of drawing either a king or a heart?

- Probability of drawing a king, $P(A) = \frac{4}{52}$
- Probability of drawing a heart, $P(B) = \frac{13}{52}$
- Probability of drawing the king of hearts (which is both a king and a heart), $P(A \text{ and } B) = \frac{1}{52}$

Using the formula:

$$P(A \text{ or } B) = \frac{4}{52} + \frac{13}{52} - \frac{1}{52} = \frac{16}{52} = \frac{4}{13}$$

Multiplication Theorem in Probability

The **Multiplication Theorem** is used to find the probability of the occurrence of two or more events together. Like the Addition Theorem, the Multiplication Theorem also has two cases:

1. **Independent Events:** Events where the occurrence of one does not affect the occurrence of the other. For example, tossing two coins.
2. **Dependent Events:** Events where the occurrence of one affects the occurrence of the other. For example, drawing cards from a deck without replacement.

1. Multiplication Theorem for Independent Events

For independent events, the probability of both events A and B occurring is the product of their individual probabilities.

Formula:

$$P(A \text{ and } B) = P(A) \times P(B)$$

Example: Consider flipping a coin and rolling a six-sided die. What is the probability of getting heads on the coin and a 3 on the die?

- Probability of getting heads, $P(A) = \frac{1}{2}$
- Probability of rolling a 3, $P(B) = \frac{1}{6}$

Since these two events are independent:

$$P(A \text{ and } B) = \frac{1}{2} \times \frac{1}{6} = \frac{1}{12}$$

2. Multiplication Theorem for Dependent Events

For dependent events, the probability of both events A and B occurring is the product of the probability of A and the probability of B given that A has occurred.

Formula:

$$P(A \text{ and } B) = P(A) \times P(B|A)$$

Example: Consider drawing two cards from a deck without replacement. What is the probability of drawing an ace first and a king second?

- Probability of drawing an ace first, $P(A) = \frac{4}{52}$
- After drawing the ace, there are 51 cards left, so the probability of drawing a king next, $P(B|A) = \frac{4}{51}$

Using the formula:

$$P(A \text{ and } B) = \frac{4}{52} \times \frac{4}{51} = \frac{16}{2652} = \frac{4}{663}$$

Here are examples for both the Addition and Multiplication Theorems in probability:

Example 1: Addition Theorem

Scenario: You are attending a party where a game involves drawing a card from a standard deck of 52 cards. You win a prize if you draw either a spade or a face card (jack, queen, or king).

Problem: What is the probability of winning the prize?

Solution:

- There are 13 spades in a deck: ($P(\{\text{Spade}\}) = 13/52$)
- There are 12 face cards (3 per suit): ($P(\{\text{Face Card}\}) = 12/52$)
- There are 3 face cards that are also spades: ($P(\{\text{Spade and Face Card}\}) = 3/52$)

Using the Addition Theorem for Non-Mutually Exclusive Events:

$$P(\text{Spade or Face Card}) = P(\text{Spade}) + P(\text{Face Card}) - P(\text{Spade and Face Card})$$

$$P(\text{Spade or Face Card}) = \frac{13}{52} + \frac{12}{52} - \frac{3}{52} = \frac{22}{52} = \frac{11}{26}$$

So, the probability of winning the prize is $\frac{11}{26}$.

Example 2: Multiplication Theorem

Scenario: You are playing a board game where you need to roll two six-sided dice. To win the game, you need to roll a 4 on the first die and a 5 on the second die.

Problem: What is the probability of rolling a 4 on the first die and a 5 on the second die?

Solution:

- Probability of rolling a 4 on the first die: $P(\{4 \text{ on first die}\}) = 1/6$
 - Probability of rolling a 5 on the second die: $P(\{5 \text{ on second die}\}) = 1/6$
- Since these are independent events:

Solution:

- Probability of rolling a 4 on the first die: $P(4 \text{ on first die}) = \frac{1}{6}$
- Probability of rolling a 5 on the second die: $P(5 \text{ on second die}) = \frac{1}{6}$

Since these are independent events:

$$P(4 \text{ on first die and } 5 \text{ on second die}) = P(4 \text{ on first die}) \times P(5 \text{ on second die}) =$$

$$\frac{1}{6} \times \frac{1}{6} = \frac{1}{36}$$

So, the probability of rolling a 4 on the first die and a 5 on the second die is $1/36$.

Bayes's Theorem: Detailed Concept

Definition

Bayes's Theorem is a fundamental concept in probability theory that describes how to update the probability of a hypothesis, H_{HH} , based on new evidence, E_{EE} . It is named after the Reverend Thomas Bayes, who first provided an equation that allows new evidence to update beliefs about the likelihood of a given event.

Bayes's Theorem links the conditional probability of the hypothesis given the evidence, $P(H|E)P(H|E)P(H|E)$, with the conditional probability of the evidence given the hypothesis, $P(E|H)P(E|H)P(E|H)$, along with the prior probability of the hypothesis, $P(H)P(H)P(H)$, and the marginal likelihood of the evidence, $P(E)P(E)P(E)$.

Concept

Bayes's Theorem is built upon the concept of **conditional probability**, which is the probability of an event occurring given that another event has already occurred. In real life, we often encounter situations where we have some prior knowledge about an event, and as we gather more information, we refine our predictions or beliefs about that event.

Bayes's Theorem is particularly useful in situations where we need to make decisions based on incomplete or evolving information. It allows us to revise our predictions or hypotheses by incorporating new data. This concept is widely used in various fields like medical diagnosis, machine learning, finance, and more.

The Formula

The mathematical formula for Bayes's Theorem is:

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)}$$

Where:

- $P(H|E)$ is the **posterior probability**, the probability of the hypothesis H_{HH} given the evidence E_{EE} .
- $P(E|H)$ is the **likelihood**, the probability of the evidence E_{EE} given that the hypothesis H_{HH} is true.
- $P(H)$ is the **prior probability**, the initial probability of the hypothesis H_{HH} before considering the evidence E_{EE} .
- $P(E)$ is the **marginal likelihood** or **evidence**, the total probability of the evidence E_{EE} under all possible hypotheses.

Example

Let's consider a classic example related to medical diagnosis.

Scenario: A patient is being tested for a rare disease. The test for the disease is 99% accurate, meaning it correctly identifies the disease 99% of the time if the patient has it, and it correctly identifies 99% of healthy patients as not having the disease. However, the disease is quite rare, affecting only 1 in 10,000 people.

Problem: If the test result comes back positive, what is the probability that the patient actually has the disease?

Solution:

- Let H be the event that the patient has the disease.
- Let E be the event that the test result is positive.

We know the following:

- $P(H) = \frac{1}{10,000} = 0.0001$ (The prior probability that the patient has the disease)
- $P(\neg H) = 1 - P(H) = 0.9999$ (The probability that the patient does not have the disease)
- $P(E|H) = 0.99$ (The probability of a positive test result given that the patient has the disease)
- $P(E|\neg H) = 0.01$ (The probability of a positive test result given that the patient does not have the disease)

First, we need to calculate the marginal likelihood $P(E)$, which is the total probability of getting a positive test result under both scenarios (having the disease or not having the disease):

$$P(E) = P(E|H) \cdot P(H) + P(E|\neg H) \cdot P(\neg H)$$

Substituting the values:

$$P(E) = (0.99 \times 0.0001) + (0.01 \times 0.9999)$$

$$P(E) = 0.000099 + 0.009999 = 0.010098$$

Now, using Bayes's Theorem to calculate the posterior probability $P(H|E)$:

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)}$$

$$P(H|E) = \frac{0.99 \times 0.0001}{0.010098} = \frac{0.000099}{0.010098} \approx 0.0098$$

So, even after a positive test result, the probability that the patient actually has the disease is only about 0.98%, which is surprisingly low. This result is due to the rarity of the disease combined with the fact that the test, while accurate, still has a small false positive rate.

Intuition and Insights

- Prior and Posterior Probability: The prior probability reflects what we know before considering new evidence, while the posterior probability updates this belief in light of the new evidence.
- Impact of Rare Events: When dealing with rare events, even highly accurate tests can lead to counterintuitive results. This is known as the base rate fallacy, where the base rate (prior probability) of the event significantly influences the outcome.
- Relevance in Decision-Making: Bayes's Theorem is crucial in decision-making processes that involve uncertainty. By systematically updating probabilities, it helps make more informed decisions.

Applications of Bayes's Theorem

Bayes's Theorem is applied in various fields:

- Medical Diagnosis: Estimating the likelihood of a disease given a test result, as illustrated in the example above.
- Machine Learning: Algorithms like Naive Bayes classifiers rely on Bayes's Theorem for classifying data.
- Finance: Estimating the likelihood of market movements based on new financial data or news.
- Law: Assessing the likelihood of a suspect's guilt given new evidence in a case.

Bayes's Theorem provides a robust framework for updating beliefs and making decisions under uncertainty, making it a powerful tool in both theoretical and applied probability.

Permutation vs Combination

The key differences between permutation and combination, some of those differences are listed as follows:

Aspect	Permutations	Combinations
Definition	Arrangements of elements in a specific order.	Selections of elements without considering the order.
Formula	${}^n P_r = n!/(n-r)!$	${}^n P_r = n!/[(n-r)! \times r!]$
Notation	${}^n P_r$ OR $P(n, r)$	${}^n C_r$ OR $C(n, r)$
Order Matters	Yes, order matters.	No, order doesn't matter.
Example	Arranging books on a shelf.	Selecting members for a committee.
Sample Problems	How many ways to arrange 3 books out of 5?	How many ways to choose 2 fruits from a basket of 7?
Application	Permutations are used when order matters, such as arranging items in a sequence or forming a code.	Combinations are used when order doesn't matter, like selecting a group of people or choosing items without caring about their order.

Unit 2

Random

Variables

Topics

Random Variable Concept

Discrete and Continuous Random Variable

Probability density function

Mathematical Expectation and their Theorem

Random Variable Concept - Introduction to Random Variables

Definition

A Random Variable is a fundamental concept in probability and statistics, representing a variable whose possible values are outcomes of a random phenomenon. It is a function that assigns a numerical value to each outcome in a sample space, which is the set of all possible outcomes of a random experiment.

In simpler terms, a random variable is a way to quantify the outcomes of a random process. For example, when you roll a die, the outcome can be any number between 1 and 6. If we define a random variable X to represent the outcome, then X can take any of these six values.

Types of Random Variables

There are two main types of random variables: Discrete and Continuous.

a. Discrete Random Variable

A **Discrete Random Variable** takes on a countable number of distinct values. These values are often integers and can be listed out. Common examples include the number of heads when flipping a coin multiple times, the number of students in a classroom, or the number rolled on a die.

Example:

Let X represent the number of heads in three flips of a fair coin. The possible values of X are 0, 1, 2, or 3, because you can get anywhere from 0 to 3 heads in three flips.

b. Continuous Random Variable

A **Continuous Random Variable** can take on an infinite number of possible values within a given range. These values are often real numbers, and they are typically measured rather than counted. Examples include the height of students in a class, the time it takes to run a race, or the temperature at a particular location.

Example:

Let Y represent the time it takes for a runner to complete a marathon. Y can take any value from, say, 2 hours to 6 hours, including any fractional value within this range (e.g., 3.5 hours, 4.1 hours).

Probability Distribution

A **Probability Distribution** describes how the probabilities are distributed over the values of a random variable. The probability distribution depends on whether the random variable is discrete or continuous.

a. Probability Mass Function (PMF) for Discrete Random Variables

For a discrete random variable, the probability distribution is described by a Probability Mass Function (PMF). The PMF gives the probability that a random variable is exactly equal to some value.

Example:

For a fair six-sided die, let X be the outcome when the die is rolled. The PMF is:

$$P(X = x) = \frac{1}{6}, \quad \text{for } x = 1, 2, 3, 4, 5, 6$$

This means each outcome (1 through 6) has an equal probability of 1/6.

b. Probability Density Function (PDF) for Continuous Random Variables

For a continuous random variable, the probability distribution is described by a **Probability Density Function (PDF)**. Unlike the PMF, the PDF does not give probabilities directly but rather describes the density of the probability at each point. The probability of the variable falling within a specific range is given by the area under the curve of the PDF over that range.

Example:

Let Y represent the height of adult men in a population, and assume Y follows a normal distribution with a mean of 70 inches and a standard deviation of 3 inches. The PDF of Y describes how heights are distributed around the mean. The probability that a randomly chosen man is between 68 and 72 inches tall is given by the area under the PDF curve between these two values.

Expectation (Mean) and Variance

a. Expectation (Mean)

The **Expectation or Mean** of a random variable is the long-run average value of repetitions of the experiment it represents. It gives a measure of the central tendency of the distribution.

- For a **Discrete Random Variable** X with possible values x_1, x_2, \dots, x_n and corresponding probabilities $P(X = x_1), P(X = x_2), \dots, P(X = x_n)$, the expectation $E(X)$ is:

$$E(X) = \sum_{i=1}^n x_i \cdot P(X = x_i)$$

- For a **Continuous Random Variable** Y with a probability density function $f(y)$, the expectation $E(Y)$ is:

$$E(Y) = \int_{-\infty}^{\infty} y \cdot f(y) dy$$

Example:

If you roll a fair die, the expected value (mean) of the outcome is:

$$E(X) = \sum_{x=1}^6 x \cdot \frac{1}{6} = \frac{1 + 2 + 3 + 4 + 5 + 6}{6} = 3.5$$

b. Variance and Standard Deviation

The **Variance** of a random variable measures the spread or dispersion of the values around the mean. It is the expected value of the squared deviation of the random variable from its mean.

- For a Discrete Random Variable X with mean μ :

$$\text{Var}(X) = E[(X - \mu)^2] = \sum_{i=1}^n (x_i - \mu)^2 \cdot P(X = x_i)$$

- For a **Continuous Random Variable** Y with mean μ :

$$\text{Var}(Y) = E[(Y - \mu)^2] = \int_{-\infty}^{\infty} (y - \mu)^2 \cdot f(y) dy$$

The Standard Deviation is the square root of the variance, providing a measure of spread in the same units as the original variable.

Example:

If you roll a fair die, the variance of the outcome can be calculated as:

$$\text{Var}(X) = \sum_{x=1}^6 (x - 3.5)^2 \cdot \frac{1}{6} = \frac{1}{6} \times (2.92 + 2.52 + 2.25 + 2.25 + 2.52 + 2.92) = \frac{35}{12} \approx 2.92$$

The standard deviation would be $\sqrt{2.92} \approx 1.71$.

Applications of Random Variables

Random variables are extensively used in statistical analysis and modeling across various fields:

- Finance: Modeling stock prices, where the price at any given time can be considered a random variable.
- Insurance: Estimating risks, where claims and losses are modeled as random variables.
- Engineering: Reliability analysis, where the failure time of a component is treated as a random variable.
- Science: Experiment outcomes, such as particle counts in physics experiments, are modeled using random variables.

Real-Life Example

Scenario: Imagine a company produces light bulbs, and historically, 5% of the bulbs are defective. Let X be the random variable representing the number of defective bulbs in a sample of 100 bulbs.

Solution:

- X is a discrete random variable because it counts the number of defective bulbs.
- The probability of a bulb being defective is $P(\text{defective}) = 0.05$.
- The expected number of defective bulbs in a sample of 100 is:

$$E(X) = n \times p = 100 * 0.05 = 5$$

- The variance can be calculated using the formula for the variance of a binomial distribution

$$\text{Var}(X) = n \times p \times (1 - p) :$$

$$\text{Var}(X) = 100 \times 0.05 \times 0.95 = 4.75$$

- The standard deviation is $\sqrt{4.75} = \text{approx. } 2.18$.

This analysis helps the company understand the expected number of defective bulbs and the variability around this expectation, which is crucial for quality control and decision-making.

Random variables are the building blocks of statistical analysis, allowing us to model and understand the uncertainty inherent in various processes. By defining, analyzing, and interpreting random variables, we can make informed decisions in fields as diverse as finance, engineering, science, and beyond.

Discrete and Continuous Random Variable

A Random Variable is a variable that takes on different values based on the outcomes of a random experiment. It quantifies the outcomes of random phenomena and is a key concept in probability and statistics. Random variables can be categorized into two main types: Discrete and Continuous.

Discrete Random Variables

Definition

A Discrete Random Variable is a random variable that can take on a countable number of distinct values. These values are typically integers and can be listed individually. The term "discrete" indicates that there are gaps between the possible values of the variable, meaning the variable can only assume specific points on the number line.

Characteristics

- **Countable Outcomes:** The values a discrete random variable can take are finite or countably infinite. For example, the number of students in a classroom or the number of heads when flipping a coin multiple times.
- **Probability Mass Function (PMF):** The probability distribution of a discrete random variable is described by a Probability Mass Function (PMF). The PMF gives the probability that the random variable is exactly equal to each possible value. The sum of all probabilities in the PMF equals 1.

Example 1: Number of Heads in Coin Flips

Let's say you flip a fair coin three times. Define the discrete random variable X as the number of heads obtained.

- Possible values of X : 0, 1, 2, 3.
- The probabilities can be calculated using the binomial distribution:

$$P(X = 0) = \frac{1}{8}, \quad P(X = 1) = \frac{3}{8}, \quad P(X = 2) = \frac{3}{8}, \quad P(X = 3) = \frac{1}{8}$$

The PMF for X is:

The PMF for X is:

$$P(X = x) = \begin{cases} \frac{1}{8}, & \text{if } x = 0, \\ \frac{3}{8}, & \text{if } x = 1, \\ \frac{3}{8}, & \text{if } x = 2, \\ \frac{1}{8}, & \text{if } x = 3. \end{cases}$$

Example 2: Rolling a Die

Consider rolling a fair six-sided die. Define the discrete random variable Y as the outcome of the roll.

- Possible values of Y : 1, 2, 3, 4, 5, 6.
- Each outcome has an equal probability of $1/6$.

The PMF for Y is:

The PMF for Y is:

$$P(Y = y) = \begin{cases} \frac{1}{6}, & \text{if } y = 1, 2, 3, 4, 5, 6, \\ 0, & \text{otherwise.} \end{cases}$$

Continuous Random Variables

Definition

A Continuous Random Variable is a random variable that can take on an infinite number of possible values within a given range. These values are uncountable and typically include real numbers, meaning the variable can assume any value within a certain interval.

Characteristics

- Uncountable Outcomes: The values of a continuous random variable are uncountable and can include any real number within a certain range. Examples include the height of people, the time taken to complete a task, or the temperature in a room.
- Probability Density Function (PDF): The probability distribution of a continuous random variable is described by a Probability Density Function (PDF). Unlike the PMF, the PDF does not give the probability that the variable takes a specific value; instead, it gives the density of probability at each point. The probability that the variable falls within a specific range is found by calculating the area under the curve of the PDF over that range.

Example 1: Height of Students

Let's say the height of students in a class is normally distributed with a mean of 170 cm and a standard deviation of 10 cm. Define the continuous random variable ZZZ as the height of a randomly selected student.

- The PDF of Z is described by the normal distribution:

$$f(z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(z - \mu)^2}{2\sigma^2}\right)$$

where $\mu=170$ and $\sigma=10$

- The probability that a student's height is between 160 cm and 180 cm is given by the area under the PDF curve from 160 to 180.

Example 2: Time to Complete a Task

Consider the time TTT it takes to complete a task, which could be any value between, say, 0 and 10 hours. If TTT is uniformly distributed, the PDF is constant over the interval.

- The PDF for T is:

$$f(t) = \frac{1}{10}, \quad \text{for } 0 \leq t \leq 10$$

- The probability that the task takes between 4 and 6 hours is the area under the PDF from 4 to 6, calculated as:

$$P(4 \leq T \leq 6) = \int_4^6 \frac{1}{10} dt = \frac{2}{10} = 0.2$$

Key Differences Between Discrete and Continuous Random Variables

Feature	Discrete Random Variable	Continuous Random Variable
Possible Values	Countable, distinct values (e.g., 0, 1, 2, 3)	Uncountable, infinite values (e.g., 1.5, 2.75, 3.14)
Probability Distribution	Probability Mass Function (PMF)	Probability Density Function (PDF)

Feature	Discrete Random Variable	Continuous Random Variable
Probability Calculation	Direct probabilities for exact values	Probability for intervals (area under the curve)
Example	Number of students in a class, number of heads	Height of students, time to complete a task
Sum of Probabilities	Sum of all probabilities equals 1	Total area under the PDF curve equals 1

Applications in Statistical Analysis

a. Discrete Random Variables

Discrete random variables are often used in situations where the data is inherently countable.

Some applications include:

- **Quality Control:** Number of defective items in a batch.
- **Epidemiology:** Number of new cases of a disease.
- **Insurance:** Number of claims filed in a year.

b. Continuous Random Variables

Continuous random variables are used when the data can take any value within a range. Some applications include:

- **Finance:** Modeling stock prices, where prices can take any real number within a range.
- **Environmental Science:** Measuring pollutants in the air, which can take any value within a given concentration range.
- **Engineering:** Analyzing the time until failure of a machine component.

Probability density function

Probability Density Function is the function of probability defined for various distributions of variables and is the less common topic in the study of probability throughout the academic journey of students. However, this function is very useful in many areas of real life such as predicting rainfall, financial modelling such as the stock market, income disparity in social sciences, etc.

This article explores the topic of the Probability Density Function in detail including its definition, condition for existence of this function, as well as various examples.

What is Probability Density Function(PDF)?

Probability Density Function is used for calculating the probabilities for continuous random variables.

When the cumulative distribution function (CDF) is differentiated we get the probability density function (PDF). Both functions are used to represent the probability distribution of a continuous random variable.

The probability density function is defined over a specific range. By differentiating CDF we get PDF and by integrating the probability density function we can get the cumulative density function.

Probability Density Function Definition

Probability density function is the function that represents the density of probability for a continuous random variable over the specified ranges.

Probability Density Function is abbreviated as PDF and for a continuous random variable X, Probability Density Function is denoted by $f(x)$.

PDF of the random variable is obtained by differentiating CDF (Cumulative Distribution Function) of X. The probability density function should be a positive for all possible values of the variable. The total area between the density curve and the x-axis should be equal to 1.

Necessary Conditions for PDF

Let X be the continuous random variable with probability density function $f(x)$. For a function to be valid probability function should satisfy below conditions.

- $f(x) \geq 0, \forall x \in R$
- $f(x)$ should be piecewise continuous.

$$\int_{-\infty}^{\infty} f(x)dx = 1$$

So, the PDF should be the non-negative and piecewise continuous function whose total value evaluates to 1.

Example of a Probability Density Function

Let X be a continuous random variable and the probability density function pdf is given by $f(x) = x - 1$, $0 < x \leq 5$. We have to find $P(1 < x \leq 2)$.

To find the probability $P(1 < x \leq 2)$ we integrate the pdf $f(x) = x - 1$ with the limits 1 and 2. This results in the probability $P(1 < x \leq 2) = 0.5$

Probability Density Function Formula

Let Y be a continuous random variable and $F(y)$ be the cumulative distribution function (CDF) of Y. Then, the probability density function (PDF) $f(y)$ of Y is obtained by differentiating the CDF of Y.

$$f(y) = \frac{d}{dy} [F(y)] = F'(y)$$

If we want to calculate the probability for X lying between the interval a and b, then we can use the following formula:

$$P(a \leq X \leq b) = F(b) - F(a) = \int_a^b f(x) dx$$

Key Points about PDF Formula

- If we differentiate CDF, we get the PDF of the random variable.

$$f(y) = \frac{d}{dy} [F(y)]$$

- If we integrate PDF, we get the CDF of the random variable.

$$F(y) = \int_{-\infty}^y f(t) dt$$

What Does a Probability Density Function (PDF) Tell Us?

A Probability Density Function (PDF) is a function that describes the likelihood of a continuous random variable taking on a particular value. Unlike discrete random variables, where probabilities are assigned to specific outcomes, continuous random variables can take on any value within a range. Probability Density Function (PDF) tells us

- Relative Likelihood
- Distribution Shape
- Expected Value and Variance, etc.

How to Find Probability from Probability Density Function

To find the probability from the probability density function we have to follow some steps.

Step 1: First check the PDF is valid or not using the necessary conditions.

Step 2: If the PDF is valid, use the formula and write the required probability and limits.

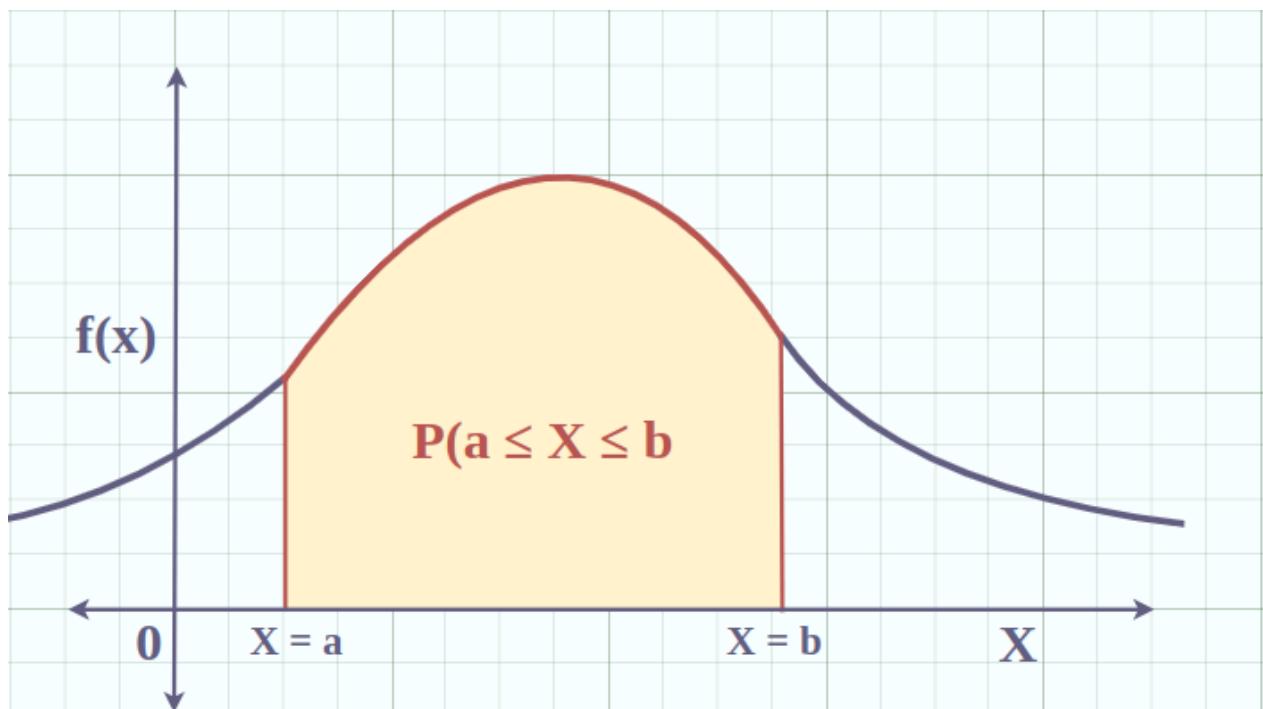
Step 3: Divide the integration according to the given PDF.

Step 4: Solve all integrations.

Step 5: The resultant value gives the required probability.

Graph for Probability Density Function

If X is continuous random variable and $f(x)$ be the probability density function. The probability for the random variable is given by area under the pdf curve. The graph of PDF looks like bell curve, with the probability of X given by area below the curve. The following graph gives the probability for X lying between interval a and b .



Probability Density Function Properties

Let $f(x)$ be the probability density function for continuous random variable x . Following are some probability density function properties:

- Probability density function is always positive for all the values of x .

$$f(x) \geq 0, \forall x \in \mathbb{R}$$

- Total area under probability density curve is equal to 1.

$$\int_{-\infty}^{\infty} f(x)dx = 1$$

- For continuous random variable X, while calculating the random variable probabilities end values of the interval can be ignored i.e., for X lying between interval a and b

$$P(a \leq X \leq b) = P(a \leq X < b) = P(a < X \leq b) = P(a < X < b)$$

- Probability density function of a continuous random variable over a single value is zero.

$$P(X = a) = P(a \leq X \leq a) = \int_a^a f(x)dx = 0$$

- Probability density function defines itself over the domain of the variable and over the range of the continuous values of the variable.

Mean of Probability Density Function

Mean of the probability density function refers to the average value of the random variable. The mean is also called as expected value or expectation. It is denoted by μ or $E[X]$ where, X is random variable.

Mean of the probability density function $f(x)$ for the continuous random variable X is given by:

$$E[X] = \mu = \int_{-\infty}^{\infty} xf(x)dx$$

Median of Probability Density Function

Median is the value which divides the probability density function graph into two equal halves. If $x = M$ is the median then, area under curve from $-\infty$ to M and area under curve from M to ∞ are equal which gives the median value = 1/2.

Median of the probability density function $f(x)$ is given by:

$$\int_{-\infty}^M f(x)dx = \int_M^{\infty} f(x)dx = \frac{1}{2}$$

Variance Probability Density Function

Variance of probability density function refers to the squared deviation from the mean of a random variable. It is denoted by $Var(X)$ where, X is random variable.

Variance of the probability density function $f(x)$ for continuous random variable X is given by:

$$Var(X) = E[(X - \mu)^2] = \int_{-\infty}^{\infty} (x - \mu)^2 f(x) dx$$

Standard Deviation of Probability Density Function

Standard Deviation is the square root of the variance. It is denoted by σ and is given by:

$$\sigma = \sqrt{Var(X)}$$

Probability Density Function Vs Cumulative Distribution Function

The key differences between Probability Density Function (PDF) and Cumulative Distribution Function (CDF) are listed in the following table:

Aspect	Probability Density Function (PDF)	Cumulative Distribution Function (CDF)
Definition	The PDF gives the probability that a random variable takes on a specific value within a certain range.	The CDF gives the probability that a random variable is less than or equal to a specific value.
Range of Values	Defined for continuous random variables.	Defined for both continuous and discrete random variables.
Mathematical Expression	$f(x)$, where $f(x) \geq 0$ and $\int_{-\infty}^{\infty} f(x) dx = 1$	$F(x)$, where $0 \leq F(x) \leq 1$ for all x , and $F(-\infty) = 0$ and $F(\infty) = 1$
Interpretation	Represents the likelihood of the random variable taking on a specific value.	Represents the probability that the random variable is less than or equal to a specific value.
Area Under	The area under the PDF curve	The value of the CDF at a

Aspect	Probability Density Function (PDF)	Cumulative Distribution Function (CDF)
the Curve	over a certain interval gives the probability that the random variable falls within that interval.	specific point gives the probability that the random variable is less than or equal to that point.
Relationship with CDF	The PDF can be obtained by differentiating the CDF with respect to the random variable.	The CDF can be obtained by integrating the PDF with respect to the random variable.
Probability Calculation	The probability of a random variable falling within a specific interval (a,b) is given by $\int_a^b f(x)dx$.	The probability of a random variable being less than or equal to a specific value x is given by $F(x)$.
Properties	The PDF is always non-negative: $f(x) \geq 0$ for all x . The total area under the PDF curve is equal to 1.	The CDF is a monotonically increasing function: $F(x_1) \leq F(x_2)$ if $x_1 \leq x_2$. $0 \leq F(x) \leq 1$ for all x .
Examples	Normal Distribution PDF: $\frac{1}{\sigma\sqrt{2\pi}}e^{-(x-\mu)^2/2\sigma^2}$ Exponential distribution PDF: $\lambda e^{-\lambda x}$	Normal Distribution CDF: $F(x) = \frac{1}{2} \operatorname{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)$ Exponential distribution CDF: $F(x) = 1 - e^{-\lambda x}$

Types of Probability Density Function

There are different types of probability density functions given below:

- Uniform Distribution
- Binomial Distribution
- Normal Distribution
- Chi-Square Distribution

Difference Between PDF and Joint PDF

The PDF is the function defined for single variable whereas joint PDF is the function defined for two or more than two variables, and other key differences between these both concepts are listed in the following table:

PDF (Probability Density Function)	Joint PDF
Probability Density Function is the probability function defined for single variable.	Joint Probability Density Function is the probability function defined for more than one variable.
It is denoted as $f(x)$.	It is denoted as $f(x, y, \dots)$.
Probability Density Function is obtained by differentiating the CDF.	Joint Probability Density Function is obtained by differentiating the joint CDF
It can be calculated by single integral.	It can be calculated using multiple integrals as there are multiple variables.

Applications of Probability Density Function

Some of the applications of Probability Density function are:

- Probability density functions are used in statistics for calculating probabilities for random variables.
- It is used in modelling various scientific data.
- **Cumulative Frequency Distribution**
- **Probability Distribution Function**

Examples on Probability Density Function

Example 1: If the probability density function is given as: $f(x) = \begin{cases} x/2 & 0 \leq x < 4 \\ 0 & x \geq 4 \end{cases}$. Find $P(1 \leq X \leq 2)$.

Solution:

Apply the formula and integrate the PDF.

$$P(1 \leq X \leq 2) = \int_1^2 f(x)dx$$

$$f(x) = x/2 \text{ for } 0 \leq x \leq 4$$

$$\Rightarrow P(1 \leq X \leq 2) = \int_1^2 (x/2)dx$$

$$\Rightarrow P(1 \leq X \leq 2) = \frac{1}{2} \times \left[\frac{x^2}{2} \right]_1^2$$

$$\Rightarrow P(1 \leq X \leq 2) = 3/4$$

Example 3: If the probability density function is given as: $f(x) = \begin{cases} \frac{5}{2}x^2 & 0 \leq x < 2 \\ 0 & \text{otherwise} \end{cases}$. Find the mean.

Solution:

Formula for mean:

$$\begin{aligned}\mu &= \int_{-\infty}^{\infty} xf(x)dx \\ \Rightarrow \mu &= \int_{-\infty}^1 x(0)dx + \int_1^2 x\left(\frac{5x^2}{2}\right)dx + \int_2^{\infty} x(0)dx \\ \Rightarrow \mu &= \frac{5}{2} \left[\frac{x^4}{4} \right]_1 \\ \Rightarrow \mu &= (5/2) \times (15/4) \\ \Rightarrow \mu &= 75/8 = 9.375\end{aligned}$$

Example 4: If the probability density function is given as: $f(x) = \begin{cases} 2x & 0 \leq x < 1 \\ 0 & \text{otherwise} \end{cases}$. Verify if this is a valid probability density function.

To verify that $f(x)$ is a valid PDF, it must satisfy two conditions:

$f(x) \geq 0$ for all x .

The integral of $f(x)$ over its entire range must equal 1.

Checking $f(x) \geq 0$:

$f(x) = 2x$ is clearly non-negative for $0 \leq x \leq 10$.

$$\begin{aligned}\text{Integrating } f(x) \text{ over its range: } &\int_{-\infty}^{\infty} f(x) dx = \int_0^1 2x dx = [x^2]_0^1 = 1 - 0 = 1. \int_{-\infty}^{\infty} f(x) dx = \\ &\int_0^1 2x dx = [x^2]_0^1 \\ &= 1^2 - 0^2 = 1.\end{aligned}$$

Since both conditions are satisfied, $f(x)$ is a valid PDF.

Mathematical Expectation and their Theorem

Because random variables are *random*, knowing the outcome on any one realisation of the random process is not possible. Instead, we can talk about what we might *expect* to happen, or what might happen *on average*.

This is the idea of *mathematical expectation*. In more usual terms, the mathematical expression of the probability distribution of a random variable is the *mean* of the random variable. Mathematical expectation goes far beyond just computing means, but we begin here as the idea of a *mean* is easily understood.

The definition looks different in detail for discrete and continuous random variables, but the intention is the same.

Definition 3.1 (Expectation) The *expectation* or *expected value* (or *mean*) of a random variable X is defined as

- $E(X) = \sum_{x \in R_X} xp_X(x)$ for a discrete random variable X with pmf $p_X(x)$;
- $E(X) = \int_{-\infty}^{\infty} xf_X(x) dx$ for a continuous random variable X with pdf $f_X(x)$.

Often we write $\mu = E(X)$, or μ_X to distinguish between random variables.

Effectively $E(X)$ is a weighted average of the points in R_X , the weights being the probabilities in the discrete case and probability densities in the continuous case.

Example - (Expectation for discrete variables) Consider the discrete random variable U with probability function

$$p_U(u) = \begin{cases} (u^2 + 1)/5 & \text{for } u = -1, 0, 1; \\ 0 & \text{elsewhere.} \end{cases}$$

The expected value of U is, by definition,

$$\begin{aligned} E(U) &= \sum_{u=-1,0,1} up_U(u) \\ &= \sum_{u=-1,0,1} u \times \left(\frac{u^2 + 1}{5} \right) \\ &= \left(-1 \times \frac{(-1)^2 + 1}{5} \right) + \left(0 \times \frac{(0)^2 + 1}{5} \right) + \left(1 \times \frac{(1)^2 + 1}{5} \right) \\ &= -2/15 + 0 + 2/15 = 0. \end{aligned}$$

The expected value of U is 0.

Example - (Expectation for continuous variables) Consider a continuous random variable X with pdf

$$f_X(x) = \begin{cases} x/4 & \text{for } 1 < x < 3; \\ 0 & \text{elsewhere.} \end{cases}$$

The expected value of X is, by definition,

$$\begin{aligned} E(X) &= \int_{-\infty}^{\infty} xf_X(x) dx = \int_1^3 x(x/4) dx \\ &= \frac{1}{12}x^3 \Big|_1^3 = 13/6. \end{aligned}$$

The expected value of X is $13/6$.

Example - (Expectation for a coin toss) Consider tossing a coin *once* and counting the number of tails. Let this random variable be T . The probability function is

$$p_T(t) = \begin{cases} 0.5 & \text{for } t = 0 \text{ or } t = 1; \\ 0 & \text{otherwise.} \end{cases}$$

The expected value of T is, by definition

$$\begin{aligned} E(T) &= \sum_{i=1}^2 tp_T(t) \\ &= \Pr(T = 0) \times 0 + \Pr(T = 1) \times 1 \\ &= (0.5 \times 0) + (0.5 \times 1) = 0.5. \end{aligned}$$

Of course, 0.5 tails can never actually be observed in practice on one toss. But it would be silly to round up (or down) and say that the expected number of tails on one toss of a coin is one (or zero). The expected value of 0.5 simply means that over a large number of repeats of this random process, we expect a tail to occur in half of those repeats.

Example (Mean not defined) Consider the distribution of Z , with the probability density function

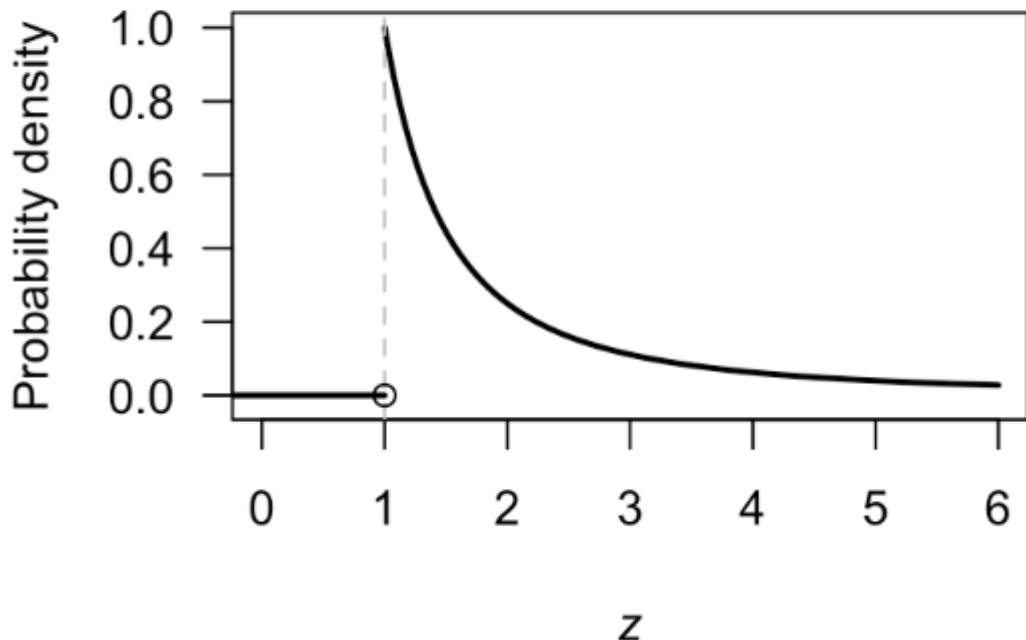
$$f_Z(z) = \begin{cases} z^{-2} & \text{for } z \geq 1; \\ 0 & \text{elsewhere} \end{cases}$$

as in Fig. 3.1. The expected value of Z is

$$E[Z] = \int_1^{-\infty} z \frac{1}{z^2} dz = \int_1^{\infty} \frac{1}{z} dz = -\log z \Big|_1^{\infty}.$$

However, $\lim_{z \rightarrow \infty} -\log z \rightarrow \infty$. The expected value of $E[Z]$ is undefined.

The probability function for Z



Expectation of a function of a random variable

While the mean can be expressed in terms of mathematical expectation, mathematical expectation is a more general concept.

Let X be a discrete random variable with a probability function $p_X(x)$, or a continuous random variable with pdf $f_X(x)$. Also assume $g(X)$ is a real-valued function of X. We can then define the expected value of $g(X)$.

Definition (Expectation for function of a random variable) The *expected value* of some function $g(\cdot)$ of a random variable X is:

- $E(g(X)) = \sum_{x \in R_X} g(x)p_X(x)$ for a discrete random variable X with pmf $p_X(x)$;
- $E(g(X)) = \int_{-\infty}^{\infty} g(x)f_X(x) dx$ for a continuous random variable X with pdf $f_X(x)$.

Unit 3

Distrubution

Topics

Distribution
Types of Data distribution
Exponential distribution
Binomial distribution
Normal distribution
Poisson distribution
Random number generation
Monte Carlo Simulation.

Distribution

In statistics and probability, the term distribution refers to the way in which values or observations are spread or arranged across a range of possibilities. It describes how probabilities or frequencies of different outcomes are distributed in a dataset or a random variable. A distribution can provide insights into the characteristics of a dataset, such as its central tendency, dispersion, shape, and outliers.

Basic Definitions

A distribution is essentially a function or a set of rules that assigns probabilities to the possible outcomes of a random variable. A random variable is a quantity that can take on different values due to randomness or uncertainty.

Distributions are broadly categorized into two types:

1. **Discrete Distribution:** This type of distribution is used when the set of possible outcomes is countable (e.g., number of heads when flipping a coin).
2. **Continuous Distribution:** This type is used when outcomes can take any value within a range (e.g., height of people, temperature, time).

Key Concepts in Distribution

1. Probability Distribution

A probability distribution is a mathematical description of the likelihood of different outcomes in an experiment. It lists all the possible values a random variable can assume and assigns probabilities to them. The sum of probabilities in a probability distribution always equals 1.

- For a discrete random variable X, the probability distribution is often represented as a probability mass function (PMF):

$$P(X = x) = p(x)$$

- For a continuous random variable Y, the distribution is represented using a probability density function (PDF):

$$f(y)$$

The PDF describes the relative likelihood of different values, but to get the actual probability, one has to integrate the PDF over an interval.

2. Cumulative Distribution Function (CDF)

The cumulative distribution function describes the probability that a random variable takes a value less than or equal to a certain value. It accumulates the probabilities or densities over the range of the distribution.

For a random variable X, the CDF is defined as:

$$F(x) = P(X \leq x)$$

This function provides the probability that X will be less than or equal to a particular value, helping to understand the spread of values in a more intuitive way.

Moments of Distribution

The moments of a distribution are statistical measures that describe different characteristics of a distribution. They include:

- **Mean (First Moment):** The average or expected value of the distribution. It provides a measure of the central tendency.

$$\mu = E(X) = \sum_x x \cdot p(x) \quad \text{for discrete variables}$$

For continuous variables:

$$\mu = E(X) = \int_{-\infty}^{\infty} x \cdot f(x) dx$$

- **Variance (Second Moment):** It measures the dispersion or spread of the distribution around the mean. A higher variance indicates that the data points are more spread out.

$$\sigma^2 = E[(X - \mu)^2]$$

- Skewness (Third Moment): It measures the asymmetry of the distribution around the mean. Positive skewness indicates that the right tail is longer, while negative skewness indicates a longer left tail.
- Kurtosis (Fourth Moment): It measures the "tailedness" of the distribution, or the likelihood of extreme values. A high kurtosis means there are more extreme values or outliers.

Normal Distribution

The normal distribution, often called the Gaussian distribution, is one of the most important continuous probability distributions. It is symmetric about the mean and characterized by its bell-shaped curve. The standard normal distribution has a mean of 0 and a standard deviation of 1.

The probability density function of the normal distribution is:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

Many natural phenomena, such as heights, test scores, and measurement errors, follow a normal distribution.

Uniform Distribution

In a uniform distribution, all outcomes are equally likely. For a discrete uniform distribution, each of the possible values has the same probability. For a continuous uniform distribution, the probability density function is constant over the range of possible outcomes.

The PDF for a continuous uniform distribution between a and b is:

$$f(x) = \frac{1}{b-a}, \quad a \leq x \leq b$$

Binomial Distribution

The binomial distribution is a discrete probability distribution that models the number of successes in a fixed number of independent trials, where each trial has two possible outcomes (success or failure) and a constant probability of success p.

The probability mass function of the binomial distribution is:

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

where n is the number of trials and k is the number of successes.

Exponential Distribution

The exponential distribution is often used to model the time between events in a Poisson process. It is characterized by a constant hazard rate, which makes it a good model for waiting times.

The PDF of the exponential distribution is:

The PDF of the exponential distribution is:

$$f(x) = \lambda e^{-\lambda x}, \quad x \geq 0$$

where λ is the rate parameter.

Examples of Distributions in Real Life

Normal Distribution Example: Heights of adult males in a population tend to follow a normal distribution. Most individuals will have a height around the mean (e.g., 170 cm), with fewer individuals being extremely short or tall. The mean height and standard deviation can describe the entire distribution of heights.

Binomial Distribution Example: Suppose a factory tests a batch of 100 items, and the probability of an item being defective is 0.05. The number of defective items in this batch follows a binomial distribution with parameters $n=100$ and $p=0.05$.

Exponential Distribution Example: The time between arrivals of buses at a bus stop might follow an exponential distribution. If buses arrive on average every 10 minutes, the time between arrivals would follow an exponential distribution with a rate $\lambda=1/10$.

Uniform Distribution Example: Rolling a fair six-sided die is an example of a discrete uniform distribution. Each outcome (1, 2, 3, 4, 5, or 6) has an equal probability of 1/6. Another example is selecting a random number between 0 and 1, which would follow a continuous uniform distribution over that interval.

Understanding distributions is fundamental to the study of statistics and probability. Distributions provide a framework for making predictions, interpreting data, and identifying patterns in randomness. From binomial to normal distributions, each type offers unique insights into how data behaves under different conditions, helping to solve real-world problems and make informed decisions based on observed data. Whether it's estimating risks, calculating probabilities, or analyzing trends, distributions are at the heart of statistical reasoning and data science.

Types of Data distribution

Distributions are at the core of understanding probability and statistical analysis. They provide the foundation for describing data, estimating probabilities, and making predictions. Different types of distributions apply to various types of data, and understanding their properties is critical for correct data analysis. In this write-up, we will explore the characteristics, applications, and examples of several key types of distributions, including normal, binomial, Poisson, uniform, exponential, and chi-square distributions.

1. Normal Distribution (Gaussian Distribution)

The **normal distribution** is one of the most well-known and widely used continuous probability distributions. It is often referred to as the Gaussian distribution after Carl Friedrich Gauss, who introduced it in the early 19th century.

Properties

- **Symmetry:** The normal distribution is symmetric about the mean. This symmetry results in the mean, median, and mode being equal.
- **Bell-Shaped Curve:** The distribution has a characteristic bell-shaped curve, where the majority of data points are concentrated around the mean, and the tails taper off symmetrically on either side.
- **Mean and Standard Deviation:** The distribution is fully described by its mean μ and standard deviation σ . The spread or width of the distribution is determined by the standard deviation.
- **68-95-99.7 Rule:** In a normal distribution, approximately 68% of the data lies within one standard deviation of the mean, 95% within two standard deviations, and 99.7% within three standard deviations.

Probability Density Function (PDF)

The PDF of the normal distribution is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Example

An example of a normal distribution is the distribution of IQ scores. The mean IQ score is typically set at 100, with a standard deviation of 15. Most people have an IQ score close to 100, and fewer individuals have extremely high or low scores.

2. Binomial Distribution

The **binomial distribution** is a discrete probability distribution that represents the number of successes in a fixed number of independent trials, where each trial has only two possible outcomes: success or failure. This distribution is particularly useful in situations involving a sequence of independent yes/no experiments.

Properties

- **Number of Trials:** The binomial distribution requires a fixed number of trials n .
- **Success Probability:** Each trial has the same probability of success p .
- **Independent Trials:** The outcome of each trial is independent of the others.
- **Discrete:** The outcomes are discrete values representing the number of successes in the trials.

Probability Mass Function (PMF)

The PMF of the binomial distribution is given by:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

Where:

- n is the number of trials.
- k is the number of successes.
- p is the probability of success in a single trial.
- $\binom{n}{k}$ is the binomial coefficient, also known as "n choose k."

Example

An example of a binomial distribution is the number of heads in 10 coin flips, where each flip has a 50% chance of landing heads. Here, $n=10$, $p=0.5$ and k is the number of heads observed in 10 flips.

3. Poisson Distribution

The **Poisson distribution** is a discrete probability distribution used to model the number of events occurring within a fixed interval of time or space. It is particularly useful for modeling rare events that occur independently of each other.

Properties

- **Rare Events:** The distribution is used for rare events or occurrences.
- **Single Parameter:** The Poisson distribution is characterized by a single parameter λ , which represents both the mean and variance of the distribution.
- **Non-Negative Integers:** The distribution models the probability of observing a non-negative integer number of events.
- **Memoryless Property:** The Poisson process is memoryless, meaning that the occurrence of an event in one interval does not affect the probability of an event in another interval.

Probability Mass Function (PMF)

The PMF of the Poisson distribution is given by:

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

Where:

- λ is the average number of occurrences in a given interval.
- k is the actual number of occurrences.

Example

An example of a Poisson distribution is the number of emails a person receives per hour. If a person receives, on average, 5 emails per hour, then the number of emails received in a given hour follows a Poisson distribution with $\lambda=5$.

3. Uniform Distribution

The **uniform distribution** is a type of probability distribution where all outcomes are equally likely. There are two types: discrete uniform distribution and continuous uniform distribution.

Properties

- **Equally Likely Outcomes:** In a uniform distribution, every outcome has the same probability of occurring.
- **Constant Probability:** The PDF or PMF is constant across the range of possible outcomes.

- **Range:** The distribution is defined over a specific range $[a,b]$, and all values within this range have the same probability.

Probability Density Function (PDF)

For a continuous uniform distribution over the interval $[a,b]$ the PDF is given by:

$$f(x) = \frac{1}{b-a}, \quad a \leq x \leq b$$

Example

An example of a continuous uniform distribution is the rolling of a fair die. Each of the six sides (1 through 6) has an equal probability of appearing, so the distribution is uniform.

4. Exponential Distribution

The **exponential distribution** is a continuous probability distribution often used to model the time between events in a Poisson process. It describes the waiting time between independent events that occur at a constant average rate.

Properties

- Memoryless: Similar to the Poisson process, the exponential distribution is memoryless, meaning the probability of an event occurring in the future does not depend on how much time has already passed.
- Mean and Variance: The mean and variance of the exponential distribution are related to the rate parameter λ . The mean is $1/\lambda$, and the variance is $1/\lambda^2$.
- Positive Values: The distribution only takes positive values, as it models time intervals.

Probability Density Function (PDF)

The PDF of the exponential distribution is given by:

$$f(x) = \lambda e^{-\lambda x}, \quad x \geq 0$$

Where:

- **λ is the rate parameter (the number of events per unit time).**

Example

An example of an exponential distribution is the time between arrivals of buses at a bus stop. If buses arrive on average every 10 minutes, the time between consecutive bus arrivals follows an exponential distribution with $\lambda=1/10$.

6. Chi-Square Distribution

The **chi-square distribution** is a continuous probability distribution that arises in statistical hypothesis testing, particularly in the context of testing the goodness-of-fit or independence between categorical variables.

Properties

- **Non-Negative Values:** The chi-square distribution only takes positive values, as it involves the sum of squared terms.
- **Degrees of Freedom:** The shape of the chi-square distribution depends on the degrees of freedom k , which is related to the number of independent variables in the analysis.
- **Right-Skewed:** The chi-square distribution is skewed to the right, with more degrees of freedom leading to a distribution that becomes more symmetric.

Probability Density Function (PDF)

The PDF of the chi-square distribution is given by:

$$f(x; k) = \frac{1}{2^{k/2}\Gamma(k/2)}x^{k/2-1}e^{-x/2}$$

Where:

- k is the degrees of freedom.
- $\Gamma(\cdot)$ is the gamma function.

Example

An example of a chi-square distribution is in hypothesis testing for categorical data. If we want to test whether the observed frequencies of different categories match the expected frequencies, we use the chi-square distribution to evaluate the goodness-of-fit.

Exponential distribution

Exponential Distribution

The *Exponential Distribution* is another important distribution and is typically used to model times between events or arrivals. The distribution has one parameter, λ which is assumed to be the average rate of arrivals or occurrences of an event in a given time interval.

If the random variable X follows an Exponential distribution then we write: $X \sim \text{Exp}(\lambda)$.

The probability density function is:

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{for } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

(Cumulative) probabilities can be calculated using:

$$P(X \leq x) = \begin{cases} 0 & \text{for } x < 0, \\ 1 - e^{-\lambda x} & \text{for } x \geq 0. \end{cases}$$

Expectation and Variance

The expectation and variance of an Exponential random variable are:

$$\begin{aligned} E[X] &= \frac{1}{\lambda} \\ \text{Var}(X) &= \frac{1}{\lambda^2} \end{aligned}$$

Example

It is assumed that the average time customers spends on hold when contacting a gas company's call centre is five minutes. The company has a policy that if a customer waits for longer than 15 minutes they are entitled to claim £5 off their next quarterly bill.

If the company employs a new team, at some expense, then the average waiting time is reduced to four minutes.

The director of the company must decide whether or not to employ a new team. He thinks the idea is only worthwhile if the probability that a customer waits for longer than 15 minutes is reduced by at least 0.025.

This situation can be modelled using Exponential distributions: one for waiting times (times on hold) under the current team and one for waiting times under a new team.

With the current team the mean waiting time is 5 minutes and so the mean rate of calls answered per minute is given by $\lambda_1=1/5=0.2$. The corresponding Exponential distribution is $\text{Exp}(0.2)$.

Similarly, with a new team, we have $\lambda_2=1/4=0.25$ and so the corresponding Exponential distribution is $\text{Exp}(0.25)$.

Determine whether the director should employ a new team or keep his current team.

Solution

Let X denote the waiting time of a customer under the current team. We know that X follows an Exponential distribution with parameter $\lambda_1=0.2$ so we have $X \sim \text{Exp}(0.2)$. Now we need to calculate $P(X > 15)$.

$$\begin{aligned} P(X > 15) &= 1 - P(X \leq 15) \\ &= 1 - (1 - e^{-0.2 \times 15}) \\ &= e^{-3} \\ &= 0.050 \text{ (to 3 d.p.)}. \end{aligned}$$

So, the probability that a customer waits for longer than 15 minutes is 0.050 (to 3 d.p.).

Now we consider the probability of this event under a new team. Let Y denote the time a customer waits under a new team $Y \sim \text{Exp}(0.25)$. We know that Y follows an Exponential

distribution with parameter $\lambda=0.25$ so we have $Y \sim \text{Exp}(0.25)$. Thus,

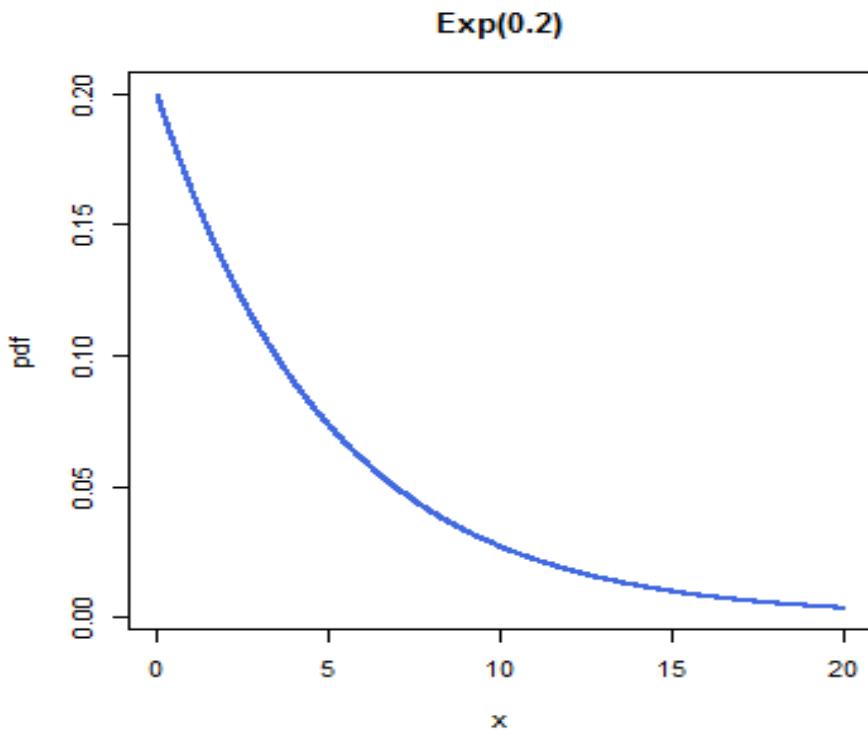
$$\begin{aligned} P(Y > 15) &= 1 - P(Y \leq 15) \\ &= 1 - (1 - e^{-0.25 \times 15}) \\ &= e^{-3.75} \\ &= 0.024 \text{ (to 3 d.p.)}. \end{aligned}$$

Recall that the director of the company would only opt for recruiting a new team if the probability that a customer waits longer than 15 minutes is reduced by at least 0.025.

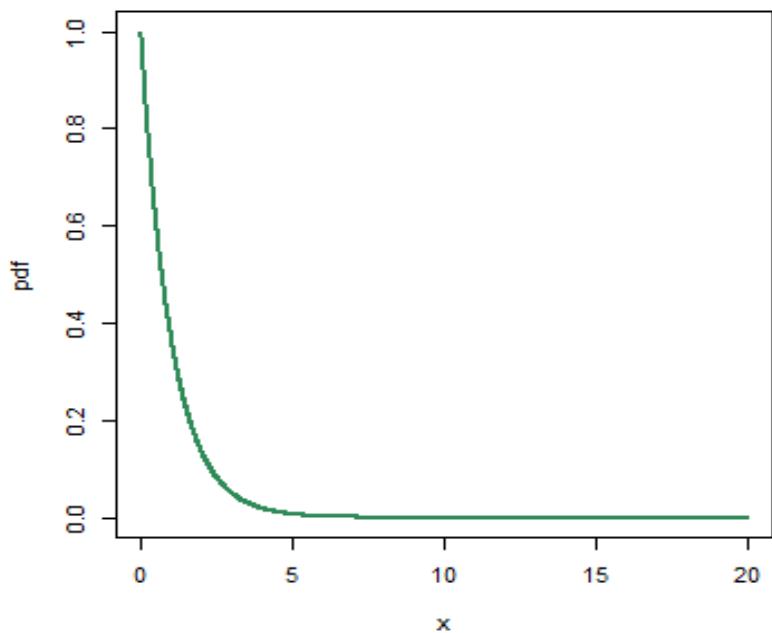
$$\begin{aligned} \text{Change in probability} &= \text{Probability without a new team} - \text{Probability with a new team} \\ &= e^{-3} - e^{-3.75} \\ &= 0.026 \text{ (to 3 d.p.)}. \end{aligned}$$

Since $0.026 > 0.025$, the director of the company should recruit a new team.

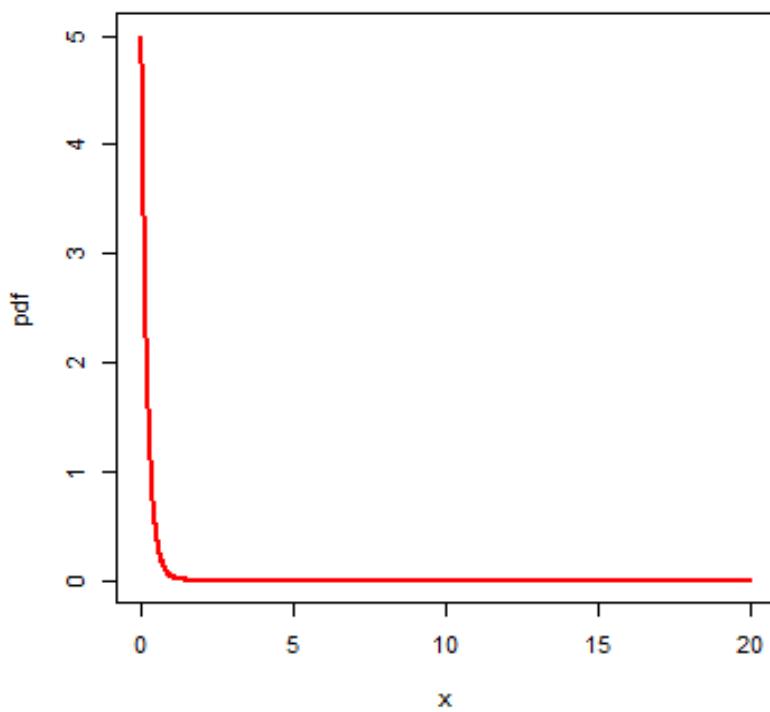
What does an Exponential distribution look like?



$\text{Exp}(1)$



$\text{Exp}(5)$



Binomial distribution

In probability theory and statistics, the binomial distribution is the discrete probability distribution that gives only two possible results in an experiment, either Success or Failure. For example, if we toss a coin, there could be only two possible outcomes: heads or tails, and if any test is taken, then there could be only two results: pass or fail. This distribution is also called a binomial probability distribution.

There are two parameters n and p used here in a binomial distribution. The variable ' n ' states the number of times the experiment runs and the variable ' p ' tells the probability of any one outcome. Suppose a die is thrown randomly 10 times, then the probability of getting 2 for anyone throw is $\frac{1}{6}$. When you throw the dice 10 times, you have a binomial distribution of $n = 10$ and $p = \frac{1}{6}$.

Binomial Probability Distribution

In binomial probability distribution, the number of 'Success' in a sequence of n experiments, where each time a question is asked for yes-no, then the boolean-valued outcome is represented either with success/yes/true/one (probability p) or failure/no/false/zero (probability $q = 1 - p$). A single success/failure test is also called a Bernoulli trial or Bernoulli experiment, and a series of outcomes is called a **Bernoulli process**. For $n = 1$, i.e. a single experiment, the binomial distribution is a **Bernoulli distribution**. The binomial distribution is the base for the famous binomial test of statistical importance.

Negative Binomial Distribution

In probability theory and statistics, the number of successes in a series of independent and identically distributed Bernoulli trials before a particularised number of failures happens. It is termed as the negative binomial distribution. Here the number of failures is denoted by ' r '. For instance, if we throw a dice and determine the occurrence of 1 as a failure and all non-1's as successes. Now, if we throw a dice frequently until 1 appears the third time, i.e., $r =$ three failures, then the probability distribution of the number of non-1s that arrived would be the negative binomial distribution.

Binomial Distribution Examples

As we already know, binomial distribution gives the possibility of a different set of outcomes. In

real life, the concept is used for:

- Finding the quantity of raw and used materials while making a product.
- Taking a survey of positive and negative reviews from the public for any specific product or place.
- By using the YES/ NO survey, we can check whether the number of persons views the particular channel.
- To find the number of male and female employees in an organisation.
- The number of votes collected by a candidate in an election is counted based on 0 or 1 probability.

Binomial Distribution Formula

The binomial distribution formula is for any random variable X, given by;

$$P(x:n,p) = {}^nC_x p^x (1-p)^{n-x}$$

Or

$$P(x:n,p) = {}^nC_x p^x (q)^{n-x}$$

Where,

n = the number of experiments

x = 0, 1, 2, 3, 4, ...

p = Probability of Success in a single experiment

q = Probability of Failure in a single experiment = 1 – p

The binomial distribution formula can also be written in the form of n-Bernoulli trials,

where ${}^nC_x = \frac{n!}{x!(n-x)!}$. Hence,

$$P(x:n,p) = \frac{n!}{x!(n-x)!} \cdot p^x \cdot (q)^{n-x}$$

Binomial Distribution Mean and Variance

For a binomial distribution, the mean, variance and standard deviation for the given number of success are represented using the formulas

Mean, $\mu = np$

Variance, $\sigma^2 = npq$

Variance, $\sigma^2 = npq$

Standard Deviation $\sigma = \sqrt{npq}$

Where p is the probability of success

q is the probability of failure, where $q = 1-p$

Binomial Distribution Vs Normal Distribution

The main difference between the binomial distribution and the normal distribution is that binomial distribution is discrete, whereas the normal distribution is continuous. It means that the binomial distribution has a finite amount of events, whereas the normal distribution has an infinite number of events. In case, if the sample size for the binomial distribution is very large, then the distribution curve for the binomial distribution is similar to the normal distribution curve.

Properties of Binomial Distribution

The properties of the binomial distribution are:

- There are two possible outcomes: true or false, success or failure, yes or no.
- There is 'n' number of independent trials or a fixed number of n times repeated trials.
- The probability of success or failure remains the same for each trial.
- Only the number of success is calculated out of n independent trials.
- Every trial is an independent trial, which means the outcome of one trial does not affect the outcome of another trial.

Binomial Distribution Examples And Solutions

Example 1: If a coin is tossed 5 times, find the probability of:

(a) Exactly 2 heads

(b) At least 4 heads.

Solution:

(a) The repeated tossing of the coin is an example of a Bernoulli trial. According to the problem:

Number of trials: $n=5$

Probability of head: $p= 1/2$ and hence the probability of tail, $q = 1/2$

For exactly two heads:

$$x=2$$

$$P(x=2) = {}^5C2 p^2 q^{5-2} = 5! / 2! 3! \times (\frac{1}{2})^2 \times (\frac{1}{2})^3$$

$$P(x=2) = 5/16$$

(b) For at least four heads,

$$x \geq 4, P(x \geq 4) = P(x = 4) + P(x=5)$$

Hence,

$$P(x = 4) = {}^5C4 p^4 q^{5-4} = 5!/4! 1! \times (\frac{1}{2})^4 \times (\frac{1}{2})^1 = 5/32$$

$$P(x = 5) = {}^5C5 p^5 q^{5-5} = (\frac{1}{2})^5 = 1/32$$

Therefore,

$$P(x \geq 4) = 5/32 + 1/32 = 6/32 = 3/16$$

Example 2: For the same question given above, find the probability of:

a) Getting at most 2 heads

Solution: $P(\text{at most 2 heads}) = P(X \leq 2) = P(X = 0) + P(X = 1) + P(X = 2)$

$$P(X = 0) = (\frac{1}{2})^5 = 1/32$$

$$P(X=1) = {}^5C1 (\frac{1}{2})^5 = 5/32$$

$$P(x=2) = {}^5C2 p^2 q^{5-2} = 5! / 2! 3! \times (\frac{1}{2})^2 \times (\frac{1}{2})^3 = 5/16$$

Therefore,

$$P(X \leq 2) = 1/32 + 5/32 + 5/16 = \frac{1}{2}$$

Example 3:

A fair coin is tossed 10 times, what are the probability of getting exactly 6 heads and at least six heads.

Solution:

Let x denote the number of heads in an experiment.

Here, the number of times the coin tossed is 10. Hence, $n=10$.

The probability of getting head, $p = \frac{1}{2}$

The probability of getting a tail, $q = 1-p = 1-\left(\frac{1}{2}\right) = \frac{1}{2}$.

The binomial distribution is given by the formula:

$$P(X=x) = {}^nC_x p^x q^{n-x}, \text{ where } x = 0, 1, 2, 3, \dots$$

$$\text{Therefore, } P(X=x) = {}^{10}C_x \left(\frac{1}{2}\right)^x \left(\frac{1}{2}\right)^{10-x}$$

(i) The probability of getting exactly 6 heads is:

$$P(X=6) = {}^{10}C_6 \left(\frac{1}{2}\right)^6 \left(\frac{1}{2}\right)^{10-6}$$

$$P(X=6) = {}^{10}C_6 \left(\frac{1}{2}\right)^{10}$$

$$P(X=6) = 105/512.$$

Hence, the probability of getting exactly 6 heads is 105/512.

(ii) The probability of getting at least 6 heads is $P(X \geq 6)$

$$P(X \geq 6) = P(X=6) + P(X=7) + P(X=8) + P(X=9) + P(X=10)$$

$$P(X \geq 6) = {}^{10}C_6 \left(\frac{1}{2}\right)^{10} + {}^{10}C_7 \left(\frac{1}{2}\right)^{10} + {}^{10}C_8 \left(\frac{1}{2}\right)^{10} + {}^{10}C_9 \left(\frac{1}{2}\right)^{10} + {}^{10}C_{10} \left(\frac{1}{2}\right)^{10}$$

$$P(X \geq 6) = 193/512.$$

Example 4 :

In a box of floppy discs it is known that 95% will work. A sample of three of the discs is selected at random.

Find the probability that (a) none (b) 1, (c) 2, (d) all 3 of the sample will work.

Solution

Let the event {the disc works} be W and the event {the disc fails} be F . The probability that a disc will work is denoted by $P(W)$ and the probability that a disc will fail is denoted by $P(F)$. Then $P(W) = 0.95$ and $P(F) = 1 - P(W) = 1 - 0.95 = 0.05$.

- (a) The probability that none of the discs works equals the probability that all 3 discs fail. This is given by:

$$\begin{aligned}P(\text{none work}) &= P(FFF) = P(F) \times P(F) \times P(F) \quad \text{as the events are independent} \\&= 0.05 \times 0.05 \times 0.05 = 0.05^3 = 0.000125\end{aligned}$$

- (b) If only one disc works then you could select the three discs in the following orders

(FFW) or (FWF) or (WFF) hence

$$\begin{aligned}P(\text{one works}) &= P(FFW) + P(FWF) + P(WFF) \\&= P(F) \times P(F) \times P(W) + P(F) \times P(W) \times P(F) + P(W) \times P(F) \times P(F) \\&= (0.05 \times 0.05 \times 0.95) + (0.05 \times 0.95 \times 0.05) + (0.95 \times 0.05 \times 0.05) \\&= 3 \times (0.05)^2 \times 0.95 = 0.007125\end{aligned}$$

- (c) If 2 discs work you could select them in order

(FWW) or (WFW) or (WWF) hence

$$\begin{aligned}P(\text{two work}) &= P(FWW) + P(WFW) + P(WWF) \\&= P(F) \times P(W) \times P(W) + P(W) \times P(F) \times P(W) + P(W) \times P(W) \times P(F) \\&= (0.05 \times 0.95 \times 0.95) + (0.95 \times 0.05 \times 0.95) + (0.95 \times 0.95 \times 0.05) \\&= 3 \times (0.05) \times (0.95)^2 = 0.135375\end{aligned}$$

- (d) The probability that all 3 discs work is given by $P(WWW) = 0.95^3 = 0.857375$.

Notice that since the 4 outcomes we have dealt with are *all possible outcomes* of selecting 3 discs, the probabilities should add up to 1. It is an easy check to verify that they do.

One of the most important assumptions above is that of **independence**. The probability of selecting a working disc remains unchanged no matter whether the previous selected disc worked or not.

Normal distribution

In probability theory and statistics, the Normal Distribution, also called the Gaussian Distribution, is the most significant continuous probability distribution. Sometimes it is also called a bell curve. A large number of random variables are either nearly or exactly represented by the normal distribution, in every physical science and economics. Furthermore, it can be used to approximate other probability distributions, therefore supporting the usage of the word ‘normal’ as in about the one, mostly used.

Normal Distribution Definition

The Normal Distribution is defined by the probability density function for a continuous random variable in a system. Let us say, $f(x)$ is the probability density function and X is the random variable. Hence, it defines a function which is integrated between the range or interval (x to $x + dx$), giving the probability of random variable X , by considering the values between x and $x+dx$.

$$f(x) \geq 0 \quad \forall x \in (-\infty, +\infty)$$

$$\text{And } \int_{-\infty}^{+\infty} f(x) = 1$$

Normal Distribution Formula

The probability density function of normal or gaussian distribution is given by;

$$f(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

Where,

- x is the variable
- μ is the mean
- σ is the standard deviation

Normal Distribution Curve

The random variables following the normal distribution are those whose values can find any unknown value in a given range. For example, finding the height of the students in the school. Here, the distribution can consider any value, but it will be bounded in the range say, 0 to 6ft. This limitation is forced physically in our query.

Whereas, the normal distribution doesn’t even bother about the range. The range can also extend to $-\infty$ to $+\infty$ and still we can find a smooth curve. These random variables are called Continuous Variables, and the Normal Distribution then provides here probability of the value lying in a

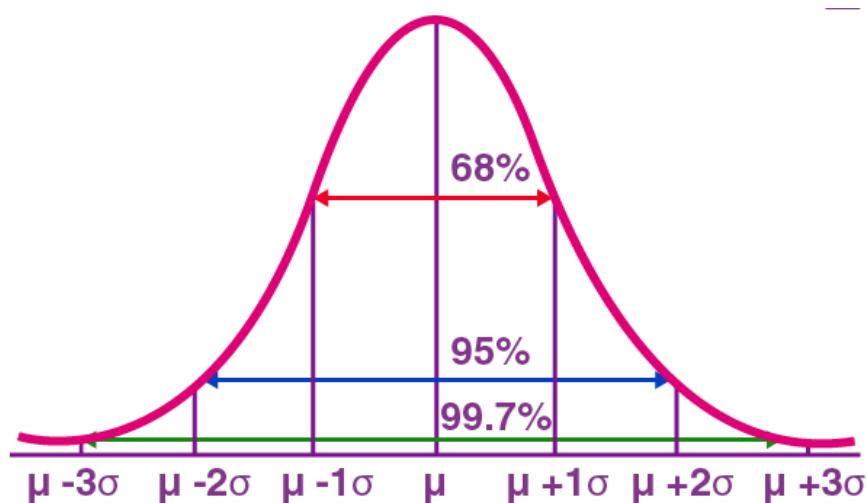
particular range for a given experiment. Also, use the [normal distribution calculator](#) to find the probability density function by just providing the mean and standard deviation value.

Normal Distribution Standard Deviation

Generally, the normal distribution has any positive standard deviation. We know that the mean helps to determine the line of symmetry of a graph, whereas the standard deviation helps to know how far the data are spread out. If the standard deviation is smaller, the data are somewhat close to each other, and the graph becomes narrower. If the standard deviation is larger, the data are dispersed more, and the graph becomes wider. The standard deviations are used to subdivide the area under the normal curve. Each subdivided section defines the percentage of data, which falls into the specific region of a graph.

Using 1 standard deviation, the Empirical Rule states that,

- Approximately 68% of the data falls within one standard deviation of the mean. (i.e., Between Mean- one Standard Deviation and Mean + one standard deviation)
- Approximately 95% of the data falls within two standard deviations of the mean. (i.e., Between Mean- two Standard Deviation and Mean + two standard deviations)
- Approximately 99.7% of the data fall within three standard deviations of the mean. (i.e., Between Mean- three Standard Deviation and Mean + three standard deviations)

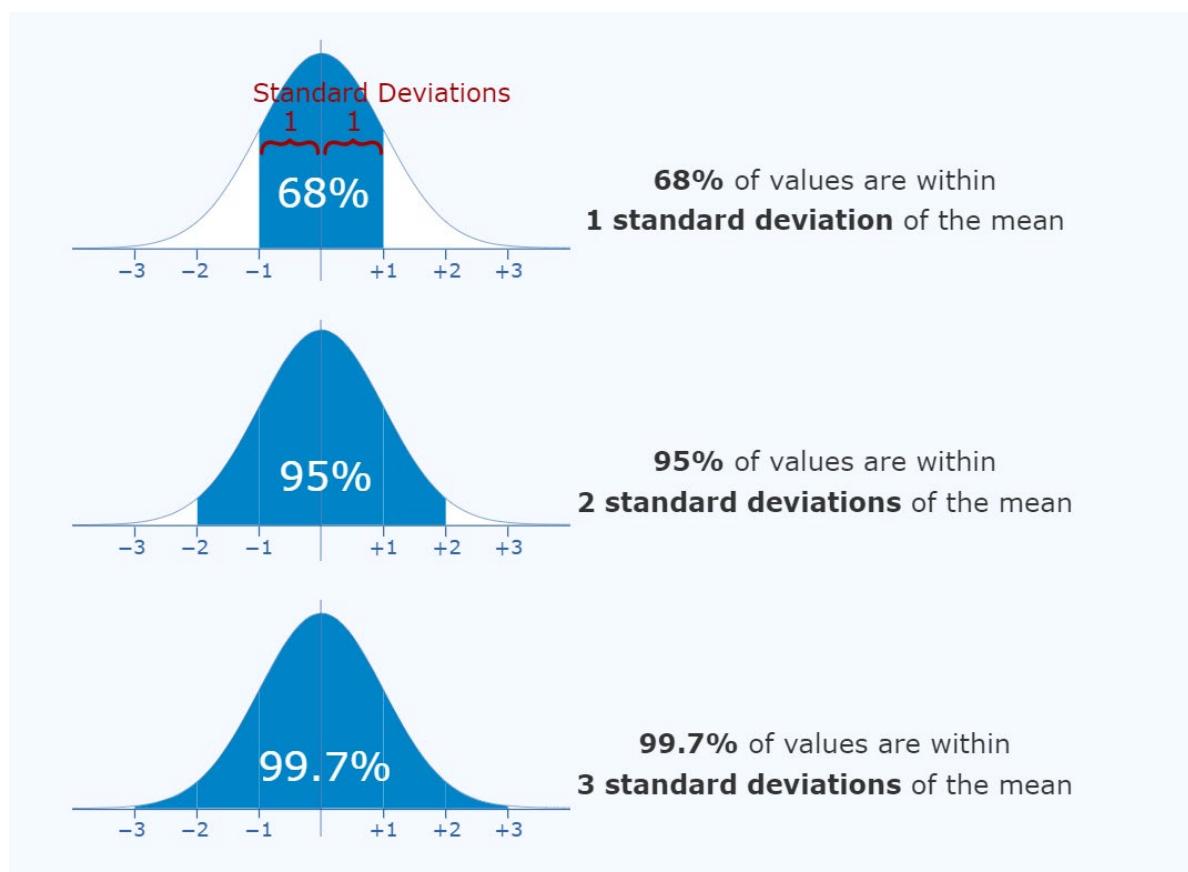


Thus, the empirical rule is also called the 68 – 95 – 99.7 rule.

Example: Using the empirical rule in a normal distribution - You collect SAT scores from students in a new test preparation course. The data follows a normal distribution with a mean score (M) of 1150 and a standard deviation (SD) of 150.

Following the empirical rule:

- Around 68% of scores are between 1,000 and 1,300, 1 standard deviation above and below the mean.
- Around 95% of scores are between 850 and 1,450, 2 standard deviations above and below the mean.
- Around 99.7% of scores are between 700 and 1,600, 3 standard deviations above and below the mean.



The empirical rule is a quick way to get an overview of your data and check for any outliers or extreme values that don't follow this pattern.

If data from small samples do not closely follow this pattern, then other distributions like the t-distribution may be more appropriate. Once you identify the distribution of your variable, you can apply appropriate statistical tests.

Central limit theorem

The central limit theorem is the basis for how normal distributions work in statistics.

In research, to get a good idea of a population mean, ideally you'd collect data from multiple random samples within the population. A **sampling distribution of the mean** is the

distribution of the means of these different samples.

The central limit theorem shows the following:

- Law of Large Numbers: As you increase sample size (or the number of samples), then the sample mean will approach the population mean.
- With multiple large samples, the sampling distribution of the mean is normally distributed, even if your original variable is not normally distributed.

Parametric statistical tests typically assume that samples come from normally distributed populations, but the central limit theorem means that this assumption isn't necessary to meet when you have a large enough sample.

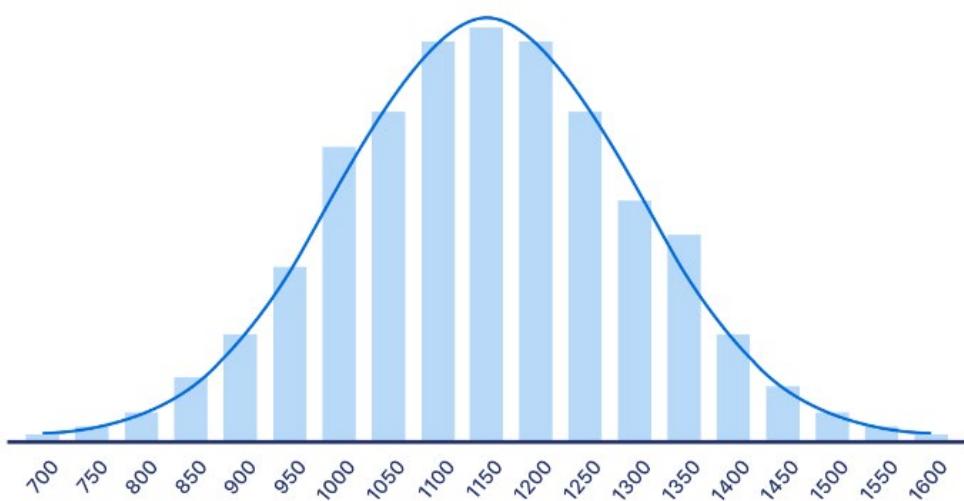
You can use parametric tests for large samples from populations with any kind of distribution as long as other important assumptions are met. A sample size of 30 or more is generally considered large.

For small samples, the assumption of normality is important because the sampling distribution of the mean isn't known. For accurate results, you have to be sure that the population is normally distributed before you can use parametric tests with small samples.

Formula of the normal curve

Once you have the mean and standard deviation of a normal distribution, you can fit a normal curve to your data using a **probability density function**.

Normal curve fitted to SAT score data



In a probability density function, the area under the curve tells you probability. The normal distribution is a probability distribution, so the total area under the curve is always 1 or 100%.

The formula for the normal probability density function looks fairly complicated. But to use it, you only need to know the population mean and standard deviation.

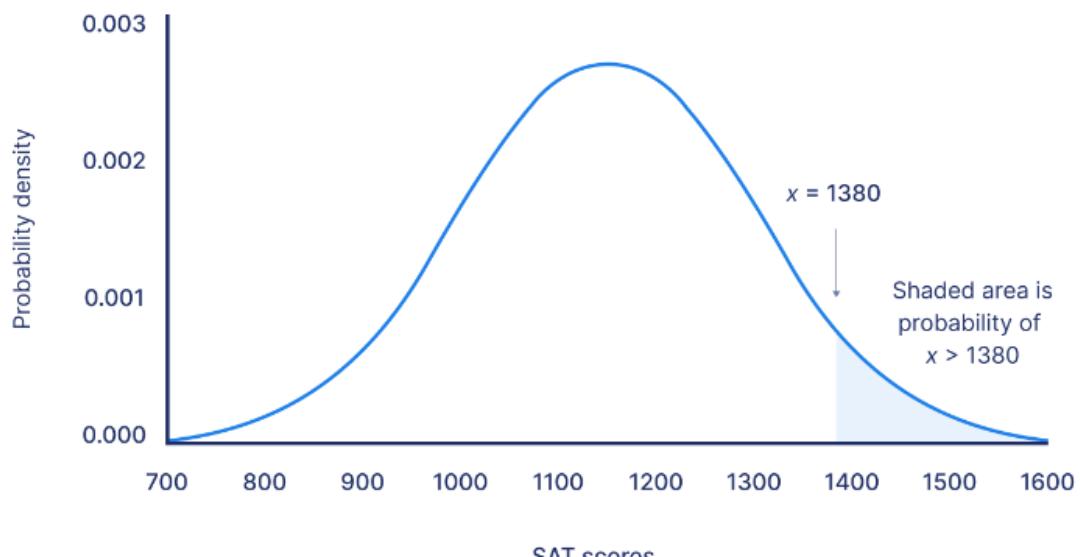
For any value of x , you can plug in the mean and standard deviation into the formula to find the probability density of the variable taking on that value of x .

Normal probability density formula	Explanation
$f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$	<ul style="list-style-type: none"> • $f(x)$ = probability • x = value of the variable • μ = mean • σ = standard deviation • σ^2 = variance

Example: Using the probability density function You want to know the probability that SAT scores in your sample exceed 1380.

On your graph of the probability density function, the probability is the shaded area under the curve that lies to the right of where your SAT scores equal 1380.

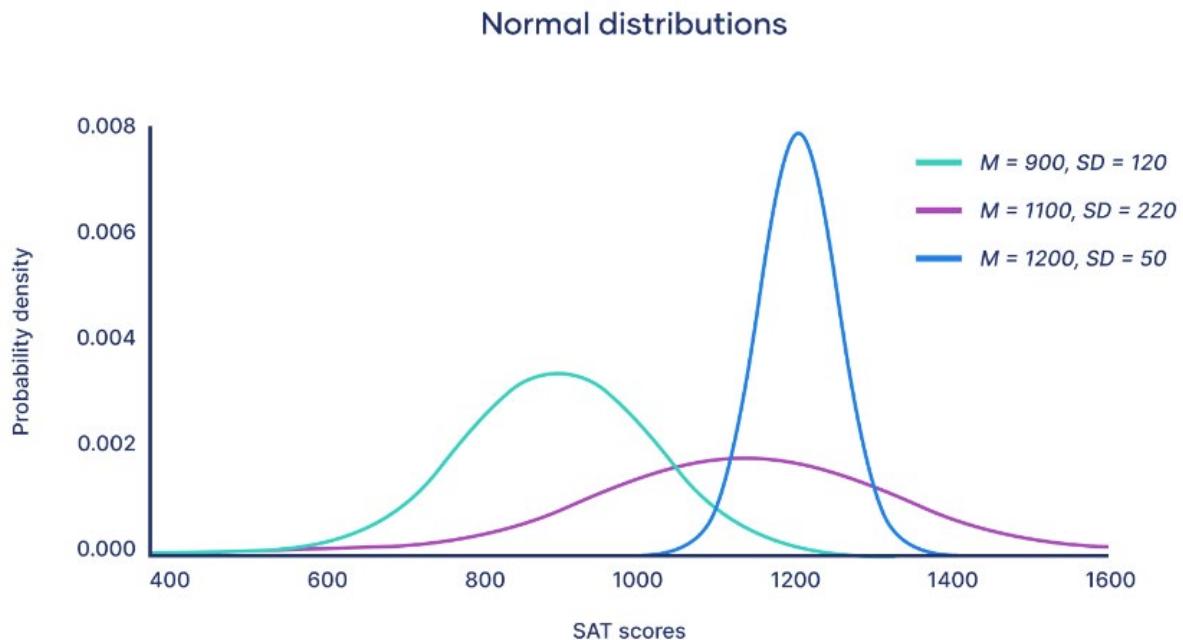
Probability density function of SAT scores



What is the standard normal distribution?

The standard normal distribution, also called the z -distribution, is a special normal distribution where the mean is 0 and the standard deviation is 1.

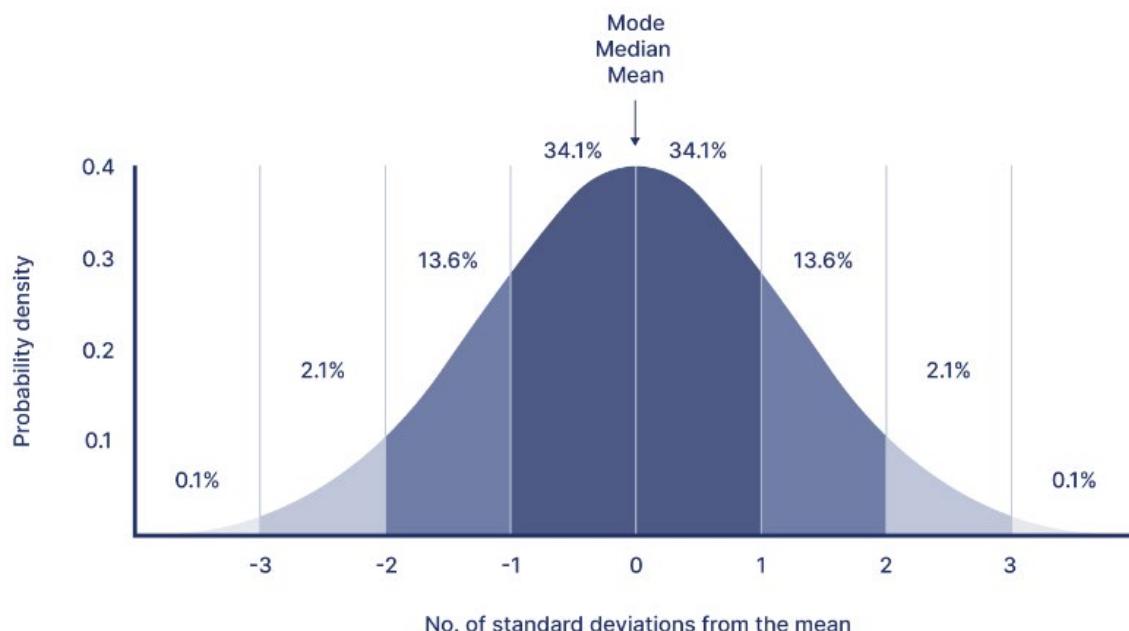
Every normal distribution is a version of the standard normal distribution that's been stretched or squeezed and moved horizontally right or left.



While individual observations from normal distributions are referred to as x , they are referred to as z in the z -distribution. Every normal distribution can be converted to the standard normal distribution by turning the individual values into z -scores.

Z -scores tell you how many standard deviations away from the mean each value lies.

Standard normal distribution



You only need to know the mean and standard deviation of your distribution to find the z -score

of a value.

Z-score Formula	Explanation
$z = \frac{x - \mu}{\sigma}$	<ul style="list-style-type: none">• x = individual value• μ = mean• σ = standard deviation

We convert normal distributions into the standard normal distribution for several reasons:

- To find the probability of observations in a distribution falling above or below a given value.
- To find the probability that a sample mean significantly differs from a known population mean.
- To compare scores on different distributions with different means and standard deviations.

Finding probability using the *z*-distribution

Each *z*-score is associated with a probability, or *p*-value, that tells you the likelihood of values below that *z*-score occurring. If you convert an individual value into a *z*-score, you can then find the probability of all values up to that value occurring in a normal distribution.

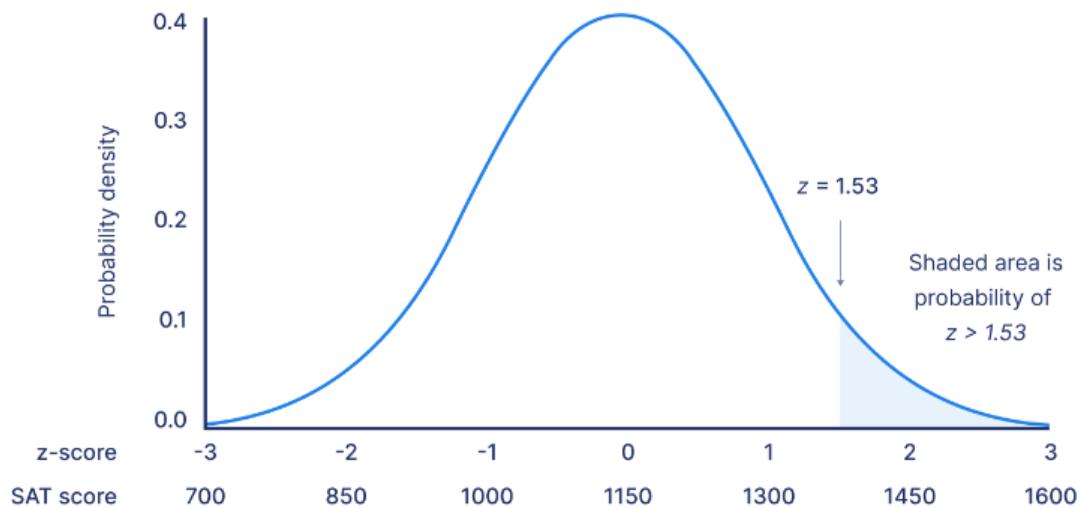
Example: Finding probability using the *z*-distribution To find the probability of SAT scores in your sample exceeding 1380, you first find the *z*-score.

The mean of our distribution is 1150, and the standard deviation is 150. The *z*-score tells you how many standard deviations away 1380 is from the mean.

Formula	Calculation
$z = \frac{x - \mu}{\sigma}$	$z = \frac{1380 - 1150}{150}$ $z = 1.53$

For a *z*-score of 1.53, the *p*-value is 0.937. This is the probability of SAT scores being 1380 or less (93.7%), and it's the area under the curve left of the shaded area.

Standard normal distribution



To find the shaded area, you take away 0.937 from 1, which is the total area under the curve.

$$\text{Probability of } x > 1380 = 1 - 0.937 = \mathbf{0.063}$$

That means it is likely that only 6.3% of SAT scores in your sample exceed 1380.

Normal Distribution Table

The table here shows the area from 0 to Z-value.

Z-Value	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.0000	0.0040	0.0080	0.0120	0.0160	0.0199	0.0239	0.0279	0.0319	0.0359
0.1	0.0398	0.0438	0.0478	0.0517	0.0557	0.0596	0.0636	0.0675	0.0714	0.0753
0.2	0.0793	0.0832	0.0871	0.0910	0.0948	0.0987	0.1026	0.1064	0.1103	0.1141
0.3	0.1179	0.1217	0.1255	0.1293	0.1331	0.1368	0.1406	0.1443	0.1480	0.1517
0.4	0.1554	0.1591	0.1628	0.1664	0.1700	0.1736	0.1772	0.1808	0.1844	0.1879
0.5	0.1915	0.1950	0.1985	0.2019	0.2054	0.2088	0.2123	0.2157	0.2190	0.2224
0.6	0.2257	0.2291	0.2324	0.2357	0.2389	0.2422	0.2454	0.2486	0.2517	0.2549
0.7	0.2580	0.2611	0.2642	0.2673	0.2704	0.2734	0.2764	0.2794	0.2823	0.2852
0.8	0.2881	0.2910	0.2939	0.2967	0.2995	0.3023	0.3051	0.3078	0.3106	0.3133
0.9	0.3159	0.3186	0.3212	0.3238	0.3264	0.3289	0.3315	0.3340	0.3365	0.3389
1.0	0.3413	0.3438	0.3461	0.3485	0.3508	0.3531	0.3554	0.3577	0.3599	0.3621
1.1	0.3643	0.3665	0.3686	0.3708	0.3729	0.3749	0.3770	0.3790	0.3810	0.3830
1.2	0.3849	0.3869	0.3888	0.3907	0.3925	0.3944	0.3962	0.3980	0.3997	0.4015
1.3	0.4032	0.4049	0.4066	0.4082	0.4099	0.4115	0.4131	0.4147	0.4162	0.4177
1.4	0.4192	0.4207	0.4222	0.4236	0.4251	0.4265	0.4279	0.4292	0.4306	0.4319
1.5	0.4332	0.4345	0.4357	0.4370	0.4382	0.4394	0.4406	0.4418	0.4429	0.4441
1.6	0.4452	0.4463	0.4474	0.4484	0.4495	0.4505	0.4515	0.4525	0.4535	0.4545
1.7	0.4554	0.4564	0.4573	0.4582	0.4591	0.4599	0.4608	0.4616	0.4625	0.4633
1.8	0.4641	0.4649	0.4656	0.4664	0.4671	0.4678	0.4686	0.4693	0.4699	0.4706
1.9	0.4713	0.4719	0.4726	0.4732	0.4738	0.4744	0.4750	0.4756	0.4761	0.4767
2.0	0.4772	0.4778	0.4783	0.4788	0.4793	0.4798	0.4803	0.4808	0.4812	0.4817
2.1	0.4821	0.4826	0.4830	0.4834	0.4838	0.4842	0.4846	0.4850	0.4854	0.4857
2.2	0.4861	0.4864	0.4868	0.4871	0.4875	0.4878	0.4881	0.4884	0.4887	0.4890
2.3	0.4893	0.4896	0.4898	0.4901	0.4904	0.4906	0.4909	0.4911	0.4913	0.4916
2.4	0.4918	0.4920	0.4922	0.4925	0.4927	0.4929	0.4931	0.4932	0.4934	0.4936
2.5	0.4938	0.4940	0.4941	0.4943	0.4945	0.4946	0.4948	0.4949	0.4951	0.4952
2.6	0.4953	0.4955	0.4956	0.4957	0.4959	0.4960	0.4961	0.4962	0.4963	0.4964
2.7	0.4965	0.4966	0.4967	0.4968	0.4969	0.4970	0.4971	0.4972	0.4973	0.4974
2.8	0.4974	0.4975	0.4976	0.4977	0.4977	0.4978	0.4979	0.4979	0.4980	0.4981
2.9	0.4981	0.4982	0.4982	0.4983	0.4984	0.4984	0.4985	0.4985	0.4986	0.4986
3.0	0.4987	0.4987	0.4987	0.4988	0.4988	0.4989	0.4989	0.4989	0.4990	0.4990

Normal Distribution Problems and Solutions

Question 1: Calculate the probability density function of normal distribution using the following data. $x = 3$, $\mu = 4$ and $\sigma = 2$.

Solution: Given, variable, $x = 3$

Mean = 4 and

Standard deviation = 2

By the formula of the probability density of normal distribution, we can write;

$$f(3, 4, 2) = \frac{1}{2\sqrt{2\pi}} e^{\frac{-(3-2)^2}{2 \times 2^2}}$$

Hence, $f(3, 4, 2) = 1.106$.

Question 2: If the value of random variable is 2, mean is 5 and the standard deviation is 4, then find the probability density function of the gaussian distribution.

Solution: Given,

Variable, $x = 2$

Mean = 5 and

Standard deviation = 4

By the formula of the probability density of normal distribution, we can write;

$$f(2, 2, 4) = \frac{1}{4\sqrt{2\pi}} e^{\frac{-(2-2)^2}{2 \times 4^2}}$$

$$f(2, 2, 4) = 1/(4\sqrt{2\pi}) e^0$$

$$f(2, 2, 4) = 0.0997$$

There are two main parameters of normal distribution in statistics namely mean and standard deviation. The location and scale parameters of the given normal distribution can be estimated using these two parameters.

Normal Distribution Properties

Some of the important properties of the normal distribution are listed below:

- In a normal distribution, the mean, median and mode are equal.(i.e., Mean = Median= Mode).
- The total area under the curve should be equal to 1.
- The normally distributed curve should be symmetric at the centre.
- There should be exactly half of the values are to the right of the centre and exactly half of the values are to the left of the centre.
- The normal distribution should be defined by the mean and standard deviation.
- The normal distribution curve must have only one peak. (i.e., Unimodal)
- The curve approaches the x-axis, but it never touches, and it extends farther away from the mean.

Applications

The normal distributions are closely associated with many things such as:

- Marks scored on the test
- Heights of different persons
- Size of objects produced by the machine
- Blood pressure and so on.

Examples

Example: 95% of students at school are between **1.1m and 1.7m tall.**

Assuming this data is **normally distributed** can you calculate the mean and standard deviation?

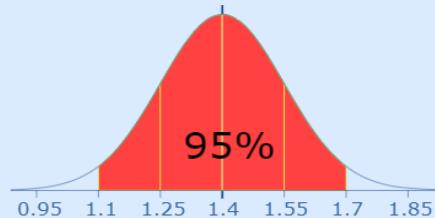
The mean is halfway between 1.1m and 1.7m:

$$\text{Mean} = (1.1\text{m} + 1.7\text{m}) / 2 = \mathbf{1.4\text{m}}$$

95% is 2 standard deviations either side of the mean (a total of 4 standard deviations) so:

$$\begin{aligned}\text{1 standard deviation} &= (1.7\text{m}-1.1\text{m}) / 4 \\ &= 0.6\text{m} / 4 \\ &= \mathbf{0.15\text{m}}\end{aligned}$$

And this is the result:

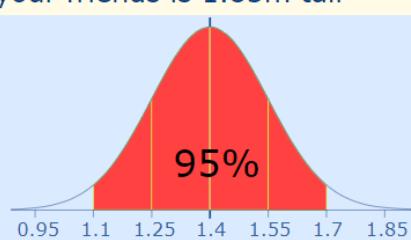


It is good to know the standard deviation, because we can say that any value is:

- **likely** to be within 1 standard deviation (68 out of 100 should be)
- **very likely** to be within 2 standard deviations (95 out of 100 should be)
- **almost certainly** within 3 standard deviations (997 out of 1000 should be)

Example: In that same school one of your friends is 1.85m tall

You can see on the bell curve that 1.85m is **3 standard deviations** from the mean of 1.4, so:



Your friend's height has a "z-score" of 3.0

It is also possible to **calculate** how many standard deviations 1.85 is from the mean

How far is 1.85 from the mean?

$$\text{It is } 1.85 - 1.4 = \mathbf{0.45\text{m from the mean}}$$

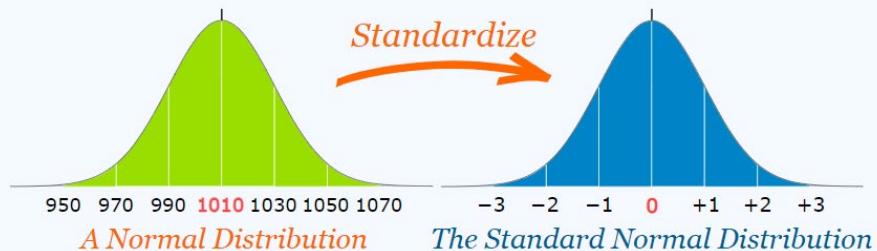
How many standard deviations is that? The standard deviation is 0.15m, so:

$$0.45\text{m} / 0.15\text{m} = \mathbf{3 \text{ standard deviations}}$$

So to convert a value to a Standard Score ("z-score"):

- first subtract the mean,
- then divide by the Standard Deviation

And doing that is called "Standardizing":



We can take any Normal Distribution and convert it to The Standard Normal Distribution.

Example: Travel Time

A survey of daily travel time had these results (in minutes):

26, 33, 65, 28, 34, 55, 25, 44, 50, 36, 26, 37, 43, 62, 35, 38, 45, 32, 28, 34

The **Mean is 38.8 minutes**, and the **Standard Deviation is 11.4 minutes** (you can copy and paste the values into the [Standard Deviation Calculator](#) if you want).

Convert the values to z-scores ("standard scores").

To convert **26**:

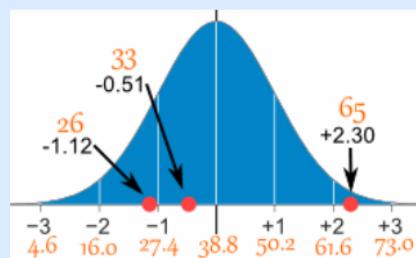
- first subtract the mean: $26 - 38.8 = -12.8$,
- then divide by the Standard Deviation: $-12.8 / 11.4 = -1.12$

So **26** is **-1.12 Standard Deviations** from the Mean

Here are the first three conversions

Original Value	Calculation	Standard Score (z-score)
26	$(26-38.8) / 11.4 =$	-1.12
33	$(33-38.8) / 11.4 =$	-0.51
65	$(65-38.8) / 11.4 =$	+2.30
...

And here they are graphically:



You can calculate the rest of the z-scores yourself!

Example: Travel Time (continued)

Here are the first three conversions using the "z-score formula":

$$z = \frac{x - \mu}{\sigma}$$

- $\mu = 38.8$
- $\sigma = 11.4$

x	$\frac{x - \mu}{\sigma}$	z (z-score)
26	$\frac{26 - 38.8}{11.4}$	= -1.12
33	$\frac{33 - 38.8}{11.4}$	= -0.51
65	$\frac{65 - 38.8}{11.4}$	= +2.30
...

The exact calculations we did before, just following the formula.

Poisson distribution

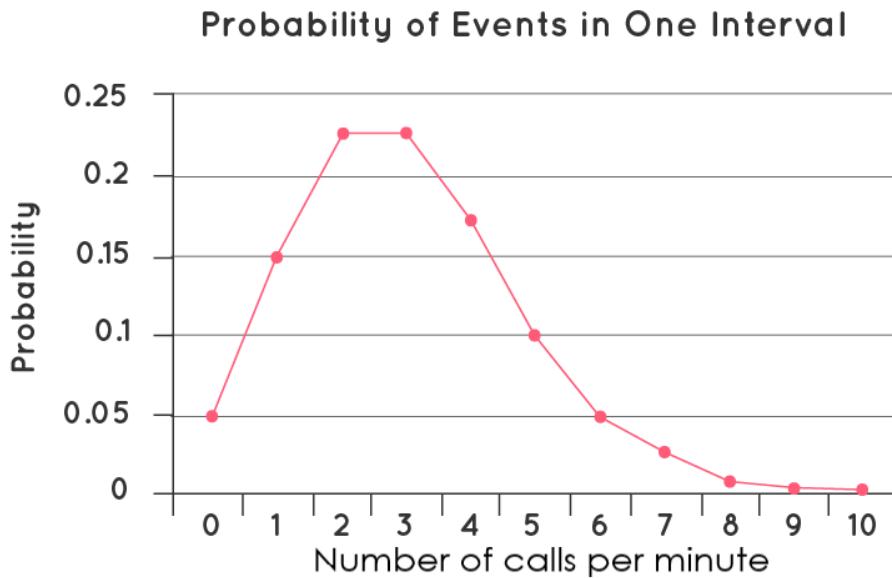
Poisson distribution is a theoretical discrete probability and is also known as the Poisson distribution probability mass function. It is used to find the probability of an independent event that is occurring in a fixed interval of time and has a constant mean rate. The Poisson distribution probability mass function can also be used in other fixed intervals such as volume, area, distance, etc. A Poisson random variable will relatively describe a phenomenon if there are few successes over many trials. The Poisson distribution is used as a limiting case of the binomial distribution when the trials are large indefinitely. If a Poisson distribution models the same binomial phenomenon, λ is replaced by np . Poisson distribution is named after the French mathematician Denis Poisson.

What is Poisson Distribution?

Poisson distribution definition is used to model a discrete probability of an event where independent events are occurring in a fixed interval of time and have a known constant mean rate. In other words, Poisson distribution is used to estimate how many times an event is likely to occur within the given period of time. λ is the Poisson rate parameter that indicates the expected value of the average number of events in the fixed time interval. Poisson distribution has wide use in the fields of business as well as in biology.

Let us try and understand this with an example, customer care center receives 100 calls per hour, 8 hours a day. As we can see that the calls are independent of each other. The probability of the number of calls per minute has a Poisson probability distribution. There can be any number of calls per minute irrespective of the number of calls received in the previous minute. Below is the curve of the probabilities for a fixed value of λ of a function following Poisson distribution:

Probability Mass Function for Poisson distribution



If we are to find the probability that more than 150 calls could be received per hour, the call center could improve its standards on customer care by employing more services and catering to the needs of its customers, based on the understanding of the Poisson distribution.

Poisson Distribution Formula

Poisson distribution formula is used to find the probability of an event that happens independently, discretely over a fixed time period, when the mean rate of occurrence is constant over time. The Poisson distribution formula is applied when there is a large number of possible outcomes. For a random discrete variable X that follows the Poisson distribution, and λ is the average rate of value, then the probability of x is given by:

$$f(x) = P(X=x) = (e^{-\lambda} \lambda^x)/x!$$

Where

- $x = 0, 1, 2, 3\dots$
- e is the Euler's number ($e = 2.718$)
- λ is an average rate of the expected value and $\lambda = \text{variance}$, also $\lambda > 0$

Poisson Distribution Mean and Variance

For Poisson distribution, which has λ as the average rate, for a fixed interval of time, then the mean of the Poisson distribution and the value of variance will be the same. So for X following Poisson distribution, we can say that λ is the mean as well as the variance of the distribution.

Hence: $E(X) = V(X) = \lambda$

where

- $E(X)$ is the expected mean
- $V(X)$ is the variance
- $\lambda > 0$
-

Properties of Poisson Distribution

The Poisson distribution is applicable in events that have a large number of rare and independent possible events. The following are the properties of the Poisson Distribution. In the Poisson distribution,

- The events are independent.
- The average number of successes in the given period of time alone can occur. No two events can occur at the same time.
- The Poisson distribution is limited when the number of trials n is indefinitely large.
- mean = variance = λ
- $np = \lambda$ is finite, where λ is constant.
- The standard deviation is always equal to the square root of the mean μ .
- The exact probability that the random variable X with mean $\mu = a$ is given by $P(X=a) = \frac{\mu^a}{a!} e^{-\mu}$
- If the mean is large, then the Poisson distribution is approximately a normal distribution.

Poisson Distribution Table

Similar to the binomial distribution, we can have a Poisson distribution table which will help us to quickly find the probability mass function of an event that follows the Poisson distribution. The Poisson distribution table shows different values of Poisson distribution for various values of λ , where $\lambda > 0$. Here in the table given below, we can see that, for $P(X=0)$ and $\lambda = 0.5$, the value of the probability mass function is 0.6065 or 60.65%.

x	.1	.2	.3	.4	.5	.6	.7	.8	.9	1.0
0	.9048	.8187	.7408	.6703	.6065	.5488	.4966	.4493	.4066	.3679
1	.0905	.1637	.2222	.2681	.3033	.3293	.3476	.3595	.3659	.3679
2	.0045	.0164	.0333	.0536	.0758	.0988	.1217	.1438	.1647	.1839
3	.0002	.0011	.0033	.0072	.0126	.0198	.0284	.0383	.0494	.0613
4	.0000	.0001	.0003	.0007	.0016	.0030	.0050	.0077	.0111	.0153
5	.0000	.0000	.0000	.0001	.0002	.0004	.0007	.0012	.0020	.0031
6	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0002	.0003	.0005
7	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0001

x	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
0	.3329	.3012	.2725	.2466	.2231	.2019	.1827	.1653	.1496	.1353
1	.3662	.3614	.3543	.3452	.3347	.3230	.3106	.2975	.2842	.2707
2	.2014	.2169	.2303	.2417	.2510	.2584	.2640	.2678	.2700	.2707
3	.0738	.0867	.0998	.1128	.1255	.1378	.1496	.1607	.1710	.1804
4	.0203	.0260	.0324	.0395	.0471	.0551	.0636	.0723	.0812	.0902
5	.0045	.0062	.0084	.0111	.0141	.0176	.0216	.0260	.0309	.0361
6	.0008	.0012	.0018	.0026	.0035	.0047	.0061	.0078	.0098	.0120
7	.0001	.0002	.0003	.0005	.0008	.0011	.0015	.0020	.0027	.0034
8	.0000	.0000	.0001	.0001	.0001	.0002	.0003	.0005	.0006	.0009
9	.0000	.0000	.0000	.0000	.0000	.0000	.0001	.0001	.0001	.0002

Applications of Poisson Distribution

There are various applications of the Poisson distribution. The random variables that follow a Poisson distribution are as follows:

- To count the number of defects of a finished product
- To count the number of deaths in a country by any disease or natural calamity
- To count the number of infected plants in the field
- To count the number of bacteria in the organisms or the radioactive decay in atoms
- To calculate the waiting time between the events.

Important Notes

- The formula for Poisson distribution is $f(x) = P(X=x) = (e^{-\lambda} \lambda^x)/x!$.
- For the Poisson distribution, λ is always greater than 0.
- For Poisson distribution, the mean and the variance of the distribution are equal.

Poisson Distribution Examples

An example to find the probability using the Poisson distribution is given below:

Example 1:

A random variable X has a Poisson distribution with parameter λ such that $P(X = 1) = (0.2) P(X = 2)$. Find $P(X = 0)$.

Solution:

For the Poisson distribution, the probability function is defined as:

$$P(X=x) = (e^{-\lambda} \lambda^x)/x!, \text{ where } \lambda \text{ is a parameter.}$$

$$\text{Given that, } P(x=1) = (0.2) P(X=2)$$

$$(e^{-\lambda} \lambda^1)/1! = (0.2)(e^{-\lambda} \lambda^2)/2!$$

$$\Rightarrow \lambda = \lambda^2 / 10$$

$$\Rightarrow \lambda = 10$$

Now, substitute $\lambda = 10$, in the formula, we get:

$$P(X=0) = (e^{-\lambda} \lambda^0)/0!$$

$$P(X=0) = e^{-10} = 0.0000454$$

$$\text{Thus, } P(X=0) = 0.0000454$$

Example 2:

Telephone calls arrive at an exchange according to the Poisson process at a rate $\lambda = 2/\text{min}$. Calculate the probability that exactly two calls will be received during each of the first 5 minutes of the hour.

Solution:

Assume that “N” be the number of calls received during a 1 minute period.

Therefore,

$$P(N=2) = (e^{-2} \cdot 2^2)/2!$$

$$P(N=2) = 2e^{-2}$$

Now, “M” be the number of minutes among 5 minutes considered, during which exactly 2 calls will be received. Thus “M” follows a binomial distribution with parameters $n=5$ and $p=2e^{-2}$.

$$P(M=5) = 32 \times e^{-10}$$

$$P(M=5) = 0.00145, \text{ where "e" is a constant, which is approximately equal to 2.718.}$$

Poisson Distribution Examples

Example 1: In a cafe, the customer arrives at a mean rate of 2 per min. Find the probability of arrival of 5 customers in 1 minute using the Poisson distribution formula.

Solution:

Given: $\lambda = 2$, and $x = 5$.

Using the Poisson distribution formula:

$$P(X = x) = (e^{-\lambda} \lambda^x)/x!$$

$$P(X = 5) = (e^{-2} 2^5)/5!$$

$$P(X = 6) = 0.036$$

Answer: The probability of arrival of 5 customers per minute is 3.6%.

Example 2: Find the mass probability of function at $x = 6$, if the value of the mean is 3.4.

Solution:

Given: $\lambda = 3.4$, and $x = 6$.

Using the Poisson distribution formula:

$$P(X = x) = (e^{-\lambda} \lambda^x)/x!$$

$$P(X = 6) = (e^{-3.4} 3.4^6)/6!$$

$$P(X = 6) = 0.072$$

Answer: The probability of function is 7.2%.

Example 3: If 3% of electronic units manufactured by a company are defective. Find the probability that in a sample of 200 units, less than 2 bulbs are defective.

Solution:

The probability of defective units $p = 3/100 = 0.03$

Give $n = 200$.

We observe that p is small and n is large here. Thus it is a Poisson distribution.

Mean $\lambda = np = 200 \times 0.03 = 6$

$P(X = x)$ is given by the Poisson Distribution Formula as $(e^{-\lambda} \lambda^x)/x!$

$$P(X < 2) = P(X = 0) + P(X = 1)$$

$$= (e^{-6} 6^0)/0! + (e^{-6} 6^1)/1!$$

$$= e^{-6} + e^{-6} \times 6$$

$$= 0.00247 + 0.0148$$

$$P(X < 2) = 0.01727$$

Answer: The probability that less than 2 bulbs are defective is 0.01727

Random number generation

Generate one or more random numbers in your custom range from 0 to 10,000. Generate positive or negative random numbers with repeats or no repeats.

About Random Number Generators

There are two main types of random number generators: pseudo-random and true random.

A **pseudo-random number generator (PRNG)** is typically programmed using a randomizing math function to select a "random" number within a set range. These random number generators are pseudo-random because the computer program or algorithm may have unintended selection bias. In other words, randomness from a computer program is not necessarily an organic, truly random event.

A **true random number generator (TRNG)** relies on randomness from a physical event that is external to the computer and its operating system. Examples of such events are blips in atmospheric noise, or points at which a radioactive material decays. A true random number generator receives information from these types of unpredictable events to produce a truly random number.

This calculator uses a randomizing computer program to produce random numbers, so it is a pseudo-random number generator.

How to Generate Random Numbers

1. What is your range? Set a minimum number and a maximum number. The random number(s) generated are selected from your range of numbers, with the min and max numbers included.
2. How many numbers? Specify how many random numbers to generate.
3. Allow repeats? If you choose No your random numbers will be unique and there is no chance of getting a duplicate number. If you choose Yes the random number generator may produce a duplicate number in your set of numbers.
4. Sort numbers? You can decide not to sort your random numbers. You can also order your random numbers ascending, lowest to highest or descending, highest to lowest.

Example: Generate a Random Number to Use as a PIN.

To generate a 6-digit PIN with or without duplicate digits choose the following settings:

- Min = 0
- Max = 9
- Generate 6 numbers
- Allow repeats = yes or no

- Sort numbers = Do not sort

Example: Randomize a Set of Numbers

Say you have a group of 10 people represented by the numbers 1 to 10. You want to shuffle them into a random order of selection for an event.

Choose the following settings to randomize order of selection:

- Min = 1
- Max = 10
- Generate 10 numbers
- Allow repeats = no
- Sort numbers = Do not sort

Example: Randomly Choose One Number From a Range of Numbers

Say you want randomly select one number from 1 to 10, like drawing a number out of a hat.

Choose the following settings:

- Min = 1
- Max = 10
- Generate 1 number
- Allow repeats = no
- Sort numbers = Do not sort

Example: Lottery Number Generator

You want to generate numbers for lottery tickets. You need to choose 5 numbers from a pool of 1 to 49 without duplicates.

Choose the following settings in the random number generator:

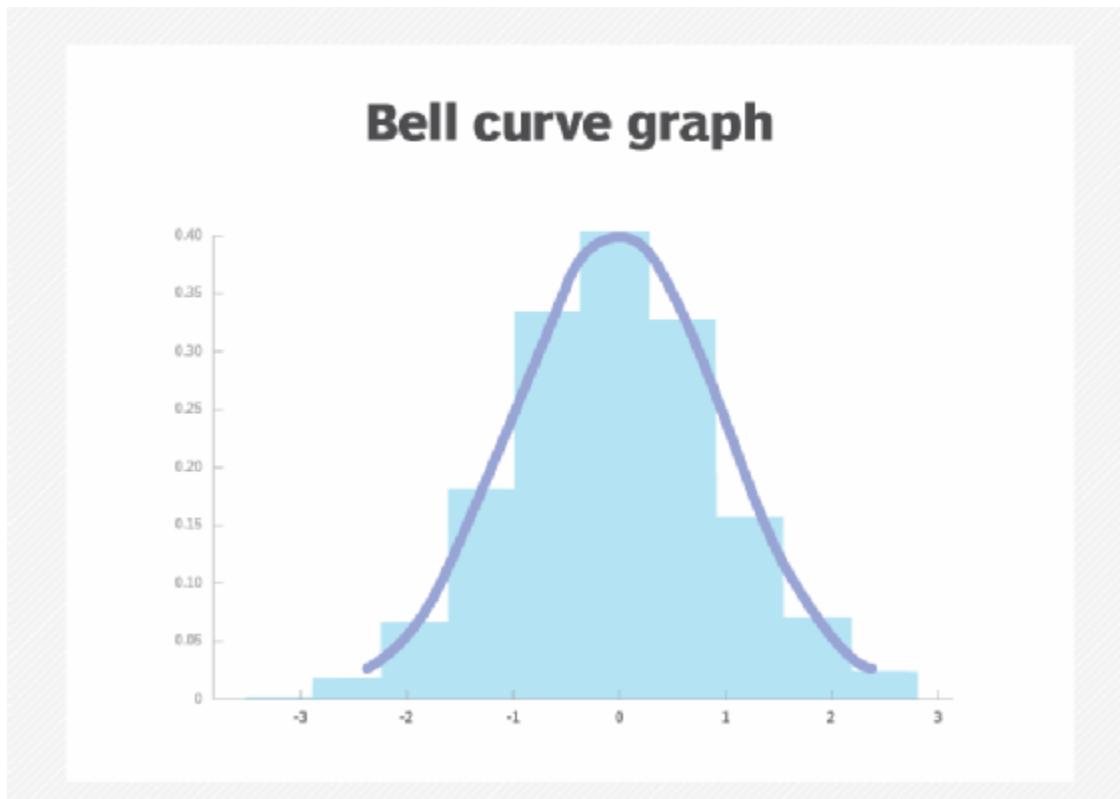
- Min = 1
- Max = 49
- Generate 5 numbers
- Allow Duplicates = no
- Sort Numbers = low to high

Monte Carlo Simulation

What is a Monte Carlo simulation?

A Monte Carlo simulation is a mathematical technique that simulates the range of possible outcomes for an uncertain event. These predictions are based on an estimated range of values instead of a fixed set of values and evolve randomly. Computers use Monte Carlo simulations to analyze data and predict a future outcome based on a course of action.

First, Monte Carlo simulations use a probability distribution for any variable that has inherent uncertainty. Then, it recalculates the results many times, using a different set of random numbers within the estimated range each time. This process generates many probable outcomes, which become more accurate as the number of inputs grows. In other words, the different outcomes form a normal distribution or bell curve, where the most common outcome is in the middle of the curve.



A normal distribution is also called a bell curve. It is always symmetrical around the mean.

The Monte Carlo method has been described as "faking it a billion times until the reality emerges." It relies on the assumption that many random samples mimic patterns in the total population.

Importance of Monte Carlo simulations

Monte Carlo simulations are simple conceptually but enable users to solve problems in complex systems. They are particularly useful for long-term predictions because of their accuracy. Monte Carlo simulations are also a good alternative to machine learning when there isn't enough data to make a machine learning model accurate. As the number of inputs increases, so does the number of forecasts.

They also enable accurate simulations involving randomness. For a simple example, someone could use a Monte Carlo simulation to calculate the probability of a particular outcome -- say, rolling a seven -- when rolling two dice. There are 36 possible combinations, and six of those combinations add up to seven. The mathematical or expected probability of rolling a seven is 6/36, or 16.67%.

External factors, such as the shape of the dice or the surface they are rolled on, cause the actual or experimental probability to be different from the mathematical probability. Rolling the dice 1,000 times and getting a seven on 170 of those times would be the actual probability -- 170/1,000, or 17%, which is close to the actual probability but not exact. Each roll would be an iteration of the Monte Carlo simulation, which gets more accurate with each iteration. This property -- that the actual probability gets closer to the exact probability with more iterations -- is known as the *law of large numbers*.

Someone could use Microsoft Excel, IBM SPSS Statistics or a similar program to run this experiment.

The 4 steps in a Monte Carlo simulation

Although they might vary from case to case, the general steps to a Monte Carlo simulation are as follows:

1. Build the model. Determine the mathematical model or transfer algorithm.
2. Choose the variables to simulate. Pick the variables, and determine an appropriate probability distribution for each random variable.
3. Run repeated simulations. Run the random variables through the mathematical model to perform many iterations of the simulation.
4. Aggregate the results, and determine the mean, standard deviation and variant to determine if the result is as expected. Visualize the results on a histogram.

Monte Carlo simulation use cases

Monte Carlo simulations can be used for a spectrum of different industries. Finance is one of the most common use case examples, but any industry that involves predicting an inherently

uncertain condition has a use for it.

Industry use cases for a Monte Carlo simulation include the following:

- **Finance**, such as risk assessment and long-term forecasting.
- **Project management**, such as estimating the duration or cost of a project.
- **Engineering and physics**, such as analyzing weather patterns, traffic flow or energy distribution.
- **Quality control and testing**, such as estimating the reliability and failure rate of a product.
- **Healthcare and biomedicine**, such as modeling the spread of diseases.

Use cases for Monte Carlo simulations also encompass different technologies. In IT alone, there are many use cases for Monte Carlo simulations. Some of those use cases specific to IT are the following:

- **Network and system design.** Monte Carlo simulations can be used to model different designs, identify potential bottlenecks, and perform capacity planning and resource allocation.
- **Artificial intelligence.** Monte Carlo simulations provide the basis for resampling techniques for estimating the accuracy of a model on a given data set.
- **Cybersecurity.** Monte Carlo simulations can be used to simulate different cyber attacks, evaluate the probability of them occurring, evaluate their hypothetical impact and identify vulnerabilities in IT systems.
- **Performance testing.** Monte Carlo simulations can be used for load testing applications and estimating the potential impact for increased usage or scaling.

Monte Carlo simulations are used in research and real-world business applications. They are specifically useful in research because of their ability to uncover data insights and enable the researcher to see multiple possible outcomes. Real-world scenarios for Monte Carlo simulations include the following:

- A researcher performing a risk assessment of potential toxic chemicals in South Korean cabbage kimchi.
- A telecom service provider gauging the ability of its network to handle swells in viewer traffic during the Olympics.
- A company tracking potential price movements of a given asset to price stock options.
- A random walk study of the spread of COVID-19.
- A smartphone manufacturer measuring a smartphone's performance in different temperatures.
- An analyst predicting the outcomes of a presidential election.

Advantages of Monte Carlo simulations

Monte Carlo simulations are used in many different areas for a reason. They are a relatively simple way to make complex predictions. They offer answers to hypothetical questions and assign a certain level of order to randomness. Other advantages of Monte Carlo simulations include the following:

- **Improve decision-making.** Monte Carlo simulations help users make decisions with a degree of confidence.
- **Solve complex problems simply.** Monte Carlo simulations show both what could happen and how likely each outcome is
- **Visualize the range of possible outcomes and their likelihood of occurring.** Monte Carlo simulations make it easy to visualize what the result of a standard decision or outcome might be next to the result of an unusual outcome.
-

Drawbacks of Monte Carlo simulations

Despite the advantages of Monte Carlo simulations, there are disadvantages. Like any simulation, it uses historical data for a future projection, which carries the risk of being inaccurate. Specific drawbacks of Monte Carlo simulations include the following:

- **Processing power.** Monte Carlo simulations require many iterations to be accurate. Running many iterations takes time and energy and can be computationally intensive.
- **Input bias.** This can be damaging if the data used for input is inaccurate or incomplete.
- **Sensitivity to the chosen probability distribution.** It is important to choose an appropriate probability distribution for the problem. Choosing the wrong one can render the results meaningless.

Common probability distributions used in Monte Carlo simulations

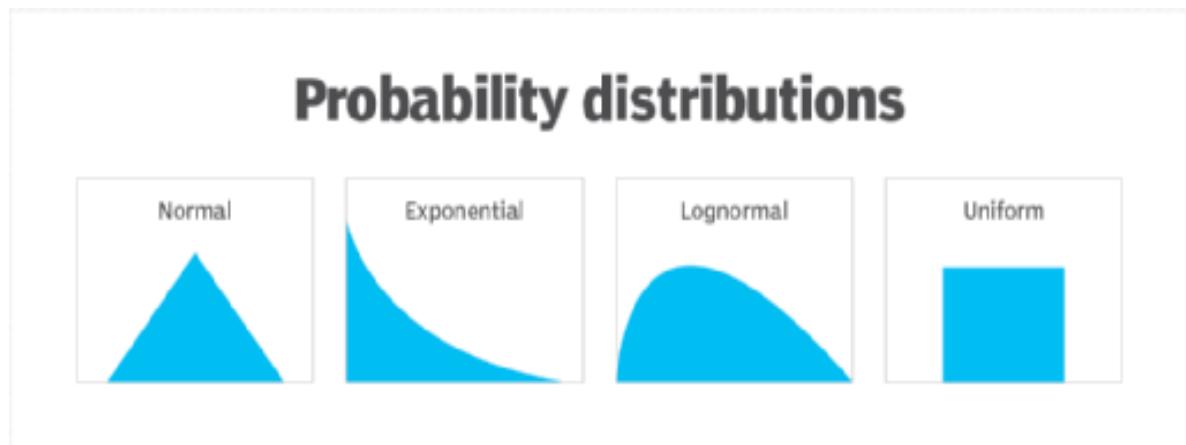
Probability distributions represent a range of values between two limits and can consist of discrete or continuous values. Discrete probability distributions are plotted as a sequence of finite numbers in a table, whereas continuous distributions are plotted as a curve between two points on a graph.

Some common probability distributions in Monte Carlo simulations are the following:

- **Normal distributions.** These are continuous distributions where the most data points cluster toward the middle. It is also called a *bell curve* or *Gaussian distribution*.
- **Triangular distributions.** These are continuous distributions with fixed minimum and maximum values. They can be either symmetrical, where the most probable value equals the

mean and the median, or asymmetrical.

- **Uniform distributions.** These are continuous distributions by known minimum and maximum values. All outcomes have the same probability of occurring.
- **Lognormal distributions.** These are continuous distributions by mean and standard deviation. The values are positive and create a curve that skews right.



- Different probability distributions have different shapes and are suitable for different contexts.
- **Exponential distributions.** These continuous distributions are used to illustrate the time between independent occurrences given the occurrence rate.
- **Weibull distributions.** These continuous distributions can model skewed data and approximate other distributions.
- **Poisson distributions.** These discrete probability distributions describe the probability of an event occurring in X periods of time.
- **Discrete distributions.** These discrete probability distributions help define the finite values of all possible outcome values.

History of the Monte Carlo simulation

The Monte Carlo simulation was invented during World War II by mathematician Stanislaw Ulam and computer scientist John von Neumann. Both von Neumann and Ulam used the Monte Carlo simulation in the invention of the hydrogen bomb as part of the Manhattan Project.

Ulam got the idea for the Monte Carlo simulation while he was sick and bored at home. He was playing the card game solitaire and wanted to be able to compute the probability of winning. His approach was to play as many hands as he could, count the number he won and divide that by the number of total hands played. He realized this would take a long time, so he involved his friend von Neumann. He asked von Neumann to run the simulation on the Electronic Numerical

Integrator and Computer machine, which was one of the first computers.

The simulation was named after a casino in Monaco. The randomness in a roulette table resembles the chance element of Monte Carlo simulations. In 1949, Ulam published the first unclassified document describing the Monte Carlo simulation.

Monte Carlo Simulation Example

Let's consider an example of a young working couple who works very hard and has a lavish lifestyle including expensive holidays every year. They have a retirement objective of spending \$170,000 per year (approx. \$14,000/month) and leaving a \$1 million estate to their children.

An analyst runs a simulation and finds that their savings-per-period is insufficient to build the desired portfolio value at retirement; however, it is achievable if the allocation to small-cap stocks is doubled (up to 50% to 70% from 25% to 35%), which will increase their risk considerably.

None of the above alternatives (higher savings or increased risk) are acceptable to the client. Thus, the analyst factors in other adjustments before running the simulation again.

The analyst delays their retirement by two years and decreases their monthly spend post-retirement to \$12,500. The resulting distribution shows that the desired portfolio value is achievable by increasing allocation to small-cap stock by only 8%. With the available insight, the analyst advises the clients to delay retirement and decrease their spending marginally, to which the couple agrees.

Monte Carlo Simulation Example

For better understanding, let's analyze the example below.

Assume that you are creating a work schedule for a research and development project. You noticed that there is some degree of uncertainty exists in the activity duration estimates. Then you decided to use the Monte Carlo Simulation to analyze the impact of risks that will affect your project.

First, you create the work schedule and estimate the duration of each activity by using the three-point estimating technique. You estimate optimistic, pessimistic and most likely durations for each activity as shown in the below table.

Activities	Optimistic	Pessimistic	Most Likely
Choose a Topic	4	7	5
Develop Research Plan	5	7	6
Complete the Research	7	9	8
Report	2	4	3

Then you calculate the duration of each activity by using PERT Formula

$$\text{PERT Estimate} = (\text{Optimistic Estimate} + 4 \times \text{Most likely Estimate} + \text{Pessimistic Estimate}) / 6$$

After calculating the duration of each activity, the table becomes as follows

Activities	Optimistic	Pessimistic	Most Likely	PERT Estimate
Choose a Topic	4	7	5	5,2
Develop Research Plan	5	7	6	6
Complete the Research	7	9	8	8
Report	2	4	3	3

Total Completion Time of the project is = $5,2 + 6 + 8 + 3 = 22,2$ Months.

For the best case, completion time of the project is ;

Total Completion Time = $4 + 5 + 7 + 2 = 18$ Months.

For the worst case, completion time of the project is ;

Total Completion Time = $7 + 7 + 9 + 4 = 27$ Months.

Now you run the Monte Carlo Simulation by using Excel or software and get the chances of completion of the project.

Let's assume that you get the results after performing the Monte Carlo Simulation. Below table shows the results.

Duration of Project (Months)	Possibility of Completion (%)
18	10%
19	17%
20	24%
21	28%
22	35%
23	46%
24	57%
25	69%
26	88%
27	100%

Note that these results are only for illustration. They are not from an actual simulation.

If you analyze the results, you will see that the possibility of completion of the project in the best case is the lowest and in the worst case, it is highest.

As it is seen from the table, this simulation provides you a number of results to improve your decision making.

Most business situations such as uncertainty in market demand, unknown quantity of sales, variable costs and many others are too complex for an analytical solution. But The Monte Carlo Simulation enables you to evaluate your plan numerically, you can change numbers, ask ‘what if’ and see the results.

Unit 4

Test of

Hypothesis

Topics

Procedure of Testing Hypothesis
Standard Error and Sampling distribution
Estimation,
Student's t-distribution
Chi-Square test and goodness of fit
F-test and analysis of variance
Factor analysis.

Hypothesis is usually considered as the principal instrument in research. Its main function is to suggest new experiments and observations. In fact, many experiments are carried out with the deliberate object of testing hypotheses. Decision-makers often face situations wherein they are interested in testing hypotheses based on available information and then take decisions based on such testing. In social science, where direct knowledge of population parameter(s) is rare, hypothesis testing is the often-used strategy for deciding whether a sample data offer such support for a hypothesis that generalization can be made. Thus, hypothesis testing enables us to make probability statements about population parameter(s). The hypothesis may not be proved absolutely, but in practice it is accepted if it has withstood a critical testing. Before we explain how hypotheses are tested through different tests meant for the purpose, it will be appropriate to explain clearly the meaning of a hypothesis and the related concepts for better understanding of the hypothesis testing techniques.

WHAT IS A HYPOTHESIS?

Ordinarily, when one talks about hypothesis, one simply means a mere assumption or some supposition to be proved or disproved. But for a researcher hypothesis is a formal question that he intends to resolve. Thus a hypothesis may be defined as a proposition or a set of proposition set forth as an explanation for the occurrence of some specified group of phenomena either asserted merely as a provisional conjecture to guide some investigation or accepted as highly probable in the light of established facts. Quite often a research hypothesis is a predictive statement, capable of being tested by scientific methods, that relates an independent variable to some dependent variable. For example, consider statements like the following ones:

“Students who receive counselling will show a greater increase in creativity than students not receiving counselling”

Or

“the automobile *A* is performing as well as automobile *B*.”

These are hypotheses capable of being objectively verified and tested. Thus, we may conclude that a hypothesis states what we are looking for and it is a proposition which can be put to a test to determine its validity.

Characteristics of hypothesis: Hypothesis must possess the following characteristics:

- (i) Hypothesis should be clear and precise. If the hypothesis is not clear and precise, the inferences drawn on its basis cannot be taken as reliable.
- (ii) Hypothesis should be capable of being tested. In a swamp of untestable hypotheses, many a

time the research programmes have bogged down. Some prior study may be done by researcher in order to make hypothesis a testable one. A hypothesis “is testable if other deductions can be made from it which, in turn, can be confirmed or disproved by observation.”¹

(iii) Hypothesis should state relationship between variables, if it happens to be a relational hypothesis.

(iv) Hypothesis should be limited in scope and must be specific. A researcher must remember that narrower hypotheses are generally more testable and he should develop such hypotheses.

(v) Hypothesis should be stated as far as possible in most simple terms so that the same is easily understandable by all concerned. But one must remember that simplicity of hypothesis has nothing to do with its significance.

(vi) Hypothesis should be consistent with most known facts i.e., it must be consistent with a substantial body of established facts. In other words, it should be one which judges accept as being the most likely.

(vii) Hypothesis should be amenable to testing within a reasonable time. One should not use even an excellent hypothesis, if the same cannot be tested in reasonable time for one cannot spend a life-time collecting data to test it.

(viii) Hypothesis must explain the facts that gave rise to the need for explanation. This means that by using the hypothesis plus other known and accepted generalizations, one should be able to deduce the original problem condition. Thus hypothesis must actually explain what it claims to explain; it should have empirical reference.

BASIC CONCEPTS CONCERNING TESTING OF HYPOTHESES

Null and research hypotheses

To carry out statistical hypothesis testing, research and null hypothesis are employed:

- Research hypothesis: this is the hypothesis that you propose, also known as the alternative hypothesis H_A . For example:

H_A : There is a relationship between intelligence and academic results.

H_A : First year university students obtain higher grades after an intensive Statistics course.

H_A : Males and females differ in their levels of stress.

- The null hypothesis (H_0) is the opposite of the research hypothesis and expresses that there is no relationship between variables, or no differences between groups; for example:

H_0 : There is no relationship between intelligence and academic results.

H_0 : First year university students do not obtain higher grades after an intensive Statistics course.

H_0 : Males and females will not differ in their levels of stress.

The purpose of hypothesis testing is to test whether the null hypothesis (there is no difference, no effect) can be rejected or approved. If the null hypothesis is rejected, then the research hypothesis can be accepted. If the null hypothesis is accepted, then the research hypothesis is rejected.

In hypothesis testing, a value is set to assess whether the null hypothesis is accepted or rejected and whether the result is statistically significant:

- A critical value is the score the sample would need to decide against the null hypothesis.
- A probability value is used to assess the significance of the statistical test. If the null hypothesis is rejected, then the alternative to the null hypothesis is accepted.

The hypothesis testing process

The hypothesis testing process can be divided into five steps:

1. Restate the research question as research hypothesis and a null hypothesis about the populations.
2. Determine the characteristics of the comparison distribution.
3. Determine the cut off sample score on the comparison distribution at which the null hypothesis should be rejected.
4. Determine your sample's score on the comparison distribution.
5. Decide whether to reject the null hypothesis.

This *example* illustrates how these five steps can be applied to test a hypothesis:

- Let's say that you conduct an experiment to investigate whether students' ability to memorise words improves after they have consumed caffeine.
- The experiment involves two groups of students: the first group consumes caffeine; the second group drinks water.
- Both groups complete a memory test.
- A randomly selected individual in the experimental condition (i.e. the group that consumes caffeine) has a score of 27 on the memory test. The scores of people in general on this memory measure are normally distributed with a mean of 19 and a

standard deviation of 4.

- The researcher predicts an effect (differences in memory for these groups) but does not predict a particular direction of effect (i.e. which group will have higher scores on the memory test). Using the 5% significance level, what should you conclude?

Step 1: There are two populations of interest.

Population 1: People who go through the experimental procedure (drink coffee).

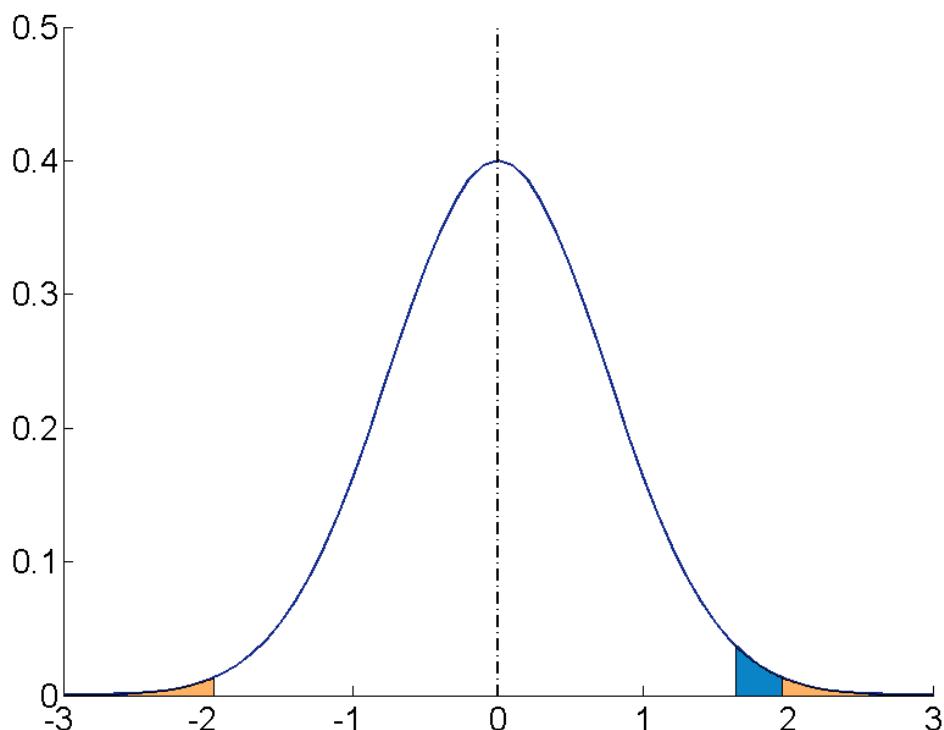
Population 2: People who do not go through the experimental procedure (drink water).

- Research hypothesis: Population 1 will score differently from Population 2.
- Null hypothesis: There will be no difference between the two populations.

Step 2: We know that the characteristics of the comparison distribution (student population) are:

Population M = 19, Population SD= 4, normally distributed. These are the mean and standard deviation of the distribution of scores on the memory test for the general student population.

Step 3: For a two-tailed test (the direction of the effect is not specified) at the 5% level (25% at each tail), the cut off sample scores are +1.96 and -1.99.



Step 4: Your sample score of 27 needs to be converted into a Z value. To calculate $Z = (27 - 19)/4 = 2$ (*check the Converting into Z scores section if you need to review how to do this process*)

Step 5: A ‘Z’ score of 2 is more extreme than the cut off Z of +1.96 (see figure above). The result is significant and, thus, the null hypothesis is rejected.

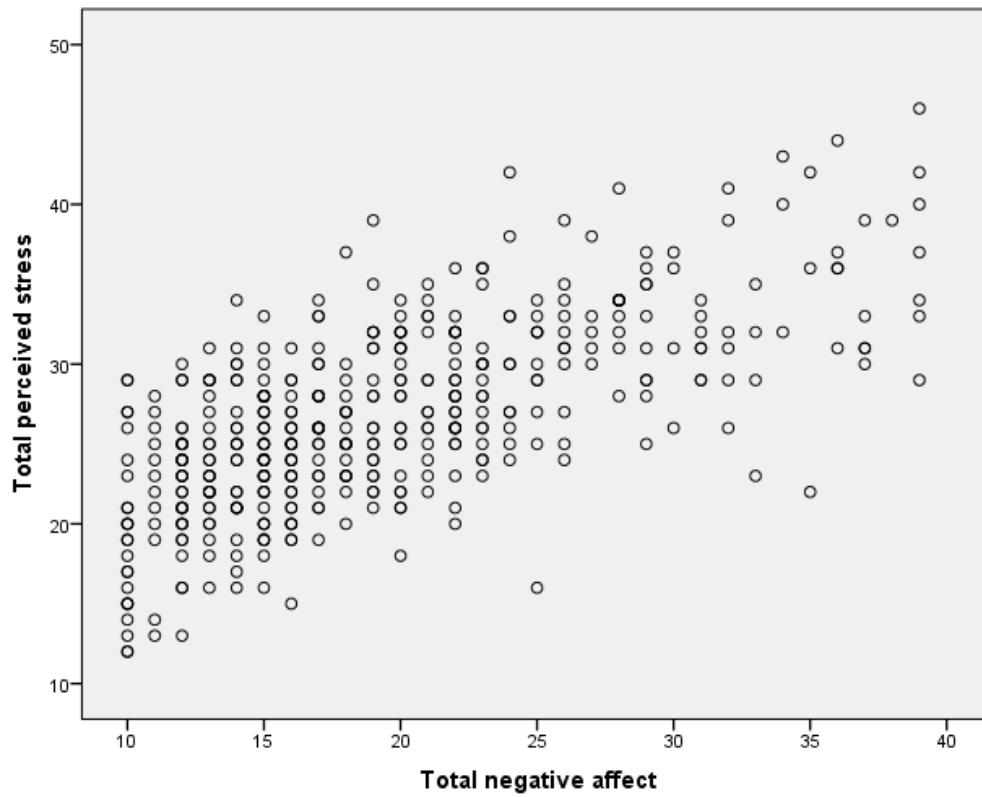
Correlation analysis

Correlation analysis explores the association between variables. The purpose of correlational analysis is to discover whether there is a relationship between variables, which is unlikely to occur by sampling error. The null hypothesis is that there is no relationship between the two variables. Correlation analysis provides information about:

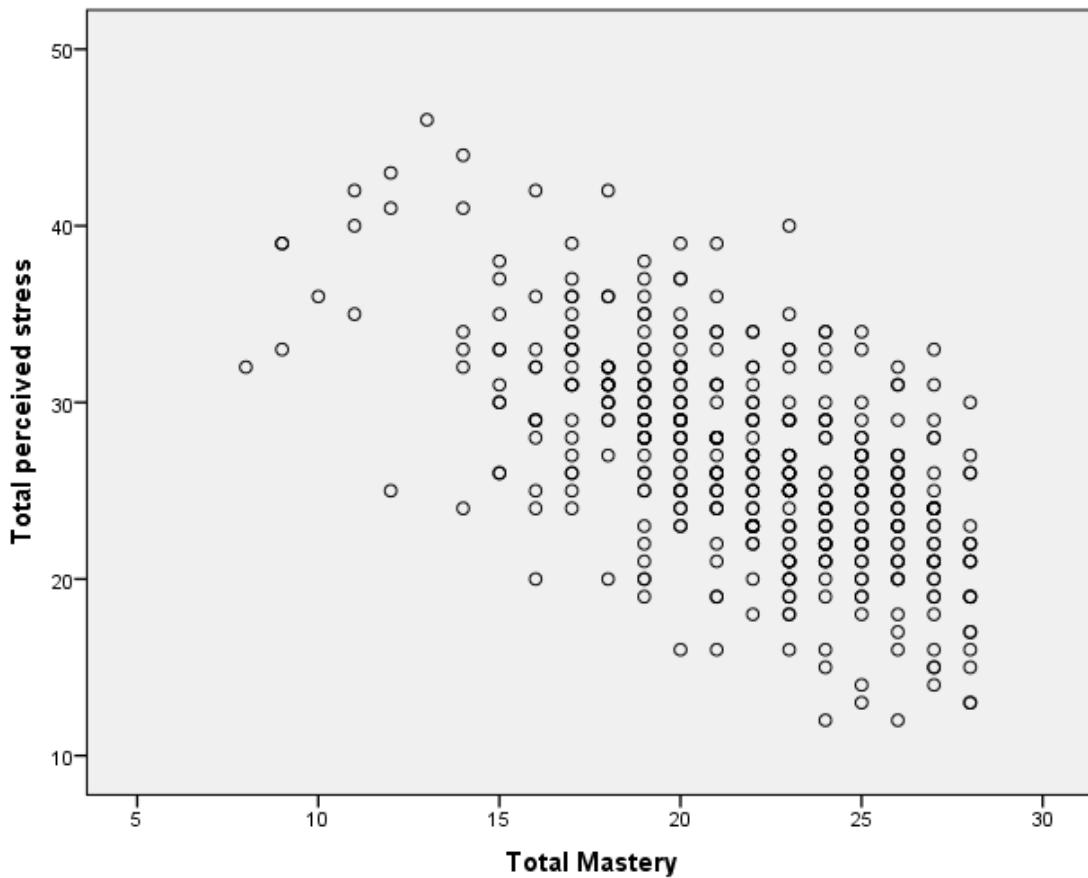
- The direction of the relationship: positive or negative- given by the sign of the correlation coefficient.
- The strength or magnitude of the relationship between the two variables- given by the correlation coefficient, which varies from 0 (no relationship between the variables) to 1 (perfect relationship between the variables).

1. Direction of the relationship.

A positive correlation indicates that high scores on one variable are associated with high scores on the other variable; low scores on one variable are associated with low scores on the second variable . For instance, in the figure below, higher scores on negative affect are associated with higher scores on perceived stress



A negative correlation indicates that high scores on one variable are associated with low scores on the other variable. The graph shows that a person who scores high on perceived stress will probably score low on mastery. The slope of the graph is downwards- as it moves to the right. In the figure below, higher scores on mastery are associated with lower scores on perceived stress.



2. The strength or magnitude of the relationship

The strength of a linear relationship between two variables is measured by a statistic known as the correlation coefficient, which varies from 0 to -1, and from 0 to +1. There are several correlation coefficients; the most widely used are Pearson's r and Spearman's ρ . The strength of the relationship is interpreted as follows:

- Small/weak: $r = .10$ to $.29$
- Medium/moderate: $r = .30$ to $.49$
- Large/strong: $r = .50$ to 1

It is important to note that correlation analysis does not imply causality. Correlation is used to explore the association between variables, however, it does not indicate that one variable causes the other. The correlation between two variables could be due to the fact that a third variable is affecting the two variables.

Example of Hypothesis Testing

Case Study: Hypothesis Testing for the Mean

Scenario:

A university claims that the average time a student spends studying per week is 25 hours. A random sample of 50 students is selected, and the sample mean is found to be 23 hours with a sample standard deviation of 6 hours. Is there enough evidence to suggest that the average study time is different from 25 hours? Use a significance level of 0.05.

Step 1: State the Hypotheses

We will perform a two-tailed hypothesis test because we are checking whether the average study time is different from 25 hours (could be either greater or less).

- Null Hypothesis (H_0): The mean study time is 25 hours.

$$H_0 : \mu = 25$$

- Alternative Hypothesis (H_a): The mean study time is different from 25 hours.

$$H_a : \mu \neq 25$$

Step 2: Choose the Significance Level

We are given a significance level of 0.05.

$$\alpha = 0.05$$

Step 3: Select the Appropriate Test and Test Statistic

Since the population standard deviation is unknown and the sample size is relatively large ($(n = 50)$), we will use the t-test for a single mean. The test statistic formula is:

$$t = \frac{\bar{x} - \mu_0}{\frac{s}{\sqrt{n}}}$$

Where:

- \bar{x} = sample mean = 23 hours
- μ_0 = hypothesized population mean = 25 hours
- s = sample standard deviation = 6 hours
- n = sample size = 50

Step 4: Compute the Test Statistic

$$t = \frac{23 - 25}{\frac{6}{\sqrt{50}}}$$

First, calculate the standard error ($\frac{s}{\sqrt{n}}$):

$$\frac{6}{\sqrt{50}} = \frac{6}{7.071} \approx 0.849$$

Now, calculate the t-statistic:

$$t = \frac{23 - 25}{0.849} = \frac{-2}{0.849} \approx -2.355$$

Step 5: Determine the Critical Value or P-Value

Since this is a two-tailed test, we will split the significance level between both tails of the t-distribution. For a significance level of 0.05 and degrees of freedom ($df = n - 1 = 50 - 1 = 49$), the critical t-value can be found from a t-distribution table or calculator.

For $\alpha = 0.05$ (two-tailed), the critical t-value is approximately:

$$t_{\text{critical}} = \pm 2.009$$

Alternatively, we can compute the p-value using statistical software or tables. For a t-statistic of -2.355 and 49 degrees of freedom, the p-value is approximately:

$$p \approx 0.022$$

Step 6: Compare the Test Statistic and Make a Decision

- The computed t-statistic is -2.355, and the critical t-value is ± 2.009 .
- Since $-2.355 < -2.009$, we reject the null hypothesis.

Alternatively, comparing the p-value to the significance level:

- The p-value is 0.022, which is less than the significance level of 0.05.
- Therefore, we reject the null hypothesis.

Interpretation:

There is sufficient evidence to suggest that the average study time of students is different from 25 hours per week at the 0.05 significance level.

The university's claim that students study an average of 25 hours per week may not be accurate. The sample data indicates that students may study less than 25 hours on average, though further investigation could provide more clarity.

Standard Error and Sampling Distribution

In statistics, the standard error is the standard deviation of the sample distribution. The sample mean of a data is generally varied from the actual population mean. It is represented as SE. It is used to measure the amount of accuracy by which the given sample represents its population. Statistics is a vast topic in which we learn about data, sample and population, mean, median, mode, dependent and independent variables, standard deviation, variance, etc. Here you will learn the standard error formula along with SE of the mean and estimation.

Standard Error Meaning

The standard error is one of the mathematical tools used in statistics to estimate the variability. It is abbreviated as SE. The standard error of a statistic or an estimate of a parameter is the standard deviation of its sampling distribution. We can define it as an estimate of that standard deviation.

Standard Error Formula

The accuracy of a sample that describes a population is identified through the SE formula. The sample mean which deviates from the given population and that deviation is given as;

$$SE_{\bar{x}} = \frac{S}{\sqrt{n}}$$

Where S is the standard deviation and n is the number of observations.

Standard Error of the Mean (SEM)

The standard error of the mean also called the standard deviation of mean, is represented as the standard deviation of the measure of the sample mean of the population. It is abbreviated as SEM. For example, normally, the estimator of the population mean is the sample mean. But, if we draw another sample from the same population, it may provide a distinct value. Thus, there would be a population of the sampled means having its distinct variance and mean. It may be defined as the standard deviation of such sample means of all the possible samples taken from the same given population. SEM defines an estimate of standard deviation which has been computed from the sample. It is calculated as the ratio of the standard deviation to the root of sample size, such as:

$$SEM = \frac{s}{\sqrt{n}}$$

Where 's' is the standard deviation and n is the number of observations.

The standard error of the mean shows us how the mean changes with different tests, estimating the same quantity. Thus if the outcome of random variations is notable, then the standard error of the mean will have a higher value. But, if there is no change observed in the data points after repeated experiments, then the value of the standard error of the mean will be zero.

Standard Error of Estimate (SEE)

The **standard error** of the **estimate** is the estimation of the accuracy of any predictions. It is denoted as SEE. The regression line depreciates the sum of squared deviations of prediction. It is also known as the sum of squares **error**. SEE is the square root of the average squared **deviation**. The deviation of some estimates from intended values is given by standard error of estimate formula.

$$SEE = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n - 2}}$$

Where x_i stands for data values, \bar{x} bar is the mean value and n is the sample size.

How to calculate Standard Error

Step 1: Note the number of measurements (n) and determine the sample mean (μ). It is the average of all the measurements.

Step 2: Determine how much each measurement varies from the mean.

Step 3: Square all the deviations determined in step 2 and add altogether: $\sum(x_i - \mu)^2$

Step 4: Divide the sum from step 3 by one less than the total number of measurements ($n-1$).

Step 5: Take the square root of the obtained number, which is the standard deviation (σ).

Step 6: Finally, divide the standard deviation obtained by the square root of the number of measurements (n) to get the standard error of your estimate.

Go through the example given below to understand the method of calculating standard error.

Standard Error Example

Calculate the standard error of the given data:

y: 5, 10, 12, 15, 20

Solution: First we have to find the mean of the given data;

Mean = $(5+10+12+15+20)/5 = 62/5 = 10.5$

Now, the standard deviation can be calculated as;

$S = \text{Summation of difference between each value of given data and the mean value}/\text{Number of values.}$

Hence,

$$S = \sqrt{\frac{(5 - 10.5)^2 + (10 - 10.5)^2 + (12 - 10.5)^2 + (15 - 10.5)^2 + (20 - 10.5)^2}{5}}$$

After solving the above equation, we get;

$$S = 5.35$$

Therefore, SE can be estimated with the formula;

$$SE = S/\sqrt{n}$$

$$SE = 5.35/\sqrt{5} = 2.39$$

Standard Error vs Standard Deviation

The below table shows how we can calculate the standard deviation (SD) using population parameters and standard error (SE) using sample parameters.

Population parameters	Formula for SD	Sample statistic	Formula for SE
Mean \bar{x}	$\frac{\sigma}{\sqrt{n}}$	Sample mean \bar{x}	$\frac{s}{\sqrt{n}}$
Sample proportion (P)	$\sqrt{\frac{P(1-P)}{n}}$	Sample proportion (p)	$\sqrt{\frac{p(1-p)}{n}}$
Difference between means $\bar{x}_1 - \bar{x}_2$	$\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}$	Difference between means $\bar{x}_1 - \bar{x}_2$	$\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$
Difference between proportions $P_1 - P_2$	$\sqrt{\frac{P_1(1-P_1)}{n_1} + \frac{P_2(1-P_2)}{n_2}}$	Difference between proportions $p_1 - p_2$	$\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}$

Importance of Standard Error

Standard errors produce simplistic measures of uncertainty in a value. They are often used because, in many cases, if the standard error of some individual quantities is known, then we can easily calculate the standard error of some function of the quantities. Also, when the probability distribution of the value is known, we can use it to calculate an exact confidence interval. However, the standard error is an essential indicator of how precise an estimate of the sample statistic's population parameter is.

Examples for Calculating the Standard Error of the Sampling Distribution of a Sample

Mean

Example 1

The mean height of all adults in a particular country is 163 cm with a standard deviation of 2.5 cm. Calculate the standard error of the sampling distribution of a sample mean if the sample size is 40. Round to three decimal places.

Step 1: Identify the standard deviation of the population, σ , and the sample size, N .

The problem states that standard deviation of the population is $\sigma=2.5$ cm, and the sample size is $N=40$.

Step 2: Calculate the standard error of the sampling distribution of a sample mean by dividing the population standard deviation by the square root of the sample size.

$$\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{N}}$$

Using the formula for the standard error of the sampling distribution of a sample mean with the information identified in step 1, we have:

$$\begin{aligned}\sigma_{\bar{X}} &= \frac{\sigma}{\sqrt{N}} \\ &= \frac{2.5 \text{ cm}}{\sqrt{40}} \\ &\approx 0.395 \text{ cm}\end{aligned}$$

The standard error is approximately 0.395 cm.

Example 2

The mean weekly grocery bill for a household in a certain large city is \$175.30 with a standard deviation of \$6.17. What is the standard error of the sampling distribution of a sample mean if the sample size is 100? Round to the nearest cent.

Step 1: Identify the standard deviation of the population, σ , and the sample size, N .

We have:

- $\sigma=\$6.17$
- $N=100$

Step 2: Calculate the standard error of the sampling distribution of a sample mean by

dividing the population standard deviation by the square root of the sample size.

$$\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{N}}$$

Using the formula for the standard error of the sampling distribution of a sample mean, we have:

$$\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{N}}$$

$$= \frac{\$6.17}{\sqrt{100}}$$

$$\approx \$0.62$$

The standard error is approximately \$0.62.

What is t-distribution?

Student's t-distribution, also known as the t-distribution, is a probability distribution that is used in statistics for making inferences about the population mean when the sample size is small or when the population standard deviation is unknown. It is similar to the standard normal distribution (Z-distribution), but it has heavier tails. Theoretical work on t-distribution was done by **W.S. Gosset**; he has published his findings under the pen name "Student". That's why it is called a **Student's t-test**. The t-score represents the number of standard deviations the sample mean is away from the population mean.

T-Score

The T-score, also known as the t-value or t-statistic, is a standardized score that quantifies how many standard deviations a data point or sample mean is from the population mean. It is commonly used in statistical hypothesis testing, particularly in scenarios where the sample size is small or the population standard deviation is unknown.

The formula for calculating the T-score in the context of a t-distribution is given by:

$$t = \frac{\bar{x} - \mu}{s\sqrt{n}}$$

where,

- t = t-score,
- \bar{x} = sample mean
- μ = population mean,
- s = standard deviation of the sample,
- n = sample size

As we know, we use t-distribution when the standard deviation of the population is unknown and the sample size is small. The formula for the t-distribution looks very similar to the normal distribution; the only difference is that instead of the standard deviation of the population, we will use the standard deviation of the sample.

When to Use the t-Distribution?

Student's t Distribution is used when :

- The sample size is 30 or less than 30.
- The population standard deviation(σ) is unknown.
- The population distribution must be unimodal and skewed.

Mathematical Derivation of t-Distribution

The t-distribution has been derived mathematically under the assumption of a normally

distributed population and the formula for the probability density function will be like this

$$f(t) = \frac{\Gamma(\frac{df+1}{2})}{\sqrt{df\pi} \Gamma(\frac{df}{2})} \left(1 + \frac{t^2}{df}\right)^{-\frac{df+1}{2}}$$

where,

- $\Gamma(.)$ is the gamma function
- df= Degrees of freedom

So, this above equation indicates the probability density function(pdf) of the t-distribution for df degrees of freedom.

Significance of the t-Distribution

1. Degrees of Freedom and Tail Heaviness:

The t-distribution degrees of freedom influence tail heaviness, with smaller values yielding heavier tails. Higher degrees of freedom make the t-distribution more akin to a standard normal distribution (mean 0, standard deviation 1), shaping its spread.

2. Small Sample Size:

The t-distribution is vital for small sample sizes, offering a precise probability distribution for statistical inferences on population parameters, especially the mean. This is crucial when the population standard deviation is unknown and must be estimated from the sample.

3. t-Score Calculation for Inference:

In situations where the standard deviation of the population is not known, the t-score (T) is calculated to make inferences about the population mean. The distinction between s and σ (population standard deviation) and the utilization of $(n - 1)$ degrees of freedom delineate the characteristics of the t-distribution.

4. Comparison with Z-Score and Normal Distribution:

Unlike the z-score, which employs the population standard deviation, the t-score uses the estimated standard deviation from the sample. This results in a t-distribution with $(n - 1)$ degrees of freedom, emphasizing the t-distribution's role in handling uncertainty when estimating the population standard deviation, especially in small sample sizes.

Interpretation of t-Distribution

A confidence interval for the mean is a statistical range computed from the data, designed to encompass a plausible “population” mean. This interval is expressed as

$$\bar{x} \pm t * s / \sqrt{n}, t$$

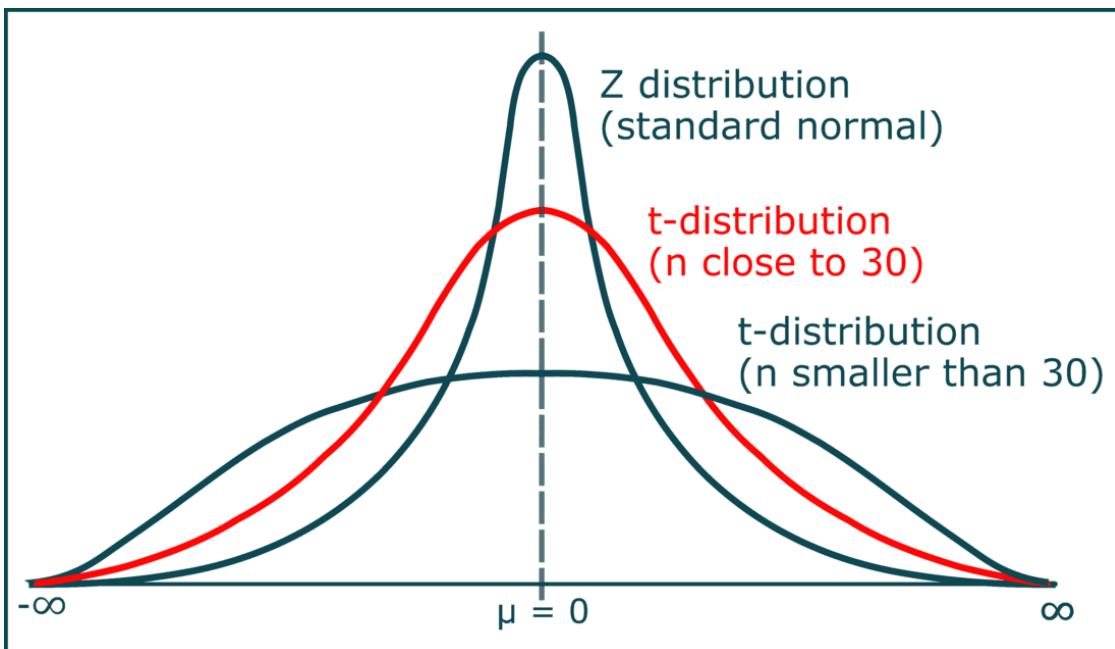
Where t represents a critical value obtained from the t-distribution.

Suppose we are investigating the mean study time for an exam by collecting data from a sample of 20 students. To establish a 90% confidence interval for the population mean study time using the above formula.

Let us say $\bar{x} = 4$ hours, $s = 1.5$ hours and $n = 20$. The critical t -value is obtained for a 90% confidence interval with 19 degrees of freedom. Assuming a critical t -value of 1.729(calculated using the t table or online calculator), the calculation results in a 90% confidence interval for the average study time, such as between 3.58 hours and 4.42 hours. This utilization of the t-distribution addresses the uncertainty linked to estimating the population mean from a sample, especially in cases where the population standard deviation is unknown.

Properties of the t-Distribution

- The variable in t-distribution ranges from $-\infty$ to $+\infty$ ($-\infty < t < +\infty$).
- t- distribution will be symmetric like the normal distribution if the power of t is even in the probability density function(pdf).
- For large values of v (i.e. increased sample size n); the t-distribution tends to a standard normal distribution. This implies that for different v values, the shape of t-distribution also differs.
- The t-distribution is less peaked than the normal distribution at the center and higher peaked in the tails. From the above diagram, one can observe that the red and green curves are less peaked at the center but higher peaked at the tails than the blue curve.
- The value of y (peak height) attains highest at $\mu = 0$ as one can observe the same in the above diagram.
- The mean of the distribution is equal to 0 for $v > 1$ where v = degrees of freedom, otherwise undefined.
- The median and mode of the distribution is equal to 0.
- The variance is equal to $v / v-2$ for $v > 2$ and ∞ for $2 < v \leq 4$ otherwise undefined.



Degrees of freedom refer to the number of independent observations in a set of data. When estimating a mean score or a proportion from a single sample, the number of independent observations is equal to the sample size minus one. Hence, the distribution of the t statistic from samples of size 10 would be described by a t distribution having $10 - 1$ or 9 degrees of freedom. Similarly, a t- distribution having 15 degrees of freedom would be used with a sample of size 16.

t-Distribution Table

t-Distribution table gives the t-value for a different level of significance and different degrees of freedom. The calculated t-value will be compared with the tabulated t-value. For example, if one is performing a student's t-test and for that performance, he has taken a 5% level of significance and he got or calculated t-value and he has taken his tabulated t-value and if the calculated t-value is higher than the tabulated t-value, in that case, it will say that there is a significant difference between the population mean and the sample means at 5% level of significance and if vice versa then, in that case, it will say that there is no significant difference between the population means and the sample means at 5% level of significance.

t Table

cum. prob	$t_{.50}$	$t_{.75}$	$t_{.50}$	$t_{.85}$	$t_{.90}$	$t_{.95}$	$t_{.975}$	$t_{.99}$	$t_{.995}$	$t_{.999}$	$t_{.9995}$
one-tail	0.50	0.25	0.20	0.15	0.10	0.05	0.025	0.01	0.005	0.001	0.0005
two-tails	1.00	0.50	0.40	0.30	0.20	0.10	0.05	0.02	0.01	0.002	0.001
df											
1	0.000	1.000	1.376	1.963	3.078	6.314	12.71	31.82	63.66	318.31	636.62
2	0.000	0.816	1.061	1.386	1.886	2.920	4.303	6.965	9.925	22.327	31.599
3	0.000	0.785	0.978	1.250	1.638	2.353	3.182	4.541	5.841	10.215	12.924
4	0.000	0.741	0.941	1.190	1.533	2.132	2.776	3.747	4.604	7.173	8.610
5	0.000	0.727	0.920	1.156	1.476	2.015	2.571	3.365	4.032	5.893	6.869
6	0.000	0.718	0.906	1.134	1.440	1.943	2.447	3.143	3.707	5.208	5.959
7	0.000	0.711	0.896	1.119	1.415	1.895	2.365	2.998	3.499	4.785	5.408
8	0.000	0.706	0.889	1.108	1.397	1.860	2.306	2.896	3.355	4.501	5.041
9	0.000	0.703	0.883	1.100	1.383	1.833	2.262	2.821	3.250	4.297	4.781
10	0.000	0.700	0.879	1.093	1.372	1.812	2.228	2.764	3.169	4.144	4.587
11	0.000	0.697	0.876	1.088	1.363	1.796	2.201	2.718	3.106	4.025	4.437
12	0.000	0.695	0.873	1.083	1.356	1.782	2.179	2.681	3.055	3.930	4.318
13	0.000	0.694	0.870	1.079	1.350	1.771	2.160	2.650	3.012	3.852	4.221
14	0.000	0.692	0.868	1.076	1.345	1.761	2.145	2.624	2.977	3.787	4.140
15	0.000	0.691	0.866	1.074	1.341	1.753	2.131	2.602	2.947	3.733	4.073
16	0.000	0.690	0.865	1.071	1.337	1.746	2.120	2.583	2.921	3.686	4.015
17	0.000	0.689	0.863	1.069	1.333	1.740	2.110	2.567	2.898	3.646	3.965
18	0.000	0.688	0.862	1.067	1.330	1.734	2.101	2.552	2.878	3.610	3.922
19	0.000	0.688	0.861	1.066	1.328	1.729	2.093	2.539	2.861	3.579	3.883
20	0.000	0.687	0.860	1.064	1.325	1.725	2.086	2.528	2.845	3.552	3.850
21	0.000	0.686	0.859	1.063	1.323	1.721	2.080	2.518	2.831	3.527	3.819
22	0.000	0.686	0.858	1.061	1.321	1.717	2.074	2.508	2.819	3.505	3.792
23	0.000	0.685	0.858	1.060	1.319	1.714	2.069	2.500	2.807	3.485	3.768
24	0.000	0.685	0.857	1.059	1.318	1.711	2.064	2.492	2.797	3.467	3.745
25	0.000	0.684	0.856	1.058	1.316	1.708	2.060	2.485	2.787	3.450	3.725
26	0.000	0.684	0.856	1.058	1.315	1.706	2.056	2.479	2.779	3.435	3.707
27	0.000	0.684	0.855	1.057	1.314	1.703	2.052	2.473	2.771	3.421	3.690
28	0.000	0.683	0.855	1.056	1.313	1.701	2.048	2.467	2.763	3.408	3.674
29	0.000	0.683	0.854	1.055	1.311	1.699	2.045	2.462	2.756	3.396	3.659
30	0.000	0.683	0.854	1.055	1.310	1.697	2.042	2.457	2.750	3.385	3.648
40	0.000	0.681	0.851	1.050	1.303	1.684	2.021	2.423	2.704	3.307	3.551
60	0.000	0.679	0.848	1.045	1.296	1.671	2.000	2.390	2.660	3.232	3.460
80	0.000	0.678	0.846	1.043	1.292	1.664	1.990	2.374	2.639	3.195	3.416
100	0.000	0.677	0.845	1.042	1.290	1.660	1.984	2.364	2.626	3.174	3.390
1000	0.000	0.675	0.842	1.037	1.282	1.646	1.962	2.330	2.581	3.098	3.300
Z	0.000	0.674	0.842	1.036	1.282	1.645	1.960	2.326	2.576	3.090	3.291
	0%	50%	60%	70%	80%	90%	95%	98%	99%	99.8%	99.9%
	Confidence Level										

t-scores and p-values

t-scores :

- It represents the deviation of a data point from the mean in a t-distribution, expressed in terms of standard deviations. Particularly useful for small sample sizes or cases with unknown population standard deviations.
- We can obtain them from a t-table or through online tools, providing a numerical measure of how atypical a data point is within the distribution.
- t-score is important in determining confidence intervals, aiding in estimating the range within which the true population parameter is likely to fall. The critical value of t is integral in confidence interval calculations, guiding the determination of upper and lower bounds.

p-value:

The p-value (probability value) is a statistical measure that helps assess the evidence against

a null hypothesis.

- p-value describes the likelihood of data occurring if the null hypothesis were true.
- You can use statistical software to directly obtain the p-value associated with the calculated t-score or you can use the t-table, which provides critical values for different levels of significance and degrees of freedom. First, find the row corresponding to your degrees of freedom and the column corresponding to your t-score to get the p-value.

Limitations of Using a T-Distribution

- **Sensitivity to Departure from Normality:** The t-distribution assumes normality in the underlying population. When data deviates significantly from a normal distribution, reliance on the t-distribution may introduce inaccuracies in statistical inferences.
- **Limited Applicability for Large Samples:** As sample sizes increase, the t-distribution converges to the normal distribution. Therefore, for sufficiently large samples and known population standard deviation, the normal distribution is more appropriate, and using the t-distribution may not offer additional benefits.
- **Impact of Outliers and Small Sample Sizes:** The t-distribution can be sensitive to outliers, and its tails can be influenced by small sample sizes. Outliers may distort results, and in cases where the sample size is very small, the t-distribution may have heavier tails, affecting the accuracy of inferences.
- **Requires Random Sampling:** The assumptions underlying the t-distribution, such as random sampling and independence of observations, need to be met for valid results. If these assumptions are violated, the accuracy of inferences drawn from the t-distribution may be compromised.

T- Distribution Applications

1. **Testing for the Hypothesis of the Population Mean:** T-distributions are commonly used in hypothesis tests regarding the population mean. This involves assessing whether a sample mean is significantly different from a hypothesized population mean.
2. **Testing for the Hypothesis of the Difference Between Two Means:** T-tests can be employed to examine if there is a significant difference between the means of two independent samples. This can be done under the assumption of equal variances or when variances are unequal. In scenarios where samples are not independent, such as paired or dependent samples, t-tests can be used to assess the significance of the mean difference between related observations.
3. **Testing for the Hypothesis about the Coefficient of Correlation:** T-distributions play a role in hypothesis testing related to correlation coefficients. This includes situations where

the population correlation coefficient is assumed to be zero ($\rho=0$) or when testing for a non-zero correlation coefficient ($\rho\neq0$).

Difference Between T-Distribution and Normal Distribution

T-Distribution	Normal Distribution
T-Distribution is defined by its degree of freedom which itself depends upon the sample size	Normal distribution is defined by its mean and standard deviation
T- distribution is used when the sample size is small	Normal distribution is used when we have large no data points in the dataset
It has a heavier tail than normal distribution which means more data points are away from the mean of the distribution	Normal distribution has a lighter tail than T-distribution which means more data points lie near the mean of the distribution
We use T-distribution in hypothesis testing when the standard variation of the population is unknown	Normal distribution is used when the standard deviation is known
T-Distribution has a larger range of critical values as compared to the normal distribution as this distribution has heavier tails	Normal distribution has a smaller range as compared to t-distribution

What Is a Chi-Square Test?

The Chi-Square test is a statistical procedure for determining the difference between observed and expected data. This test can also be used to decide whether it correlates to our data's categorical variables. It helps to determine whether a difference between two categorical variables is due to chance or a relationship between them.

A chi-square test or comparable nonparametric test is required to test a hypothesis regarding the distribution of a categorical variable. Categorical variables, which indicate categories such as animals or countries, can be nominal or ordinal. They cannot have a normal distribution since they only have a few particular values. Chi-Square Test Formula

$$x_c^2 = \frac{\sum (O_i - E_i)^2}{E_i}$$

Where

c = Degrees of freedom

O = Observed Value

E = Expected Value

The degrees of freedom in a statistical calculation represent the number of variables that can vary. The degrees of freedom can be calculated to ensure that chi-square tests are statistically valid. These tests are frequently used to compare observed data with data expected to be obtained if a particular hypothesis were true.

The Observed values are those you gather yourselves.

The expected values are the anticipated frequencies, based on the null hypothesis.

Fundamentals of Hypothesis Testing

Hypothesis testing is a technique for interpreting and drawing inferences about a population based on sample data. It aids in determining which sample data best support mutually exclusive population claims.

Null Hypothesis (H₀) - The Null Hypothesis is the assumption that the event will not occur. A null hypothesis has no bearing on the study's outcome unless it is rejected.

H₀ is the symbol for it, and it is pronounced H-naught.

Alternate Hypothesis(H₁ or H_a) - The Alternate Hypothesis is the logical opposite of the null hypothesis. The acceptance of the alternative hypothesis follows the rejection of the null hypothesis. H₁ is the symbol for it.

Types of Chi-Square Tests

There are two main types of Chi-Square tests:

1. Independence
2. Goodness-of-Fit

Independence

The Chi-Square Test of Independence is a derivable (also known as inferential) statistical test which examines whether the two sets of variables are likely to be related with each other or not. This test is used when we have counts of values for two nominal or categorical variables and is considered as non-parametric test. A relatively large sample size and independence of observations are the required criteria for conducting this test.

Example:

In a movie theatre, suppose we made a list of movie genres. Let us consider this as the first variable. The second variable is whether or not the people who came to watch those genres of movies have bought snacks at the theatre. Here the null hypothesis is that the genre of the film and whether people bought snacks or not are unrelated. If this is true, the movie genres don't impact snack sales.

Goodness-Of-Fit

In statistical hypothesis testing, the Chi-Square Goodness-of-Fit test determines whether a variable is likely to come from a given distribution or not. We must have a set of data values and the idea of the distribution of this data. We can use this test when we have value counts for categorical variables. This test demonstrates a way of deciding if the data values have a “good enough” fit for our idea or if it is a representative sample data of the entire population.

Example:

Suppose we have bags of balls with five different colours in each bag. The given condition is that the bag should contain an equal number of balls of each colour. The idea we would like to test here is that the proportions of the five colours of balls in each bag must be exact.

Chi-Square Test Examples

1. Chi-Square Test for Independence

Example: A researcher wants to determine if there is an association between gender (male/female) and preference for a new product (like/dislike). The test can assess whether preferences are independent of gender.

2. Chi-Square Test for Goodness of Fit

Example: A dice manufacturer wants to test if a six-sided die is fair. They roll the die 60

times and expect each face to appear 10 times. The test checks if the observed frequencies match the expected frequencies.

3. Chi-Square Test for Homogeneity

Example: A fast-food chain wants to see if the preference for a particular menu item is consistent across different cities. The test can compare the distribution of preferences in multiple cities to see if they are homogeneous.

4. Chi-Square Test for a Contingency Table

Example: A study investigates whether smoking status (smoker/non-smoker) is related to the presence of lung disease (yes/no). The test can evaluate the relationship between smoking and lung disease in the sample.

5. Chi-Square Test for Population Proportions

Example: A political analyst wants to see if voter preference (candidate A vs. candidate B) is the same across different age groups. The test can determine if the proportions of preferences differ significantly between age groups.

Real-Life Practical Examples of Goodness of Fit and F-Test

Goodness of Fit Examples

1. Marketing Survey Responses

A company launches a survey to gauge customer preferences for four different product colors: red, blue, green, and yellow. The company expects that 25% of the customers will choose each color. A goodness-of-fit test can be used to compare the observed preferences from the survey to the expected distribution (25% for each color) and determine if the actual preferences significantly differ from the expected.

2. Genetic Traits in Offspring

In genetics, a researcher might expect that the offspring of two heterozygous parents will display dominant and recessive traits in a 3:1 ratio. After collecting data on the observed number of offspring showing each trait, a goodness-of-fit test can be used to see if the observed data aligns with the expected Mendelian inheritance ratio of 3:1.

3. Manufacturing Quality Control

A manufacturer produces items that are expected to meet certain specifications, like weight

or size. Suppose the company claims that 10% of the products are defective. To test this claim, a random sample of items is taken, and a goodness-of-fit test is used to compare the observed defect rate to the expected 10% rate and check whether the defect rate is statistically consistent with the company's claim.

4. Website Traffic Analysis

A website admin expects traffic to their website to be evenly distributed across weekdays. If they collect data on the number of visitors each day for a month, they can use a goodness-of-fit test to check if the actual traffic pattern follows the expected even distribution across the five days.

5. Election Polling Results

In an election, a political analyst may expect that voters' preferences are split evenly across three candidates (e.g., 33.3% for each candidate). After polling a sample of voters, a goodness-of-fit test can be used to determine whether the observed proportions significantly differ from the expected equal distribution.

F-Test Examples

1. Comparing the Effectiveness of Two Medications

A pharmaceutical company wants to compare the effectiveness of two drugs for treating high blood pressure. The company collects data on the blood pressure reduction in patients taking Drug A and Drug B. An F-test can be used to determine whether the variances in blood pressure reduction are significantly different between the two groups, which could suggest that the drugs affect patients differently.

2. Manufacturing Process Improvement

A factory implements two different methods for producing a product. Management wants to test whether the variability (variance) in product dimensions is the same between the two methods. An F-test is conducted to compare the variances and determine if one process is more consistent than the other, indicating which method produces more reliable results.

3. Comparing Income Variability Between Regions

An economist is interested in whether income variability (variance) differs between urban

and rural populations. The economist collects income data from both populations and conducts an F-test to compare the variances, helping to understand whether income distribution is more unequal in one area compared to the other.

4. Analyzing Teacher Effectiveness

In an educational setting, a school administrator may want to compare the variability in test scores between two teachers who teach the same subject. An F-test can be used to check whether the variance in student test scores significantly differs between the two teachers, indicating whether one teacher's students perform more consistently than the other's.

5. Quality Control in Food Processing

A food processing company wants to compare the variance in the shelf life of two different food preservation techniques. Using an F-test, the company can compare the variability in shelf life for the two techniques to determine which method provides more consistent product longevity.

How to Perform a Chi-Square Test?

Let's say you want to know if gender has anything to do with political party preference. You poll 440 voters in a simple random sample to find out which political party they prefer. The results of the survey are shown in the table below:

	Republican	Democrat	Independent	Total
Male	100	70	30	200
Female	140	60	20	220
Total	240	130	50	440

To see if gender is linked to political party preference, perform a Chi-Square test of independence using the steps below.

Step 1: Define the Hypothesis

H0: There is no link between gender and political party preference.

H1: There is a link between gender and political party preference.

Step 2: Calculate the Expected Values

Now you will calculate the expected frequency.

$$\text{Expected Value} = \frac{(\text{Row Total}) * (\text{Column Total})}{\text{Total Number Of Observations}}$$

For example, the expected value for Male Republicans is:

$$= \frac{(240) * (200)}{440} = 109$$

Similarly, you can calculate the expected value for each of the cells.

		Expected Values			
		Republican	Democrat	Independent	Total
Male		109	59	22.72	200
Female		120	65	25	220
Total		240	130	50	440

Step 3: Calculate $(O - E)^2 / E$ for Each Cell in the Table

Now you will calculate the $(O - E)^2 / E$ for each cell in the table.

Where

O = Observed Value

E = Expected Value

		$(O - E)^2 / E$			
		Republican	Democrat	Independent	Total
Male		0.74311927	2.050847	2.332676056	200
Female		3.333333333	0.384615	1	220
Total		240	130	50	440

Step 4: Calculate the Test Statistic X²

X² is the sum of all the values in the last table

$$= 0.743 + 2.05 + 2.33 + 3.33 + 0.384 + 1$$

$$= 9.837$$

Before you can conclude, you must first determine the critical statistic, which requires determining our degrees of freedom. The degrees of freedom in this case are equal to the table's number of columns minus one multiplied by the table's number of rows minus one, or $(r-1)(c-1)$. We have $(3-1)(2-1) = 2$.

Finally, you compare our obtained statistic to the critical statistic found in the chi-square table. As you can see, for an alpha level of 0.05 and two degrees of freedom, the critical statistic is 5.991, which is less than our obtained statistic of 9.83. You can reject our null hypothesis because the critical statistic is higher than your obtained statistic.

This means you have sufficient evidence to say that there is an association between gender and political party preference.

Example: Bird species at a bird feeder

Frequency of visits by bird species at a bird feeder during a 24-hour period

Bird species	Frequency
House sparrow	15
House finch	12
Black-capped chickadee	9
Common grackle	8
European starling	8
Mourning dove	6

A chi-square test (a [chi-square goodness of fit test](#)) can test whether these observed frequencies are significantly different from what was expected, such as equal frequencies.

Example: Handedness and nationality

Contingency table of the handedness of a sample of Americans and Canadians

	Right-handed	Left-handed
American	236	19
Canadian	157	16

A chi-square test (a test of independence) can test whether these observed frequencies are significantly different from the frequencies expected if handedness is unrelated to nationality.

Critical values of the Chi-square distribution with d degrees of freedom

Probability of exceeding the critical value							
d	0.05	0.01	0.001	d	0.05	0.01	0.001
1	3.841	6.635	10.828	11	19.675	24.725	31.264
2	5.991	9.210	13.816	12	21.026	26.217	32.910
3	7.815	11.345	16.266	13	22.362	27.688	34.528
4	9.488	13.277	18.467	14	23.685	29.141	36.123
5	11.070	15.086	20.515	15	24.996	30.578	37.697
6	12.592	16.812	22.458	16	26.296	32.000	39.252
7	14.067	18.475	24.322	17	27.587	33.409	40.790
8	15.507	20.090	26.125	18	28.869	34.805	42.312
9	16.919	21.666	27.877	19	30.144	36.191	43.820
10	18.307	23.209	29.588	20	31.410	37.566	45.315

INTRODUCTION TO POPULATION GENETICS, Table D.1
 © 2013 Sinauer Associates, Inc.

What Are Categorical Variables?

Categorical variables belong to a subset of variables that can be divided into discrete categories. Names or labels are the most common categories. These variables are also known as qualitative variables because they depict the variable's quality or characteristics.

Categorical variables can be divided into two categories:

1. Nominal Variable: A nominal variable's categories have no natural ordering. Example: Gender, Blood groups
2. Ordinal Variable: A variable that allows the categories to be sorted is an ordinal variable. An example is customer satisfaction (Excellent, Very Good, Good, Average, Bad, and so on).

How to Solve Chi-Square Problems?

1. State the Hypotheses

- Null hypothesis (H_0): There is no association between the variables
- Alternative hypothesis (H_1): There is an association between the variables.

2. Calculate the Expected Frequencies

- Use the formula: $E = (\text{Row Total} \times \text{Column Total}) / \text{Grand Total}$

$$E = \frac{\text{Row Total} \times \text{Column Total}}{\text{Grand Total}}$$

3. Compute the Chi-Square Statistic

- Use the formula: $\chi^2 = \sum \frac{(O - E)^2}{E}$, where O is the observed frequency and E is the expected frequency.

4. Determine the Degrees of Freedom (df)

- Use the formula: $df = (\text{number of rows} - 1) \times (\text{number of columns} - 1)$

5. Find the Critical Value and Compare

- Use the chi-square distribution table to find the critical value for the given df and significance level (usually 0.05).
- Compare the chi-square statistic to the critical value to decide whether to reject the null hypothesis.

These practice problems help you understand how chi-square analysis tests hypotheses and explores relationships between categorical variables in various fields.

When to Use a Chi-Square Test?

A Chi-Square Test is used to examine whether the observed results are in order with the expected values. When the data to be analysed is from a random sample, and when the variable is the question is a categorical variable, then Chi-Square proves the most appropriate test for the same. A categorical variable consists of selections such as breeds of dogs, types of cars, genres of movies, educational attainment, male v/s female etc. Survey responses and questionnaires are the primary sources of these types of data. The Chi-square test is most commonly used for analysing this kind of data. This type of analysis is helpful for

researchers who are studying survey response data. The research can range from customer and marketing research to political sciences and economics.

Chi-Square Distribution

Chi-square distributions (X^2) are a type of continuous probability distribution. They're commonly utilized in hypothesis testing, such as the chi-square goodness of fit and independence tests. The parameter k , which represents the degrees of freedom, determines the shape of a chi-square distribution.

Very few real-world observations follow a chi-square distribution. Chi-square distributions aim to test hypotheses, not to describe real-world distributions. In contrast, other commonly used distributions, such as normal and Poisson distributions, may explain important things like birth weights or illness cases per year.

Chi-square distributions are excellent for hypothesis testing because of its close resemblance to the conventional normal distribution. Many essential statistical tests rely on the traditional normal distribution.

In statistical analysis, the Chi-Square distribution is used in many hypothesis tests and is determined by the parameter k degree of freedom. It belongs to the family of continuous probability distributions. The Sum of the squares of the k -independent standard random variables is called the Chi-Squared distribution. Pearson's Chi-Square Test formula is -

$$X^2 = \sum \frac{(O-E)^2}{E}$$

Where X^2 is the Chi-Square test symbol

Σ is the summation of observations

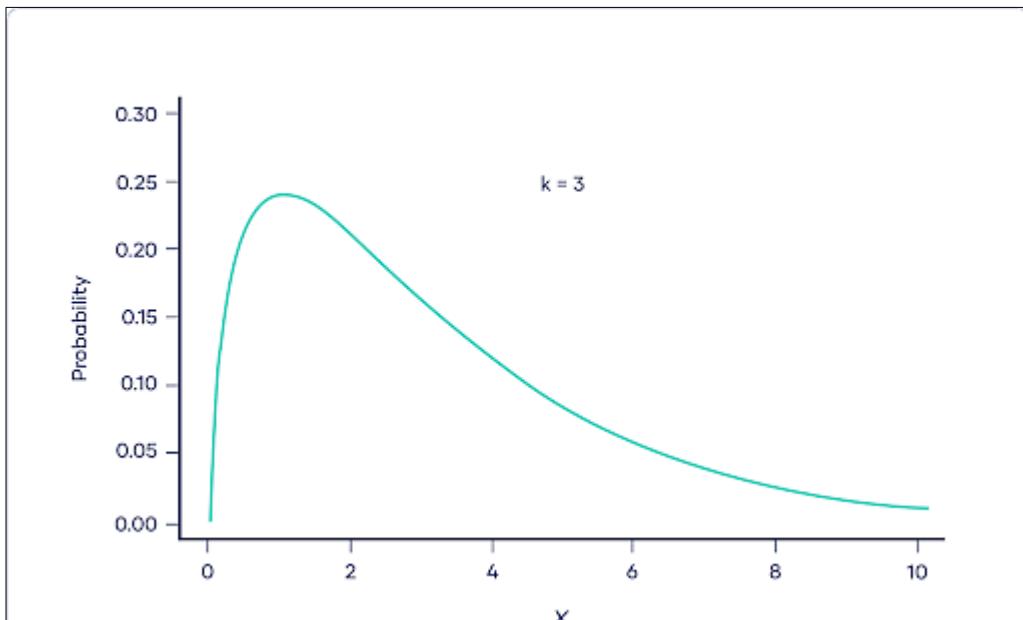
O is the observed results

E is the expected results

The shape of the distribution graph changes with the increase in the value of k , i.e., the degree of freedom.

When k is 1 or 2, the Chi-square distribution curve is shaped like a backwards 'J'. It means there is a high chance that X^2 becomes close to zero.

When k is greater than 2, the shape of the distribution curve looks like a hump and has a low probability that X^2 is very near to 0 or very far from 0. The distribution occurs much longer on the right-hand side and shorter on the left-hand side. The probable value of X^2 is $(X^2 - 2)$.



When k is greater than ninety, a normal distribution is seen, approximating the Chi-square distribution.

What is the P-Value in a Chi-Square Test?

The P-Value in a Chi-Square test is a statistical measure that helps to assess the importance of your test results.

Here P denotes the probability; hence for the calculation of p-values, the Chi-Square test comes into the picture. The different p-values indicate different types of hypothesis interpretations.

1. $P \leq 0.05$ (Hypothesis interpretations are rejected)
2. $P \geq 0.05$ (Hypothesis interpretations are accepted)

The concepts of probability and statistics are entangled with Chi-Square Test. Probability is the estimation of something that is most likely to happen. Simply put, it is the possibility of an event or outcome of the sample. Probability can understandably represent bulky or complicated data. And statistics involves collecting and organising, analysing, interpreting and presenting the data.

Finding P-Value

When you run all of the Chi-square tests, you'll get a test statistic called X^2 . You have two options for determining whether this test statistic is statistically significant at some alpha level:

1. Compare the test statistic X^2 to a critical value from the Chi-square distribution table.
2. Compare the p-value of the test statistic X^2 to a chosen alpha level.

Test statistics are calculated by taking into account the sampling distribution of the test statistic under the null hypothesis, the sample data, and the approach which is chosen for performing the test.

The p-value will be as mentioned in the following cases.

- A lower-tailed test is specified by: $P(TS \leq ts | H_0 \text{ is true})$ p-value = $cdf(ts)$
- Lower-tailed tests have the following definition: $P(TS \leq ts | H_0 \text{ is true})$ p-value = $cdf(ts)$
- A two-sided test is defined as follows, if we assume that the test statistic's distribution is symmetric about 0. $2 * P(TS \geq |ts| | H_0 \text{ is true}) = 2 * (1 - cdf(|ts|))$

Where:

P: probability Event

TS: Test statistic is computed observed value of the test statistic from your sample $cdf()$: Cumulative distribution function of the test statistic's distribution (TS)

Tools and Software for Chi-Square Analysis

Here are some commonly used tools and software for performing Chi-Square analysis:

1. SPSS (Statistical Package for the Social Sciences) is a widely used software for statistical analysis, including Chi-Square tests. It provides an easy-to-use interface for performing Chi-Square tests for independence, goodness of fit, and other statistical analyses.
2. R is a powerful open-source programming language and software environment for statistical computing. The `chisq.test()` function in R allows for easy conducting of Chi-Square tests.
3. The SAS suite is used for advanced analytics, including Chi-Square tests. It is often used in research and business environments for complex data analysis.
4. Microsoft Excel offers a Chi-Square test function (`CHISQ.TEST`) for users who prefer working within spreadsheets. It's a good option for basic Chi-Square analysis with smaller datasets.
5. Python (with libraries like SciPy or Pandas) offers robust tools for statistical analysis. The `scipy.stats.chisquare()` function can be used to perform Chi-Square tests.

Properties of Chi-Square Test

1. Variance is double the times the number of degrees of freedom.
2. Mean distribution is equal to the number of degrees of freedom.
3. When the degree of freedom increases, the Chi-Square distribution curve becomes normal.

Limitations of Chi-Square Test

There are two limitations to using the chi-square test that you should be aware of.

- The chi-square test, for starters, is extremely sensitive to sample size. Even insignificant

relationships can appear statistically significant when a large enough sample is used. Keep in mind that "statistically significant" does not always imply "meaningful" when using the chi-square test.

- Be mindful that the chi-square can only determine whether two variables are related. It does not necessarily follow that one variable has a causal relationship with the other. It would require a more detailed analysis to establish causality.

Example: McNemar's test

Suppose that a sample of 100 people is offered two flavors of ice cream and asked whether they like the taste of each.

Contingency table of ice cream flavor preference

	Like chocolate	Dislike chocolate
Like vanilla	47	32
Dislike vanilla	8	13

- **Null hypothesis (H_0):** The proportion of people who like chocolate is **the same** as the proportion of people who like vanilla.
- **Alternative hypothesis (H_A):** The proportion of people who like chocolate is **different** from the proportion of people who like vanilla.

Advanced Chi-Square Test Techniques

1. Chi-Square Test with Yates' Correction (Continuity Correction)

This technique is used in 2x2 contingency tables to reduce the Chi-Square value and correct for the overestimation of statistical significance when sample sizes are small. The correction is achieved by subtracting 0.5 from the absolute difference between each observed and expected frequency.

2. Mantel-Haenszel Chi-Square Test

This technique is used to assess the association between two variables while controlling for one or more confounding variables. It's particularly useful in stratified analyses where the goal is to examine the relationship between variables across different strata (e.g., age groups, geographic locations).

3. Chi-Square Test for Trend (Cochran-Armitage Test)

This test is used when the categorical variable is ordinal, and you want to assess whether there is

a linear trend in the proportions across the ordered groups. It's commonly used in epidemiology to analyze trends in disease rates over time or across different exposure levels.

4. Monte Carlo Simulation for Chi-Square Test

When the sample size is very small or when expected frequencies are too low, the Chi-Square distribution may not provide accurate p-values. In such cases, Monte Carlo simulation can be used to generate an empirical distribution of the test statistic, providing a more accurate significance level.

5. Bayesian Chi-Square Test

In Bayesian statistics, the Chi-Square test can be adapted to incorporate prior knowledge or beliefs about the data. This approach is useful when existing information should influence the analysis, leading to potentially more accurate conclusions.

Problem 1: Goodness of Fit for Dice Rolls

A die is rolled 120 times, and the following frequencies of outcomes are observed:

Outcome	Frequency
1	20
2	22
3	18
4	15
5	25
6	20

Check if the die is fair using a significance level of 0.05.

Solution:

- **Step 1: Set up hypotheses**
 - H₀: The die is fair (expected frequencies are equal).
 - H_a: The die is not fair (expected frequencies differ).
- **Step 2: Calculate expected frequencies** A fair die has an equal probability for each outcome, so:

$$\text{Expected frequency} = \frac{120}{6} = 20$$

- **Step 3: Calculate the test statistic (Chi-Square)**

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

Where O_i is the observed frequency, and $E_i = 20$ is the expected frequency.

$$\begin{aligned}\chi^2 &= \frac{(20 - 20)^2}{20} + \frac{(22 - 20)^2}{20} + \frac{(18 - 20)^2}{20} + \frac{(15 - 20)^2}{20} + \frac{(25 - 20)^2}{20} + \frac{(20 - 20)^2}{20} \\ \chi^2 &= 0 + 0.2 + 0.2 + 1.25 + 1.25 + 0 = 2.9\end{aligned}$$

- **Step 4: Compare with critical value** Degrees of freedom (df) = 6 - 1 = 5. The critical value from the Chi-Square table for $\alpha=0.05$ and df = 5 is 11.07.
Since $\chi^2=2.9 < 11.07$, we fail to reject H0.
 - **Conclusion:** The die is fair.
-

Problem 2: Goodness of Fit for Survey Responses

A survey is conducted to see how many people prefer four different brands of a product. The expected preferences are equal across all brands. Out of 400 people, the responses are as follows:

Brand	Observed Frequency
A	90
B	110
C	100
D	100

Test at the 0.05 significance level if the preferences differ.

Solution:

- **Step 1: Set up hypotheses**
 - H_0 : Preferences are equally distributed.
 - H_a : Preferences are not equally distributed.
- **Step 2: Calculate expected frequencies**

$$E_i = \frac{400}{4} = 100$$

- **Step 3: Calculate the Chi-Square statistic**

$$\begin{aligned}\chi^2 &= \frac{(90 - 100)^2}{100} + \frac{(110 - 100)^2}{100} + \frac{(100 - 100)^2}{100} + \frac{(100 - 100)^2}{100} \\ \chi^2 &= \frac{100}{100} + \frac{100}{100} + 0 + 0 = 2\end{aligned}$$

Step 4: Compare with critical value Degrees of freedom = 4 - 1 = 3. The critical value for df = 3 and $\alpha=0.05$ is 7.815.

Since $\chi^2 = 2 < 7.815$, we fail to reject H_0 .

Conclusion: Preferences do not significantly differ.

F-Test (Variance Comparison)

Problem 3: F-Test for Two Sample Variances

Two machines are producing metal rods. A random sample of 10 rods from machine 1 has a variance of 4, and a random sample of 12 rods from machine 2 has a variance of 6. Test at the 0.05 significance level if the variances of the machines differ.

Solution:

- **Step 1: Set up hypotheses**

- $H_0: \sigma_1^2 = \sigma_2^2$ (variances are equal).
- $H_a: \sigma_1^2 \neq \sigma_2^2$ (variances are not equal).

- **Step 2: Compute the F-statistic**

$$F = \frac{s_2^2}{s_1^2} = \frac{6}{4} = 1.5$$

- **Step 3: Find critical value** The degrees of freedom are $df1=n1-1=10-1=9$ and $df2=n2-1=12-1=11$. The critical value at $\alpha=0.05$ is 3.18 for $df1=9$ and $df2=11$.

Since $F=1.5 < 3.18$, we fail to reject H_0 .

- **Conclusion:** There is no significant difference in the variances of the two machines.
-

Problem 4: F-Test for Exam Scores

A teacher wants to know if two classes have different variability in their exam scores. A random sample from class 1 ($n=15$) has a variance of 16, and a sample from class 2 ($n=18$) has a variance of 9. Test the hypothesis at the 0.05 significance level.

Solution:

- **Step 1: Set up hypotheses**

- $H_0: \sigma_1^2 = \sigma_2^2$ (variances are equal).
- $H_a: \sigma_1^2 \neq \sigma_2^2$ (variances are not equal).

- **Step 2: Compute the F-statistic**

$$F = \frac{s_1^2}{s_2^2} = \frac{16}{9} \approx 1.78$$

- **Step 3: Find critical value** Degrees of freedom: df1=14, df2=17. The critical value from the F-distribution table at $\alpha=0.05$ is approximately 2.46.

Since $F=1.78 < 2.46$, we fail to reject H_0

Conclusion: The variability in exam scores is not significantly different between the two classes.

ANALYSIS OF VARIANCE (ANOVA)

Analysis of variance (abbreviated as ANOVA) is an extremely useful technique concerning researches in the fields of economics, biology, education, psychology, sociology, business/industry and in researches of several other disciplines. This technique is used when multiple sample cases are involved. As stated earlier, the significance of the difference between the means of two samples can be judged through either z -test or the t -test, but the difficulty arises when we happen to examine the significance of the difference amongst more than two sample means at the same time. The ANOVA technique enables us to perform this simultaneous test and as such is considered to be an important tool of analysis in the hands of a researcher. Using this technique, one can draw inferences about whether the samples have been drawn from populations having the same mean.

The ANOVA technique is important in the context of all those situations where we want to compare more than two populations such as in comparing the yield of crop from several varieties of seeds, the gasoline mileage of four automobiles, the smoking habits of five groups of university students and so on. In such circumstances one generally does not want to consider all possible combinations of two populations at a time for that would require a great number of tests before we would be able to arrive at a decision. This would also consume lot of time and money, and even then certain relationships may be left unidentified (particularly the interaction effects). Therefore, one quite often utilizes the ANOVA technique and through it investigates the differences among the means of all the populations simultaneously.

WHAT IS ANOVA?

Professor R.A. Fisher was the first man to use the term ‘Variance’* and, in fact, it was he who developed a very elaborate theory concerning ANOVA, explaining its usefulness in practical field. Later on Professor Snedecor and many others contributed to the development of this technique. ANOVA is essentially a procedure for testing the difference among different groups of data for homogeneity. “The essence of ANOVA is that the total amount of variation in a set of data is broken down into two types, that amount which can be attributed to chance and that amount which can be attributed to specified causes.”¹ There may be variation between samples and also within sample items. ANOVA consists in splitting the variance for analytical purposes. Hence, it is a method of analysing the variance to which a response is subject into its

various components corresponding to various sources of variation. Through this technique one can explain whether various varieties of seeds or fertilizers or soils differ significantly so that a policy decision could be taken accordingly, concerning a particular variety in the context of agriculture researches. Similarly, the differences in various types of feed prepared for a particular class of animal or various types of drugs manufactured for curing a specific disease may be studied and judged to be significant or not through the application of ANOVA technique. Likewise, a manager of a big concern can analyse the performance of various salesmen of his concern in order to know whether their performances differ significantly

THE BASIC PRINCIPLE OF ANOVA

The basic principle of ANOVA is to test for differences among the means of the populations by examining the amount of variation within each of these samples, relative to the amount of variation between the samples. In terms of variation within the given population, it is assumed that the values of (X_{ij}) differ from the mean of this population only because of random effects i.e., there are influences on (X_{ij}) which are unexplainable, whereas in examining differences between populations we assume that the difference between the mean of the j th population and the grand mean is attributable to what is called a ‘specific factor’ or what is technically described as treatment effect. Thus while using ANOVA, we assume that each of the samples is drawn from a normal population and that each of these populations has the same variance. We also assume that all factors other than the one or more being tested are effectively controlled. This, in other words, means that we assume the absence of many factors that might affect our conclusions concerning the factor(s) to be studied.

In short, we have to make two estimates of population variance viz., one based on between samples variance and the other based on within samples variance. Then the said two estimates of population variance are compared with F -test, wherein we work out.

$$F = \frac{\text{Estimate of population variance based on between samples variance}}{\text{Estimate of population variance based on within samples variance}}$$

<i>Source of variation</i>	<i>Sum of squares (SS)</i>	<i>Degrees of freedom (d.f.)</i>	<i>Mean Square (MS)</i> <i>(This is SS divided by d.f.) and is an estimation of variance to be used in F-ratio</i>	<i>F-ratio</i>
Between samples or categories	$n_1(\bar{X}_1 - \bar{\bar{X}})^2 + \dots + n_k(\bar{X}_k - \bar{\bar{X}})^2$	$(k - 1)$	$\frac{SS \text{ between}}{(k - 1)}$	$\frac{MS \text{ between}}{MS \text{ within}}$
Within samples or categories	$\sum(X_{1i} - \bar{X}_1)^2 + \dots + \sum(X_{ki} - \bar{X}_k)^2$ $i = 1, 2, 3, \dots$	$(n - k)$	$\frac{SS \text{ within}}{(n - k)}$	
Total	$\sum(X_{ij} - \bar{\bar{X}})^2$ $i = 1, 2, \dots$ $j = 1, 2, \dots$	$(n - 1)$		

Illustration 1

Set up an analysis of variance table for the following per acre production data for three varieties of wheat, each grown on 4 plots and state if the variety differences are significant.

<i>Plot of land</i>	<i>Per acre production data</i>		
	<i>Variety of wheat</i>		
	<i>A</i>	<i>B</i>	<i>C</i>
1	6	5	5
2	7	5	4
3	3	3	3
4	8	7	4

Solution: We can solve the problem by the direct method or by short-cut method, but in each case we shall get the same result. We try below both the methods.

Solution through direct method: First we calculate the mean of each of these samples:

$$\bar{X}_1 = \frac{6 + 7 + 3 + 8}{4} = 6$$

$$\bar{X}_2 = \frac{5 + 5 + 3 + 7}{4} = 5$$

$$\bar{X}_3 = \frac{5 + 4 + 3 + 4}{4} = 4$$

Mean of the sample means or $\bar{\bar{X}} = \frac{\bar{X}_1 + \bar{X}_2 + \bar{X}_3}{k}$

$$= \frac{6 + 5 + 4}{3} = 5$$

Now we work out SS between and SS within samples:

$$\begin{aligned} SS \text{ between} &= n_1(\bar{X}_1 - \bar{\bar{X}})^2 + n_2(\bar{X}_2 - \bar{\bar{X}})^2 + n_3(\bar{X}_3 - \bar{\bar{X}})^2 \\ &= 4(6 - 5)^2 + 4(5 - 5)^2 + 4(4 - 5)^2 \\ &= 4 + 0 + 4 \\ &= 8 \end{aligned}$$

$$\begin{aligned} SS \text{ within} &= \sum(X_{1i} - \bar{X}_1)^2 + \sum(X_{2i} - \bar{X}_2)^2 + \sum(X_{3i} - \bar{X}_3)^2, \quad i = 1, 2, 3, 4 \\ &= \{(6 - 6)^2 + (7 - 6)^2 + (3 - 6)^2 + (8 - 6)^2\} \\ &\quad + \{(5 - 5)^2 + (5 - 5)^2 + (3 - 5)^2 + (7 - 5)^2\} \\ &\quad + \{(5 - 4)^2 + (4 - 4)^2 + (3 - 4)^2 + (4 - 4)^2\} \\ &= \{0 + 1 + 9 + 4\} + \{0 + 0 + 4 + 4\} + \{1 + 0 + 1 + 0\} \\ &= 14 + 8 + 2 \\ &= 24 \end{aligned}$$

$$\begin{aligned}
SS \text{ for total variance} &= \sum \left(X_{ij} - \bar{\bar{X}} \right)^2 \quad i = 1, 2, 3\dots \\
&\quad j = 1, 2, 3\dots \\
&= (6 - 5)^2 + (7 - 5)^2 + (3 - 5)^2 + (8 - 5)^2 \\
&\quad + (5 - 5)^2 + (5 - 5)^2 + (3 - 5)^2 \\
&\quad + (7 - 5)^2 + (5 - 5)^2 + (4 - 5)^2 \\
&\quad + (3 - 5)^2 + (4 - 5)^2 \\
&= 1 + 4 + 4 + 9 + 0 + 0 + 4 + 4 + 0 + 1 + 4 + 1 \\
&= 32
\end{aligned}$$

Alternatively, it (SS for total variance) can also be worked out thus:

$$SS \text{ for total} = SS \text{ between} + SS \text{ within}$$

$$\begin{aligned}
&= 8 + 24 \\
&= 32
\end{aligned}$$

<i>Source of variation</i>	<i>SS</i>	<i>d.f.</i>	<i>MS</i>	<i>F-ratio</i>	<i>5% F-limit (from the F-table)</i>
Between sample	8	$(3 - 1) = 2$	$8/2 = 4.00$	$4.00/2.67 = 1.5$	$F(2, 9) = 4.26$
Within sample	24	$(12 - 3) = 9$	$24/9 = 2.67$		
Total	32	$(12 - 1) = 11$			

The above table shows that the calculated value of F is 1.5 which is less than the table value of 4.26 at 5% level with d.f. being $v_1 = 2$ and $v_2 = 9$ and hence could have arisen due to chance. This analysis supports the null-hypothesis of no difference in sample means. We may, therefore, conclude that the difference in wheat output due to varieties is insignificant and is just a matter of chance.

Factor Analysis

What is Factor Analysis?

Factor analysis, a method within the realm of statistics and part of the general linear model (GLM), serves to condense numerous variables into a smaller set of factors. By doing so, it captures the maximum shared variance among the variables and condenses them into a unified score, which can subsequently be utilized for further analysis. Factor analysis operates under several assumptions: linearity in relationships, absence of multicollinearity among variables, inclusion of relevant variables in the analysis, and genuine correlations between variables and factors. While multiple methods exist, principal component analysis stands out as the most prevalent approach in practice.

What does Factor mean in Factor Analysis?

In the context of factor analysis, a “factor” refers to an underlying, unobserved variable or latent construct that represents a common source of variation among a set of observed variables. These observed variables, also known as indicators or manifest variables, are the measurable variables that are directly observed or measured in a study.

How to do Factor Analysis (Factor Analysis Steps)?

Factor analysis is a statistical method used to describe variability among observed, correlated variables in terms of a potentially lower number of unobserved variables called factors. Here are the general steps involved in conducting a factor analysis:

1. Determine the Suitability of Data for Factor Analysis

- **Bartlett’s Test:** Check the significance level to determine if the correlation matrix is suitable for factor analysis.
- **Kaiser-Meyer-Olkin (KMO) Measure:** Verify the sampling adequacy. A value greater than 0.6 is generally considered acceptable.

2. Choose the Extraction Method

- **Principal Component Analysis (PCA):** Used when the main goal is data reduction.
- **Principal Axis Factoring (PAF):** Used when the main goal is to identify underlying factors.

3. Factor Extraction

- Use the chosen extraction method to identify the initial factors.
- Extract eigenvalues to determine the number of factors to retain. Factors with eigenvalues greater than 1 are typically retained in the analysis.
- Compute the initial factor loadings.

4. Determine the Number of Factors to Retain

- **Scree Plot:** Plot the eigenvalues in descending order to visualize the point where the plot levels off (the “elbow”) to determine the number of factors to retain.

- **Eigenvalues:** Retain factors with eigenvalues greater than 1.

5. Factor Rotation

- **Orthogonal Rotation (Varimax, Quartimax):** Assumes that the factors are uncorrelated.
- **Oblique Rotation (Promax, Oblimin):** Allows the factors to be correlated.
- Rotate the factors to achieve a simpler and more interpretable factor structure.
- Examine the rotated factor loadings.

6. Interpret and Label the Factors

- Analyze the rotated factor loadings to interpret the underlying meaning of each factor.
- Assign meaningful labels to each factor based on the variables with high loadings on that factor.

7. Compute Factor Scores (if needed)

- Calculate the factor scores for each individual to represent their value on each factor.

8. Report and Validate the Results

- Report the final factor structure, including factor loadings and communalities.
- Validate the results using additional data or by conducting a confirmatory factor analysis if necessary.

Factor Analysis in R programming

Factor Analysis (FA) is a statistical method that is used to analyze the underlying structure of a set of variables. It is a method of data reduction that seeks to explain the correlations among many variables in terms of a smaller number of unobservable (latent) variables, known as factors. In R Programming Language, the *psych package* provides a variety of functions for performing factor analysis.

Factor analysis involves several steps:

1. **Data preparation:** The data are usually standardized (i.e., scaled) to make sure that the variables are on a common scale and have equal weight in the analysis.
2. **Factor Extraction:** The factors are identified based on their ability to explain the variance in the data. There are several methods for extracting factors, including principal components analysis (PCA), maximum likelihood estimate(MLE), and minimum residuals (MR).
3. **Factor Rotation:** The factors are usually rotated to make their interpretation easier. The most common method of rotation is Varimax rotation, which tries to maximize the variance of the factor loadings.
4. **Factor interpretation:** The final step involves interpreting the factors and their loadings (i.e., the correlation between each variable and each factor). The loadings represent the

degree to which each variable is associated with each factor.

Loading the Data

First, we need to load the data that we want to analyze. For this example, we will use the iris dataset that comes with R. This dataset contains measurements of the sepal length, sepal width, petal length, and petal width of three different species of iris flowers.

```
# Load the dataset
data(iris)

# View the first few rows of the dataset
head(iris)

> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5         1.4         0.2  setosa
2          4.9         3.0         1.4         0.2  setosa
3          4.7         3.2         1.3         0.2  setosa
4          4.6         3.1         1.5         0.2  setosa
5          5.0         3.6         1.4         0.2  setosa
6          5.4         3.9         1.7         0.4  setosa
>
```

First five rows of the dataset

Data Preparation

Before conducting factor analysis, we need to prepare the data by scaling the variables to have a mean of zero and a standard deviation of one. This is important because factor analysis is sensitive to differences in scale between variables.

```
# Scale the data
iris_scaled <- scale(iris[,1:4])
```

Determining the Number of Factors

The next step is to determine the number of factors to extract from the data. This can be done using a variety of methods, such as the Kaiser criterion, scree plot, or parallel analysis. In this example, we will use the Kaiser criterion, which suggests extracting factors with eigenvalues greater than one.

```
# Perform factor analysis
library(psych)
fa <- fa(r = iris_scaled,
          nfactors = 4,
          rotate = "varimax")
summary(fa)
```

Output:

```
Factor analysis with Call: fa(r = iris_scaled, nfactors = 4, rotate = "varimax")

Test of the hypothesis that 4 factors are sufficient.
The degrees of freedom for the model is -4 and the objective function was  0
The number of observations was 150 with Chi Square = 0 with prob < NA

The root mean square of the residuals (RMSA) is 0
The df corrected root mean square of the residuals is NA

Tucker Lewis Index of factoring reliability = 1.009
```

The output of the summary() function shows the results of the factor analysis, including the number of factors extracted, the eigenvalues for each factor, and the percentage of variance explained by each factor.

This summary shows that the factor analysis extracted 2 factors, and provides the standardized loadings (or factor loadings) for each variable on each factor. It also shows the eigenvalues and proportion of variance explained by each factor, as well as the results of a test of the hypothesis that 2 factors are sufficient. The goodness of fit statistic is also reported.

Interpreting the Results of Factor Analysis

Once the factor analysis is complete, we can interpret the results by examining the factor loadings, which represent the correlations between the observed variables and the extracted factors. In general, loadings greater than 0.4 or 0.5 are considered significant.

```
# View the factor loadings
fa$loadings
```

Output:

```
Loadings:
      MR1    MR2    MR3    MR4
Sepal.Length  0.997
Sepal.Width   -0.108  0.757
Petal.Length  0.861 -0.413  0.288
Petal.Width   0.801 -0.317  0.492

      MR1    MR2    MR3    MR4
SS loadings   2.389  0.844  0.332  0.000
Proportion Var 0.597  0.211  0.083  0.000
Cumulative Var 0.597  0.808  0.891  0.891
```

The output of the loadings function shows the factor loadings for each variable and factor. We can interpret these loadings to identify the underlying factors that explain the correlations among the observed variables. In this example, it appears that the first factor is strongly associated with petal length and petal width, while the second factor is strongly associated with sepal length and sepal width.

Validating the Results of Factor Analysis

Finally, it is important to validate the results of the factor analysis by checking the assumptions of the technique, such as normality and linearity. Additionally, it is important to examine the factor structure for different subsets of the data to ensure that the results are consistent and stable.

```
# examine factor structure for
# different subsets of the data
subset1 <- subset(iris[,1:4],
                  iris$Sepal.Length < mean(iris$Sepal.Length))
fa1 <- fa(subset1, nfactors = 4)
print(fa1)
```

Output:

```
Factor Analysis using method = minres
Call: fa(r = subset1, nfactors = 4)
Standardized loadings (pattern matrix) based upon correlation matrix
      MR1   MR2   MR3  MR4   h2    u2 com
Sepal.Length  0.66  0.61 -0.12   0  0.82  0.178 2.1
Sepal.Width   -0.68  0.61  0.11   0  0.85  0.150 2.0
Petal.Length   1.00  0.00  0.00   0  1.00  0.005 1.0
Petal.Width    0.97  0.01  0.16   0  0.97  0.031 1.1

      MR1   MR2   MR3  MR4
SS loadings     2.85  0.74  0.05  0.00
Proportion Var   0.71  0.18  0.01  0.00
Cumulative Var   0.71  0.90  0.91  0.91
Proportion Explained  0.78  0.20  0.01  0.00
Cumulative Proportion 0.78  0.99  1.00  1.00

Mean item complexity = 1.5
Test of the hypothesis that 4 factors are sufficient.
```

The degrees of freedom for the null model are 6 and the objective function was 4.57 with Chi Square of 351.02

The degrees of freedom for the model are -4 and the objective function was 0

The root mean square of the residuals (RMSR) is 0

The df corrected root mean square of the residuals is NA

The harmonic number of observations is 80 with the empirical chi square 0 with prob < NA

The total number of observations was 80 with Likelihood Chi Square = 0 with prob < NA

Tucker Lewis Index of factoring reliability = 1.018

Fit based upon off diagonal values = 1

Measures of factor score adequacy

	MR1	MR2	MR3	MR4
Correlation of (regression) scores with factors	1.00	0.91	0.69	0
Multiple R square of scores with factors	1.00	0.82	0.47	0
Minimum correlation of possible factor scores	0.99	0.64	-0.05	-1

```
subset2 <- subset(iris[,1:4],
                   iris$Sepal.Length >= mean(iris$Sepal.Length))
fa2 <- fa(subset2, nfactors = 4)
print(fa2)
```

Output:

```
Factor Analysis using method = minres
Call: fa(r = subset2, nfactors = 4)
Standardized loadings (pattern matrix) based upon correlation matrix
      MR1    MR2    MR3   MR4   h2   u2 com
Sepal.Length 0.76 -0.37  0.26   0  0.78 0.222 1.7
Sepal.Width  0.50  0.36  0.34   0  0.49 0.507 2.6
Petal.Length 0.95 -0.23 -0.22   0  1.00 0.005 1.2
Petal.Width  0.82  0.39 -0.20   0  0.86 0.144 1.6

      MR1    MR2    MR3   MR4
SS loadings     2.39  0.46  0.27 0.00
Proportion Var  0.60  0.12  0.07 0.00
Cumulative Var 0.60  0.71  0.78 0.78
Proportion Explained 0.76  0.15  0.09 0.00
Cumulative Proportion 0.76  0.91  1.00 1.00

Mean item complexity = 1.8
Test of the hypothesis that 4 factors are sufficient.
```

```
The degrees of freedom for the null model are 6 and the objective function was
1.97 with Chi Square of 131.96
The degrees of freedom for the model are -4 and the objective function was 0

The root mean square of the residuals (RMSR) is 0
The df corrected root mean square of the residuals is NA

The harmonic number of observations is 70 with the empirical chi square 0 with
prob < NA
The total number of observations was 70 with Likelihood Chi Square = 0 with
prob < NA

Tucker Lewis Index of factoring reliability = 1.05
Fit based upon off diagonal values = 1
Measures of factor score adequacy
      MR1    MR2    MR3   MR4
Correlation of (regression) scores with factors 0.98  0.86  0.75   0
Multiple R square of scores with factors       0.96  0.75  0.57   0
Minimum correlation of possible factor scores 0.92  0.49  0.14  -1
```

```
# display variance explained by each factor
print(fa$Vaccounted)
```

Output:

	MR1	MR2	MR3	MR4
SS loadings	2.8853608	0.5816336	0.09819492	4.000000e-30
Proportion Var	0.7213402	0.1454084	0.02454873	1.000000e-30
Cumulative Var	0.7213402	0.8667486	0.89129733	8.912973e-01
Proportion Explained	0.8093149	0.1631424	0.02754269	1.121960e-30
Cumulative Proportion	0.8093149	0.9724573	1.00000000	1.000000e+00

Factor Analysis using factanal() function:

The *factanal()* function is used to perform factor analysis on a data set. The factanal() function takes several arguments described below

Syntax:

factanal(x, factors, rotation, scores, covmat)

where,

- *x* – The data set to be analyzed.
- *factors* – The number of factors to extract.
- *rotation* – The rotation method to use. Popular rotation methods include varimax, oblimin, and promax.
- *scores* – Whether to compute factor scores for each observation.
- *covmat* – A covariance matrix to use instead of the default correlation matrix.

The output of factanal() function includes several pieces of information, including:

- **Uniquenesses:** The amount of variance in each variable that is not accounted for by the factors.
- **Loadings:** The correlations between each variable and each factor.
- **Communalities:** The amount of variance in each variable that is accounted for by the factors.
- **Eigenvalues:** The amount of variance explained by each factor.
- **Factor Correlations:** The correlations between the factors.

Here is an example code snippet that demonstrates how to use factanal() function in R:

```

# Install the required package
install.packages("psych")

# Load the psych package for
# data analysis and visualization
library(psych)

# Load the mtcars dataset
data(mtcars)

# Perform factor analysis on the mtcars dataset
factor_analysis <- factanal(mtcars, factors = 3, rotation = "varimax")

# Print the results
print(factor_analysis)

```

Output:

```

Call:
factanal(x = mtcars, factors = 3, rotation = "varimax")

Uniquenesses:
  mpg   cyl  disp    hp   drat    wt   qsec    vs     am   gear   carb
0.135 0.055 0.090 0.127 0.290 0.060 0.051 0.223 0.208 0.125 0.158

Loadings:
  Factor1 Factor2 Factor3
mpg    0.643 -0.478 -0.473
cyl   -0.618  0.703  0.261
disp  -0.719  0.537  0.323
hp    -0.291  0.725  0.513
drat   0.804 -0.241
wt    -0.778  0.248  0.524
qsec -0.177 -0.946 -0.151
vs     0.295 -0.805 -0.204
am     0.880
gear   0.908        0.224
carb   0.114  0.559  0.719

```

```

      Factor1 Factor2 Factor3
SS loadings     4.380   3.520   1.578
Proportion Var  0.398   0.320   0.143
Cumulative Var 0.398   0.718   0.862

Test of the hypothesis that 3 factors are sufficient.
The chi square statistic is 30.53 on 25 degrees of freedom.
The p-value is 0.205

```

In this example, we load the psych package, which provides functions for data analysis and visualization, and the mtcars data set, which contains information about different car models. We then use the factanal() function to perform factor analysis on the mtcars data set, specifying that we want to extract three factors and use the varimax rotation method. Finally, we print the results of the factor analysis.

Conclusion

In conclusion, factor analysis is a useful statistical technique for identifying underlying factors or latent variables that explain the correlations among a set of observed variables. In R programming, the psych package provides a range of functions for conducting factor analysis, which can be used to extract meaningful insights from complex datasets.

Unit 5

Introduction

to R

Programming

language

Topics

Getting R

Managing R

Arithmetic and Matrix Operations

Introduction to Functions

Control structures

Working with Objects and Data :

Introduction to Objects,

Manipulating Objects,

Constructing Data Objects,

types of Data items,

Structure of Data items,

Reading and Getting Data,

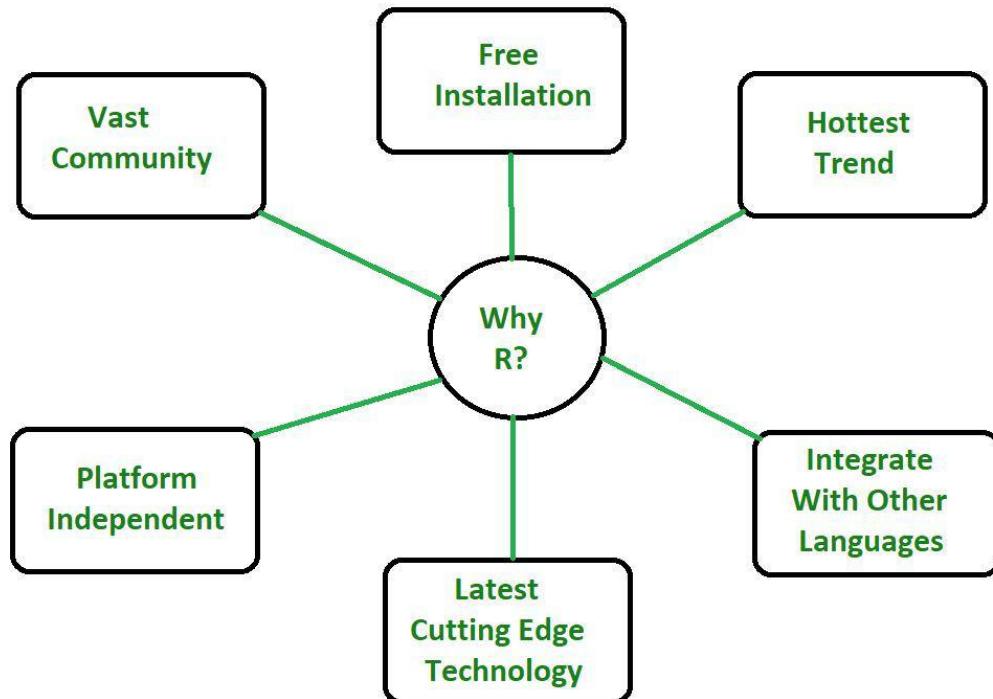
Manipulating Data,

Storing data

R Programming Language – Introduction

The R Language stands out as a powerful tool in the modern era of statistical computing and data analysis. Widely embraced by statisticians, data scientists, and researchers, the R Language offers an extensive suite of packages and libraries tailored for data manipulation, statistical modeling, and visualization. In this article, we explore the features, benefits, and applications of the R Programming Language, shedding light on why it has become an indispensable asset for data-driven professionals across various industries.

R programming language is an implementation of the S programming language. It also combines with lexical scoping semantics inspired by Scheme. Moreover, the project was conceived in 1992, with an initial version released in 1995 and a stable beta version in 2000.



R programming is a leading tool for machine learning, statistics, and data analysis, allowing for the easy creation of objects, functions, and packages. Designed by Ross Ihaka and Robert Gentleman at the University of Auckland and developed by the R Development Core Team, R Language is platform-independent and open-source, making it accessible for use across all operating systems without licensing costs. Beyond its capabilities as a statistical package, R integrates with other languages like C and C++, facilitating interaction with various data sources and statistical tools. With a growing community of users and high demand in the Data Science job market, R is one of the most sought-after programming languages today. Originating as an implementation of the S programming language with influences from Scheme, R has evolved since its conception in 1992, with its first stable beta version released in 2000.

Why Use R Language?

The R Language is a powerful tool widely used for data analysis, statistical computing, and machine learning. Here are several reasons why professionals across various fields prefer R:

1. Comprehensive Statistical Analysis:

- R language is specifically designed for statistical analysis and provides a vast array of statistical techniques and tests, making it ideal for data-driven research.

2. Extensive Packages and Libraries:

- The R Language boasts a rich ecosystem of packages and libraries that extend its capabilities, allowing users to perform advanced data manipulation, visualization, and machine learning tasks with ease.

3. Strong Data Visualization Capabilities:

- R language excels in data visualization, offering powerful tools like ggplot2 and plotly, which enable the creation of detailed and aesthetically pleasing graphs and plots.

4. Open Source and Free:

- As an open-source language, R is free to use, which makes it accessible to everyone, from individual researchers to large organizations, without the need for costly licenses.

5. Platform Independence:

- The R Language is platform-independent, meaning it can run on various operating systems, including Windows, macOS, and Linux, providing flexibility in development environments.

6. Integration with Other Languages:

- R can easily integrate with other programming languages such as C, C++, Python, and Java, allowing for seamless interaction with different data sources and statistical packages.

7. Growing Community and Support:

- R language has a large and active community of users and developers who contribute to its continuous improvement and provide extensive support through forums, mailing lists, and online resources.

8. High Demand in Data Science:

- R is one of the most requested programming languages in the Data Science job market, making it a valuable skill for professionals looking to advance their careers in this field.

Features of R Programming Language

The R Language is renowned for its extensive features that make it a powerful tool for data analysis, statistical computing, and visualization. Here are some of the key features of R:

1. Comprehensive Statistical Analysis:

- R language provides a wide array of statistical techniques, including linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, and clustering.

2. Advanced Data Visualization:

- With packages like ggplot2, plotly, and lattice, R excels at creating complex and aesthetically pleasing data visualizations, including plots, graphs, and charts.

3. Extensive Packages and Libraries:

- The Comprehensive R Archive Network (CRAN) hosts thousands of packages that extend R's capabilities in areas such as machine learning, data manipulation, bioinformatics, and more.

4. Open Source and Free:

- R is free to download and use, making it accessible to everyone. Its open-source nature encourages community contributions and continuous improvement.

5. Platform Independence:

- R is platform-independent, running on various operating systems, including Windows, macOS, and Linux, which ensures flexibility and ease of use across different environments.

6. Integration with Other Languages:

- R language can integrate with other programming languages such as C, C++, Python, Java, and SQL, allowing for seamless interaction with various data sources and computational processes.

7. Powerful Data Handling and Storage:

- R efficiently handles and stores data, supporting various data types and structures, including vectors, matrices, data frames, and lists.

8. Robust Community and Support:

- R has a vibrant and active community that provides extensive support through forums, mailing lists, and online resources, contributing to its rich ecosystem of packages and documentation.

9. Interactive Development Environment (IDE):

- RStudio, the most popular IDE for R, offers a user-friendly interface with features like syntax highlighting, code completion, and integrated tools for plotting, history, and debugging.

10. Reproducible Research:

- R supports reproducible research practices with tools like R Markdown and Knitr, enabling users to create dynamic reports, presentations, and documents that combine code, text, and visualizations.

Advantages of R language

- R is the most comprehensive statistical analysis package. As new technology and concepts often appear first in R.
- As R programming language is an open source. Thus, you can run R anywhere and at any time.
- R programming language is suitable for GNU/Linux and Windows operating systems.
- R programming is cross-platform and runs on any operating system.
- In R, everyone is welcome to provide new packages, bug fixes, and code enhancements.

Disadvantages of R language

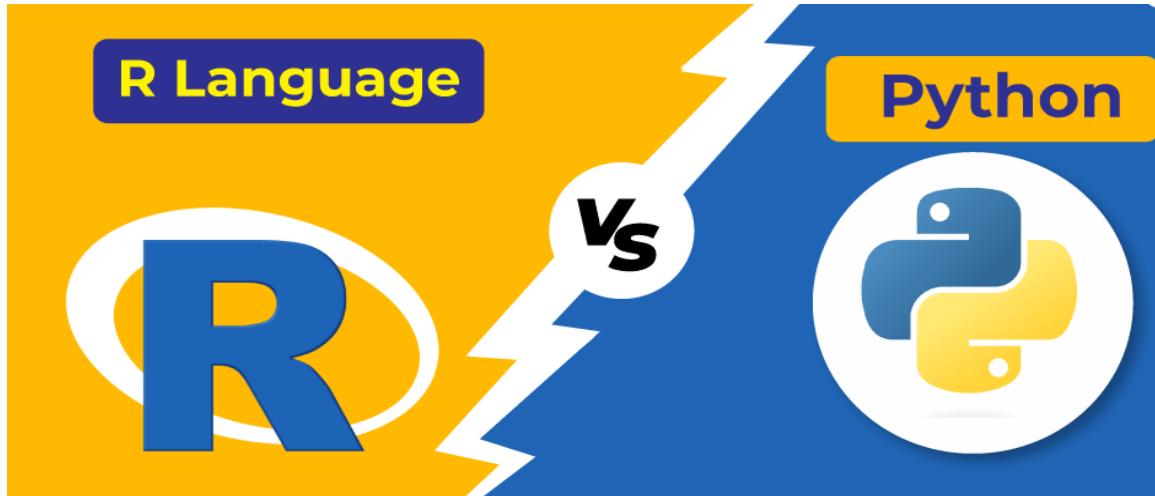
- In the R programming language, the standard of some packages is less than perfect.
- Although, R commands give little pressure on memory management. So R programming language may consume all available memory.
- In R basically, nobody to complain if something doesn't work.
- R programming language is much slower than other programming languages such as Python and MATLAB.

Applications of R language

- We use R for Data Science. It gives us a broad variety of libraries related to statistics. It also provides the environment for statistical computing and design.
- R is used by many quantitative analysts as its programming tool. Thus, it helps in data importing and cleaning.
- R is the most prevalent language. So many data analysts and research programmers use it. Hence, it is used as a fundamental tool for finance.
- Tech giants like Google, Facebook, Bing, Twitter, Accenture, Wipro, and many more using R nowadays.

R vs Python

R Programming Language and **Python** are both used extensively for Data Science. Both are very useful and open-source languages as well. For data analysis, statistical computing, and machine learning Both languages are strong tools with sizable communities and huge libraries for data science jobs. A theoretical comparison between R and Python is provided below:



R Programming Language is used for machine learning algorithms, linear regression, time series, statistical inference, etc. It was designed by Ross Ihaka and Robert Gentleman in 1993. R is an open-source programming language that is widely used as a statistical software and data analysis tool. R generally comes with the Command-line interface. R is available across widely used platforms like Windows, Linux, and macOS. Also, the R programming language is the latest cutting-edge tool.

Python Programming Language

Python is a widely-used general-purpose, high-level programming language. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

Difference between R Programming and Python Programming

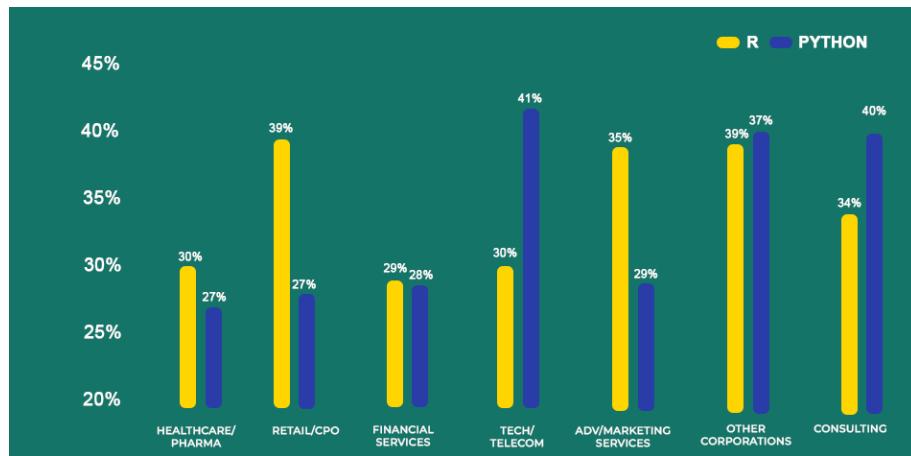
Below are some major differences between R and Python:

Feature	R	Python
Introduction	R is a language and environment for statistical programming which includes statistical computing and graphics.	Python is a general-purpose programming language for data analysis and scientific computing
Objective	It has many features which are useful for statistical analysis and representation.	It can be used to develop GUI applications and web applications as well as with embedded systems
Workability	It has many easy-to-use packages for performing tasks	It can easily perform matrix computation as well as optimization
Integrated development environment	Various popular R IDEs are Rstudio, RKward, R commander, etc.	Various popular Python IDEs are Spyder, Eclipse+Pydev, Atom, etc.
Libraries and packages	There are many packages and libraries like <u>ggplot2</u> , <u>caret</u> , etc.	Some essential packages and libraries are <u>Pandas</u> , <u>Numpy</u> , <u>Scipy</u> , etc.
Scope	It is mainly used for complex data analysis in data science.	It takes a more streamlined approach for data science projects.

Ecosystem in R Programming and Python Programming

Python supports a very large community of general-purpose data science. One of the most basic uses for data analysis, primarily because of the fantastic ecosystem of data-centric Python packages. Pandas and NumPy are one of those packages that make importing and analyzing, and visualization of data much easier.

R Programming has a rich ecosystem to use in standard machine learning and data mining techniques. It works in statistical analysis of large datasets, and it offers a number of different options for exploring data and It makes it easier to use probability distributions, apply different statistical tests.



R vs Python

Features	R	Python
Data collection	It is used for data analysts to import data from Excel, CSV, and text files.	It is used in all kinds of data formats including SQL tables
Data exploration	It optimized for the statistical analysis of large datasets	You can explore data with Pandas

Features	R	Python
Data modeling	It supports Tidyverse, making it easy to import, manipulate, visualize, and report on data.	You can use NumPy, SciPy, scikit-learn , TensorFlow
Data visualization	You can use ggplot2 and ggplot tools to plots complex scatter plots with regression lines.	You can use Matplotlib , Pandas, Seaborn

Statistical Analysis and Machine Learning In R and Python

Statistical analysis and machine learning are critical components of data science, involving the application of statistical methods, models, and techniques to extract insights, identify patterns, and draw meaningful conclusions from data. Both R and Python have widely used programming languages for statistical analysis, each offering a variety of libraries and packages to perform diverse statistical and machine learning tasks. Some comparison of statistical analysis and modeling capabilities in R and Python.

Capability	R	Python
Basic Statistics	Built-in functions (mean, median, etc.)	NumPy (mean, median, etc.)
Linear Regression	lm() function and Formulas	Statsmodels (OLS) Ordinary Least Squares (OLS) Method

Capability	R	Python
Generalized Linear Models (GLM)	glm() function	Statsmodels (GLM)
Time Series Analysis	Time Series packages (forecast)	Statsmodels (Time Series)
ANOVA and t-tests	Built-in functions (aov, t.test)	SciPy (ANOVA, t-tests)
Hypothesis Tests	Built-in functions (wilcox.test, etc.)	SciPy (Mann-Whitney, Kruskal-Wallis)
Principal Component Analysis (PCA)	princomp() function	scikit-learn (PCA)
Clustering (K-Means, Hierarchical)	kmeans(), hclust()	scikit-learn (KMeans, AgglomerativeClustering)
Decision Trees	rpart() function	scikit-learn (DecisionTreeClassifier)
Random Forest	randomForest() function	scikit-learn (RandomForestClassifier)

Advantages in R Programming and Python Programming

R Programming	Python Programming
It supports a large dataset for statistical analysis	General-purpose programming to use data analyze
Primary users are Scholar and R&D	Primary users are Programmers and developers
Support packages like <u>tidyverse</u> , ggplot2, caret, zoo	Support packages like pandas, scipy, scikit-learn, TensorFlow, caret
Support <u>RStudio</u> and It has a wide range of statistics and general data analysis and visualization capabilities.	Support Conda environment with Spyder, Ipython Notebook

Disadvantages in R Programming and Python Programming

R Programming	Python Programming
R is much more difficult as compared to Python because it mainly uses for statistics purposes.	Python does not have too many libraries for data science as compared to R.
R might not be as fast as languages like Python, especially for computationally intensive tasks and large-scale data processing.	Python might not be as specialized for statistics and data analysis as R. Some statistical functions and visualization capabilities might be more streamlined in R.
Memory management in R might not be	Python visualization capabilities

R Programming	Python Programming
as efficient as in some other languages, which can lead to performance issues and memory-related errors	might not be as polished and streamlined as those offered by R's ggplot2.

R and Python usages in Data Science

Python and R programming language is most useful in data science and it deals with identifying, representing, and extracting meaningful information from data sources to be used to perform some business logic with these languages. It has a popular package for Data collection, Data exploration, Data modeling, Data visualization, and statical analysis.

Introduction to R Studio

R Studio is an integrated development environment(IDE) for R. IDE is a GUI, where you can write your quotes, see the results and also see the variables that are generated during the course of programming.

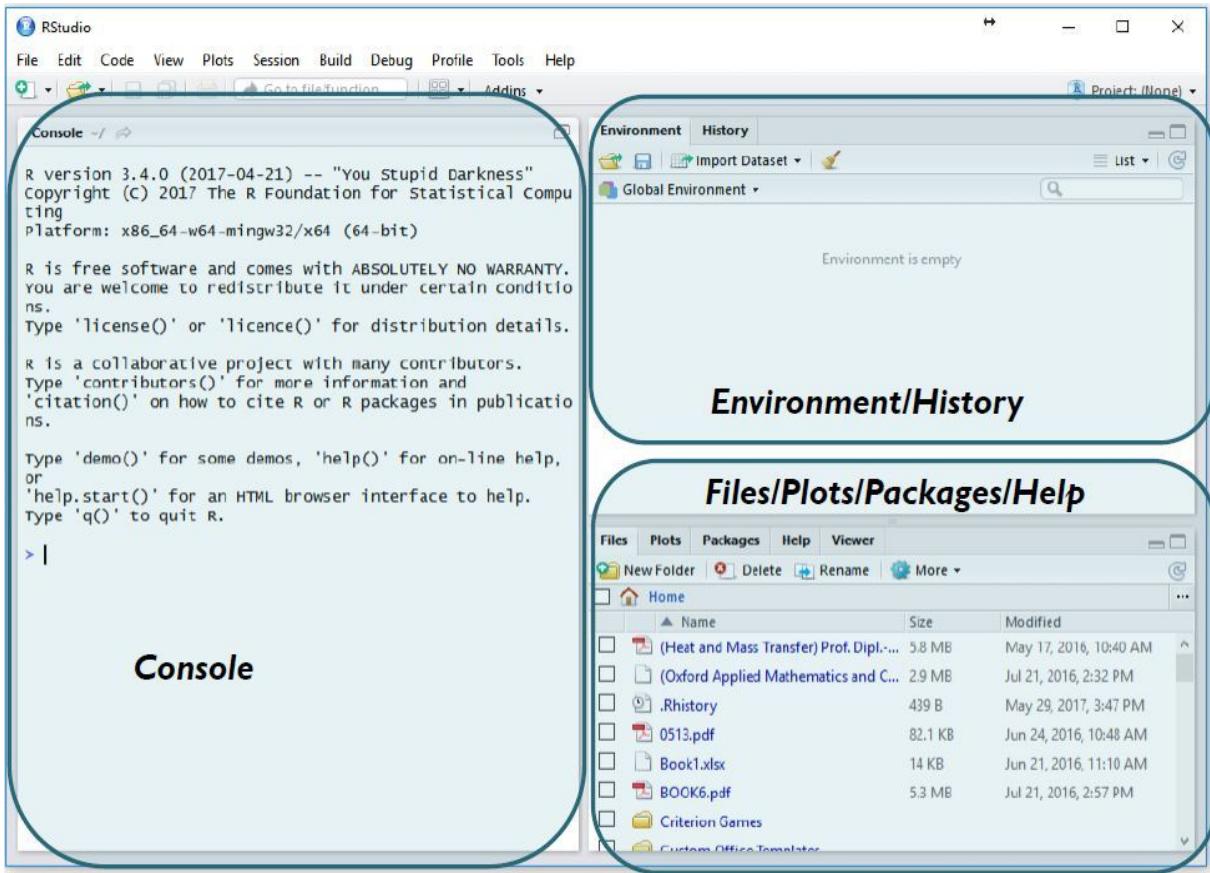
- R Studio is available as both Open source and Commercial software.
- R Studio is also available as both Desktop and Server versions.
- R Studio is also available for various platforms such as Windows, Linux, and macOS.

Introduction to R studio for beginners:

Rstudio is an open-source tool that provides Ide to use R language, and enterprise-ready professional software for data science teams to develop share the work with their team.

R Studio can be downloaded from its official Website (<https://rstudio.com/>) and instructions for installation are available on

After the installation process is over, the R Studio interface looks like:



- The console panel(left panel) is the place where R is waiting for you to tell it what to do, and see the results that are generated when you type in the commands.
- To the top right, you have the Environmental/History panel. It contains 2 tabs:
 - **Environment tab:** It shows the variables that are generated during the course of programming in a workspace that is temporary.
 - **History tab:** In this tab, you'll see all the commands that are used till now from the start of usage of R Studio.
- To the right bottom, you have another panel, which contains multiple tabs, such as files, plots, packages, help, and viewer.
 - The **Files tab** shows the files and directories that are available within the default workspace of R.
 - The **Plots tab** shows the plots that are generated during the course of programming.
 - The **Packages tab** helps you to look at what are the packages that are already installed in the R Studio and it also gives a user interface to install new packages.
 - The **Help tab** is the most important one where you can get help from the R Documentation on the functions that are in built-in R.
 - The final and last tab is that the **Viewer tab** which can be used to see the local web content that's generated using R.

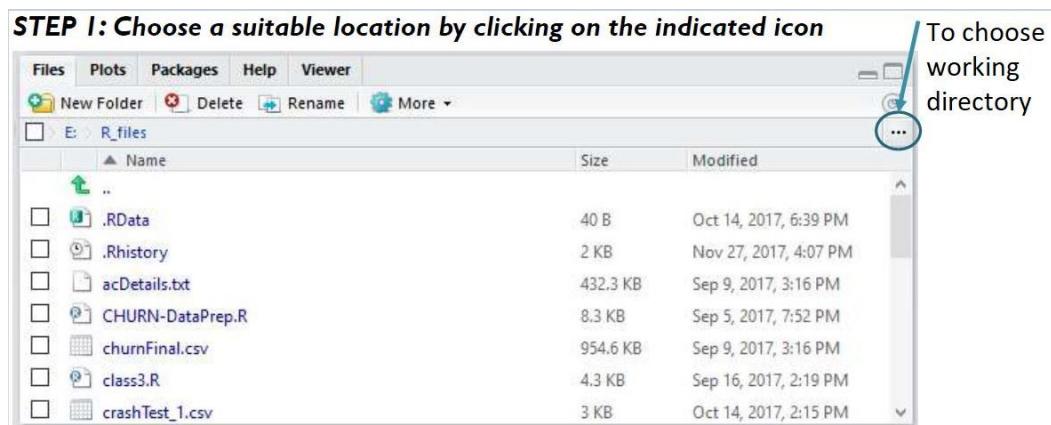
Features of R Studio

- A friendly user interface
- writing and storing reusable programmes
- All imported data and newly created objects (such as variables, functions, etc.) are easily accessible.
- Comprehensive assistance for any item Code autocompletion
- The capacity to organise and share your work with your partners more effectively through the creation of projects.
- Plot snippets
- Simple terminal and console switching
- Tracking of operational history
- There are numerous articles from RStudio Support on using the IDE.

Set the working directory in R Studio

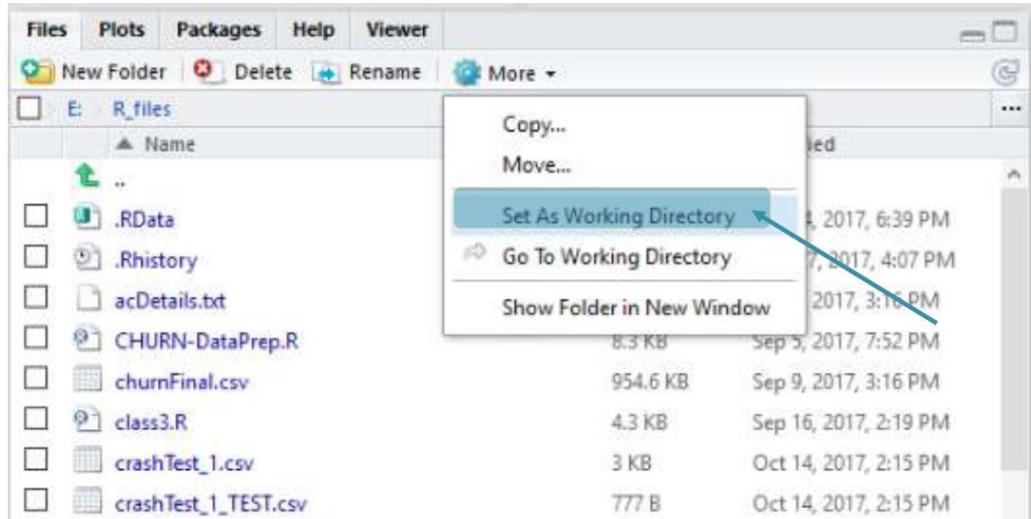
R is always pointed at a directory on our computer. We can find out which directory by running the getwd() function. Note: this function has no arguments. We can set the working directory manually in two ways:

- **The first way is to use the console and using the command setwd("directorypath").**
You can use this function setwd() and give the path of the directory which you want to be the working directory for R studio, in the double codes.
- **The second way is to set the working directory from the GUI.**
To set the working directory from the GUI you have to click on this 3 dots button. When you click this, this will open up a file browser, which will help you to choose your working directory.



Once you choose your working directory, you need to use this setting button in the more tab and click it and then you get a popup menu, where you need to select “Set as working directory”.

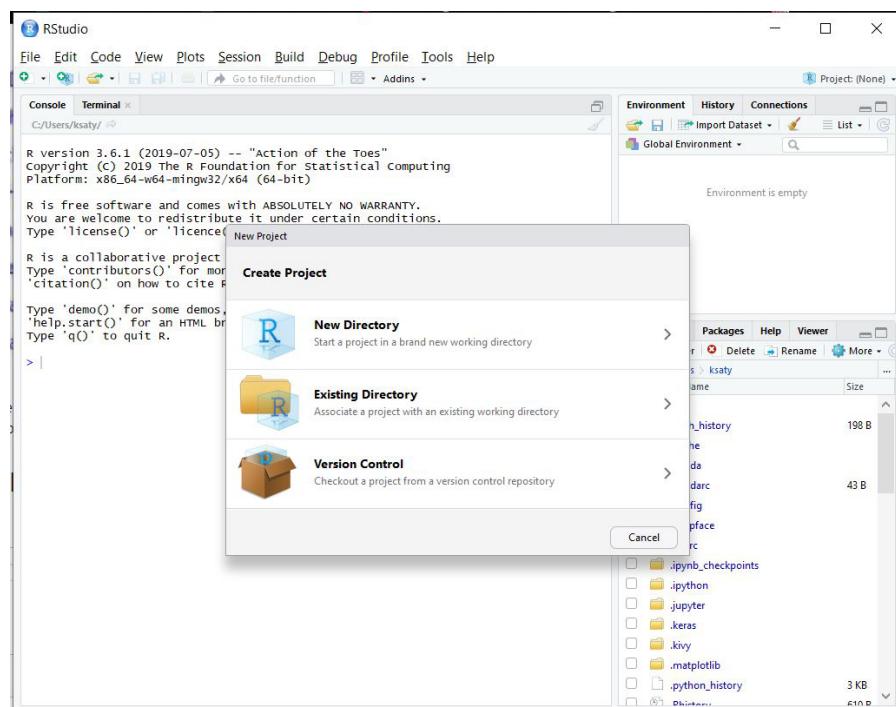
STEP 2: Once directory is chosen, select the more icon and choose “Set as Working Directory”



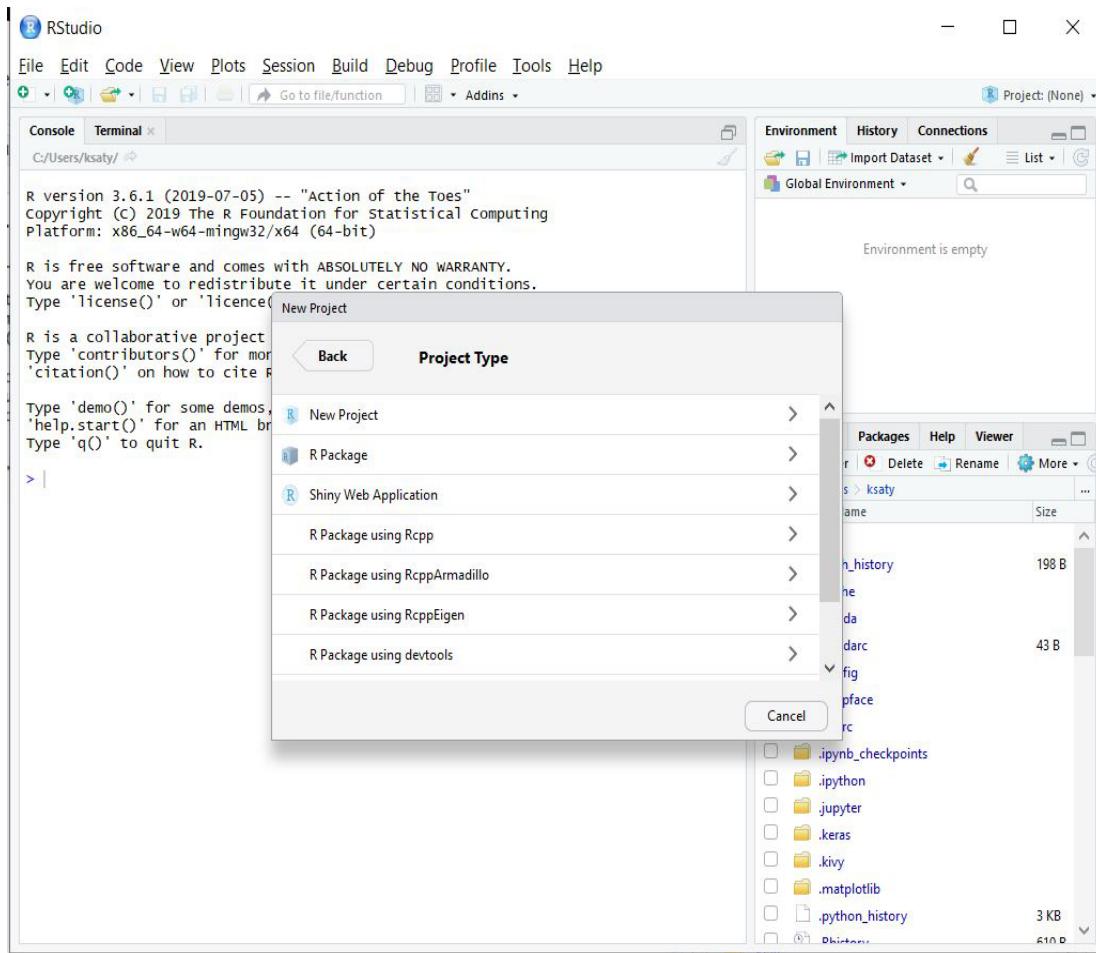
This will select the current directory, which you have chosen using this file browser as your working directory. Once you set the working directory, you are ready to program in R Studio.

Create an RStudio project

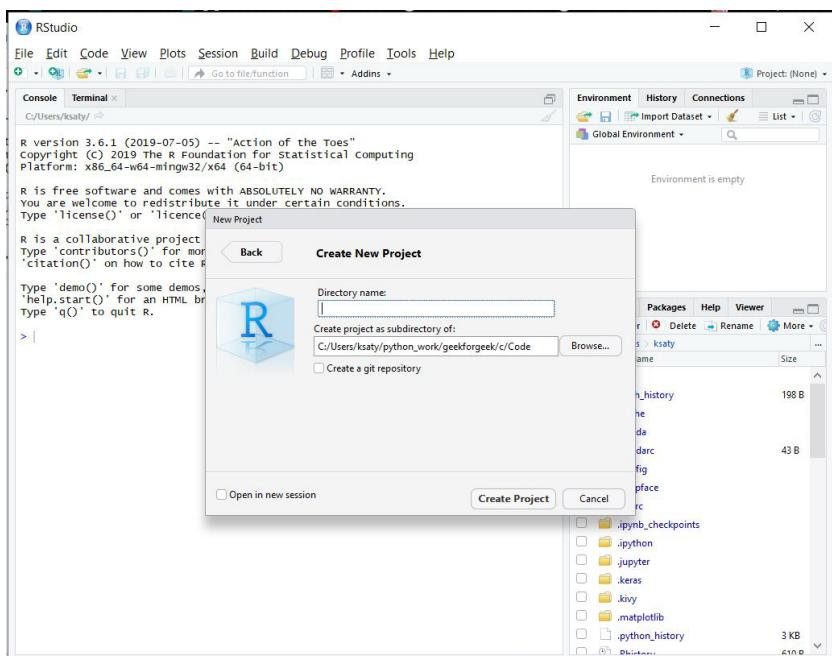
Step 1: Select the FILE option and select create option.



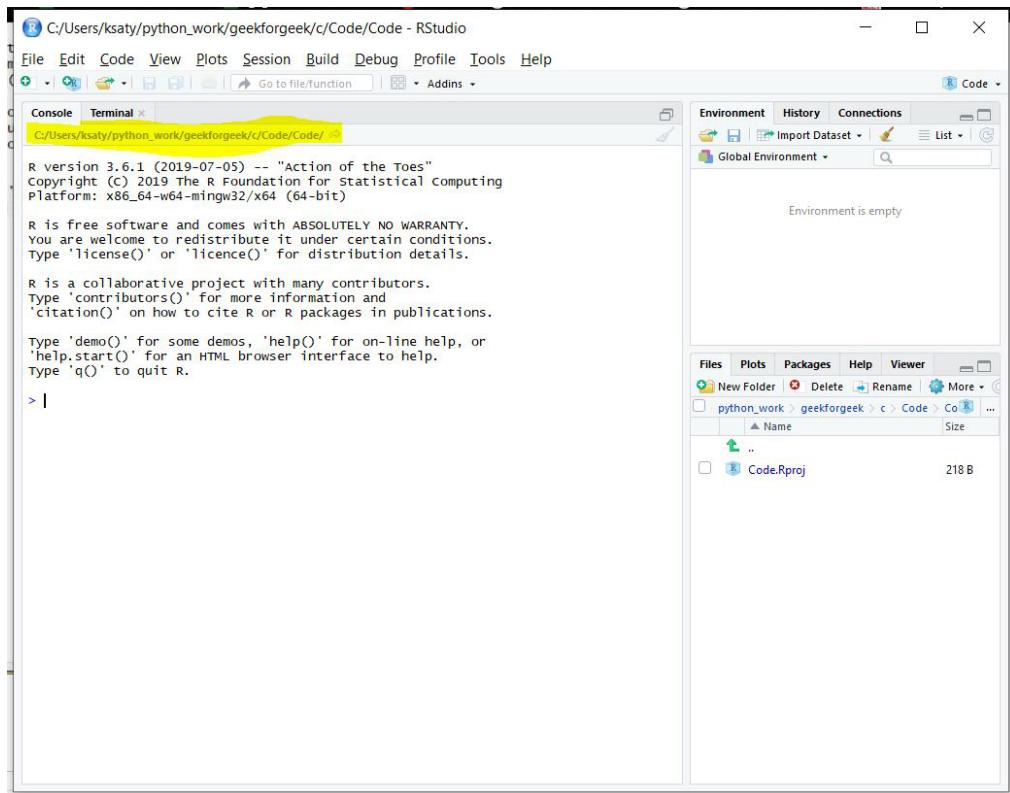
Step 2: Then select the New Project option.



Step 3: Then choose the path and directory name.



Finally, project are created in a specific location:

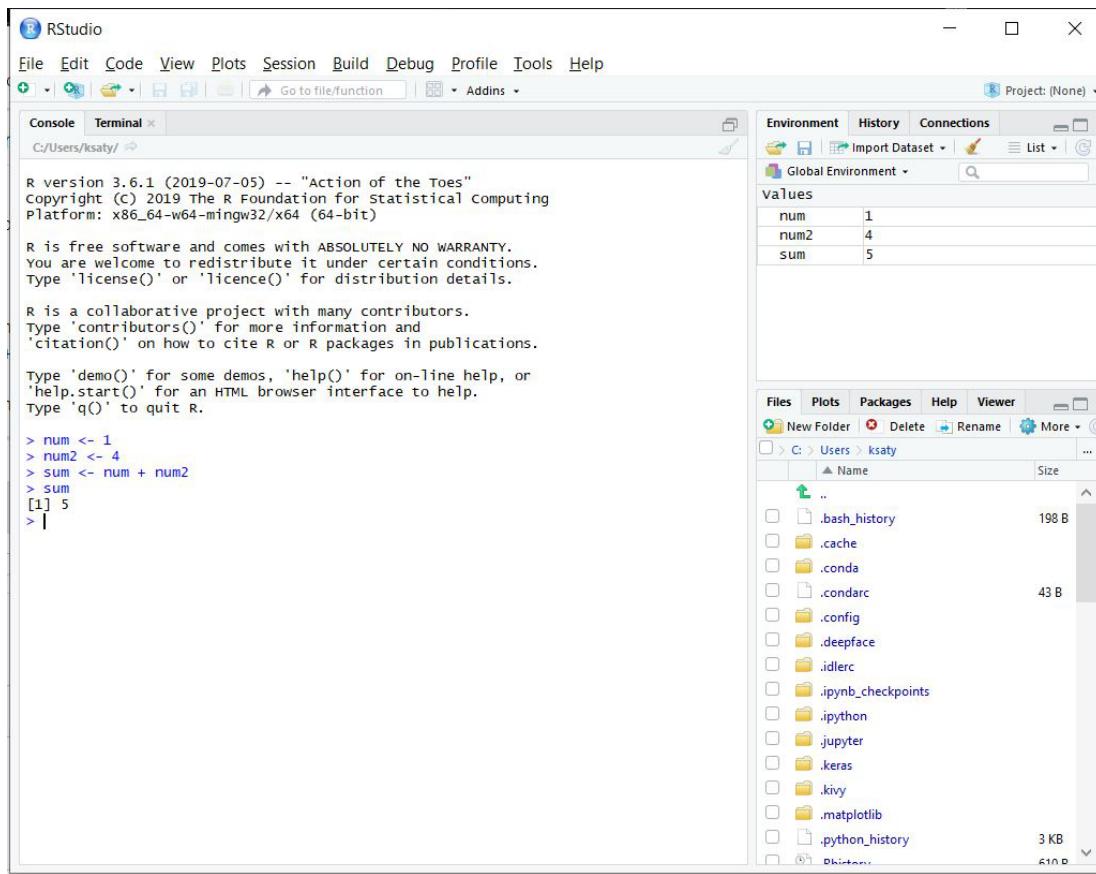


Navigating directories in R studio

- **getwd()**: Returns the current working directory.
- **setwd()**: Set the working directory.
- **dir()**: Return the list of the directory.
- **sessionInfo()**: Return the session of the windows.
- **date()**: Return the current date.

Creating your first R script

Here we are adding two numbers in R studio.



How to Perform Various Operations in RStudio

We'll see some common tasks, their codes in R Studio

Installing R packages

Syntax:

```
install.packages('package_name')
```

Loading R package

Syntax:

```
library(package_name)
```

Help on an R package

```
help(package_name)
```

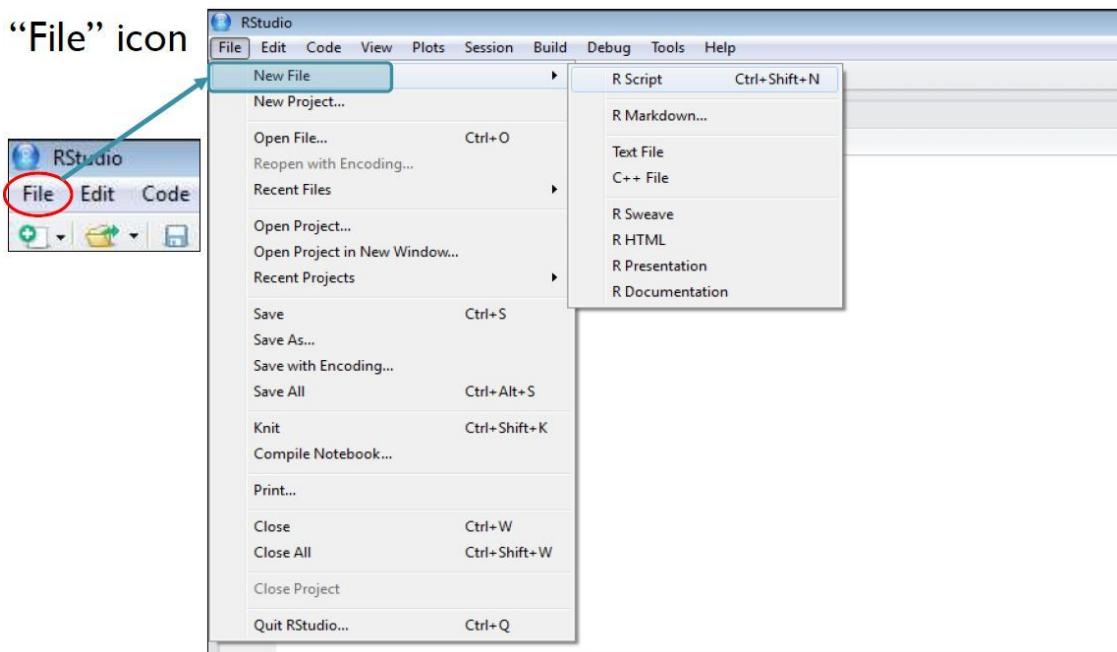
Creation and Execution of R File in R Studio

R Studio is an integrated development environment(IDE) for R. IDE is a GUI, where you can write your quotes, see the results and also see the variables that are generated during the course of programming. R is available as an Open Source software for Client as well as Server Versions.

Creating an R file

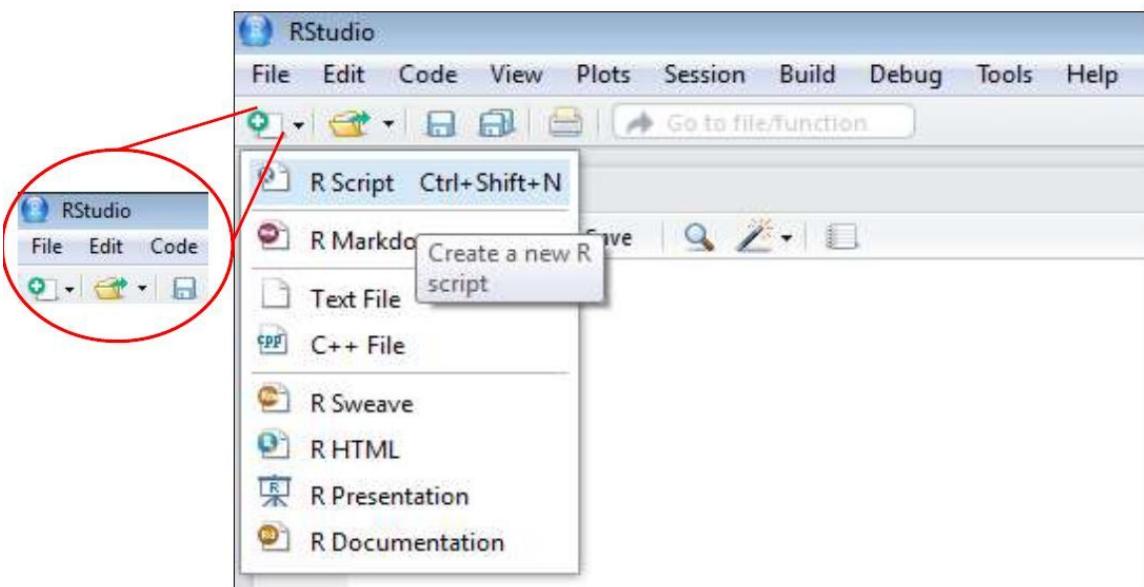
There are two ways to create an R file in R studio:

- You can click on the File tab, from there when you click it will give a drop-down menu, where you can select the new file and then R script, so that, you will get a new file open.

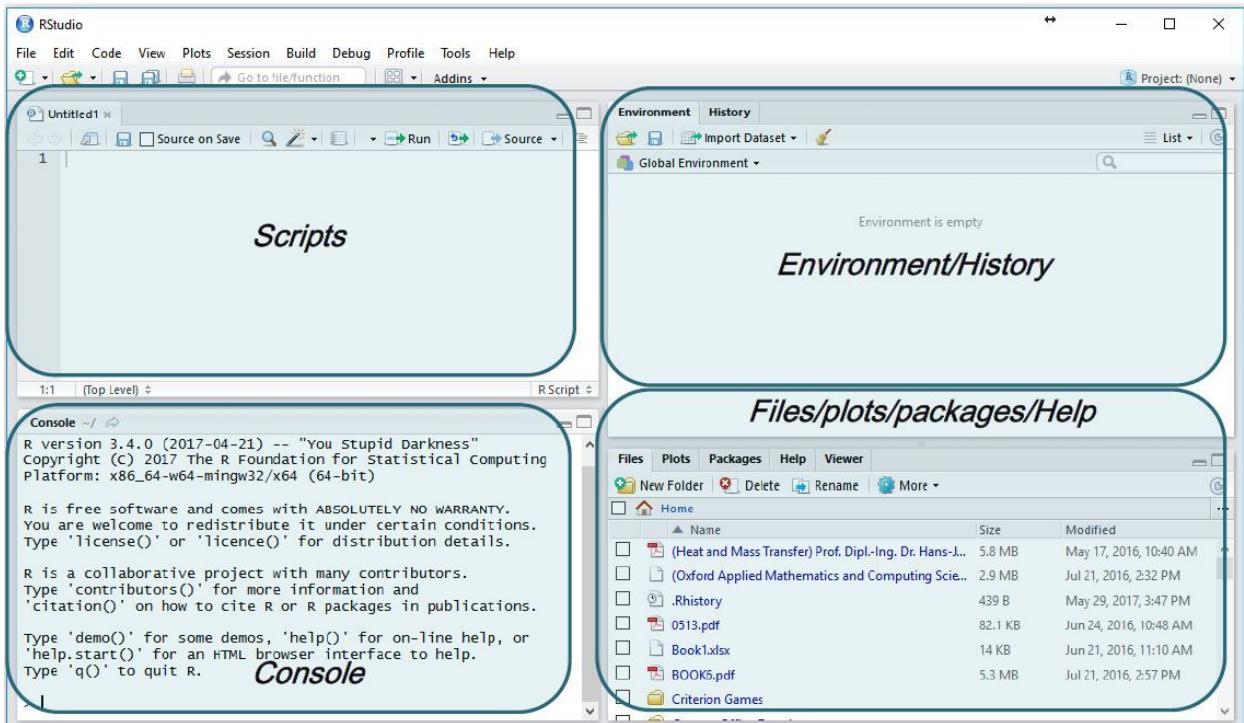


- Use the plus button, which is just below the file tab and you can choose R script, from there, to open a new R script file.

By clicking the icon “  ”below the toolbar



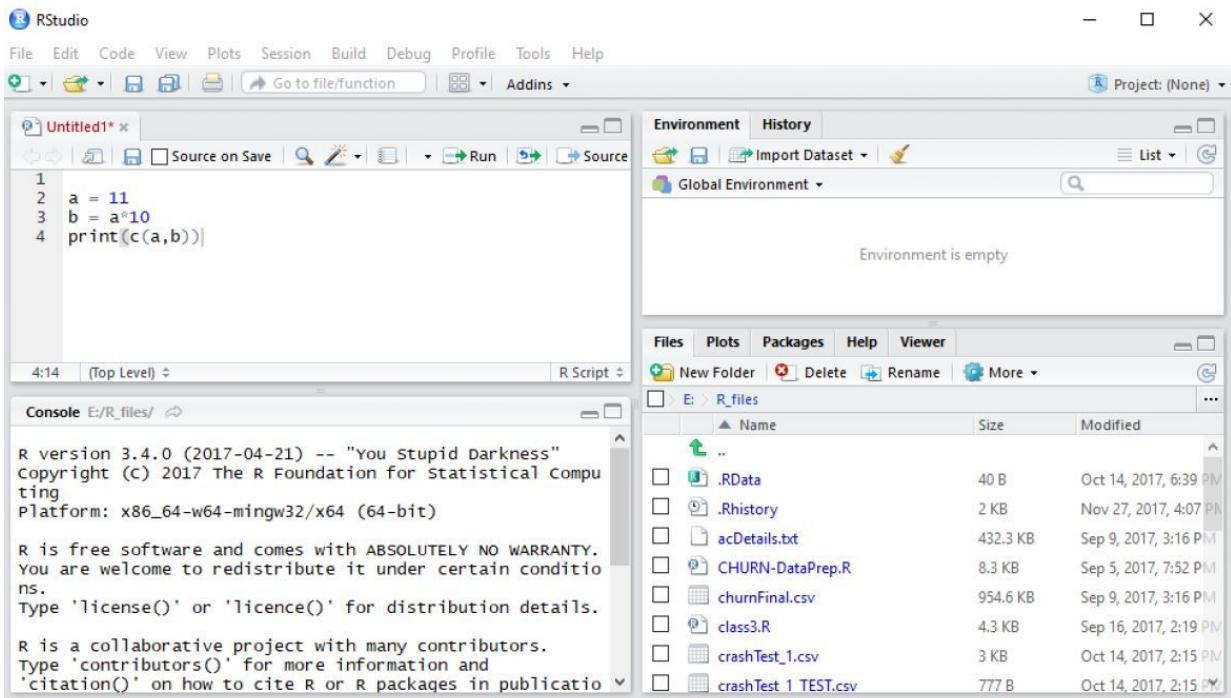
Once you open an R script file, this is how an R Studio with the script file open looks like.



So, 3 panels console, environment/history and file/plots panels are there. On top left you have a new window, which is now being opened as a script file. Now you are ready to write a script file or some program in R Studio.

Writing Scripts in an R File

Writing scripts to an R file is demonstrated here with an example:

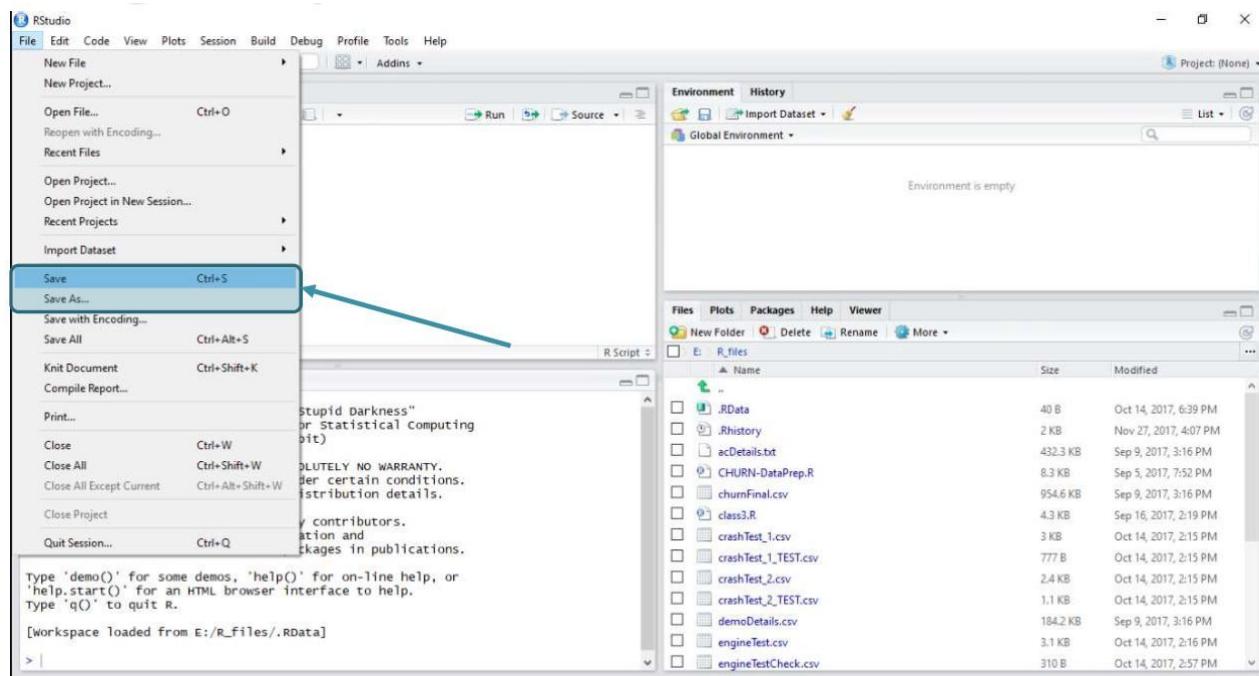


In the above example, a variable 'a' is assigned with a value 11, in the first line of the code and

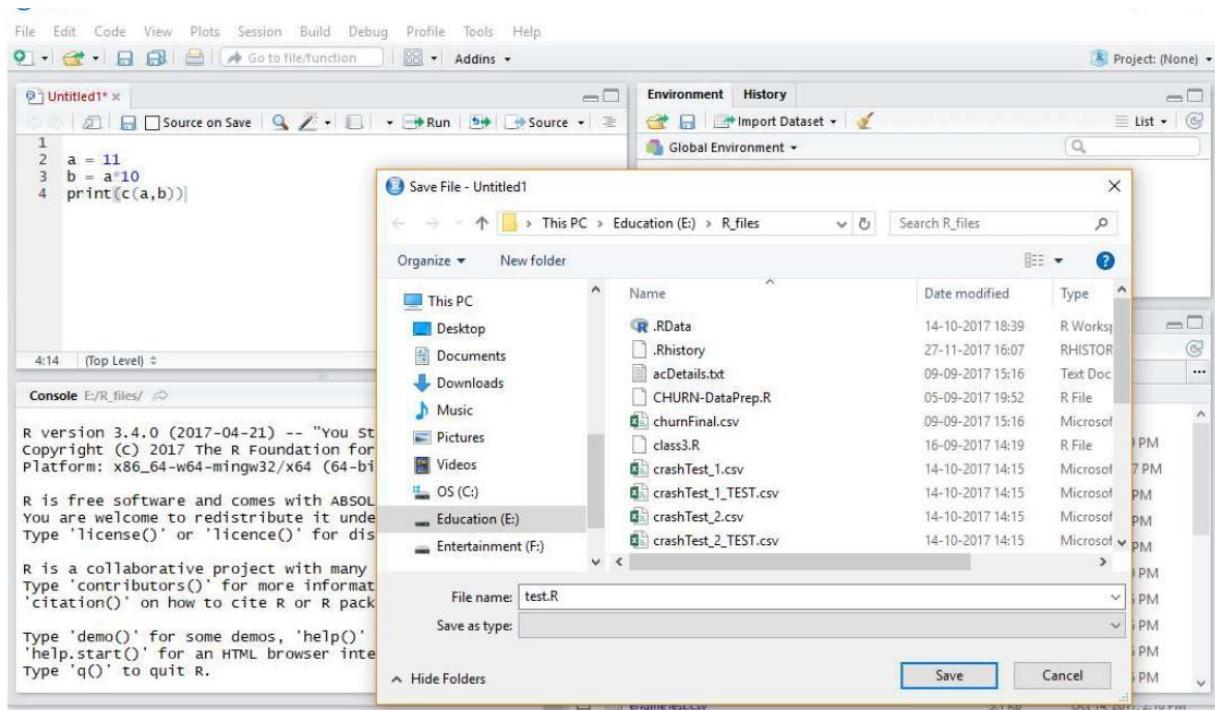
there is b which is ‘a’ times 10, that is the second command. Here, the code is evaluating the value of a times 10 and assign the value to the b and the third statement, which is print(c(a, b)) means concatenates this a and b and print the result. So, this is how a script file is written in R. After writing a script file, there is a need to save this file before execution.

Saving an R File

Let us see, how to save the R file. From the file menu if you click the file tab you can either save or save as button. When you want to save the file if you click the save button, it will automatically save the file has untitled x. So, this x can be 1 or 2 depending upon how many R scripts you have already opened.



Or, it is a nice idea to use the Save as button, just below the Save one, so that, you can rename the script file according to your wish. Let us suppose we have clicked the Save as button. This will pop out a window like this, where you can rename the script file as test.R. Once you rename, then by clicking the save button you can save the script file.

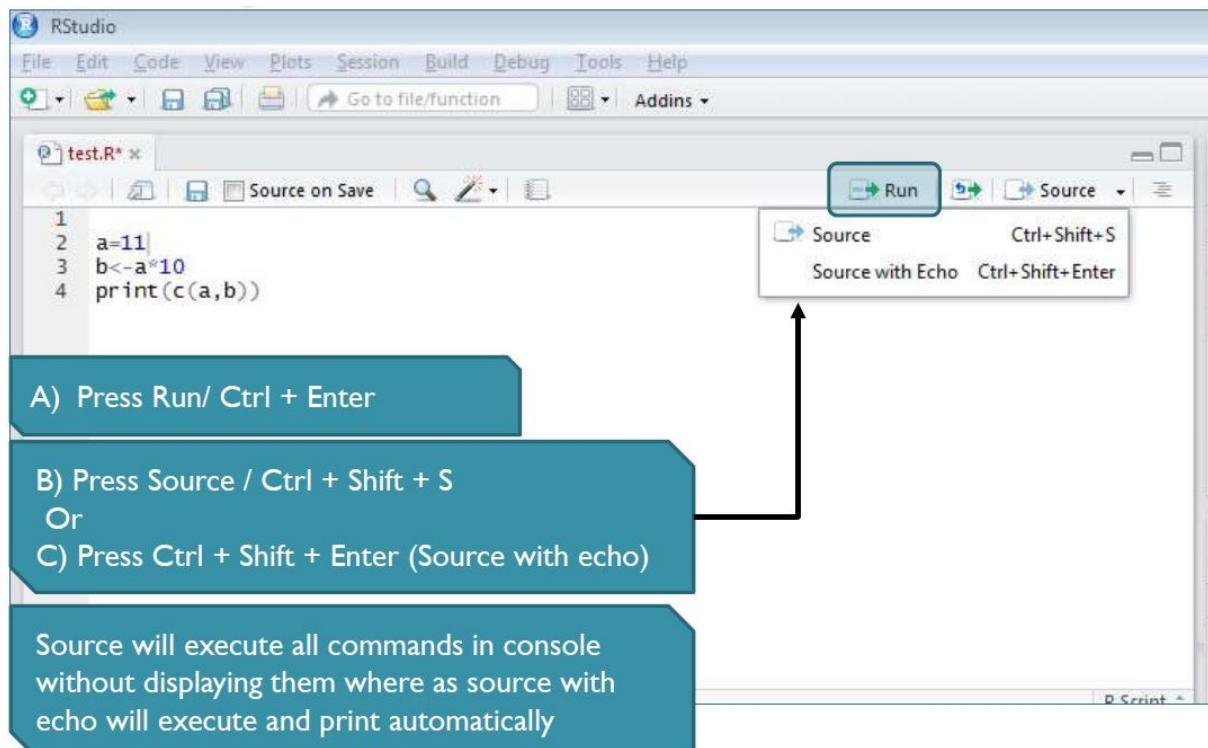


So now, we have seen how to open an R script and how to write some code in the R script file and save the file.

The next task is to execute the R file.

Execution of an R file

There are several ways in which the execution of the commands that are available in the R file is done.



- **Using the run command:** This “run” command can be executed using the GUI, by pressing the run button there, or you can use the Shortcut key control + enter.

What does it do?

It will execute the line in which the cursor is there.

Using the source command:

This “source” command can be executed using the GUI, by pressing the source button there, or you can use the Shortcut key control + shift + S.

What does it do?

It will execute the whole R file and only print the output which you wanted to print.

Using the source with echo command:

This “source with echo” command can be executed using the GUI, by pressing the source with echo button there, or you can use the Shortcut key control + shift + enter.

What does it do?

It will print the commands also, along with the output you are printing.

So, this is an example, where R file is executed, using the source with echo command.

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Project: (None)

test.R *
Source on Save Source
1
2 a = 11
3 b = a*10
4 print(c(a,b))

4:14 (Top Level) R Script

Console E/R_files/
OR
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[workspace loaded from E:/R_files/.RData]
> source('E:/R_files/test.R', echo=TRUE)
> a = 11
> b = a*10
> print(c(a,b))
[1] 11 110
> |
```

RESULT

Environment History
Global Environment
values
a 11
b 110

Files Plots Packages Help Viewer
New Folder Delete Rename More ...
E: R_files
.. .RData 40 B Oct 14, 2017, 6:59 PM
.Rhistory 2 KB Nov 27, 2017, 4:07 PM
acDetails.txt 432.3 KB Sep 9, 2017, 3:16 PM
CHURN-DataPrep.R 8.3 KB Sep 5, 2017, 7:52 PM
churnFinal.csv 954.6 KB Sep 9, 2017, 3:16 PM
class3.R 4.3 KB Sep 16, 2017, 2:19 PM
crashTest_1.csv 3 KB Oct 14, 2017, 2:15 PM
crashTest_1_TEST.csv 777 B Oct 14, 2017, 2:15 PM
crashTest_2.csv 2.4 KB Oct 14, 2017, 2:15 PM

It can be seen in the console, that it printed the command `a = 11` and the command `b = a*10` and also the output `print(c(a, b))` with the values.

So, `a` is 11 and `b` is 11 times 10, this is 110. This is how the output will be printed in the console. Values of `a` and `b` are also shown in the environment panel.

Run command over Source command:

- Run can be used to execute the selected lines of R code.
- Source and Source with echo can be used to run the whole file.
- The advantage of using Run is, you can troubleshoot or debug the program when something is not behaving according to your expectations.
- The disadvantages of using run command are, it populates the console and makes it messy unnecessarily.

Clear the Console and the Environment in R Studio

R Studio is an integrated development environment(IDE) for R. IDE is a GUI, where you can write your quotes, see the results and also see the variables that are generated during the course of programming.

Clearing the Console

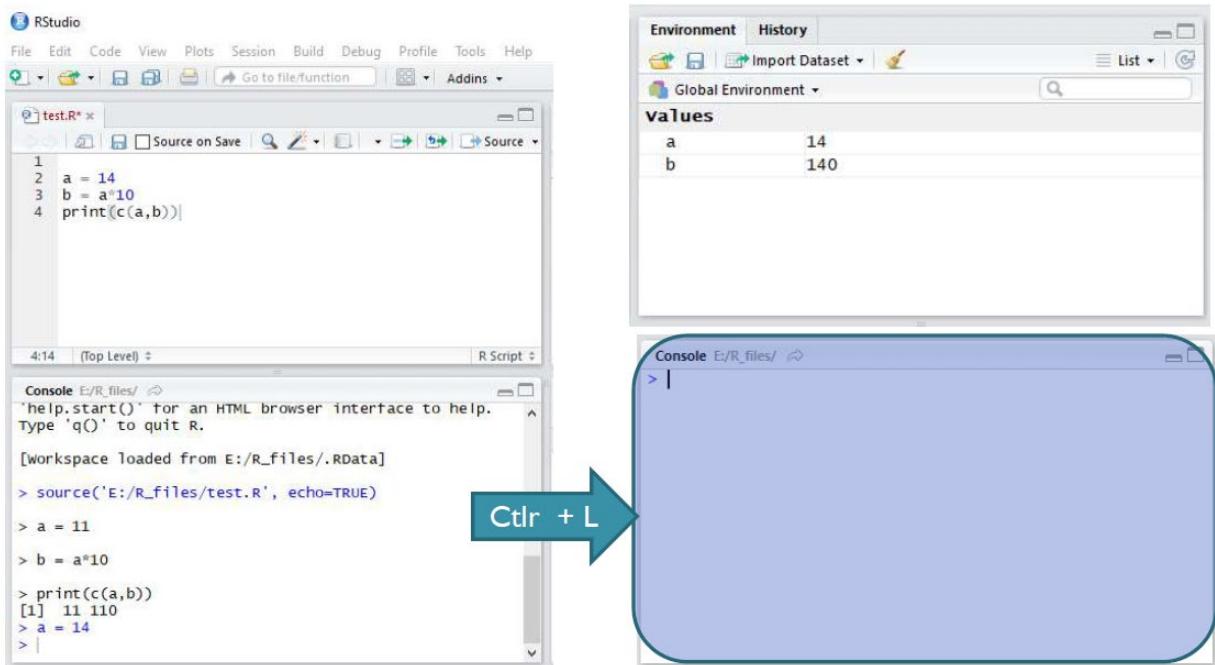
We Clear console in R and RStudio, In some cases when you run the codes using “source” and “source with echo” your console will become messy. And it is needed to clear the console. So let’s now look at how to clear the console. The console can be cleared using the

shortcut key “**ctrl + L**“.

Example:

In this below screenshot, an R code is written in the script tab defined **a** and calculated **b** and printed **a, b**. When this code is executed using “**source with echo**” all the commands will get printed in the console tab. Now, to clear this console click on the console tab and enter the key combination “**ctrl + L**“. Once it is done the console will get cleared.

“control +L”



Note: Remember that clearing the console will not delete the variables that are there in the workspace. You can see that in the environment tab even though we have cleared the console in the workspace we still have the variables that are created earlier.

Clearing the Environment

Variables on the R environment can be cleared in two ways:

Using **rm()** command:

When you want to clear a single variable from the R environment you can use the “**rm()**” command followed by the variable you want to remove.

-> **rm(variable)**

variable: that variable name you want to remove.

If you want to delete all the variables that are there in the environment what you can do is you can use the “**rm**” with an argument “list” is equal to “**ls**” followed by a parenthesis.

-> **rm(list=ls())**

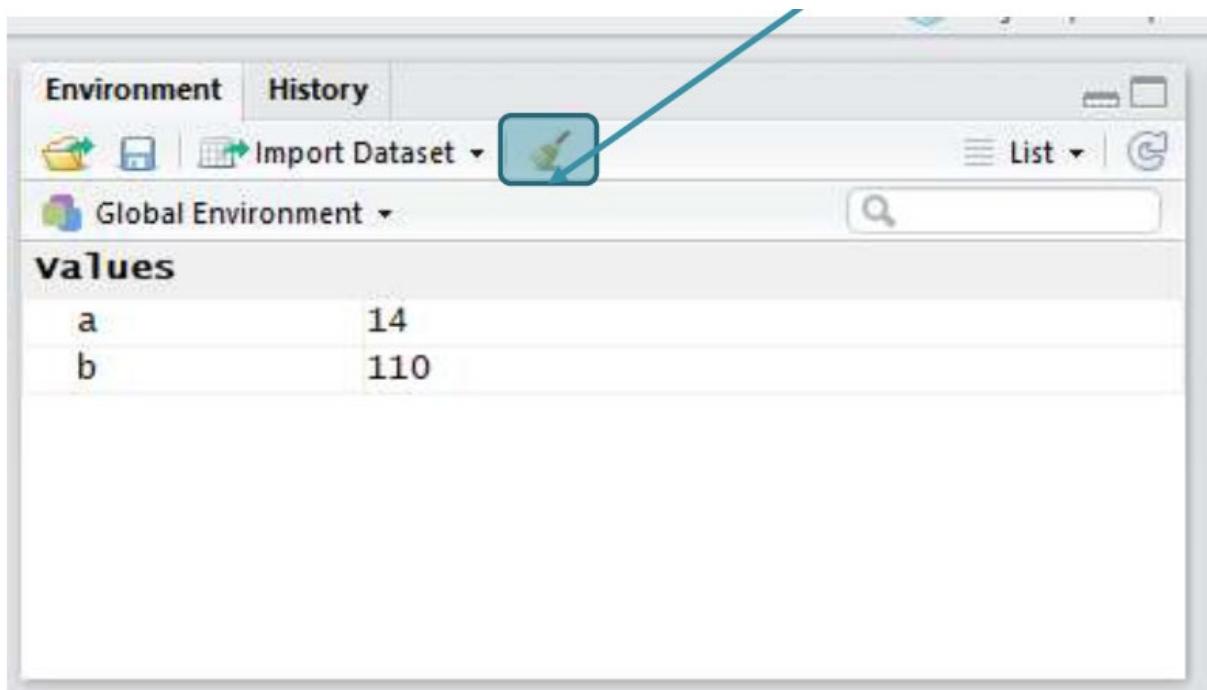
Using the GUI:

We can also clear all the variables in the environment using the GUI in the environment

pane.

How does it works?

You see this brush button in the environment pane.

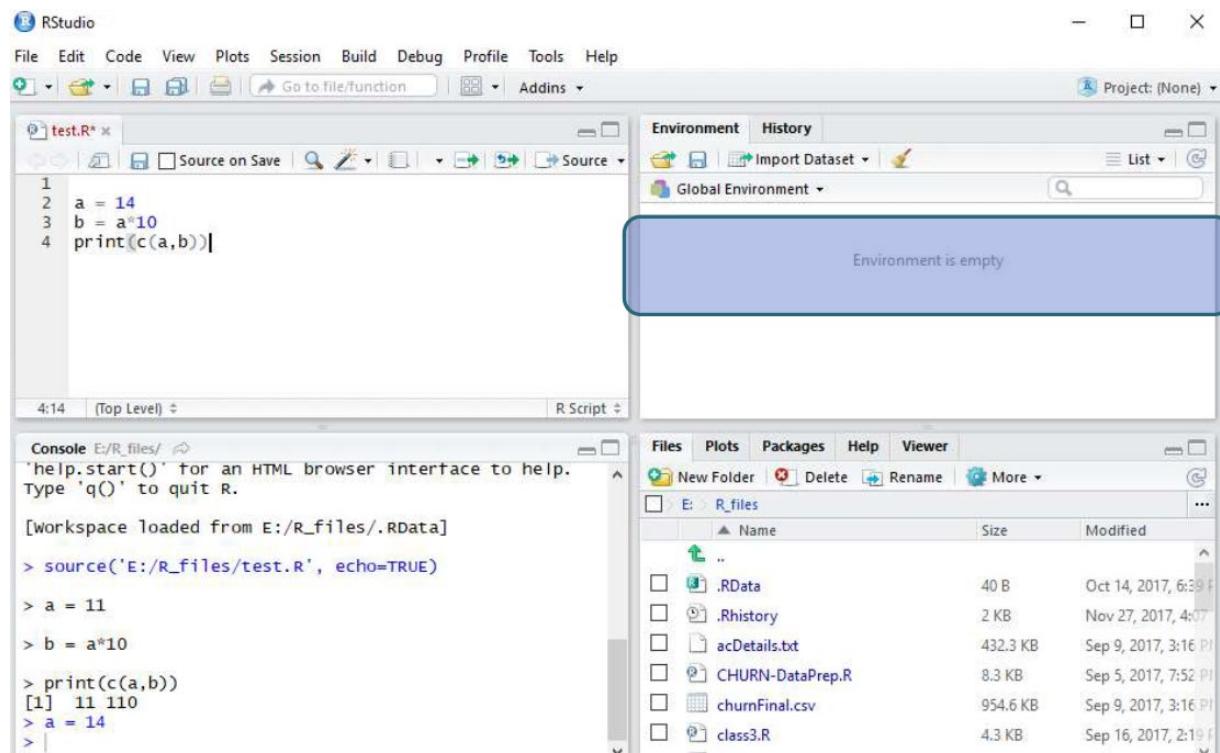


So when you press the brush button it will pop up a window saying “you want to remove all the objects from the environment?”

A screenshot of the RStudio interface with a 'Confirm Remove Objects' dialog box in the foreground. The dialog box contains a warning icon and the text: 'Are you sure you want to remove all objects from the environment? This operation cannot be undone.' There is a checked checkbox labeled 'Include hidden objects'. At the bottom are 'Yes' and 'No' buttons. In the background, the Environment pane shows variables 'a' and 'b' with values '14' and '110' respectively. The Console pane shows R code being run, and the Files pane shows a list of files in the workspace.

And if you say yes it will clear all the variables which are shown here and you can see

the environment is empty now.



Matrix in R – Arithmetic Operations

Arithmetic operations include addition (+), subtraction (-), multiplication(*), division (/) and modulus(%). In this article we are going to see the matrix creation and arithmetic operations on the matrices in R programming language.

- Create first matrix

Syntax:

matrix_name <- matrix(data , nrow = value, ncol = value) .

Parameters:

- *data=includes a list/vector of elements passed as data to an matrix.*
- *nrow= nrow represent the number of rows specified.*
- *ncol= ncol represent the number of columns specified.*

```
# create a vector of elements
vector1=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)

# create a matrix with 4* 4 by passing this vector1
matrix1 <- matrix(vector1, nrow = 4, ncol = 4)

# display matrix
print(matrix1)

# create a vector of elements
vector2=c(1,2,3,2,4,5,6,3,4,1,2,7,8,9,4,5)

# create a matrix with 4* 4 by passing vector2
matrix2 <- matrix(vector2, nrow = 4, ncol = 4)

# display matrix
print(matrix2)

# add matrices
print(matrix1+matrix2)
```

```

[,1] [,2] [,3] [,4]
[1,] 1 5 9 13
[2,] 2 6 10 14
[3,] 3 7 11 15
[4,] 4 8 12 16
[,1] [,2] [,3] [,4]
[1,] 1 4 4 8
[2,] 2 5 1 9
[3,] 3 6 2 4
[4,] 2 3 7 5
[,1] [,2] [,3] [,4]
[1,] 2 9 13 21
[2,] 4 11 11 23
[3,] 6 13 13 19
[4,] 6 11 19 21

```

Subtraction

```
# create a vector of elements
vector1=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
```

```
# create a matrix with 4* 4 by passing this vector1
matrix1 <- matrix(vector1, nrow = 4, ncol = 4)
```

```
# display matrix
print(matrix1)
```

```
# create a vector of elements
vector2=c(1,2,3,2,4,5,6,3,4,1,2,7,8,9,4,5)
```

```
# create a matrix with 4* 4 by passing vector2
matrix2 <- matrix(vector2, nrow = 4, ncol = 4)
```

```
# display matrix
print(matrix2)
```

```
print(" subtraction result")
```

```
# subtract matrices
```

```
print(matrix1-matrix2)
```

```
      [,1] [,2] [,3] [,4]
[1,]    1    5    9   13
[2,]    2    6   10   14
[3,]    3    7   11   15
[4,]    4    8   12   16
      [,1] [,2] [,3] [,4]
[1,]    1    4    4    8
[2,]    2    5    1    9
[3,]    3    6    2    4
[4,]    2    3    7    5
[1] " subtraction result"
      [,1] [,2] [,3] [,4]
[1,]    0    1    5    5
[2,]    0    1    9    5
[3,]    0    1    9   11
[4,]    2    5    5   11
```

Multiplication

```
# create a vector of elements
```

```
vector1=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
```

```
# create a matrix with 4* 4 by passing this vector1
```

```
matrix1 <- matrix(vector1, nrow = 4, ncol = 4)
```

```
# display matrix
```

```
print(matrix1)
```

```
# create a vector of elements
```

```
vector2=c(1,2,3,2,4,5,6,3,4,1,2,7,8,9,4,5)
```

```
# create a matrix with 4* 4 by passing vector2
```

```

matrix2 <- matrix(vector2, nrow = 4, ncol = 4)

# display matrix
print(matrix2)
print(" multiplication result")

# multiply matrices
print(matrix1*matrix2)

```

```

[,1] [,2] [,3] [,4]
[1,]    1    5    9   13
[2,]    2    6   10   14
[3,]    3    7   11   15
[4,]    4    8   12   16
[,1] [,2] [,3] [,4]
[1,]    1    4    4    8
[2,]    2    5    1    9
[3,]    3    6    2    4
[4,]    2    3    7    5
[1] " multiplication result"
[,1] [,2] [,3] [,4]
[1,]    1   20   36  104
[2,]    4   30   10  126
[3,]    9   42   22   60
[4,]    8   24   84   80

```

Division

```

# create a vector of elements
vector1=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)

# create a matrix with 4* 4 by passing this vector1
matrix1 <- matrix(vector1, nrow = 4, ncol = 4)

# display matrix
print(matrix1)

# create a vector of elements
vector2=c(1,2,3,2,4,5,6,3,4,1,2,7,8,9,4,5)

```

```

# create a matrix with 4* 4 by passing vector2
matrix2 <- matrix(vector2, nrow = 4, ncol = 4)

# display matrix
print(matrix2)
print(" Division result")

# divide the matrices
print(matrix1/matrix2)

```

```

[,1] [,2] [,3] [,4]
[1,] 1 5 9 13
[2,] 2 6 10 14
[3,] 3 7 11 15
[4,] 4 8 12 16
[,1] [,2] [,3] [,4]
[1,] 1 4 4 8
[2,] 2 5 1 9
[3,] 3 6 2 4
[4,] 2 3 7 5
[1] " Division result"
[,1] [,2] [,3] [,4]
[1,] 1 1.250000 2.250000 1.625000
[2,] 1 1.200000 10.000000 1.555556
[3,] 1 1.166667 5.500000 3.750000
[4,] 2 2.666667 1.714286 3.200000

```

Modulo operation

Modulo returns the remainder of the elements in a matrix. The operator used: `%%`. The main difference between division and modulo operator is that division returns quotient and modulo returns remainder.

```

# create a vector of elements
vector1=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)

# create a matrix with 4* 4 by passing this vector1
matrix1 <- matrix(vector1, nrow = 4, ncol = 4)

# display matrix
print(matrix1)

# create a vector of elements
vector2=c(1,2,3,2,4,5,6,3,4,1,2,7,8,9,4,5)

# create a matrix with 4* 4 by passing vector2
matrix2 <- matrix(vector2, nrow = 4, ncol = 4)

# display matrix
print(matrix2)
print(" modulo result")

print(matrix1%%matrix2)

```

	[,1]	[,2]	[,3]	[,4]
[1,]	1	5	9	13
[2,]	2	6	10	14
[3,]	3	7	11	15
[4,]	4	8	12	16
	[,1]	[,2]	[,3]	[,4]
[1,]	1	4	4	8
[2,]	2	5	1	9
[3,]	3	6	2	4
[4,]	2	3	7	5
[1]	" modulo result"			
	[,1]	[,2]	[,3]	[,4]
[1,]	0	1	1	5
[2,]	0	1	0	5
[3,]	0	1	1	3
[4,]	0	2	5	1

Algebraic Operations on a Matrix in R

What is Matrix?

A Matrix is a rectangular arrangement of numbers in rows and columns. In a matrix, as we know rows are the ones that run horizontally and columns are the ones that run vertically. In R Programming Language matrices are two-dimensional, homogeneous data structures. These are some examples of matrices.

$$\begin{pmatrix} 1 & 5 & 3 \\ 4 & 9 & 2 \\ 5 & 6 & 7 \end{pmatrix} \quad \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad [1 \ 4 \ 5]$$

What are the Algebraic Operations?

Basic algebraic operations are any one of the traditional operations of arithmetic, which are addition, subtraction, multiplication, division, raising to an integer power, and taking roots. These operations may be performed on numbers, in which case they are often called arithmetic operations. We can perform many more algebraic operations on a matrix in R. Algebraic operations that can be performed on a matrix in R:

- Operations on a single matrix
- Unary operations
- Binary operations
- Linear algebraic operations
 - Rank, determinant, transpose, inverse, trace of a matrix
 - Nullity of a matrix
 - Eigenvalues and eigenvectors of matrices
 - Solve a linear matrix equation

Operations on a single matrix

We can use overloaded arithmetic operators to do element-wise operation on a matrix to create a new matrix. In case of $+=$, $-=$, $*=$ operators, the existing matrix is modified.

```
# R program to demonstrate  
# basic operations on a single matrix
```

```

# Create a 3x3 matrix
a = matrix(
c(1, 2, 3, 4, 5, 6, 7, 8, 9),
nrow = 3,
ncol = 3,
byrow = TRUE
)
cat("The 3x3 matrix:\n")
print(a)

# add 1 to every element
cat("Adding 1 to every element:\n")
print(a + 1)

# subtract 3 from each element
cat("Subtracting 3 from each element:\n")
print(a - 3)

# multiply each element by 10
cat("Multiplying each element by 10:\n")
print(a * 10)

# square each element
cat("Squaring each element:\n")
print(a ^ 2)

# modify existing matrix
cat("Doubled each element of original matrix:\n")
print(a * 2)

```

Output:

```
The 3x3 matrix:  
[, 1] [, 2] [, 3]  
[1, ]    1    2    3  
[2, ]    4    5    6  
[3, ]    7    8    9  
Adding 1 to every element:  
[, 1] [, 2] [, 3]  
[1, ]    2    3    4  
[2, ]    5    6    7  
[3, ]    8    9    10  
Subtracting 3 from each element:  
[, 1] [, 2] [, 3]  
[1, ]   -2   -1    0  
[2, ]    1    2    3  
[3, ]    4    5    6
```

Multiplying each element by 10:

```
[, 1] [, 2] [, 3]  
[1, ]    10   20   30  
[2, ]    40   50   60  
[3, ]    70   80   90
```

Squaring each element:

```
[, 1] [, 2] [, 3]  
[1, ]    1    4    9  
[2, ]   16   25   36  
[3, ]   49   64   81
```

Doubled each element of original matrix:

```
[, 1] [, 2] [, 3]  
[1, ]    2    4    6  
[2, ]    8   10   12  
[3, ]   14   16   18
```

Unary operations

Many unary operations can be performed on a matrix in R. This includes sum, min, max, etc.

```
# R program to demonstrate
# unary operations on a matrix
# Create a 3x3 matrix
a = matrix(
  c(1, 2, 3, 4, 5, 6, 7, 8, 9),
  nrow = 3,
  ncol = 3,
  byrow = TRUE
)
cat("The 3x3 matrix:\n")
print(a)
# maximum element in the matrix
cat("Largest element is:\n")
print(max(a))
# minimum element in the matrix
cat("Smallest element is:\n")
print(min(a))
# sum of element in the matrix
cat("Sum of elements is:\n")
print(sum(a))
```

Output:

```
The 3x3 matrix:
 [, 1] [, 2] [, 3]
 [1, ]    1    2    3
 [2, ]    4    5    6
 [3, ]    7    8    9
 Largest element is:
 [1] 9
 Smallest element is:
 [1] 1
 Sum of elements is:
 [1] 45
```

Binary operations

These operations apply on a matrix elementwise and a new matrix is created. You can use all basic arithmetic operators like `+`, `-`, `*`, `/`, etc. In case of `+=`, `-=`, `=` operators, the existing matrix is modified.

```
# R program to demonstrate
# binary operations on a matrix
# Create a 3x3 matrix
a = matrix(c(1, 2, 3, 4, 5, 6, 7, 8, 9), nrow = 3, ncol = 3, byrow = TRUE )
cat("The 3x3 matrix:\n")
print(a)
# Create another 3x3 matrix
b = matrix( c(1, 2, 5, 4, 6, 2, 9, 4, 3), nrow = 3, ncol = 3, byrow = TRUE )
cat("The another 3x3 matrix:\n")
print(b)

cat("Matrix addition:\n")
print(a + b)

cat("Matrix subtraction:\n")
print(a-b)

cat("Matrix element wise multiplication:\n")
print(a * b)

cat("Regular Matrix multiplication:\n")
print(a %*% b)

cat("Matrix elementwise division:\n")
print(a / b)
```

Output:

```
The 3x3 matrix:  
[, 1] [, 2] [, 3]  
[1, ] 1 2 3  
[2, ] 4 5 6  
[3, ] 7 8 9  
The another 3x3 matrix:  
[, 1] [, 2] [, 3]  
[1, ] 1 2 5  
[2, ] 4 6 2  
[3, ] 9 4 3  
Matrix addition:  
[, 1] [, 2] [, 3]  
[1, ] 2 4 8  
[2, ] 8 11 8  
[3, ] 16 12 12
```

Matrix element wise multiplication:

```
[, 1] [, 2] [, 3]  
[1, ] 1 4 15  
[2, ] 16 30 12  
[3, ] 63 32 27
```

Regular Matrix multiplication:

```
[, 1] [, 2] [, 3]  
[1, ] 36 26 18  
[2, ] 78 62 48  
[3, ] 120 98 78
```

Matrix elementwise division:

```
[, 1] [, 2] [, 3]  
[1, ] 1.0000000 1.0000000 0.6  
[2, ] 1.0000000 0.8333333 3.0  
[3, ] 0.7777778 2.0000000 3.0
```

Linear algebraic operations

One can perform many linear algebraic operations on a given matrix In R. Some of them are as follows:

Rank, determinant, transpose, inverse, trace of a matrix

- **Rank:** The rank of a matrix is the maximum number of linearly independent rows or columns in the matrix. In other words, it is the dimension of the column space (or equivalently, the row space) of the matrix.
- **Determinant:** The determinant has various geometric interpretations, such as measuring the volume scaling factor of a linear transformation represented by the matrix.
- **Transpose:** The transpose of a matrix is obtained by swapping its rows and columns.
- **Inverse:** The inverse of a square matrix is a matrix that, when multiplied with the original matrix, results in the identity matrix.
- **Trace:** The trace of a square matrix is the sum of its diagonal elements.

```
# R program to demonstrate  
# Linear algebraic operations on a matrix  
  
# Importing required library  
# For rank of matrix  
library(pracma)  
# For trace of matrix  
library(psych)  
  
# Create a 3x3 matrix  
A = matrix(  
  c(6, 1, 1, 4, -2, 5, 2, 8, 7),  
  nrow = 3,  
  ncol = 3,  
  byrow = TRUE  
)  
cat("The 3x3 matrix:\n")  
print(A)  
  
# Rank of a matrix
```

```
cat("Rank of A:\n")
print(Rank(A))

# Trace of matrix A
cat("Trace of A:\n")
print(tr(A))

# Determinant of a matrix
cat("Determinant of A:\n")
print(det(A))

# Transpose of a matrix
cat("Transpose of A:\n")
print(t(A))

# Inverse of matrix A
cat("Inverse of A:\n")
print(inv(A))
```

Output:

```
The 3x3 matrix:
 [, 1] [, 2] [, 3]
[1, ]    6    1    1
[2, ]    4   -2    5
[3, ]    2    8    7

Rank of A:
[1] 3

Trace of A:
[1] 11

Determinant of A:
[1] -306

Transpose of A:
 [, 1] [, 2] [, 3]
[1, ]    6    4    2
[2, ]    1   -2    8
[3, ]    1    5    7

Inverse of A:
 [, 1]      [, 2]      [, 3]
[1, ]  0.17647059 -0.003267974 -0.02287582
[2, ]  0.05882353 -0.130718954  0.08496732
[3, ] -0.11764706  0.150326797  0.05228758
```



Nullity of a matrix

The nullity of a matrix is the dimension of the null space, also known as the kernel, of the matrix.

The null space of a matrix.

```
# R program to demonstrate
```

```
# nullity of a matrix
```

```
# Importing required library
```

```
library(pracma)
```

```
# Create a 3x3 matrix
```

```
a = matrix(
```

```
c(1, 2, 3, 4, 5, 6, 7, 8, 9),  
nrow = 3,  
ncol = 3,  
byrow = TRUE  
)  
cat("The 3x3 matrix:\n")  
print(a)
```

```
# No of column  
col = ncol(a)
```

```
# Rank of matrix  
rank = Rank(a)
```

```
# Calculating nullity  
nullity = col - rank
```

```
cat("Nullity of matrix is:\n")  
print(nullity)
```

Output:

```
The 3x3 matrix:  
 [, 1] [, 2] [, 3]  
[1, ]    1    2    3  
[2, ]    4    5    6  
[3, ]    7    8    9  
Nullity of matrix is:  
[1] 1
```

Functions in R Programming

A function accepts input arguments and produces the output by executing valid R commands that are inside the function.

Functions are useful when you want to perform a certain task multiple times.

In R Programming Language when you are creating a function the function name and the file in which you are creating the function need not be the same and you can have one or more functions in R.

Creating a Function in R Programming

Functions are created in R by using the command **function()**. The general structure of the function file is as follows:

```
f = function(arguments){  
    statements  
}
```

Here f = function name

Functions in R Programming

Note: In the above syntax f is the function name, this means that you are creating a function with name f which takes certain arguments and executes the following statements.

Parameters or Arguments in R Functions:

Parameters and arguments are same term in functions.

Parameters or arguments are the values passed into a function.

A function can have any number of arguments, they are separated by comma in parenthesis.

Example:

```
# function to add 2 numbers
```

```
add_num <- function(a,b)
```

```
{
```

```
    sum_result <- a+b
```

```
    return(sum_result)
```

```
}
```

```
# calling add_num function
```

```
sum = add_num(35,34)
```

```
#printing result  
print(sum)
```

Output

```
[1] 69
```

No. of Parameters:

Function should be called with right no. of parameters, neither less nor more or else it will give error.

Default Value of Parameter:

Some functions have default values, and you can also give default value in your user-defined functions. These values are used by functions if user doesn't pass any parameter value while calling a function.

Return Value:

You can use **return()** function if you want your function to return the result.

Calling a Function in R

After creating a Function, you have to call the function to use it.

Calling a function in R is very easy, you can call a function by writing it's name and passing possible parameters value.

Passing Arguments to Functions in R Programming Language

There are several ways you can pass the arguments to the function:

- **Case 1:** Generally in R, the arguments are passed to the function in the same order as in the function definition.
- **Case 2:** If you do not want to follow any order what you can do is you can pass the arguments using the names of the arguments in any order.
- **Case 3:** If the arguments are not passed the default values are used to execute the function.

Now, let us see the examples for each of these cases in the following R code:

```
# A simple R program to demonstrate  
# passing arguments to a function
```

```
Rectangle = function(length=5, width=4){  
  area = length * width  
  return(area)  
}
```

```
# Case 1:  
print(Rectangle(2, 3))
```

```
# Case 2:  
print(Rectangle(width = 8, length = 4))
```

```
# Case 3:  
print(Rectangle())
```

Output

```
[1] 6  
[1] 32  
[1] 20
```

Types of Function in R Language

1. **Built-in Function:** Built-in functions in R are pre-defined functions that are available in R programming languages to perform common tasks or operations.
2. **User-defined Function:** R language allow us to write our own function.

Built-in Function in R Programming Language

Built-in Function are the functions that are already existing in R language and you just need to call them to use.

Here we will use built-in functions like **sum()**, **max()** and **min()**.

```
# Find sum of numbers 4 to 6.
```

```

print(sum(4:6))

# Find max of numbers 4 and 6.

print(max(4:6))

# Find min of numbers 4 and 6.

print(min(4:6))

```

Output

```

[1] 15
[1] 6
[1] 4

```

Other Built-in Functions in R:

Let's look at the list of built-in R functions and their uses:

Functions	Syntax
Mathematical Functions	
<u>abs()</u>	calculates a number's absolute value.
<u>sqrt()</u>	calculates a number's square root.
<u>round()</u>	rounds a number to the nearest integer.
<u>exp()</u>	calculates a number's exponential value
<u>log()</u>	which calculates a number's natural logarithm.
<u>cos()</u> , <u>sin()</u> , and <u>tan()</u>	calculates a number's cosine, sine, and tangent.
Statistical Functions	

Functions	Syntax
<u>mean()</u>	A vector's arithmetic mean is determined by the mean() function.
<u>median()</u>	A vector's median value is determined by the median() function.
<u>cor()</u>	calculates the correlation between two vectors.
<u>var()</u>	calculates the variance of a vector and calculates the standard deviation of a vector.
Data Manipulation Functions	
<u>unique()</u>	returns the unique values in a vector.
<u>subset()</u>	subsets a data frame based on conditions.
<u>aggregate()</u>	groups data according to a grouping variable.
<u>order()</u>	uses ascending or descending order to sort a vector.
File Input/Output Functions	
<u>read.csv()</u>	reads information from a CSV file.
<u>Write.csv()</u>	publishes information to write a CSV file.
<u>Read.table()</u>	reads information from a tabular.
<u>Write.table()</u>	creates a tabular file with data.

User-defined Functions in R Programming Language

User-defined functions are the functions that are **created by the user**.

User defines the working, parameters, default parameter, etc. of that user-defined function. They can be only used in that specific code.

Example

```
# A simple R function to check
```

```
# whether x is even or odd
```

```
evenOdd = function(x){
```

```
  if(x %% 2 == 0)
```

```
    return("even")
```

```
  else
```

```
    return("odd")
```

```
}
```

```
print(evenOdd(4))
```

```
print(evenOdd(3))
```

Output

```
[1] "even"
```

```
[1] "odd"
```

R Function Examples

Now let's look at some use cases of functions in R with some examples.

1. Single Input Single Output

Now create a function in R that will take a single input and gives us a single output.

Following is an example to create a function that calculates the area of a circle which takes in the arguments the radius. So, to create a function, name the function as “areaOfCircle” and the arguments that are needed to be passed are the “radius” of the circle.

```
# A simple R function to calculate  
# area of a circle
```

```
areaOfCircle = function(radius){  
  area = pi*radius^2  
  return(area)  
}
```

```
print(areaOfCircle(2))
```

Output

```
[1] 12.56637
```

2. Multiple Input Multiple Output

Now create a function in R Language that will take multiple inputs and gives us multiple outputs using a list.

The functions in R Language take multiple input objects but returned only one object as output, this is, however, not a limitation because you can create lists of all the outputs which you want to create and once the list is created you can access them into the elements of the list and get the answers which you want.

Let us consider this example to create a function “Rectangle” which takes “length” and “width” of the rectangle and returns area and perimeter of that rectangle. Since R Language can return only one object. Hence, create one object which is a list that contains “area” and “perimeter” and return the list.

```
# A simple R function to calculate  
# area and perimeter of a rectangle
```

```
Rectangle = function(length, width){  
  area = length * width  
  perimeter = 2 * (length + width)
```

```
# create an object called result which is
```

```

# a list of area and perimeter
result = list("Area" = area, "Perimeter" = perimeter)
return(result)
}

resultList = Rectangle(2, 3)
print(resultList["Area"])
print(resultList["Perimeter"])

```

Output

\$Area

[1] 6

\$Perimeter

[1] 10

3. Inline Functions in R Programming Language

Sometimes creating an R script file, loading it, executing it is a lot of work when you want to just create a very small function. So, what we can do in this kind of situation is an inline function.

To create an inline function you have to use the function command with the argument x and then the expression of the function.

```

# A simple R program to
# demonstrate the inline function

f = function(x) x^2*4+x/3

print(f(4))
print(f(-2))
print(0)

```

Output

```
[1] 65.33333  
[1] 15.33333  
[1] 0
```

Lazy Evaluations of Functions in R Programming Language

In R the functions are executed in a lazy fashion. When we say lazy what it means is if some arguments are missing the function is still executed as long as the execution does not involve those arguments.

Example

In the function “Cylinder” given below. There are defined three-argument “diameter”, “length” and “radius” in the function and the volume calculation does not involve this argument “radius” in this calculation. Now, when you pass this argument “diameter” and “length” even though you are not passing this “radius” the function will still execute because this radius is not used in the calculations inside the function.

Let's illustrate this in an R code given below:

```
# A simple R program to demonstrate  
# Lazy evaluations of functions  
  
Cylinder = function(diameter, length, radius ){  
  volume = pi*diameter^2*length/4  
  return(volume)  
}  
  
# This'll execute because this  
# radius is not used in the  
# calculations inside the function.  
print(Cylinder(5, 10))
```

Output

```
[1] 196.3495
```

If you do not pass the argument and then use it in the definition of the function it will throw an error that this “radius” is not passed and it is being used in the function definition.

Example

```
# A simple R program to demonstrate  
# Lazy evaluations of functions  
  
Cylinder = function(diameter, length, radius ){  
  volume = pi*diameter^2*length/4  
  print(radius)  
  return(volume)  
}  
  
# This'll throw an error  
print(Cylinder(5, 10))
```

Output

Error in print(radius) : argument "radius" is missing, with no default

Control Statements in R Programming

Control statements are expressions used to control the execution and flow of the program based on the conditions provided in the statements. These structures are used to make a decision after assessing the variable. In this article, we'll discuss all the control statements with the examples.

In R programming, there are 8 types of control statements as follows:

- if condition
- if-else condition
- for loop
- nested loops
- while loop
- repeat and break statement
- return statement
- next statement

if condition

This control structure checks the expression provided in parenthesis is true or not. If true, the execution of the statements in braces {} continues.

Syntax:

```
if(expression){  
    statements  
    ....  
    ....  
}
```

Example:

```
x <- 100
```

```
if(x > 10){  
    print(paste(x, "is greater than 10"))  
}
```

Output:

```
[1] "100 is greater than 10"
```

if-else condition

It is similar to **if** condition but when the test expression in if condition fails, then statements in **else** condition are executed.

Syntax:

```
if(expression){  
    statements  
    ....  
    ....  
}  
else{  
    statements  
    ....  
    ....  
}
```

Example:

```
x <- 5
```

```
# Check value is less than or greater than 10  
if(x > 10){  
    print(paste(x, "is greater than 10"))  
} else{  
    print(paste(x, "is less than 10"))  
}
```

Output:

```
[1] "5 is less than 10"
```

for loop

It is a type of loop or sequence of statements executed repeatedly until exit condition is reached.

Syntax:

```
for(value in vector){  
    statements
```

....

....

}

Example:

```
x <- letters[4:10]
```

```
for(i in x){  
  print(i)  
}
```

Output:

```
[1] "d"  
[1] "e"  
[1] "f"  
[1] "g"  
[1] "h"  
[1] "i"  
[1] "j"
```

Nested loops

Nested loops are similar to simple loops. Nested means loops inside loop. Moreover, nested loops are used to manipulate the matrix.

Example:

```
# Defining matrix  
m <- matrix(2:15, 2)
```

```
for (r in seq(nrow(m))) {  
  for (c in seq(ncol(m))) {  
    print(m[r, c])  
  }  
}
```

Output:

```
[1] 2  
[1] 4  
[1] 6  
[1] 8  
[1] 10  
[1] 12  
[1] 14  
[1] 3  
[1] 5  
[1] 7  
[1] 9  
[1] 11  
[1] 13  
[1] 15
```

while loop

while loop is another kind of loop iterated until a condition is satisfied. The testing expression is checked first before executing the body of loop.

Syntax:

```
while(expression){  
    statement  
    ....  
    ....  
}
```

Example:

```
x = 1  
# Print 1 to 5  
while(x <= 5){  
    print(x)  
    x = x + 1  
}
```

Output:

```
[1] 1  
[1] 2  
[1] 3  
[1] 4  
[1] 5
```

repeat loop and break statement

repeat is a loop which can be iterated many number of times but there is no exit condition to come out from the loop. So, **break** statement is used to exit from the loop. **break** statement can be used in any type of loop to exit from the loop.

Syntax:

```
repeat {  
    statements  
    ....  
    ....  
    if(expression) {  
        break  
    }  
}
```

Example:

```
x = 1
```

```
# Print 1 to 5  
repeat{  
    print(x)  
    x = x + 1  
    if(x > 5){  
        break  
    }  
}
```

Output:

```
[1] 1  
[1] 2  
[1] 3  
[1] 4  
[1] 5
```

return statement

return statement is used to return the result of an executed function and returns control to the calling function.

Syntax:

```
return(expression)
```

Example:

```
# Checks value is either positive, negative or zero
```

```
func <- function(x){  
  if(x > 0){  
    return("Positive")  
  } else if(x < 0){  
    return("Negative")  
  } else{  
    return("Zero")  
  }  
}
```

```
func(1)  
func(0)  
func(-1)
```

Output:

```
[1] "Positive"  
[1] "Zero"  
[1] "Negative"
```

next statement

next statement is used to skip the current iteration without executing the further statements and continues the next iteration cycle without terminating the loop.

Example:

```
# Defining vector
```

```
x <- 1:10
```

```
# Print even numbers
```

```
for(i in x){  
  if(i%%2 != 0){  
    next #Jumps to next loop  
  }  
  print(i)  
}
```

Output:

```
[1] 2
```

```
[1] 4
```

```
[1] 6
```

```
[1] 8
```

```
[1] 10
```

R – Objects

Every programming language has its own data types to store values or any information so that the user can assign these data types to the variables and perform operations respectively.

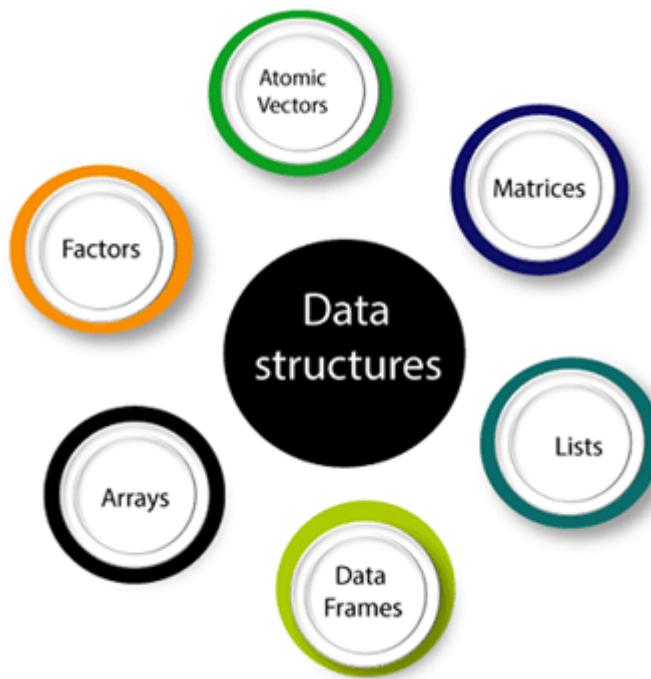
Operations are performed accordingly to the data types.

These data types can be character, integer, float, long, etc. Based on the data type, memory/storage is allocated to the variable. For example, in C language character variables are assigned with 1 byte of memory, integer variable with 2 or 4 bytes of memory and other data types have different memory allocation for them.

Unlike other programming languages, variables are assigned to objects rather than data types in R programming.

Type of Objects

There are 5 basic types of objects in the R language:



Vectors

Atomic vectors are one of the basic types of objects in R programming. Atomic vectors can store homogeneous data types such as character, doubles, integers, raw, logical, and complex. A single element variable is also said to be vector.

Example:

```

# Create vectors
x <- c(1, 2, 3, 4)
y <- c("a", "b", "c", "d")
z <- 5

# Print vector and class of vector
print(x)
print(class(x))

print(y)
print(class(y))

print(z)
print(class(z))

```

Output:

```

[1] 1 2 3 4
[1] "numeric"
[1] "a" "b" "c" "d"
[1] "character"
[1] 5
[1] "numeric"

```

Lists

List is another type of object in R programming. List can contain heterogeneous data types such as vectors or another lists.

Example:

```

# Create list
ls <- list(c(1, 2, 3, 4), list("a", "b", "c"))

# Print
print(ls)

```

```
print(class(ls))
```

Output:

```
[[1]]  
[1] 1 2 3 4
```

```
[[2]]  
[[2]][[1]]  
[1] "a"
```

```
[[2]][[2]]  
[1] "b"
```

```
[[2]][[3]]  
[1] "c"
```

```
[1] "list"
```

Matrices

To store values as 2-Dimensional array, matrices are used in R. Data, number of rows and columns are defined in the matrix() function.

Syntax:

```
matrix(data = NA, nrow = 1, ncol = 1, byrow = FALSE, dimnames = NULL)
```

Example:

```
x <- c(1, 2, 3, 4, 5, 6)  
  
# Create matrix  
mat <- matrix(x, nrow = 2)  
  
print(mat)  
print(class(mat))
```

Output:

```
[, 1] [, 2] [, 3]  
[1, ] 1 3 5  
[2, ] 2 4 6
```

```
[1] "matrix"
```

Factors

Factor object encodes a vector of unique elements (levels) from the given data vector.

Example:

```
# Create vector  
s <- c("spring", "autumn", "winter", "summer",  
      "spring", "autumn")  
  
print(factor(s))  
print(nlevels(factor(s)))
```

Output:

```
[1] spring autumn winter summer spring autumn
```

```
Levels: autumn spring summer winter
```

```
[1] 4
```

Arrays

array() function is used to create n-dimensional array. This function takes dim attribute as an argument and creates required length of each dimension as specified in the attribute.

Syntax:

```
array(data, dim = length(data), dimnames = NULL)
```

Example:

```
# Create 3-dimensional array  
# and filling values by column  
arr <- array(c(1, 2, 3), dim = c(3, 3, 3))  
  
print(arr)
```

Output:

```
,, 1
```

```
[, 1] [, 2] [, 3]  
[1, ] 1 1 1  
[2, ] 2 2 2  
[3, ] 3 3 3,, 2
```

```
[, 1] [, 2] [, 3]  
[1, ] 1 1 1  
[2, ] 2 2 2  
[3, ] 3 3 3,, 3
```

```
[, 1] [, 2] [, 3]  
[1, ] 1 1 1  
[2, ] 2 2 2  
[3, ] 3 3 3
```

Data Frames

Data frames are 2-dimensional tabular data object in R programming. Data frames consists of multiple columns and each column represents a vector. Columns in data frame can have different modes of data unlike matrices.

Example:

```
# Create vectors
x <- 1:5
y <- LETTERS[1:5]
z <- c("Albert", "Bob", "Charlie", "Denver", "Elie")

# Create data frame of vectors
df <- data.frame(x, y, z)

# Print data frame
print(df)
```

Output:

	x	y	z
1	1	A	Albert
2	2	B	Bob
3	3	C	Charlie
4	4	D	Denver
5	5	E	Elie

Data Manipulation in R with Dplyr Package

In order to manipulate the data, R provides a library called dplyr which consists of many built-in methods to manipulate the data. So to use the data manipulation function, first need to import the dplyr package using *library(dplyr)* line of code. Below is the list of a few data manipulation functions present in dplyr package.

Function Name	Description
filter()	Produces a subset of a Data Frame.
Distinct()	Removes duplicate rows in a Data Frame
arrange()	Reorder the rows of a Data Frame
select()	Produces data in required columns of a Data Frame
rename()	Renames the variable names
mutate()	Creates new variables without dropping old ones.
Transmute()	Creates new variables by dropping the old.
Summarize()	Gives summarized data like Average, Sum, etc.

filter() method

The filter() function is used to produce the subset of the data that satisfies the condition specified in the filter() method. In the condition, we can use conditional operators, logical operators, NA values, range operators etc. to filter out data. Syntax of filter() function is given below-

filter(dataframeName, condition)

Example:

In the below code we used filter() function to fetch the data of players who scored more than 100 runs from the “stats” data frame.

```
# import dplyr package
library(dplyr)

# create a data frame
stats <- data.frame(player=c('A', 'B', 'C', 'D'),
                     runs=c(100, 200, 408, 19),
                     wickets=c(17, 20, NA, 5))

# fetch players who scored more
# than 100 runs
filter(stats, runs>100)
```

Output

	player	runs	wickets
1	B	200	20
2	C	408	NA

distinct() method

The distinct() method removes duplicate rows from data frame or based on the specified columns. The syntax of distinct() method is given below-

distinct(dataframeName, col1, col2, ..., .keep_all=TRUE)

Example:

Here in this example, we used distinct() method to remove the duplicate rows from the data frame and also remove duplicates based on a specified column.

```
# import dplyr package
library(dplyr)
```

```

# create a data frame
stats <- data.frame(player=c('A', 'B', 'C', 'D', 'A', 'A'),
                     runs=c(100, 200, 408, 19, 56, 100),
                     wickets=c(17, 20, NA, 5, 2, 17))

# removes duplicate rows
distinct(stats)

#remove duplicates based on a column
distinct(stats, player, .keep_all = TRUE)

```

Output

player runs wickets

1	A	100	17
2	B	200	20
3	C	408	NA
4	D	19	5
5	A	56	2

player runs wickets

1	A	100	17
2	B	200	20
3	C	408	NA
4	D	19	5

arrange() method

In R, the `arrange()` method is used to order the rows based on a specified column. The syntax of `arrange()` method is specified below-

arrange(dataframeName, columnName)

Example:

In the below code we ordered the data based on the runs from low to high using `arrange()` function.

```

# import dplyr package
library(dplyr)
# create a data frame
stats <- data.frame(player=c('A', 'B', 'C', 'D'),
                     runs=c(100, 200, 408, 19),
                     wickets=c(17, 20, NA, 5))
# ordered data based on runs
arrange(stats, runs)

```

Output

	player	runs	wickets
1	D	19	5
2	A	100	17
3	B	200	20
4	C	408	NA

select() method

The select() method is used to extract the required columns as a table by specifying the required column names in select() method. The syntax of select() method is mentioned below-

select(dataframeName, col1,col2,...)

Example:

Here in the below code we fetched the player, wickets column data only using select() method.

```

# import dplyr package
library(dplyr)

# create a data frame
stats <- data.frame(player=c('A', 'B', 'C', 'D'),
                     runs=c(100, 200, 408, 19),
                     wickets=c(17, 20, NA, 5))

# fetch required column data
select(stats, player,wickets)

```

Output

```
player wickets  
1 A 17  
2 B 20  
3 C NA  
4 D 5
```

rename() method

The rename() function is used to change the column names. This can be done by the below syntax-

```
rename(dataframeName, newName=oldName)
```

Example:

In this example, we change the column name “runs” to “runs_scored” in stats data frame.

```
# import dplyr package  
library(dplyr)  
# create a data frame  
stats <- data.frame(player=c('A', 'B', 'C', 'D'),  
                      runs=c(100, 200, 408, 19),  
                      wickets=c(17, 20, NA, 5))  
# renaming the column  
rename(stats, runs_scored=runs)
```

Output

```
player runs_scored wickets  
1 A 100 17  
2 B 200 20  
3 C 408 NA  
4 D 19 5
```

mutate() & transmute() methods

These methods are used to create new variables. The mutate() function creates new variables without dropping the old ones but transmute() function drops the old variables and creates new variables. The syntax of both methods is mentioned below-

mutate(dataframeName, newVariable=formula)
transmute(dataframeName, newVariable=formula)

Example:

In this example, we created a new column avg using mutate() and transmute() methods.

```
# import dplyr package
library(dplyr)
# create a data frame
stats <- data.frame(player=c('A', 'B', 'C', 'D'),
                     runs=c(100, 200, 408, 19),
                     wickets=c(17, 20, 7, 5))
# add new column avg
mutate(stats, avg=runs/4)
# drop all and create a new column
transmute(stats, avg=runs/4)
```

Output

	player	runs	wickets	avg
1	A	100	17	25.00
2	B	200	20	50.00
3	C	408	7	102.00
4	D	19	5	4.75

	avg
1	25.00
2	50.00
3	102.00
4	4.75

Here mutate() functions adds a new column for the existing data frame without dropping the old ones where as transmute() function created a new variable but dropped all the old columns.
summarize() method

Using the summarize method we can summarize the data in the data frame by using aggregate functions like sum(), mean(), etc. The syntax of summarize() method is specified below-

summarize(dataframeName, aggregate_function(columnName))

Example:

In the below code we presented the summarized data present in the runs column using summarize() method.

```
# import dplyr package
library(dplyr)

# create a data frame
stats <- data.frame(player=c('A', 'B', 'C', 'D'),
                     runs=c(100, 200, 408, 19),
                     wickets=c(17, 20, 7, 5))

# summarize method
summarize(stats, sum(runs), mean(runs))
```

Output

```
sum(runs) mean(runs)
1      727    181.75
```

R Data Types

Different forms of data that can be saved and manipulated are defined and categorized using data types in computer languages, including R. Each R data type has unique properties and associated operations.

What are R Data types?

R Data types are used to specify the kind of data that can be stored in a variable.

For effective memory consumption and precise computation, the right data type must be selected.

Each R data type has its own set of regulations and restrictions.

Variables are not needed to be declare with a data type in R, data type even can be changed.

Example of R data Type:

```
#numeric data type  
var <- 30  
#integer data type  
var <- 80L
```

Data Types in R Programming Language

Each variable in R has an associated data type. Each R-Data Type requires different amounts of memory and has some specific operations which can be performed over it.

Data Types in R are:

1. numeric – (3,6.7,121)
2. Integer – (2L, 42L; where ‘L’ declares this as an integer)
3. logical – (“True”)
4. complex – (7 + 5i; where ‘i’ is imaginary number)
5. character – (“a”, “B”, “c is third”, “69”)
6. raw – (as.raw(55); raw creates a raw vector of the specified length)

R Programming language has the following basic R-data types and the following table shows the data type and the values that each data type can take.

Basic Data Types	Values	Examples
Numeric	Set of all real numbers	"numeric_value <- 3.14"
Integer	Set of all integers, \mathbb{Z}	"integer_value <- 42L"
Logical	TRUE and FALSE	"logical_value <- TRUE"
Complex	Set of complex numbers	"complex_value <- 1 + 2i"
Character	“a”, “b”, “c”, ..., “@”, “#”, “\$”, ..., “1”, “2”, ...etc	"character_value <- "Hello Geeks"
raw	as.raw()	"single_raw <- as.raw(255)"

1. Numeric Data type in R

Decimal values are called numeric in R. It is the default R data type for numbers in R.

If you assign a decimal value to a variable x as follows, x will be of numeric type.

Real numbers with a decimal point are represented using this data type in R. It uses a format for double-precision floating-point numbers to represent numerical values.

```
# A simple R program
# to illustrate Numeric data type
# Assign a decimal value to x
x = 5.6
# print the class name of variable
print(class(x))
# print the type of variable
print(typeof(x))
```

Output

```
[1] "numeric"  
[1] "double"
```

Even if an integer is assigned to a variable y, it is still saved as a numeric value.

```
# A simple R program  
# to illustrate Numeric data type  
# Assign an integer value to y  
y = 5  
# print the class name of variable  
print(class(y))  
# print the type of variable  
print(typeof(y))
```

Output

```
[1] "numeric"  
[1] "double"
```

When R stores a number in a variable, it converts the number into a “double” value or a decimal type with at least two decimal places.

This means that a value such as “5” here, is stored as 5.00 with a type of double and a class of numeric. And also y is not an integer here can be confirmed with the **is.integer()** function.

```
# A simple R program  
# to illustrate Numeric data type  
# Assign a integer value to y  
y = 5  
# is y an integer?  
print(is.integer(y))
```

Output

```
[1] FALSE
```

2. Integer Data type in R

R supports integer data types which are the set of all integers.

You can create as well as convert a value into an integer type using the **as.integer()** function.

You can also use the capital ‘L’ notation as a suffix to denote that a particular value is of the integer R data type.

```
# A simple R program  
# to illustrate integer data type  
# Create an integer value  
x = as.integer(5)  
# print the class name of x  
print(class(x))  
  
# print the type of x  
print(typeof(x))  
# Declare an integer by appending an L suffix.  
y = 5L  
# print the class name of y  
print(class(y))  
  
# print the type of y  
print(typeof(y))
```

Output

```
[1] "integer"  
[1] "integer"  
[1] "integer"  
[1] "integer"
```

3. Logical Data type in R

R has logical data types that take either a value of **true** or **false**.

A logical value is often created via a comparison between variables.

Boolean values, which have two possible values, are represented by this R data type: FALSE or

TRUE

```
# A simple R program
# to illustrate logical data type
# Sample values
x = 4
y = 3
# Comparing two values
z = x > y
# print the logical value
print(z)

# print the class name of z
print(class(z))

# print the type of z
print(typeof(z))
```

Output

```
[1] TRUE
[1] "logical"
[1] "logical"
```

4. Complex Data type in R

R supports complex data types that are set of all the complex numbers. The complex data type is to store numbers with an imaginary component.

```
# A simple R program
# to illustrate complex data type
# Assign a complex value to x
x = 4 + 3i
# print the class name of x
print(class(x))
```

```
# print the type of x  
print(typeof(x))
```

Output

```
[1] "complex"  
[1] "complex"
```

5. Character Data type in R

R supports character data types where you have all the alphabets and special characters. It stores character values or strings. Strings in R can contain alphabets, numbers, and symbols. The easiest way to denote that a value is of character type in R data type is to wrap the value inside single or double inverted commas.

```
# A simple R program  
# to illustrate character data type  
# Assign a character value to char  
char = "Geeksforgeeks"  
# print the class name of char  
print(class(char))  
# print the type of char  
print(typeof(char))
```

Output

```
[1] "character"  
[1] "character"
```

There are several tasks that can be done using R data types. Let's understand each task with its action and the syntax for doing the task along with an R code to illustrate the task.

6. Raw data type in R

To save and work with data at the byte level in R, use the raw data type. By displaying a series of unprocessed bytes, it enables low-level operations on binary data. Here are some speculative data on R's raw data types:

```
# Create a raw vector  
x <- as.raw(c(0x1, 0x2, 0x3, 0x4, 0x5))  
print(x)
```

Output

```
[1] 01 02 03 04 05
```

Five elements make up this raw vector x, each of which represents a raw byte value.

Find Data Type of an Object in R

To find the data type of an object you have to use **class()** function. The syntax for doing that is you need to pass the object as an argument to the function **class()** to find the data type of an object.

Syntax

```
class(object)
```

Example

```
# A simple R program  
# to find data type of an object  
# Logical  
print(class(TRUE))  
# Integer  
print(class(3L))  
# Numeric  
print(class(10.5))  
# Complex  
print(class(1+2i))  
# Character  
print(class("12-04-2020"))
```

Output

```
[1] "logical"  
[1] "integer"  
[1] "numeric"  
[1] "complex"  
[1] "character"
```

Type verification

You can verify the data type of an object, if you doubt about it's data type.

To do that, you need to use the prefix “is.” before the data type as a command.

Syntax:

```
is.data_type(object)
```

Example

```
# A simple R program  
  
# Verify if an object is of a certain datatype  
  
# Logical  
print(is.logical(TRUE))  
  
# Integer  
print(is.integer(3L))  
  
# Numeric  
print(is.numeric(10.5))  
  
# Complex  
print(is.complex(1+2i))  
  
# Character  
print(is.character("12-04-2020"))  
print(is.integer("a"))  
print(is.numeric(2+3i))
```

Output

```
[1] TRUE  
[1] TRUE
```

```
[1] TRUE  
[1] TRUE  
[1] TRUE  
[1] FALSE  
[1] FALSE
```

Coerce or Convert the Data Type of an Object to Another

The process of altering the data type of an object to another type is referred to as coercion or data type conversion. This is a common operation in many programming languages that is used to alter data and perform various computations.

When coercion is required, the language normally performs it automatically, whereas conversion is performed directly by the programmer.

Coercion can manifest itself in a variety of ways, depending on the R programming language and the context in which it is employed.

In some circumstances, the coercion is implicit, which means that the language will change one type to another without the programmer having to expressly request it.

Syntax

```
as.data_type(object)
```

Note: All the coercions are not possible and if attempted will be returning an “NA” value.

For Detailed Explanation – Data Type Conversion in R

Example

```
# A simple R program  
# convert data type of an object to another  
# Logical  
print(as.numeric(TRUE))  
# Integer  
print(as.complex(3L))  
# Numeric  
print(as.logical(10.5))  
# Complex
```

```
print(as.character(1+2i))  
# Can't possible  
print(as.numeric("12-04-2020"))
```

Output

```
[1] 1  
[1] 3+0i  
[1] TRUE  
[1] "1+2i"  
[1] NA  
Warning message:  
In print(as.numeric("12-04-2020")) : NAs introduced by coercion
```

Reading and Writing Data to and from R

Functions for Reading Data into R:

There are a few very useful functions for reading data into R.

1. **read.table()** and **read.csv()** are two popular functions used for reading tabular data into R.
2. **readLines()** is used for reading lines from a text file.
3. **source()** is a very useful function for reading in R code files from another R program.
4. **dget()** function is also used for reading in R code files.
5. **load()** function is used for reading in saved workspaces
6. **unserialize()** function is used for reading single R objects in binary format.

Functions for Writing Data to Files:

There are similar functions for writing data to files

1. **write.table()** is used for writing tabular data to text files (i.e. CSV).
2. **writeLines()** function is useful for writing character data line-by-line to a file or connection.
3. **dump()** is a function for dumping a textual representation of multiple R objects.
4. **dput()** function is used for outputting a textual representation of an R object.
5. **save()** is useful for saving an arbitrary number of R objects in binary format to a file.
6. **serialize()** is used for converting an R object into a binary format for outputting to a connection (or file).

Reading Data Files with `read.table()`:

The `read.table()` function is one of the most commonly used functions for reading data in R. TO get the help file for `read.table()` just type **?read.table** in R console.

The `read.table()` function has a few important arguments:

- file, the name of a file, or a connection
- header, logical indicating if the file has a header line
- sep, a string indicating how the columns are separated
- colClasses, a character vector indicating the class of each column in the dataset
- nrow, the number of rows in the dataset. By default `read.table()` reads an entire file.
- comment.char, a character string indicating the comment character. This defalts to “#”. If there are no commented lines in your file, it’s worth setting this to be the empty string “”.

- skip, the number of lines to skip from the beginning
- stringsAsFactors, should character variables be coded as factors? This defaults to TRUE because back in the old days, if you had data that were stored as strings, it was because those strings represented levels of a categorical variable. Now we have lots of data that is text data and they don't always represent categorical variables. So you may want to set this to be FALSE in those cases. If you always want this to be FALSE, you can set a global option via options(stringsAsFactors = FALSE). I've never seen so much heat generated on discussion forums about an R function argument than the stringsAsFactors argument.

Check the following example how to work with read.table() in r. For this example a data set called **wine data set** will be used. You can download the data set by clicking [here](#). The data set was originally taken from UCI Repository. You can get more details about the data set from [here](#).

Download the Wine Data set

```
w<-read.table("https://makemeanalyst.com/wp-
content/uploads/2017/05/wine.txt",sep=",",header = TRUE)
head(w)
View(w)
```

Writing Data Files with write.table():

To write a R object into a file check the following code.

```
write.table(w,"E:/MakeMeAnalyst/wine.txt") #Give your own path here.
```

To learn more about data output using write.table().

readLines() and writeLines() function in R:

readLines() function is mainly used for reading lines from a text file and writeLines() function is useful for writing character data line-by-line to a file or connection. Check the following example to deal with readLines() and writeLines(). First, download the sample text from [here](#) and then read it into R.

Download the Sample Text

```
con <- file("https://makemeanalyst.com/wp-content/uploads/2017/05/Sample.txt", "r")
```

```
w<-readLines(con)
```

```
close(con)
```

```
w[1]
```

```
w[2]
```

```
w[3]
```

Output:

```
> w[1]
```

```
[1] "This is a sample text file."
```

```
> w[2]
```

```
[1] "Read this file using readLines() function."
```

```
> w[3]
```

```
[1] "And you can write a file using writeLines() function."
```

You can also write contents into a file using writeLines() function in R. Following example shows how to do that.

```
sample<-c("Class,Alcohol,Malic acid,Ash","1,14.23,1.71,2.43","1,13.2,1.78,2.14")  
writeLines(sample,"F://sample.csv")
```

You can write them into tsv file also using below code.

```
sample<-c("Class,Alcohol,Malic acid,Ash","1,14.23,1.71,2.43","1,13.2,1.78,2.14")  
t<- gsub(",","\\t", sample)  
writeLines(t, "F://Sample.tsv")
```

dput() and dget() Function in R:

You can create a more descriptive representation of an R object by using the **dput()** or **dump()** functions. Unlike writing out a table or CSV file, **dump()** and **dput()** preserve the metadata, so that another user doesn't have to specify it all over again. For example, we can preserve the class of each column of a table or the levels of a factor variable.

```
# Create a data frame
```

```
x <- data.frame(Name = "Mr. A", Gender = "Male", Age=35)
```

```
#Print 'dput' output to your R console
```

```
dput(x)
```

```
#Write the 'dput' output to a file
```

```
dput(x, file = "F://w.R")
```

```
# Now read in 'dput' output from the file  
y <- dget("F:/w.R")  
y
```

dump() Function in R:

You can dump() R objects to a file by passing its names.

```
x<-1:10  
d <- data.frame(Name = "Mr. A", Gender = "Male", Age=35)  
dump(c("x", "d"), file = "F://dump_data.R")  
rm(x, d) #After dumping just remove the variables from environment.
```

source() Function in R:

The inverse of dump() is **source()** function. Now you can import that dump_data.R into R using following code.

```
source("F://dump_data.R")  
x  
d  
str(d)
```

Output:

```
> x  
[1] 1 2 3 4 5 6 7 8 9 10  
> d  
Name Gender Age  
1 Mr. A Male 35  
> str(d)  
'data.frame': 1 obs. of 3 variables:  
 $ Name : Factor w/ 1 level "Mr. A": 1  
 $ Gender: Factor w/ 1 level "Male": 1  
 $ Mobile: num 35
```

Binary Formats in R:

The complement to the textual format is the binary format. Binary format is sometimes useful for efficiency purposes. Sometimes, it may happen that there is no useful way to represent your data in a textual manner then binary format helps to import and export data in R. The main functions for converting R objects into a binary format are **save()**, **save.image()**, and **serialize()**. Individual R objects can be saved to a file using the **save()** function.

```
x <- data.frame(col1 = rep(10,10), col2 = runif(10,min=0,max=10))
y<-rnorm(10)
z<-100:110
#Save 'x', 'y' and 'z' to a file
save(x,y,z,file="F:/testdata.rda")
#OR
save(x,y,z,file="F:/testdata.rData")
#Load 'x', 'y' and 'z' into your workspace
load("F:/testdata.rda")
#OR
load("F:/testdata.rData")
```

If you have a lot of objects that you want to save to a file in one run, you can save all objects in your workspace using the **save.image()** function.

```
# Save everything to a file
save.image(file = "F://mydata.RData")
#load all objects in this file
load("F://mydata.RData")
```

serialize() and unserialize() function in R:

The **serialize()** function is used to convert individual R objects into a binary format that can be communicated across an arbitrary connection. When you call **serialize()** on an R object, the output will be a raw vector coded in hexadecimal format. The benefit of the **serialize()** function is that it is the only way to perfectly represent an R object in an exportable format, without losing precision or any metadata. If that is what you need, then **serialize()** is the function for you.

```
x<-list(1,2,3)
s<-serialize(x, NULL)
```

```
s  
save(s,file="F:/test_serialization.rda")  
load("F:/test_serialization.rda")  
unserialize(s)
```

saveRDS() and readRDS() in R:

Now you are familiar with save() and load() function in R. They allow you to save a named R object to a file or other connection and restore that object again. When loaded the named object is restored to the current environment with the same name it had when saved. This is annoying for example when you have a saved model object resulting from a previous fit and you want to compare it with the model object returned when the R code is rerun. Unless you change the name of the model fit object in your script you can't have both the saved object and the newly created one available in the same environment at the same time. **saveRDS()** provides a far better solution to this problem and to the general one of saving and loading objects created with R. saveRDS() serializes an R object into a format that can be saved.

save() does the same thing, but with one important difference; saveRDS() doesn't save the both the object and its name it just saves a representation of the object. As a result, the saved object can be loaded into a named object within R that is different from the name it had when originally serialized. The main difference is that save() can save many objects to a file in a single call, whilst saveRDS(), being a lower-level function, works with a single object at a time.

```
# save a single object to file  
women  
saveRDS(women, "F://women.rds")  
# restore it under a different name  
women2 <- readRDS("F://women.rds")  
identical(women, women2)
```

Output:

```
> women
```

```
height weight
```

```
1 58 115
```

```
2 59 117
```

```
3 60 120
4 61 123
5 62 126
6 63 129
7 64 132
8 65 135
9 66 139
10 67 142
11 68 146
12 69 150
13 70 154
14 71 159
15 72 164
identical(women, women2)
[1] TRUE
```

CSV Files in R

In R can read and write into various file formats like csv, excel,json, xml etc. The csv file is a text file in which the values in the columns are separated by a comma. **read.csv()** function is used to read a CSV file in your working directory. Similarly, **write.csv()** function is used to write the csv file. You can download the sample data set by clicking [here](#) and then read it using **read.csv()** function.

Download the IRIS Data Set

Reading a CSV File in R:

```
mydata <- read.csv(file="https://makemeanalyst.com/wp-
content/uploads/2017/06/iris.csv", header=TRUE, sep=",")
head(mydata)
dim(mydata)
summary(mydata)
```

Output:

```
> head(mydata)
X Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1 1 5.1 3.5 1.4 0.2 setosa
2 2 4.9 3.0 1.4 0.2 setosa
```

```

3 3 4.7 3.2 1.3 0.2 setosa
4 4 4.6 3.1 1.5 0.2 setosa
5 5 5.0 3.6 1.4 0.2 setosa
6 6 5.4 3.9 1.7 0.4 setosa
> dim(mydata)
[1] 150 6
> summary(mydata)
X Sepal.Length Sepal.Width Petal.Length Petal.Width Species
Min. : 1.00 Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100 setosa :50
1st Qu.: 38.25 1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300 versicolor:50
Median : 75.50 Median :5.800 Median :3.000 Median :4.350 Median :1.300 virginica :50
Mean : 75.50 Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199
3rd Qu.:112.75 3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800
Max. :150.00 Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500

```

Writing into a CSV File:

The write.csv() function is used to create the csv file.

```

mydata <- read.csv(file="https://makemeanalyst.com/wp-
content/uploads/2017/06/iris.csv", header=TRUE, sep=",")
t<-tail(mydata)
t
write.csv(t,"Iris_tail.csv", row.names = FALSE)

```

Output:

```

X Sepal.Length Sepal.Width Petal.Length Petal.Width Species
145 145 6.7 3.3 5.7 2.5 virginica
146 146 6.7 3.0 5.2 2.3 virginica
147 147 6.3 2.5 5.0 1.9 virginica
148 148 6.5 3.0 5.2 2.0 virginica
149 149 6.2 3.4 5.4 2.3 virginica
150 150 5.9 3.0 5.1 1.8 virginica

```

Unit 6

Graphical

analysis in R

Topics

Basic Plotting in R

Scatter Plots

Line Plot

Histogram

Boxplot

Bar Plot

Graphical analysis in R involves using visual representations to explore, analyze, and interpret data. R is a powerful statistical computing language with extensive capabilities for creating various types of plots and charts. Here's a detailed guide to performing graphical analysis using R, covering the basics, common types of plots, and practical examples.

Data visualization is a technique used for the graphical representation of data. By using elements like scatter plots, charts, graphs, histograms, maps, etc., we make our data more understandable. Data visualization makes it easy to recognize patterns, trends, and exceptions in our data. It enables us to convey information and results in a quick and visual way. It is easier for a human brain to understand and retain information when it is represented in a pictorial form. Therefore, Data Visualization helps us interpret data quickly, examine different variables to see their effects on the patterns, and derive insights from our data.

Getting Started

Before creating plots, ensure you have R and a suitable IDE like RStudio installed. You can use R's built-in plotting functions or packages like `ggplot2` for more advanced and aesthetically pleasing graphics.

Base R Graphics

There are some key elements of a statistical graphic. These elements are the basics of the grammar of graphics. R provides some built-in functions which are included in the graphics package for data visualization in R. Let's discuss each of the elements one by one to gain the basic knowledge of graphics.

Now we are going to use the default mtcars dataset for data visualization in R.

```
#To load graphics package  
library("graphics")  
  
#To load datasets package  
library("datasets")  
  
#To load mtcars dataset  
data(mtcars)  
  
#To analyze the structure of the dataset  
str(mtcars)
```

Output:

```
1 | 'data.frame': 32 obs. of 11 variables:  
2 | $ mpg : num 21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...  
3 | $ cyl : num 6 6 4 6 8 6 8 4 4 6 ...  
4 | $ disp: num 160 160 108 258 360 ...  
5 | $ hp : num 110 110 93 110 175 105 245 62 95 123 ...  
6 | $ drat: num 3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...  
7 | $ wt : num 2.62 2.88 2.32 3.21 3.44 ...  
8 | $ qsec: num 16.5 17 18.6 19.4 17 ...  
9 | $ vs : num 0 0 1 1 0 1 0 1 1 1 ...  
10 | $ am : num 1 1 1 0 0 0 0 0 0 0 ...  
11 | $ gear: num 4 4 4 3 3 3 3 4 4 4 ...  
12 | $ carb: num 4 4 1 1 2 1 4 2 2 4 ...
```

It contains data about the design, performance and fuel economy of 32 automobiles from 1973 to 1974, extracted from the 1974 Motor Trend US magazine.

Basic Plotting in R

R provides several base plotting functions. Here's a quick overview:

The plot() Function

The plot() function is used to plot R objects.

The basic syntax for the plot() function is given below:

```
1 | plot(x,y,type,main,sub,xlab,ylab,asp,col,..)
```

x:— The x coordinate of the plot, a single plotting structure, a function, or an R object

y:— The Y coordinate points in the plot (optional if x coordinate is a single structure)

type:— ‘p’ for points, ‘l’ for lines, ‘b’ for both, ‘h’ for high-density vertical lines, etc.

main:— Title of the plot

sub:— Subtitle of the plot

xlab:— Title for the x-axis

ylab:— Title for the y-axis

asp :- Aspect ratio(y/x)

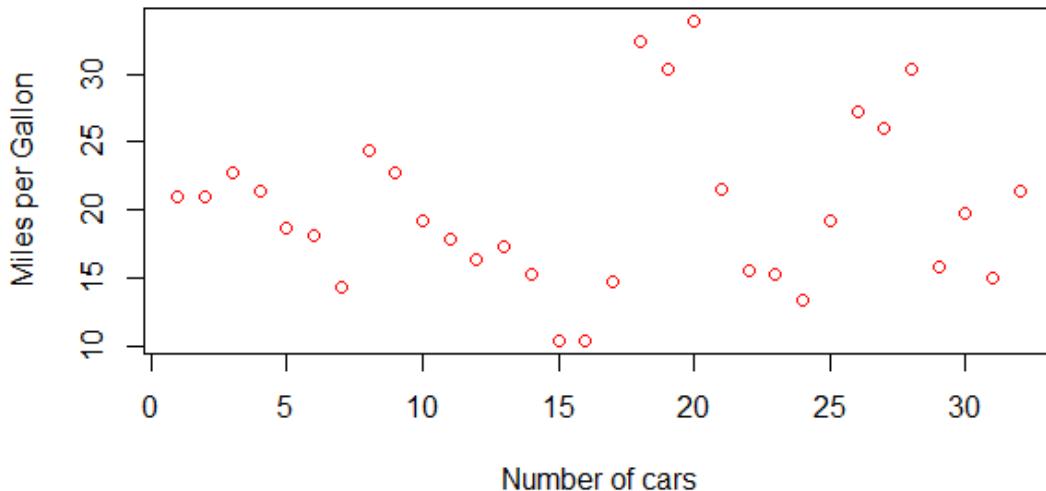
col:— Color of the plot(points, lines, etc.)

For example:

```
#To plot mpg(Miles per Gallon) vs Number of cars
```

```
plot(mtcars$mpg, xlab = "Number of cars", ylab = "Miles per Gallon", col = "red")
```

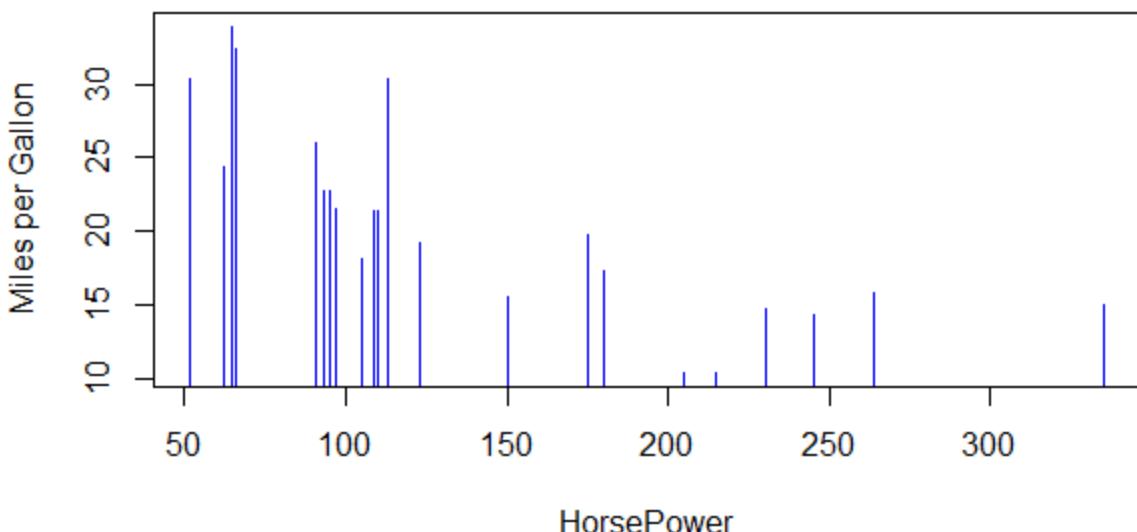
Output:



Here, we get a scatter/dot plot wherein we can observe that there are only six cars with miles per gallon (mpg) more than 25.

Plot the correlation between 2 variables

```
#To find relation between hp (Horse Power) and mpg (Miles per Gallon)
plot(mtcars$hp,mtcars$mpg, xlab = "HorsePower", ylab = "Miles per Gallon", type = "h", col =
"blue")
```



Here, we can observe that hp and mpg have a negative correlation, which means that as Horse Power increases Miles per Gallon decreases.

1. Scatter Plot

Scatter plots are used to visualize the relationship between two continuous variables.

#Sample data

```
x <- c(1, 2, 3, 4, 5)
```

```
y <- c(2, 4, 6, 8, 10)
```

Basic scatter plot

```
plot(x, y, main="Scatter Plot", xlab="X-axis", ylab="Y-axis", pch=19)
```

2. Line Plot

Line plots are used to visualize data trends over time or other continuous variables.

Sample data

```
x <- 1:10
```

```
y <- c(2, 3, 5, 7, 11, 13, 17, 19, 23, 29)
```

Basic line plot

```
plot(x, y, type="l", main="Line Plot", xlab="X-axis", ylab="Y-axis")
```

3. Histogram

Histograms display the distribution of a single continuous variable.

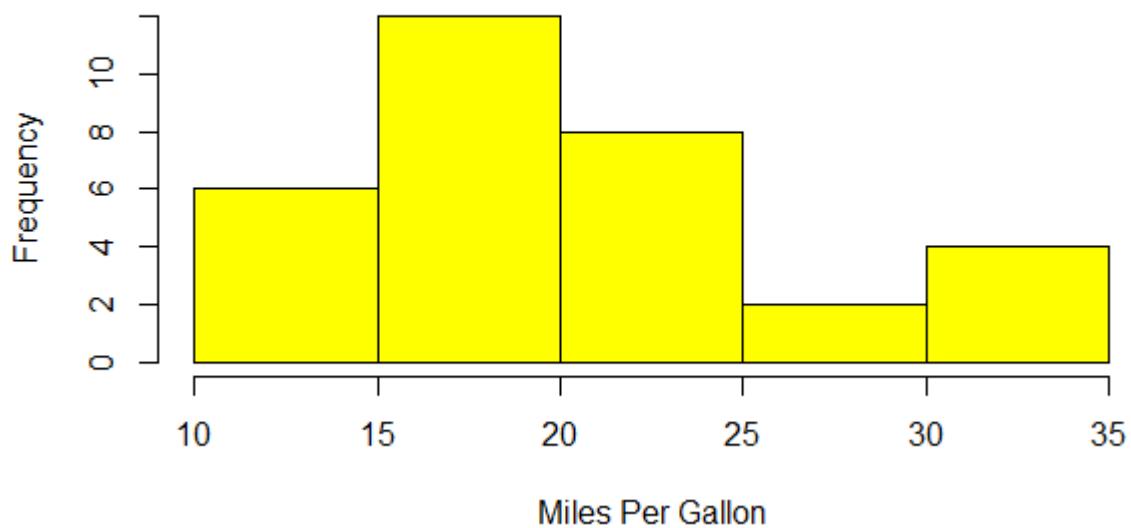
It is used to divide values into groups of continuous ranges measured against the frequency range of the variable.

For example:

```
#To find histogram for mpg (Miles per Gallon)
```

```
hist(mtcars$mpg,xlab = "Miles Per Gallon", main = "Histogram for MPG", col = "yellow")
```

Histogram for MPG



#Sample data

```
data <- rnorm(1000) Generating random data
```

Basic histogram

```
hist(data,      main="Histogram",      xlab="Values",      ylab="Frequency",      col="lightblue",
border="black")
```

4. Bar Plot

Bar plots are used to compare different categories.

Sample data

```
categories <- c("A", "B", "C")
values <- c(3, 7, 5)
```

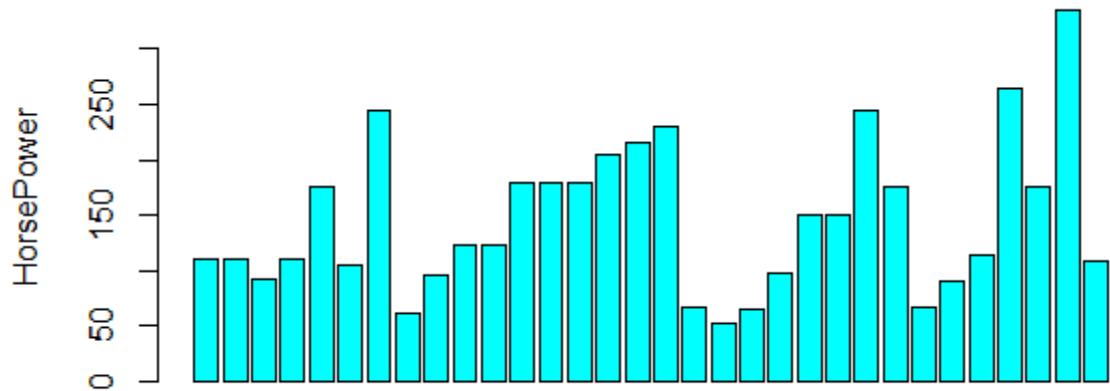
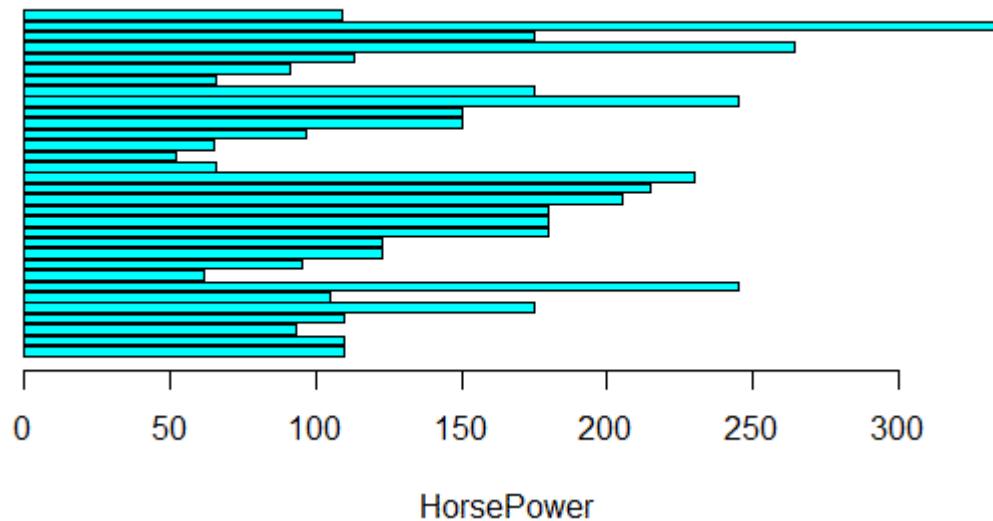
Basic bar plot

It is used to represent data in the form of rectangular bars, both in vertical and horizontal ways, and the length of the bar is proportional to the value of the variable. For example:

```
#To draw a barplot of hp
#Horizontal
barplot(mtcars$hp,xlab = "HorsePower", col = "cyan", horiz = TRUE)
```

```
#Vertical
barplot(mtcars$hp, ylab = "HorsePower", col = "cyan", horiz = FALSE)

barplot(values, names.arg=categories, main="Bar Plot", xlab="Categories", ylab="Values",
col="lightgreen")
```



5. Box Plot

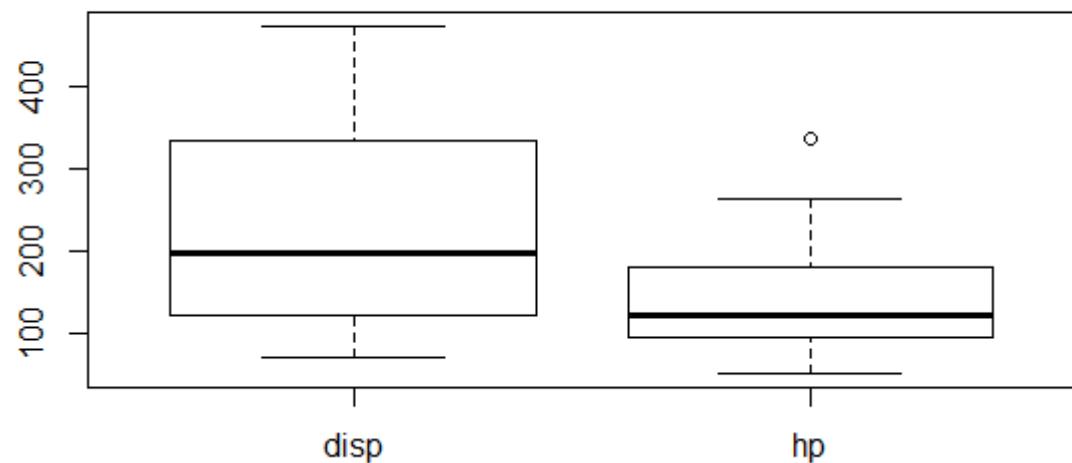
Box plots are used to show the distribution of a continuous variable and highlight outliers.

It is used to represent descriptive statistics of each variable in a dataset. It represents the minimum, first quartile, median, third quartile, and the maximum values of a variable.

```
#To draw boxplots for disp (Displacement) and hp (Horse Power)
```

```
boxplot(mtcars[,3:4])
```

Box Plots



Sample data

```
data <- list(Group1=rnorm(50), Group2=rnorm(50, mean=5))
```

#Basic box plot

```
boxplot(data, main="Box Plot", xlab="Groups", ylab="Values", col=c("lightpink", "lightblue"))
```

Advanced Plotting with ggplot2

The **ggplot2** package in R is based on the **grammar of graphics**, which is a set of rules for describing and building graphs. By breaking up graphs into semantic components such as scales and layers, **ggplot2** implements the grammar of graphics.

The ggplot2 grammar of graphics is composed of the following:

- Data
- Layers
- Scales
- Coordinates
- Faceting
- Themes

ggplot2 is one of the most sophisticated packages in R for data visualization, and it helps create the most elegant and versatile print-quality plots with minimal adjustments. It is very simple to create single- and **multivariable** graphs with the help of the **ggplot2** package.

'**ggplot2**' is a popular package for creating sophisticated and customizable plots. It uses a grammar of graphics approach.

The three basic components to build a ggplot are as follows:

- **Data**:- Dataset to be plotted
 - **Aesthetics**:- Mapping of data to visualization
 - **Geometry/Layers**:- Visual elements used for the data
-
- The basic syntax for **ggplot** is given below:

```
ggplot(data = NULL, mapping = aes()) + geom_function()
```

```
#To Install and load the ggplot2 package
```

```
install.packages("ggplot2")
```

```
library(ggplot2)
```

we are going to use the **mtcars** dataset from the datasets package in R that can be loaded as follows:

```
#To load datasets package  
  
library("datasets")  
  
#To load iris dataset  
  
data(mtcars)  
  
#To analyze the structure of the dataset  
  
str(mtcars)
```

Scatter Plot with ggplot2

```
# Sample data  
df <- data.frame(x=1:10, y=c(2, 3, 5, 7, 11, 13, 17, 19, 23, 29))  
  
# Scatter plot  
ggplot(df, aes(x=x, y=y)) +  
  geom_point() +  
  ggtitle("Scatter Plot") +  
  xlab("X-axis") +  
  ylab("Y-axis")
```

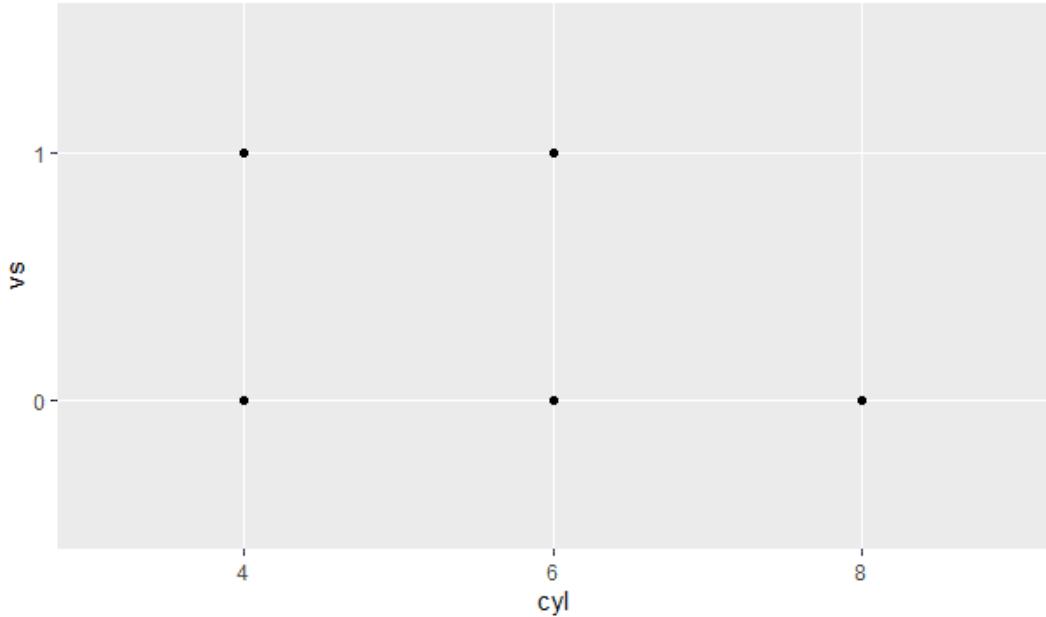
To draw a scatter plot of cyl(Number of Cylinders) and vs(Engine Type(0 = V-shaped, 1 = straight)), run the code below:

```
#Since the following columns have discrete(categorical) set of values, So we can  
convert them to factors for optimal plotting  
mtcars$am <- as.factor(mtcars$am)  
mtcars$cyl <- as.factor(mtcars$cyl)
```

```

mtcars$vs <- as.factor(mtcars$vs)
mtcars$gear <- as.factor(mtcars$gear)
#To draw scatter plot
ggplot(mtcars, aes(x= cyl , y= vs)) + geom_point()

```

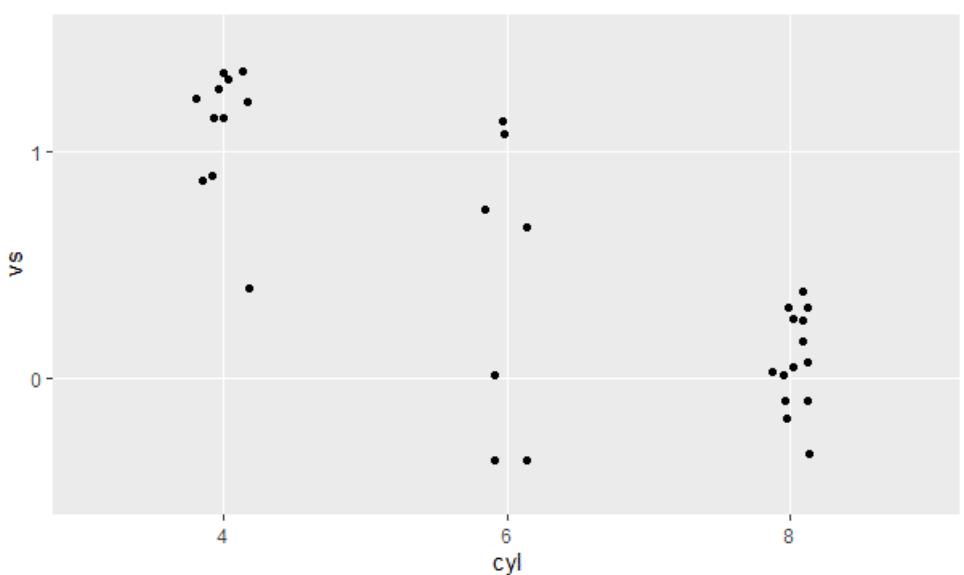


Since this plot has a lot of overlapped values, which is known as **overplotting**, we will use **geom_jitter()** function to add a certain amount of noise to avoid it.

```

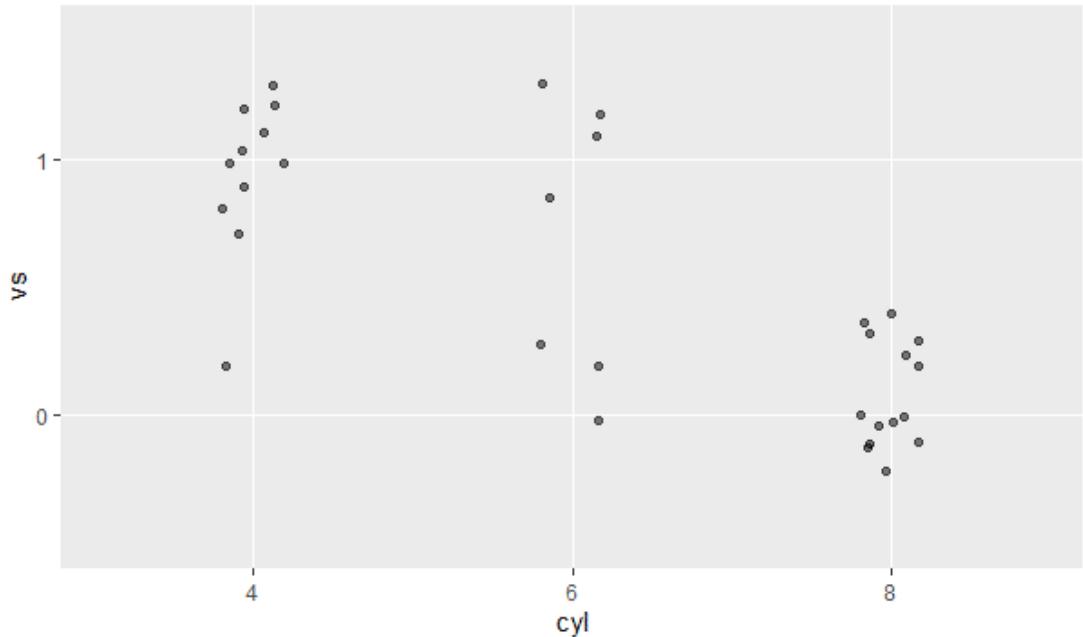
#Here width argument is used to set the amount of jitter
ggplot(mtcars, aes(x= cyl , y= vs)) + geom_jitter(width = 0.1)

```



Here, we can also use the argument **alpha** to set the transparency of the points to further reduce overplotting for data visualization in R.

```
#Transparency set to 50%
ggplot(mtcars, aes(x= cyl , y= vs)) + geom_jitter(width = 0.1, alpha = 0.5)
```

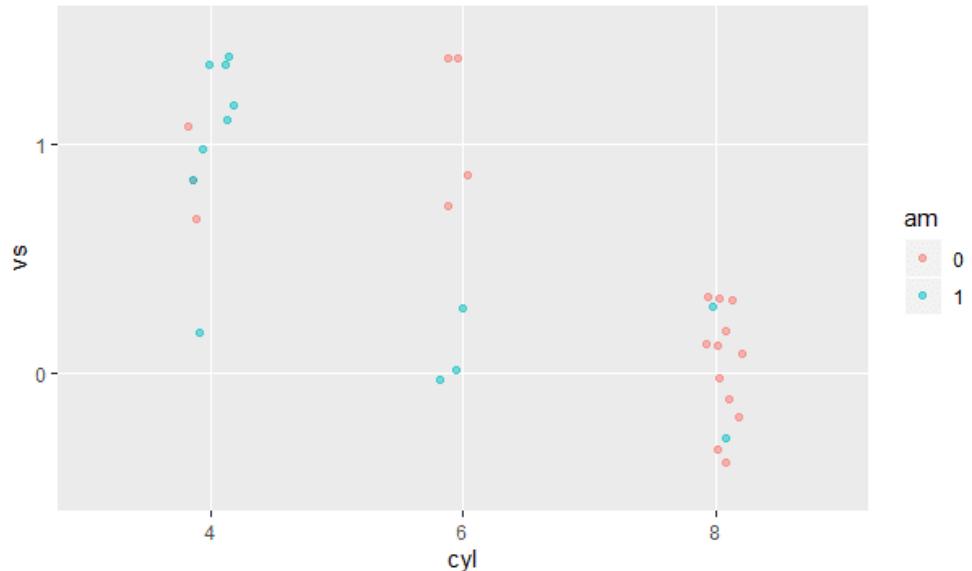


With **ggplot2**, we can plot multivariate plots effectively.

For example:

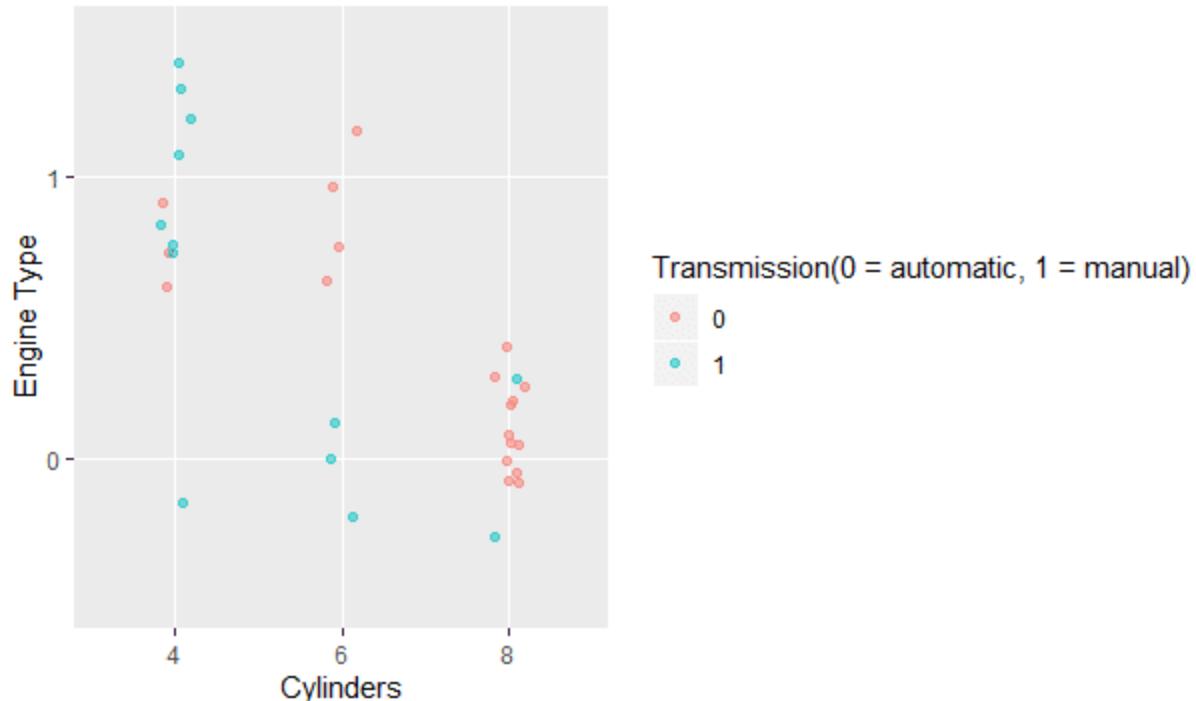
To draw a scatter plot of cyl(Number of Cylinders) and vs(Engine Type(0 = V-shaped, 1 = straight)) according to **am** Transmission (0 = automatic, 1 = manual), run the following code:-

```
#We use the color aesthetic to introduce third variable with a legend on the right side
ggplot(mtcars, aes(x= cyl,y= vs,color = am)) + geom_jitter(width = 0.1, alpha = 0.5)
```



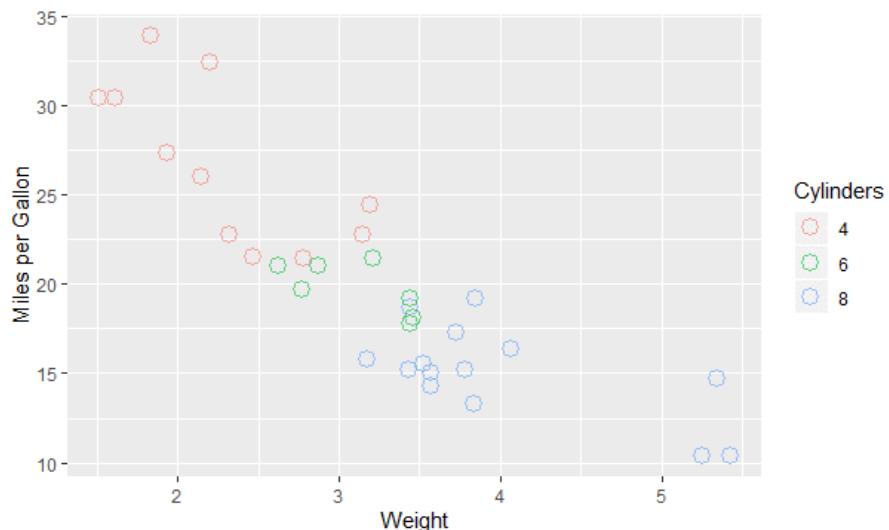
#To add the labels

```
ggplot(mtcars, aes(x= cyl , y= vs ,color = am)) +  
  geom_jitter(width = 0.1, alpha = 0.5) +  
  labs(x = "Cylinders",y = "Engine Type", color = "Transmission(0 = automatic, 1 = manual)")
```



#To plot with shape =1 and size = 4

```
ggplot(mtcars, aes(x = wt, y = mpg, col = cyl)) +  
  geom_point(size = 4, shape = 1, alpha = 0.6) +  
  labs(x = "Weight",y = "Miles per Gallon", color = "Cylinders")
```



Line Plot with ggplot2

```
# Sample data
```

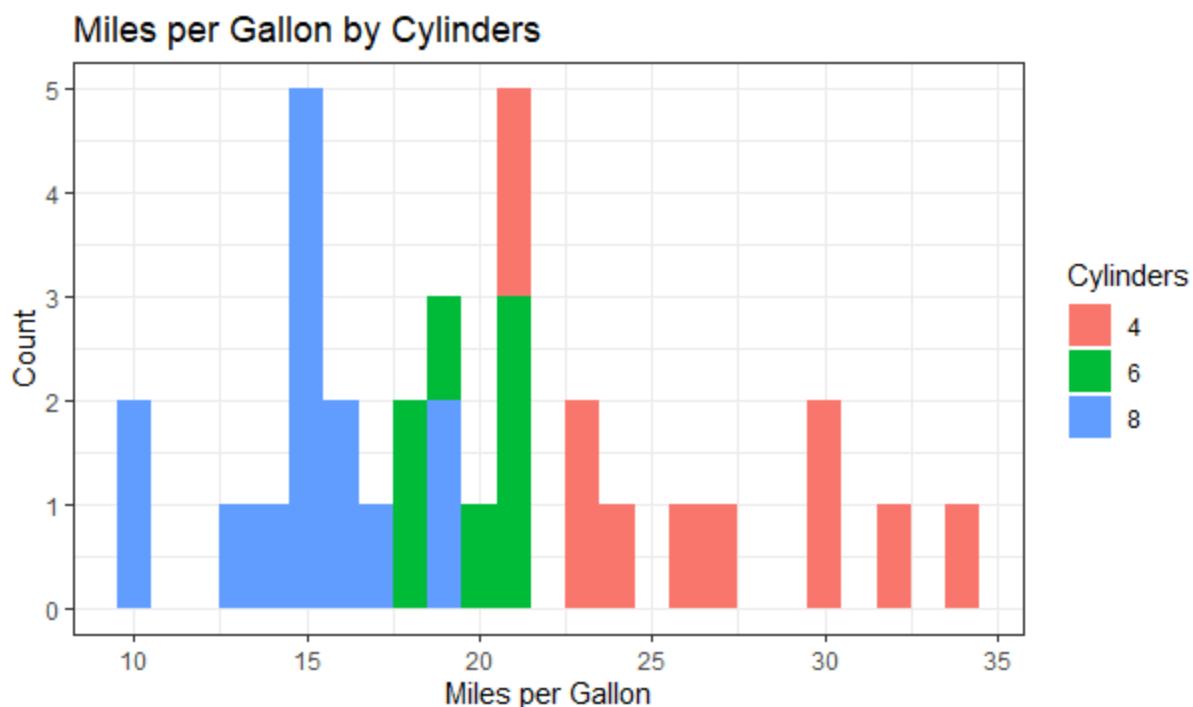
```
df<- data.frame(x=1:10, y=c(2, 3, 5, 7, 11, 13, 17, 19, 23, 29))
```

```
# Line plot
```

```
ggplot(df, aes(x=x, y=y)) + geom_line() + ggtitle("Line Plot") + xlab("X-axis") + ylab("Y-axis")
```

Histogram with ggplot2

```
#To plot a histogram for mpg (Miles per Gallon),  
according to cyl(Number of Cylinders), we use the geom_histogram() functiong  
gplot(mtcars, aes(mpg,fill = cyl)) +  
geom_histogram(binwidth = 1)+  
theme_bw() +  
labs(title = "Miles per Gallon by Cylinders",x = "Miles per Gallon",y =  
"Count",fill = "Cylinders")
```



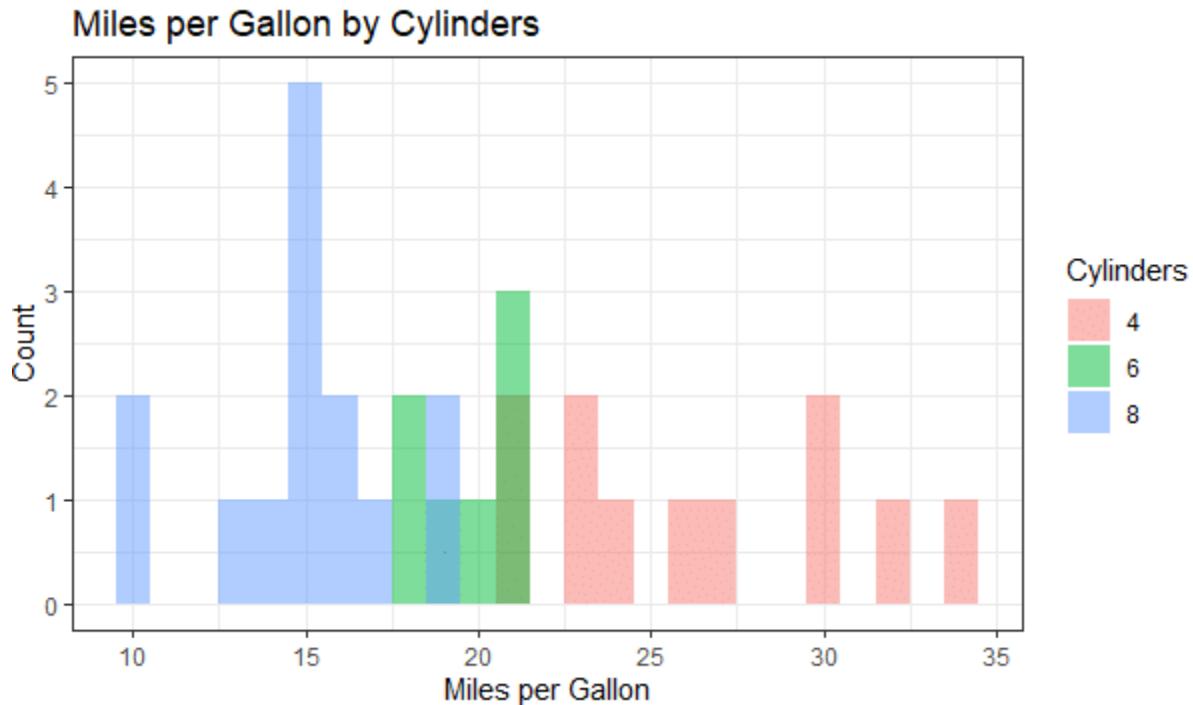
```
#To show overlapping, we set position to identity and alpha to 0.5
```

```
ggplot(mtcars, aes(mpg,fill = cyl)) +
```

```

geom_histogram(binwidth = 1, position = "identity", alpha = 0.5) +
theme_bw() +
labs(title = "Miles per Gallon by Cylinders", x = "Miles per Gallon", y = "Count", fill =
"Cylinders")

```



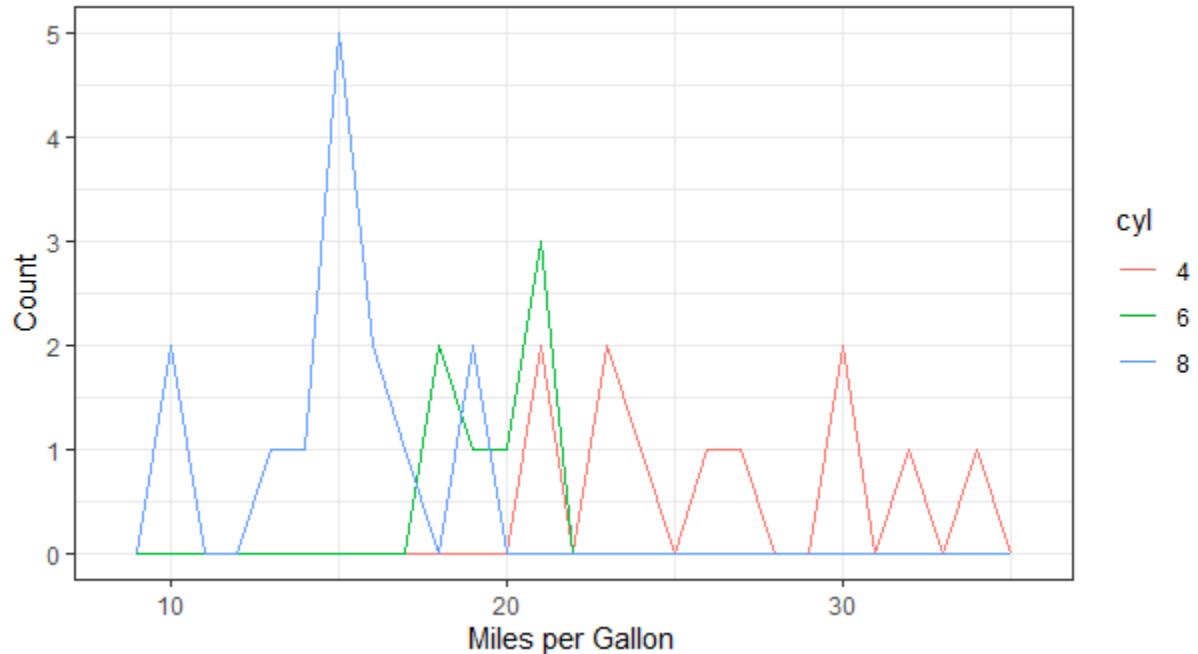
#To overcome overlapping, we can use the frequency polygon, as follows:

```

ggplot(mtcars, aes(mpg, color = cyl)) + geom_freqpoly(binwidth = 1) +
theme_bw() + labs(title = "Miles per Gallon by Cylinders", x = "Miles per Gallon", y = "Count",
fill = "Cylinders")

```

Miles per Gallon by Cylinders



Sample data

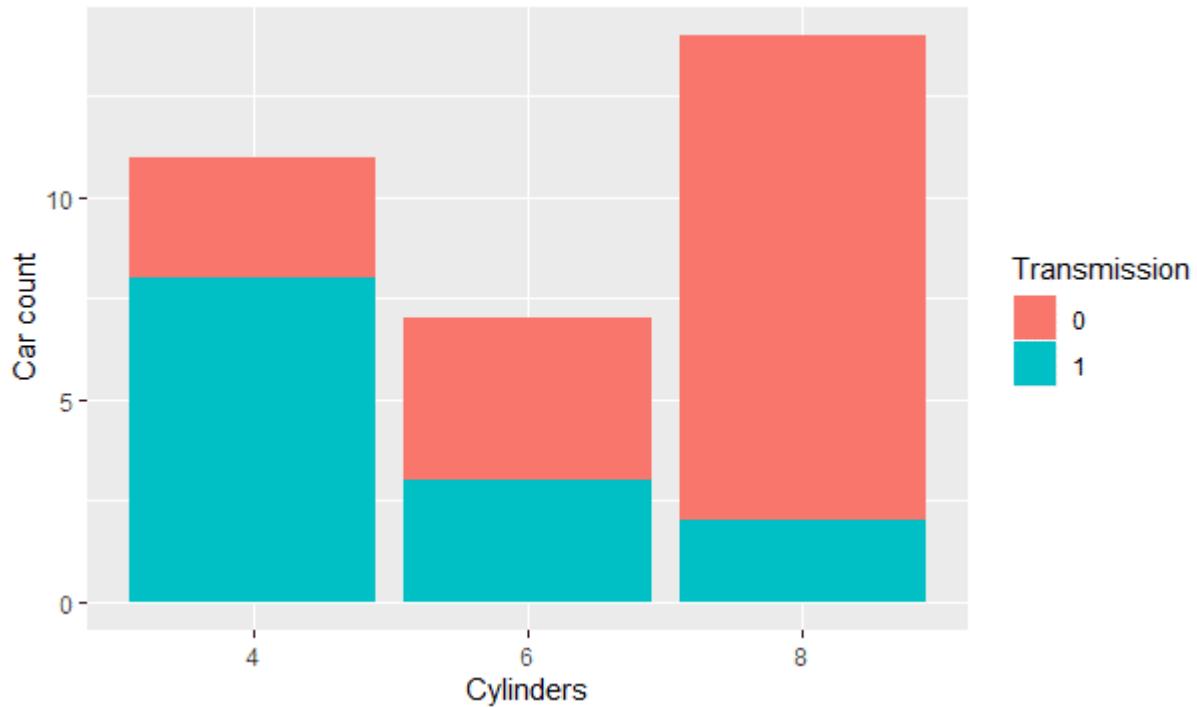
```
df <- data.frame(values=rnorm(1000))
```

Histogram

```
ggplot(df, aes(x=values)) +  
  geom_histogram(binwidth=0.5, fill="lightblue", color="black") +  
  ggtitle("Histogram") +  
  xlab("Values") +  
  ylab("Frequency")
```

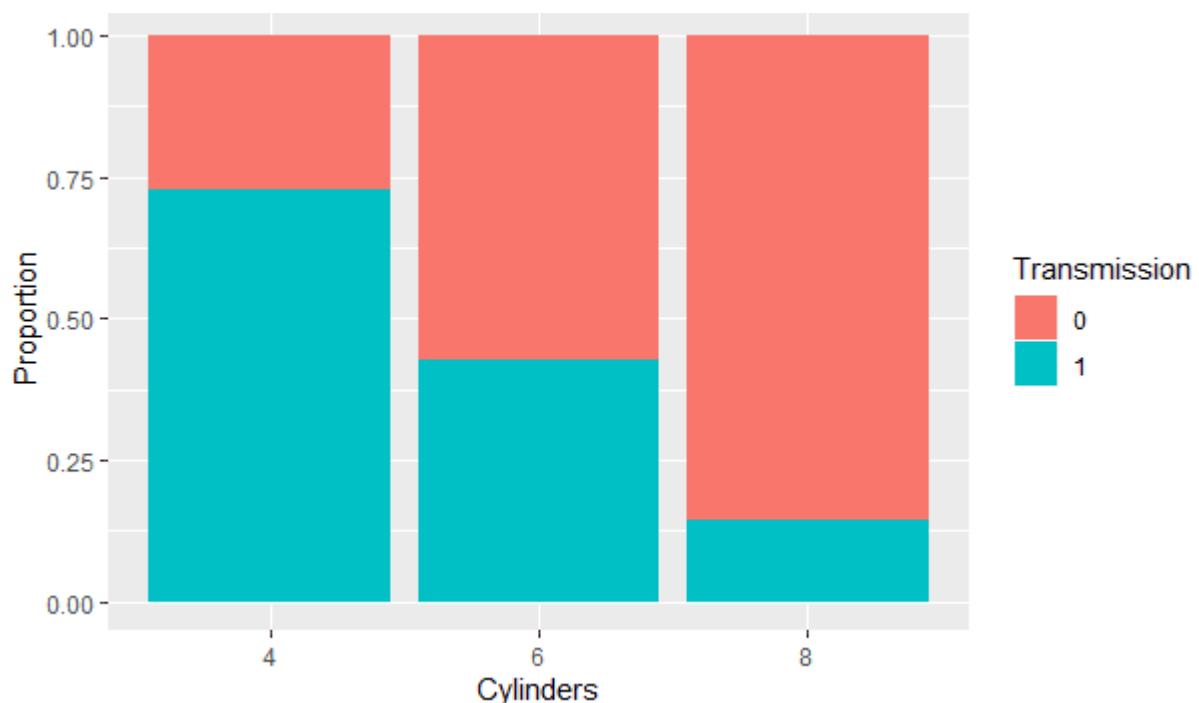
Bar Plot with ggplot2

```
#To draw a bar plot of cyl(Number of Cylinders) according to the Transmission type  
using <strong>geom_bar() and fill()</strong>  
ggplot(mtcars, aes(x = cyl, fill = am)) +  
  geom_bar() +  
  labs(x = "Cylinders", y = "Car count", fill = "Transmission")
```



#To find the proportion, we use position argument,as follows:

```
ggplot(mtcars, aes(x = cyl, fill = am)) +
  geom_bar(position = "fill") +
  labs(x = "Cylinders",y = "Proportion",fill = "Transmission")
```



Themes

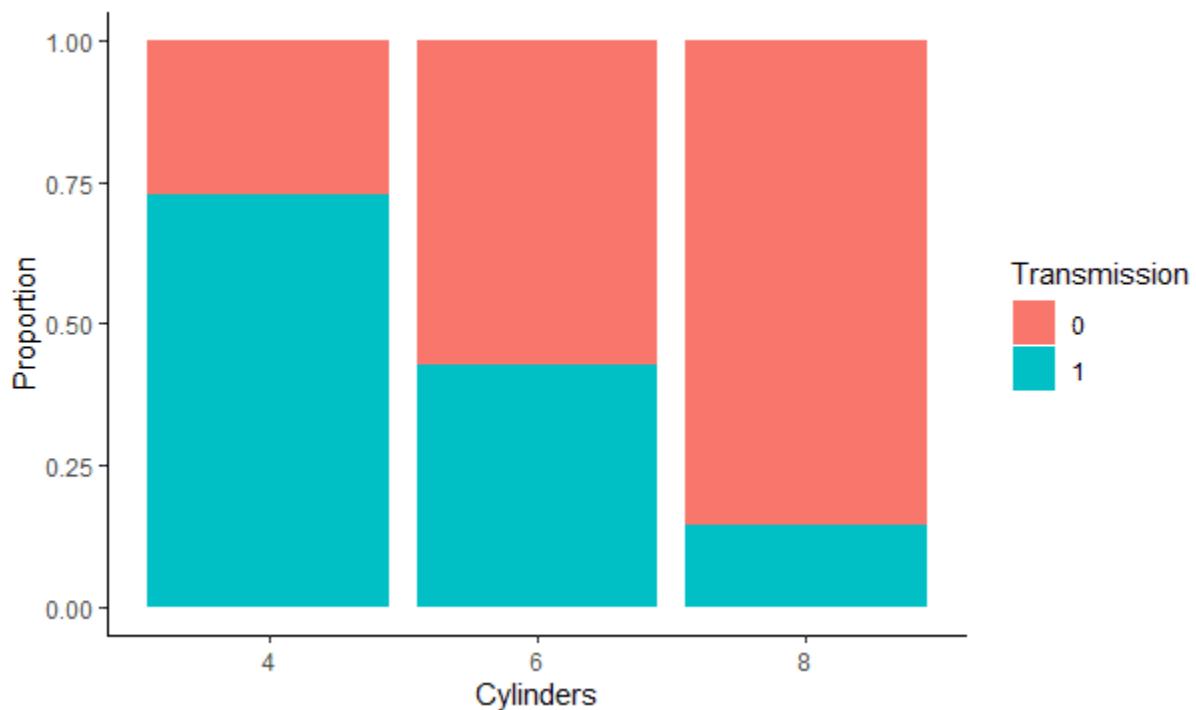
It is used to change the attributes of non-data elements of our plot like text, lines, background, etc. We use the **theme_function()** to make changes to these elements for data

visualization in R.

Some of the commonly used theme function is as follows:

- **theme_bw()** :- For white background and gray grid lines
- **theme_gray:-** For gray background and white grid lines
- **theme_linedraw:-** For black lines around the plot
- **theme_light:-** For light gray lines and axis
- **theme_void:-** An empty theme, useful for plots with non-standard coordinates or for drawings
- **theme_dark():-** A dark background designed to make colors pop out

```
ggplot(mtcars, aes(x = cyl, fill = am)) +  
  geom_bar(position = "fill") +  
  theme_classic() +  
  labs(x = "Cylinders", y = "Proportion", fill = "Transmission")
```

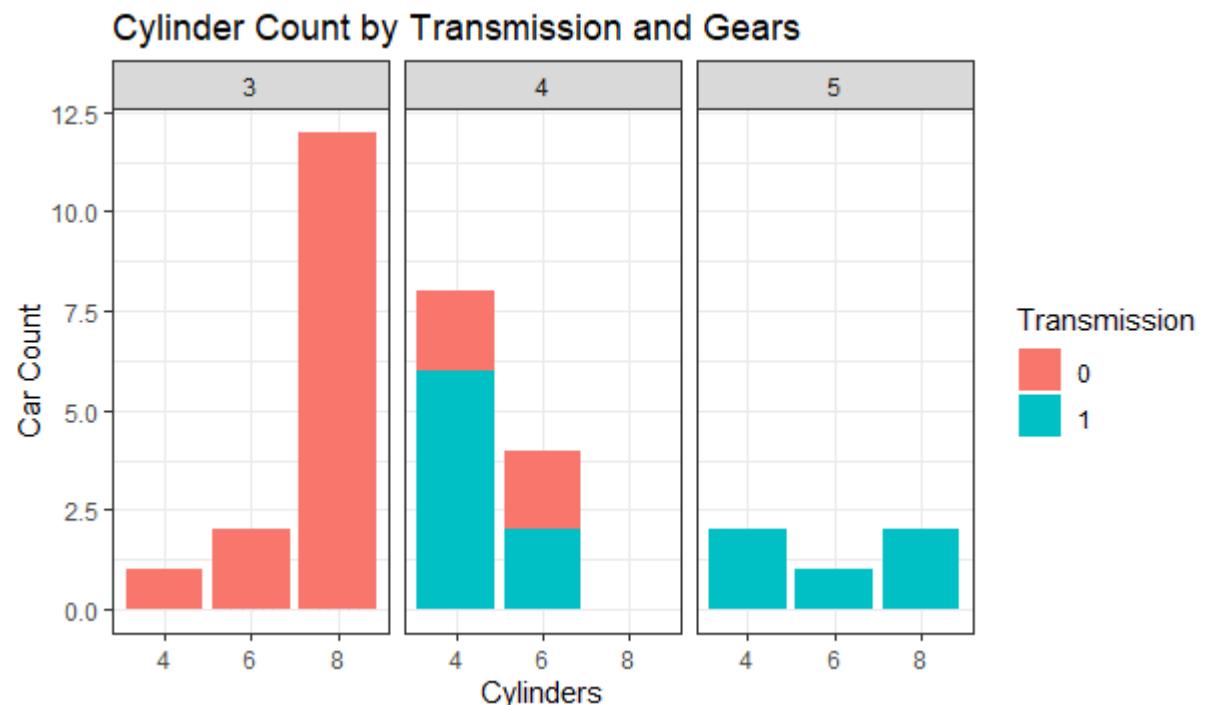


Faceting

It is used to further drill down data and split the data by one or more variables, and then plot the subsets of the data altogether for optimum data visualization in R. For example:

```
#To facet the following plot according to gear(Number of Gears(3,4,5)), we use  
facet_grid() function as follows:
```

```
ggplot(mtcars, aes(x = cyl, fill = am)) +  
  geom_bar() +  
  facet_grid(.~gear)+  
#facet_grid(rows ~ columns) theme_bw() + labs(title = "Cylinder count by transmission and Gears",  
x = "Cylinders", y = "Count",fill = "Transmission")
```



Sample data

```
df <- data.frame(Category=c("A", "B", "C"), Value=c(3, 7, 5))
```

Bar plot

```
ggplot(df, aes(x=Category, y=Value)) +  
  geom_bar(stat="identity", fill="lightgreen") +  
  ggtitle("Bar Plot") +  
  xlab("Category") +
```

```
ylab("Value")
```

Box Plot with ggplot2

```
# Sample data
```

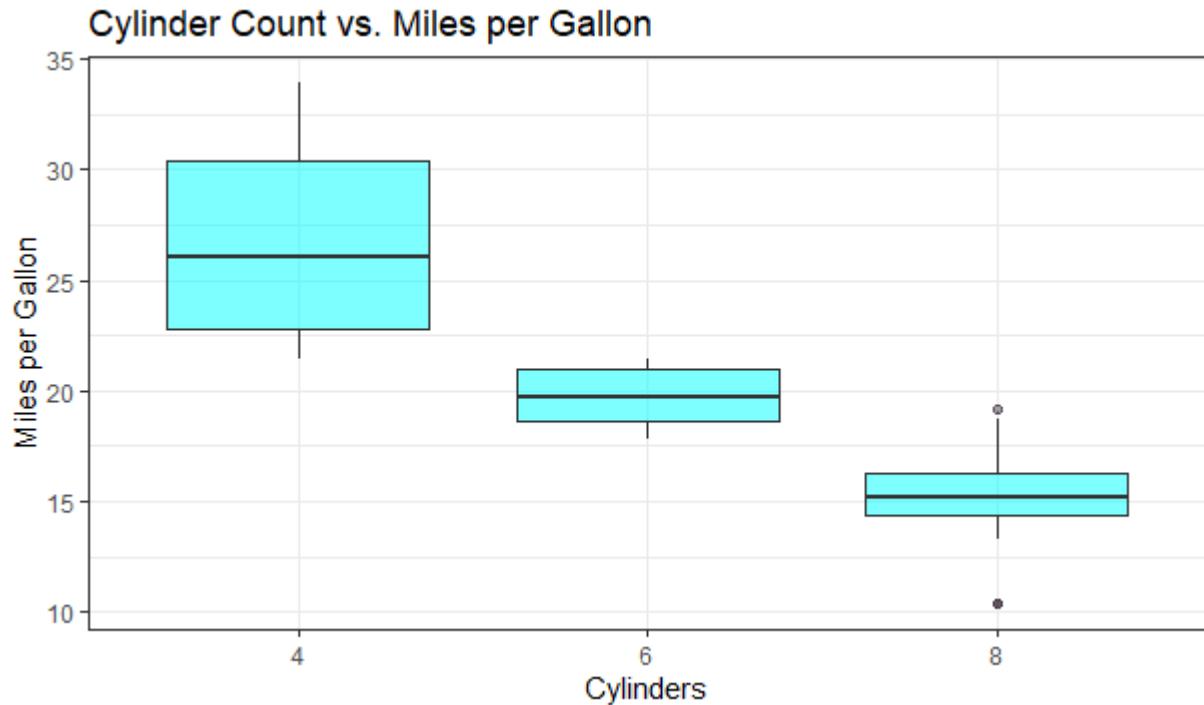
```
df <- data.frame(Group=rep(c("Group1", "Group2"), each=50), Value=c(rnorm(50), rnorm(50, mean=5)))
```

```
# Box plot
```

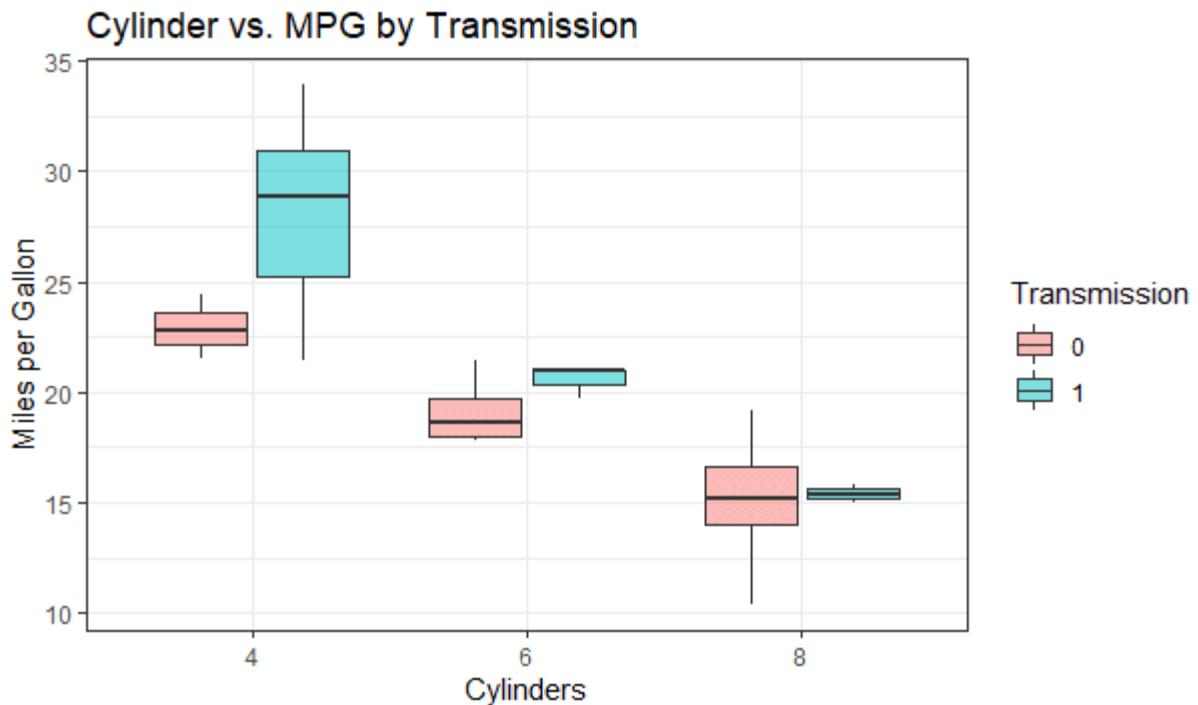
```
ggplot(df, aes(x=Group, y=Value, fill=Group)) +  
  geom_boxplot() +  ggttitle("Box Plot") +  xlab("Group") +  ylab("Value")
```

```
#To draw a Box plot
```

```
ggplot(mtcars, aes(x = cyl,y = mpg)) +  geom_boxplot(fill = "cyan", alpha = 0.5) +  
  theme_bw() +  labs(title = "Cylinder count vs Miles per Gallon",x = "Cylinders",  
y = "Miles per Gallon")
```



```
#To draw a Box plot
ggplot(mtcars, aes(x = cyl,y = mpg,fill = am)) +
  geom_boxplot( alpha = 0.5) +
  theme_bw() +
  labs(title = "Cylinder vs MPG by Transmission",x = "Cylinders",
       y = "Miles per Gallon",fill = "Transmission")
```



Customizing Plots

Both base R graphics and `ggplot2` offer extensive customization options:

- Titles and Labels: Add titles and axis labels to make your plots more informative.
- Colors and Themes: Customize colors and themes to enhance visual appeal.
- Legends and Annotations: Add legends, annotations, and text to provide more context.

Saving Plots

You can save plots to files using `ggsave` in `ggplot2` or functions like `png()`, `jpeg()`, and `pdf()` in base R.

Saving with ggplot2

```
# Save a ggplot2 plot
ggplot(df, aes(x=x, y=y)) +
  geom_point() +
  ggtitle("Scatter Plot") +
  xlab("X-axis") +
  ylab("Y-axis") +
  ggsave("scatter_plot.png")
```

Saving with base R

```
# Save a base R plot
png("scatter_plot.png")
plot(x, y, main="Scatter Plot", xlab="X-axis", ylab="Y-axis", pch=19)
dev.off()
```

Advantages of Data Visualization in R:

Some of the advantages of R over other tools for data visualization include:

- R offers a broad collection of visualization libraries in addition to 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.

Disadvantages of Data Visualization in R:

There are also few disadvantages of R in Data Visualization as follows:

- 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.

Unit 7

Advance R

Topics

Advanced R

Statistical models in R

Correlation and regression analysis

Analysis of Variance (ANOVA)

Creating data for complex analysis

Summarizing data, and case studies.

Correlation and Regression Analysis in R

This section contains R methods for computing and visualizing correlation analyses. Recall that, correlation analysis is used to investigate the association between two or more variables. A simple example, is to evaluate whether there is a link between maternal age and child's weight at birth.

Correlation analysis

1. Pearson correlation (r), which measures a linear dependence between two variables (x and y). It's also known as a parametric correlation test because it depends to the distribution of the data. It can be used only when x and y are from normal distribution. The plot of $y=f(x)$ is named the linear regression curve. The Pearson correlation formula is:

$$r = \frac{\sum (x - m_x)(y - m_y)}{\sqrt{\sum (x - m_x)^2 \sum (y - m_y)^2}}$$

where m_x and m_y are means of the distributions x and y respectively.

2. Kendall tau and Spearman rho, which are rank-based correlation coefficients (non-parametric)
3. The Spearman correlation method computes the correlation between the rank of x and the rank of y variables.

$$\rho = \frac{\sum (x' - m_{x'})(y'_i - m_{y'})}{\sqrt{\sum (x' - m_{x'})^2 \sum (y' - m_{y'})^2}}$$

where $x'=\text{rank}(x)$ and $y'=\text{rank}(y)$.

The Kendall correlation method measures the correspondence between the ranking of x and y variables. The total number of possible pairings of x with y observations is $n(n-1)/2$, where n is the size of x and y.

The procedure is as follow:

1. Begin by ordering the pairs by the x values. If x and y are correlated, then they would have the same relative rank orders.
2. Now, for each y_i , count the number of $y_j > y_i$ (concordant pairs (c)) and the number of $y_j < y_i$ (discordant pairs (d)).

Kendall correlation distance is defined as follow:

$$tau = \frac{n_c - n_d}{\frac{1}{2}n(n - 1)}$$

Where

- n_c : total number of concordant pairs
- n_d : total number of discordant pairs
- n : size of x and y

R method to find correlation coefficient

Correlation coefficient can be computed using the functions cor() or cor.test():

Syntax:

```
cor(x, y, method = c("pearson", "kendall", "spearman")) cor.test(x, y, method=c("pearson",  
"kendall", "spearman"))
```

Note: If your data contain missing values, use the following R code to handle missing values by case-wise deletion.

```
cor(x, y, method = "pearson", use = "complete.obs")
```

As an illustration Here, we'll use the built-in R data set mtcars as the first example.

The R code below computes the correlation between mpg and wt variables in mtcars data set:

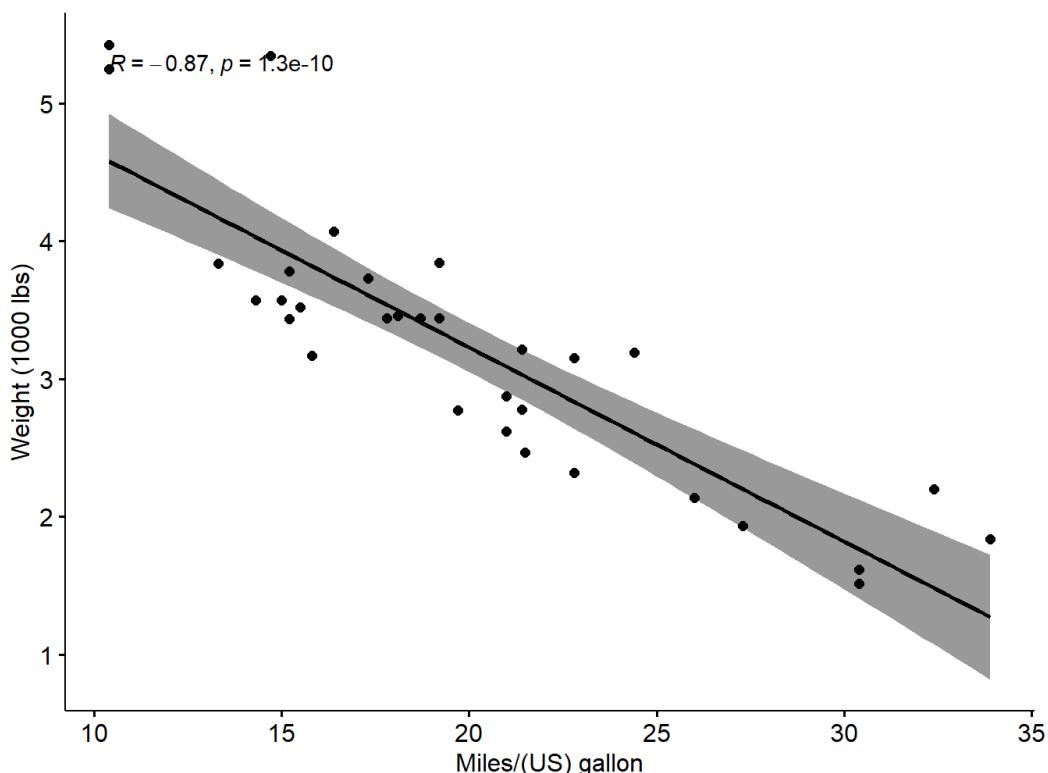
```
my_data <- mtcars  
head(my_data, 6)
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
## Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
## Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
## Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
## Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
## Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
## Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Visualizing the relationship using scatter plot

We can show the correlation in the form of scatter plot as follows:

```
library("ggpubr")
## Loading required package: ggplot2
ggscatter(my_data, x = "mpg", y = "wt",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Miles/(US) gallon", ylab = "Weight (1000 lbs)")
## `geom_smooth()` using formula 'y ~ x'
```



Scatter plot with smooth fit curve

Preliminary checks before finding the Pearson correlation coefficient

Inorder to apply the Pearson correlation test, the data should satisfy some conditions

Preleminary test to check the test assumptions

1. Is the covariation linear? Yes, from the plot above, the relationship is linear. In the situation where the scatter plots show curved patterns, we are dealing with nonlinear association between the two variables.
2. Are the data from each of the 2 variables (x, y) follow a normal distribution?

Pearson correlation test

Correlation test between mpg and wt variables:

```
res <- cor.test(my_data$wt, my_data$mpg, method = "pearson")  
res  
##  
## Pearson's product-moment correlation  
##  
## data: my_data$wt and my_data$mpg  
## t = -9.559, df = 30, p-value = 1.294e-10  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.9338264 -0.7440872  
## sample estimates:  
##       cor  
## -0.8676594
```

Similarly the Kendall rank correlation coefficient or Kendall's tau statistic is used to estimate a rank-based measure of association. This test may be used if the data do not necessarily come from a bivariate normal distribution.

```
res2 <- cor.test(my_data$wt, my_data$mpg, method="kendall")  
## Warning in cor.test.default(my_data$wt, my_data$mpg, method = "kendall"): Cannot  
## compute exact p-value with ties  
res2  
##
```

```

## Kendall's rank correlation tau
##
## data: my_data$wt and my_data$mpg
## z = -5.7981, p-value = 6.706e-09
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##      tau
## -0.7278321

```

Further Spearman's rho statistic is also used to estimate a rank-based measure of association. This test may be used if the data do not come from a bivariate normal distribution.

```

res3 <- cor.test(my_data$wt, my_data$mpg, method = "spearman")
## Warning in cor.test.default(my_data$wt, my_data$mpg, method = "spearman"):
## Cannot compute exact p-value with ties
res3
##
## Spearman's rank correlation rho
##
## data: my_data$wt and my_data$mpg
## S = 10292, p-value = 1.488e-11
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## -0.886422

```

Interpreting correlation test

Result of a correlation test can be interpreted using the correlation coefficient and p-value of the test. Correlation coefficient is comprised between -1 and 1:

1. -1 indicates a strong negative correlation : this means that every time x increases, y decreases
2. 0 means that there is no association between the two variables (x and y)
3. 1 indicates a strong positive correlation : this means that y increases with x

Use of p-value statistics: If the p-value of correlation test is less than 0.05, the null hypothesis of no significant correlation will be accepted at 5% significant level.

Problem 1: From the following data, compute Karl Pearson's coefficient of correlation.

Price(Rupees):	10	20	30	40	50	60	70
Supply(Units):	8	6	14	16	10	20	24

Solution: As the first step read the variables price and supply and use cor.test function on the variable pair. The R code and the result are shown below:

```
price=c(10,20,30,40,50,60,70)
supply=c(8,6,14,16,10,20,24)
resp1=cor.test(price,supply,method='pearson')
resp1
##
## Pearson's product-moment correlation
##
## data: price and supply
## t = 3.6145, df = 5, p-value = 0.01531
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.2707625 0.9774828
## sample estimates:
## cor
## 0.8504201
```

Interpretation: Since the Pearson coefficient is 0.8504201. Also p-value is 0.015308 < 0.05. So the null hypothesis is accepted. So it is statistically reasonable to conclude that there is a significant positive correlation between the price and supply based on the sample.

Problem: From the following data compute correlation between height of father and height of daughters by Karl Pearson's coefficient of correlation.

Height of Father(Cms)	65	66	67	67	68	69	71	73
Height of Daughter(Cms)	67	68	64	69	72	70	69	73

Solution: As the first step read the variables price and supply and use cor.test function on the variable pair. The R code and the result are shown below:

```

height_F=c(65,66,67,67,68,69,71,73)
height_D=c(67,68,64,69,72,70,69,73)
resp2=cor.test(height_F,height_D,method='pearson')
resp2
##
## Pearson's product-moment correlation
##
## data: height_F and height_D
## t = 2.0717, df = 6, p-value = 0.08369
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.1080788 0.9281049
## sample estimates:
##      cor
## 0.6457766

```

Interpretation: Since the Pearson coefficient is 0.6457766. Also p-value is 0.0836865 >0.05. So the null hypothesis is accepted. So it is statistically reasonable to conclude that there is no significant positive correlation between the price and supply based on the sample.

We can show the correlation in the form of scatter plot as follows:

```

library("ggpubr")
data1=data.frame(height_F,height_D)
ggscatter(data1, x = "height_F", y = "height_D",
add = "reg.line", conf.int = TRUE,

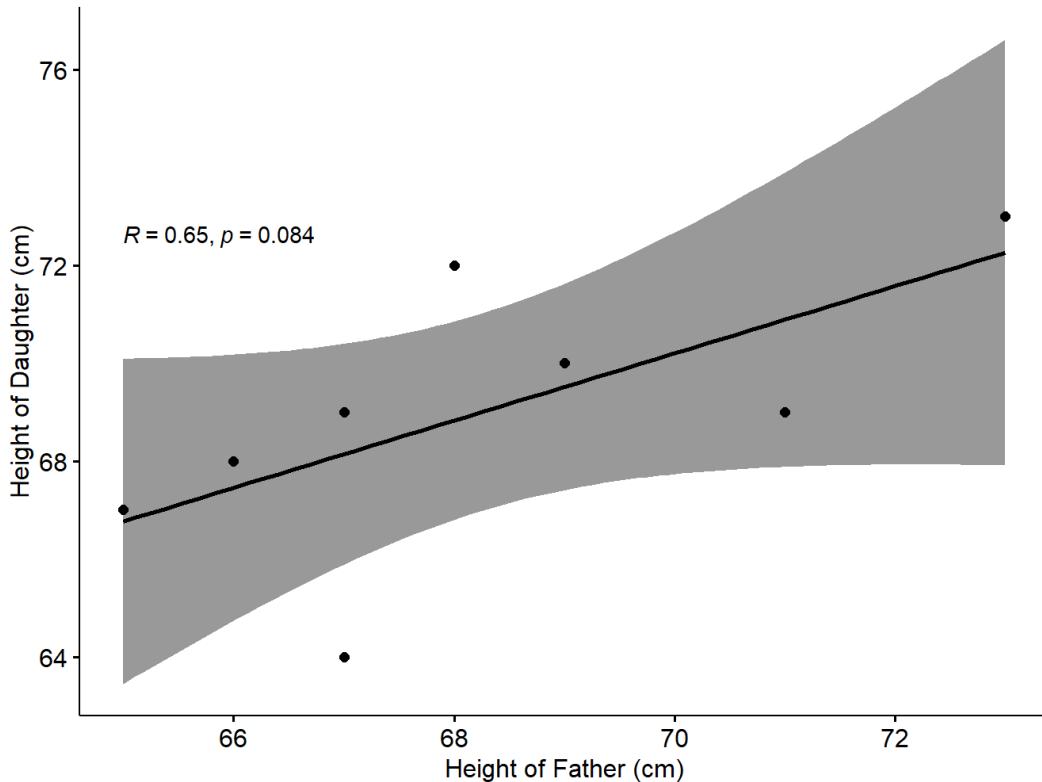
```

```

cor.coef = TRUE, cor.method = "pearson",
xlab = "Height of Father (cm)", ylab = "Height of Daughter (cm)")

## `geom_smooth()` using formula 'y ~ x'

```



Scatter plot with smooth fit curve

Problem: The scores for nine students in history and algebra are as follows:

History:	35	23	47	17	10	43	9	6	28
Algebra:	30	33	45	23	8	49	12	4	31

Compute the Spearman rank correlation.

SOlution: As the first step read the variables price and supply and use cor.test function on the variable pair. The R code and the result are shown below:

```
History=c(35,23,47,17,10,43,9,6,28)
```

```
Algebra=c(30,33,45,23,8,49,12,4,31)
```

```
ress3=cor.test(History,Algebra,method='spearman')
```

```
ress3
```

```

## 
## Spearman's rank correlation rho
##
## data: History and Algebra
## S = 12, p-value = 0.002028
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
## rho
## 0.9

```

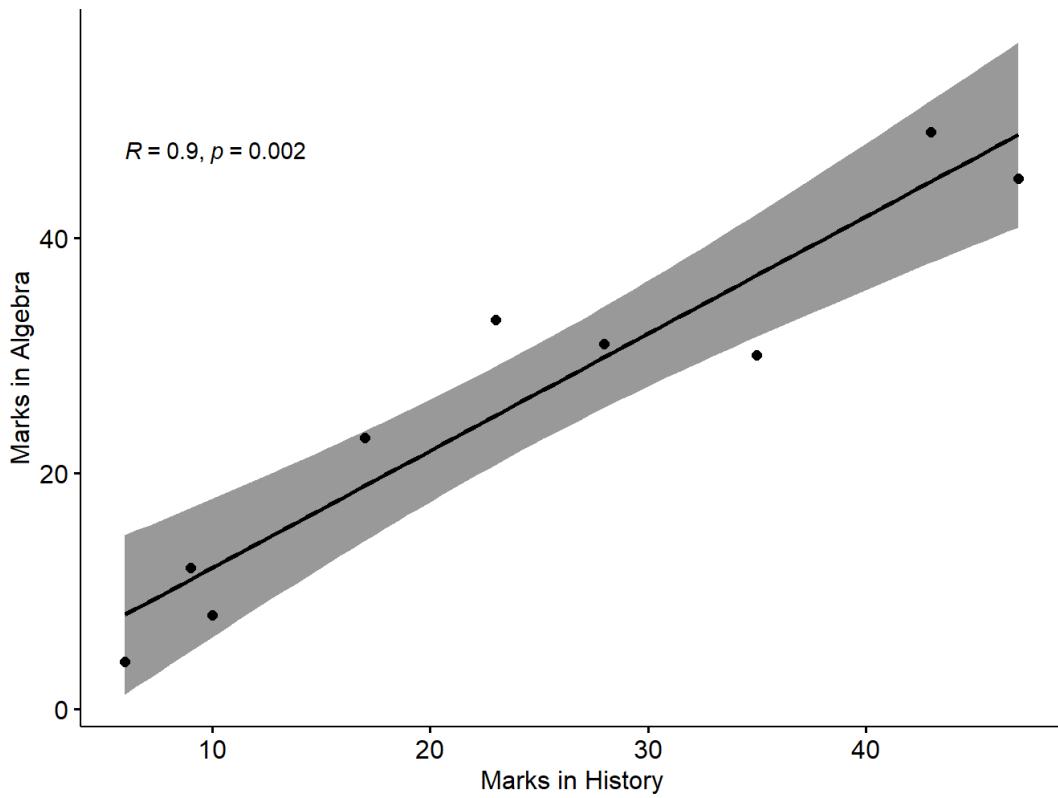
Interpretation: Since the correlation coefficient is 0.9. Also p-value is 0.0020282 < 0.05. So the null hypothesis is rejected. So it is statistically reasonable to conclude that there is a significant positive correlation between the price and supply based on the sample.

We can show the correlation in the form of scatter plot as follows ([Kassambara 2020](#)):

```

library("ggpubr")
data2=data.frame(History,Algebra)
ggsscatter(data2, x = "History", y = "Algebra",
           add = "reg.line", conf.int = TRUE,
           cor.coef = TRUE, cor.method = "spearman",
           xlab = "Marks in History", ylab = "Marks in Algebra")
## `geom_smooth()` using formula 'y ~ x'

```



Scatter plot with smooth fit curve

Correlation Matrix

Previously, we described how to perform correlation test between two variables. In this section, you'll learn how to compute a correlation matrix, which is used to investigate the dependence between multiple variables at the same time. The result is a table containing the correlation coefficients between each variable and the others.

There are different methods for correlation analysis : Pearson parametric correlation test, Spearman and Kendall rank-based correlation analysis.

Compute correlation matrix in R

As you may know, The R function `cor()` can be used to compute a correlation matrix. A simplified format of the function is :

syntax `cor(x, method = c("pearson", "kendall", "spearman"))`

Example: Here, we'll use a data (few numeric columns) derived from the built-in R data set `mtcars` as the first example:

```
# Load data
data("mtcars")
my_data <- mtcars[, c(1,3,4,5,6,7)]
```

```
# print the first 6 rows
head(my_data, 6)

##          mpg   disp    hp drat    wt  qsec
## Mazda RX4     21.0   160 110 3.90 2.620 16.46
## Mazda RX4 Wag 21.0   160 110 3.90 2.875 17.02
## Datsun 710    22.8   108  93 3.85 2.320 18.61
## Hornet 4 Drive 21.4   258 110 3.08 3.215 19.44
## Hornet Sportabout 18.7   360 175 3.15 3.440 17.02
## Valiant       18.1   225 105 2.76 3.460 20.22
```

Compute correlation matrix

The R code to compute the correlation matrix is:

```
rescm <- cor(my_data)
round(rescm, 2)
```

```
##          mpg   disp    hp drat    wt  qsec
## mpg     1.00 -0.85 -0.78  0.68 -0.87  0.42
## disp   -0.85  1.00  0.79 -0.71  0.89 -0.43
## hp     -0.78  0.79  1.00 -0.45  0.66 -0.71
## drat    0.68 -0.71 -0.45  1.00 -0.71  0.09
## wt     -0.87  0.89  0.66 -0.71  1.00 -0.17
## qsec    0.42 -0.43 -0.71  0.09 -0.17  1.00
```

Unfortunately, the function `cor()` returns only the correlation coefficients between variables. In the next section, we will use Hmisc R package to calculate the correlation p-values. The function `rcorr()` [in Hmisc package] can be used to compute the significance levels for pearson and spearman correlations. It returns both the correlation coefficients and the p-value of the

correlation for all possible pairs of columns in the data table.

Syntax rcorr(x, type = c("pearson","spearman"))

The following R code illustrate the use of rcorr() function on my_data.

```
library("Hmisc")

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

## 

## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':

## 

## format.pval, units

resH <- rcorr(as.matrix(my_data))

resH
```

```
##      mpg   disp     hp   drat     wt   qsec
## mpg    1.00 -0.85 -0.78  0.68 -0.87  0.42
## disp  -0.85  1.00  0.79 -0.71  0.89 -0.43
## hp    -0.78  0.79  1.00 -0.45  0.66 -0.71
## drat   0.68 -0.71 -0.45  1.00 -0.71  0.09
## wt    -0.87  0.89  0.66 -0.71  1.00 -0.17
## qsec   0.42 -0.43 -0.71  0.09 -0.17  1.00
## 
## n= 32
```

```

## 
## 
## P

##      mpg      disp      hp      drat      wt      qsec
## mpg          0.0000  0.0000  0.0000  0.0000  0.0171
## disp  0.0000          0.0000  0.0000  0.0000  0.0131
## hp    0.0000  0.0000          0.0100  0.0000  0.0000
## drat 0.0000  0.0000  0.0100          0.0000  0.6196
## wt    0.0000  0.0000  0.0000  0.0000          0.3389
## qsec  0.0171  0.0131  0.0000  0.6196  0.3389

```

Below is the code to compute the correlation

1. Loading the dataset

```
> data1<-swiss
```

```
> head(data1, 4)
```

	Fertility	Agriculture	Examination	Education	Catholic	Infant.Mortality
Courtelary	80.2	17.0	15	12	9.96	22.2
Delemont	83.1	45.1	6	9	84.84	22.2
Franches-Mnt	92.5	39.7	5	5	93.40	20.2
Moutier	85.8	36.5	12	7	33.77	20.3

2.

Creating a scatter plot using ggplot2 library

```
> library(ggplot2)
> ggplot(data1, aes(x = Fertility, y = Infant.Mortality)) + geom_point() +
+ geom_smooth(method = "lm", se = TRUE, color = 'black')
```

3. Testing the assumptions (Linearity and Normalcy)

Linearity[#]: Visible from the plot itself (True, the relationship is linear)

Normality^{\$}: Using Shapiro test (This is a test of normality, here we are checking whether the variables are normally distributed or not)

```
> shapiro.test(data1$Fertility)
```

Shapiro-Wilk normality test

data: data1\$Fertility

W = 0.97307, p-value = 0.3449

```
> shapiro.test(data1$Infant.Mortality)
```

Shapiro-Wilk normality test

data: data1\$Infant.Mortality

W = 0.97762, p-value = 0.4978

p-value is greater than 0.05, so we can assume the normality

4. Correlation Coefficient

```
> cor(data1$Fertility,data1$Infant.Mortality)
```

[1] 0.416556

5. Checking for the significance

```
> Tes<- cor.test(swiss$Fertility,swiss$Infant.Mortality,method = "pearson")
```

>

> Tes

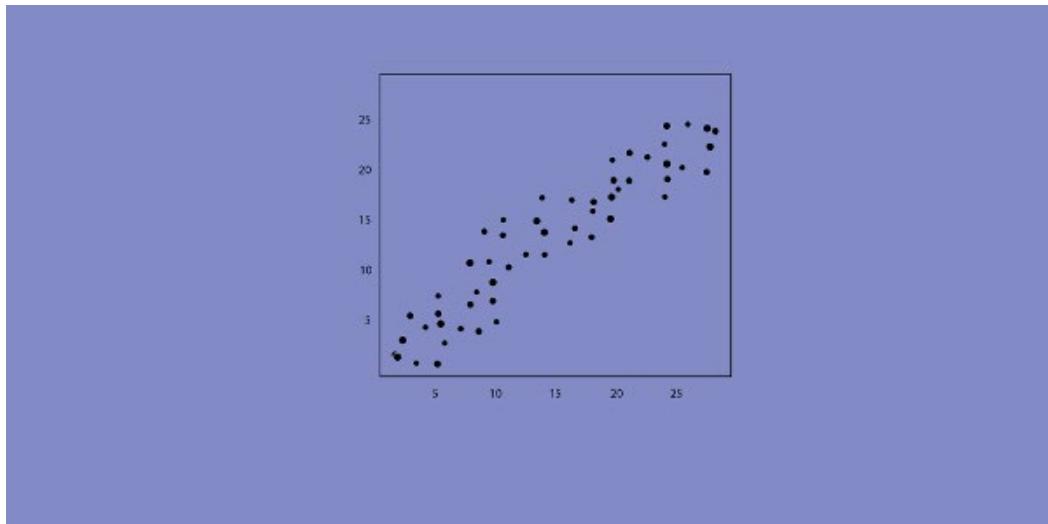
```
Pearson's product-moment correlation

data: swiss$Fertility and swiss$Infant.Mortality
t = 3.0737, df = 45, p-value = 0.003585
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
0.1469699 0.6285366
sample estimates:
cor
0.416556
```

Since the p-value is less than 0.05 (here it is 0.003585, we can conclude that Fertility and Infant Mortality are significantly correlated with a value of 0.41 and a p-value of 0.003585.

Regression analysis

Can you predict a company's revenue by analyzing the budget it allocates to its marketing team? Yes, you can. Do you know how to predict using linear regression in R? Not yet? Well, let me show you how. In this article, we will discuss one of the simplest machine-learning techniques, linear regression in r. Regression in r is almost a 200-year-old tool that is still effective in data science. It is one of the oldest statistical tools used in machine learning predictive analysis.



What Is Linear Regression in R?

Simple linear regression in R is a powerful technique to uncover associations between two variables. In this method, the dependent variable (response variable) reacts to changes in the independent variable (predictor variable). It's important to note that we are not just calculating the dependency of the dependent variable on the independent variable, but also exploring the nuanced association. This makes linear regression in R a valuable tool for understanding and interpreting relationships in your data.

For example, a firm is investing some amount of money in the marketing of a product, and it has also collected sales data throughout the years. Now, by analyzing the correlation between the marketing budget and the sales data, we can predict next year's sales if the company allocates a certain amount of money to the marketing department. The above idea of prediction sounds magical, but it's pure statistics. The linear regression algorithm is basically fitting a straight line to our dataset using the least squares method so that we can predict future events. One limitation of linear regression in r programming is that it is sensitive to outliers. The best-fit line would be of the form:

$$Y = B_0 + B_1 X$$

Where, Y – Dependent variable

X – Independent variable

B_0 and B_1 – Regression parameter

Practical Application of Linear Regression in R

Let's try to understand the practical application of linear Regression in R with another example.

Let's say we have a dataset of the blood pressure and age of a certain group of people. With the help of this data, we can train a simple linear regression model in R, which will be able to predict blood pressure at ages that are not present in our dataset.

You can download the Dataset from [below](#):

Equation of the regression line in our dataset.

$$BP = 98.7147 + 0.9709 \text{ Age}$$

where y is BP

Now let's see how to do this

Step 1: Import the Dataset

Import the dataset of Age vs. Blood Pressure, a CSV file using function `read.csv()` in R, and store this dataset into a data frame `bp`.

```
bp <- read.csv("bp.csv")
```

Step 2: Create the Data Frame for Predicting Values

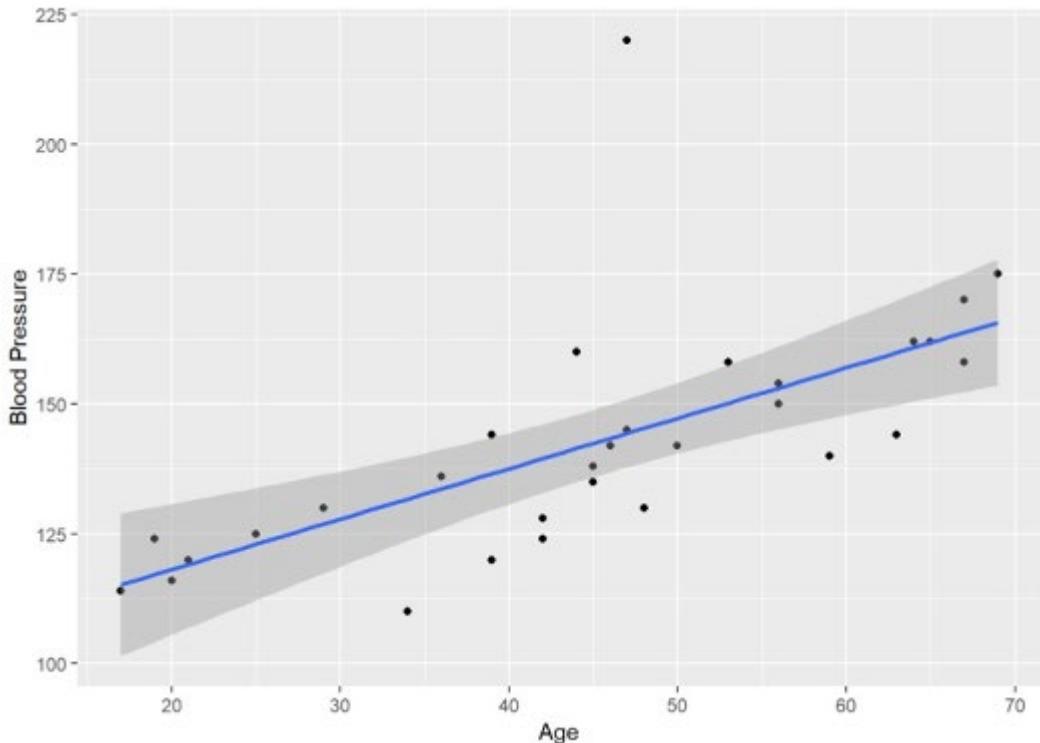
Create a data frame that will store Age 53. This data frame will help us predict blood pressure at Age 53 after creating a linear regression model.

```
p <- as.data.frame(53)
```

```
colnames(p) <- "Age"
```

Step 3: Create a Scatter Plot using the `ggplot2` Library

Taking the help of the `ggplot2` library in R, we can see that there is a correlation between Blood Pressure and Age, as we can see that the increase in Age is followed by an increase in blood pressure.



We can also use the plot function In R for scatterplot and abline function to plot straight lines.

It is quite evident from the graph that the distribution on the plot is scattered in a manner that we can fit a straight line through the data points.

Step 4: Calculate the Correlation Between Age and Blood Pressure

We can also verify our above analysis that there is a correlation between Blood Pressure and Age by taking the help of the cor() function in R, which is used to calculate the correlation between two variables.

```
cor(bp$BP,bp$Age)
```

```
[1] 0.6575673
```

Step 5: Create a Linear Regression Model

Now, leveraging the lm() function in R, let's build a linear model. Using 'BP ~ Age' as the formula, with Age as the independent variable and Blood Pressure as the dependent variable, we apply this to our dataset named 'bp'. The model seamlessly fits the data, showcasing the power of R linear Regression.

```
model <- lm(BP ~ Age, data = bp)
```

Summary of Our Linear Regression Model

```
summary(model)
```

Output:

```

## Call:
## lm(formula = BP ~ Age, data = bp)

## Residuals:
## Min 1Q Median 3Q Max
## -21.724 -6.994 -0.520 2.931 75.654

## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 98.7147 10.0005 9.871 1.28e-10 ***
## Age 0.9709 0.2102 4.618 7.87e-05 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 17.31 on 28 degrees of freedom
## Multiple R-squared: 0.4324, Adjusted R-squared: 0.4121
## F-statistic: 21.33 on 1 and 28 DF, p-value: 7.867e-05

```

Interpretation of the Model

```

## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 98.7147 10.0005 9.871 1.28e-10 ***
## Age 0.9709 0.2102 4.618 7.87e-05 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

$B_0 = 98.7147$ (Y- intercept)

$B_1 = 0.9709$ (Age coefficient)

$BP = 98.7147 + 0.9709 \text{ Age}$

It means a change in one unit in Age will bring 0.9709 units to change in Blood Pressure.

Standard Error

The standard error is variability to expect in coefficient, which captures sampling variability, so the variation in intercept can be up to 10.0005, and the variation in Age will be 0.2102, not more.

T value

The T value is the coefficient divided by the standard error. It is basically how big the estimate is relative to the error. The bigger the coefficient relative to standard error, the bigger the t score. The t score comes with a p-value because a distribution p-value is how statistically significant the variable is to the model for a confidence level of 95%. We will compare this value with alpha which will be 0.05, so in our case, the p-values of both intercept and Age are less than alpha ($\alpha = 0.05$). This implies that both are statistically significant to our model.

We can calculate the confidence interval using the `confint(model, level=.95)` method.

```
## Residual standard error: 17.31 on 28 degrees of freedom  
## Multiple R-squared: 0.4324, Adjusted R-squared: 0.4121  
## F-statistic: 21.33 on 1 and 28 DF, p-value: 7.867e-05
```

Residual Standard Error

Residual standard error or the standard error of the model is basically the average error for the model, which is 17.31 in our case, and it means that our model can be off by an average of 17.31 while predicting the blood pressure. The lesser the error, the better the model while predicting.

Multiple R-squared

Multiple R-squared is the ratio of $(1 - (\text{sum of squared error} / \text{sum of squared total}))$

Adjusted R-squared

Suppose we add variables, no matter if it's significant in prediction or not. In that case, the value of the R-squared will increase, which is the reason adjusted R-squared is used because if the variable added isn't significant for the prediction of the model, the value of the adjusted R-squared will reduce. It is one of the most helpful tools to avoid overfitting the model.

F – statistics

F – statistics is the ratio of the mean square of the model and the mean square of the error. In other words, it is the ratio of how well the model is doing and what the error is doing, and the higher the F value is, the better the model is doing compared to the error.

One is the degree of freedom of the numerator of the F – statistic, and 28 is the degree of freedom of the errors.

Step 6: Run a Sample Test

Now, let's try using our model to predict the value of blood pressure for someone at age 53.

$$BP = 98.7147 + 0.9709 \text{ Age}$$

The above formula will be used to calculate blood pressure at the age of 53, and this will be achieved by using the predict function(). First, we will write the name of the linear regression model, separated by a comma, giving the value of the new data set at p as the Age 53 is earlier saved in data frame p.

```
predict(model, newdata = p)
```

Output:

```
## 1  
## 150.1708
```

So, the predicted value of blood pressure is 150.17 at age 53

As we have predicted Blood Pressure with the association of Age, now there can be more than one independent variable involved, which shows a correlation with a dependent variable. This is called Multiple Regression.

Multiple Linear Regression Model

Multi-Linear regression analysis is a statistical technique to find the association of multiple independent variables with the dependent variable. For example, revenue generated by a company is dependent on various factors, including market size, price, promotion, competitor's price, etc. basically Multiple linear regression in R establishes a linear relationship between a dependent variable and multiple independent variables.

The equation of Multiple Linear Regression is as follows:

$$Y = B_0 + B_1X_1 + B_2X_2 + \dots + B_nX_k + E$$

Where

Y – Dependent variable

X – Independent variable

B₀, B₁, B₃, . – Multiple linear regression coefficients

E- Error

Taking another example of the Wine dataset and with the help of AGST, HarvestRain, we are going to predict the price of wine. Here AGST and HarvestRain are fitted values.

Here's how we can build a multiple R linear regression model.

Step 1: Import the Dataset

Using the function `read.csv()`, import both data sets `wine.csv` and `wine_test.csv`, into the data

frame wine and wine_test, respectively.

```
wine <- read.csv("wine.csv")
wine_test <- read.csv("wine_test.csv")
```

You can download the dataset below.

Step 2: Find the Correlation Between Different Variables

Using the cor() function and round() function, we can round off the correlation between all variables of the dataset wine to two decimal places.

```
round(cor(wine),2)
```

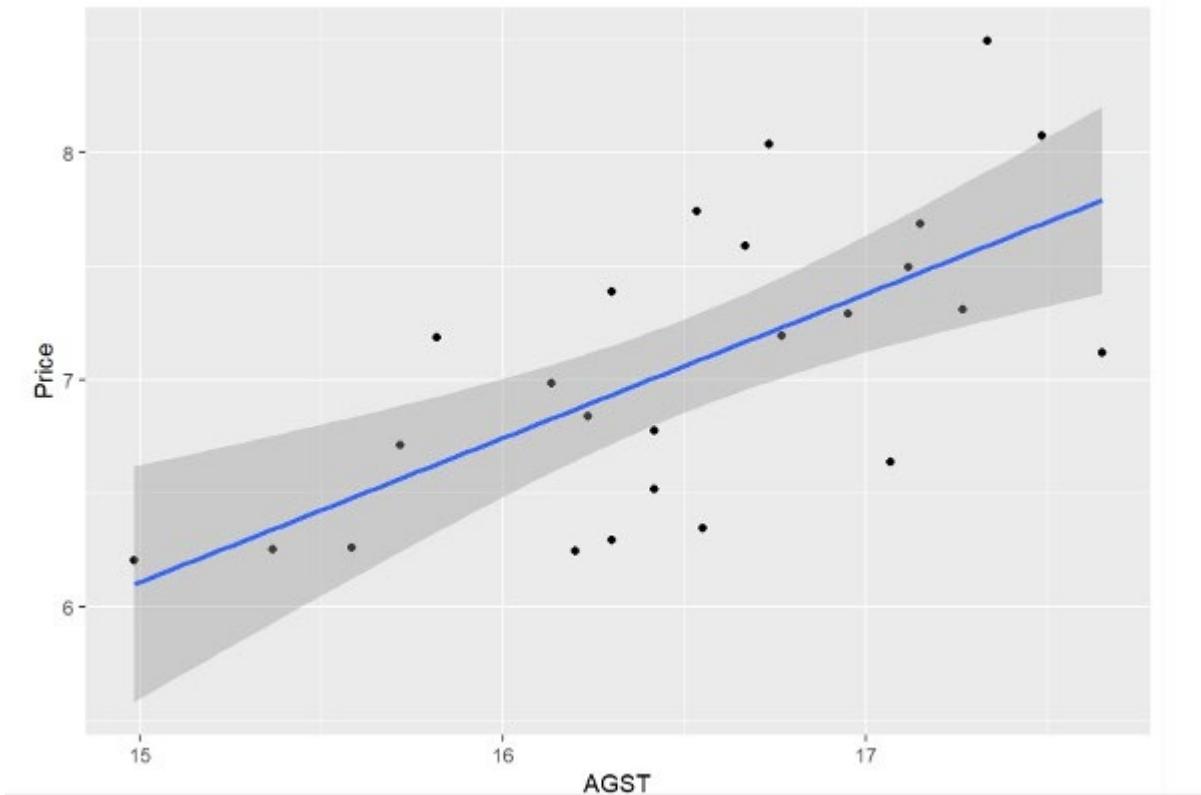
Output:

	Year	Price	WinterRain	AGST	HarvestRain	Age	FrancePop
## Year	1.00	-0.45	0.02	-0.25	0.03	-1.00	0.99
## Price	-0.45	1.00	0.14	0.66	-0.56	0.45	-0.47
## WinterRain	0.02	0.14	1.00	-0.32	-0.28	-0.02	0.00
## AGST	-0.25	0.66	-0.32	1.00	-0.06	0.25	-0.26
## HarvestRain	0.03	-0.56	-0.28	-0.06	1.00	-0.03	0.04
## Age	-1.00	0.45	-0.02	0.25	-0.03	1.00	-0.99
## FrancePop	0.99	-0.47	0.00	-0.26	0.04	-0.99	1.00

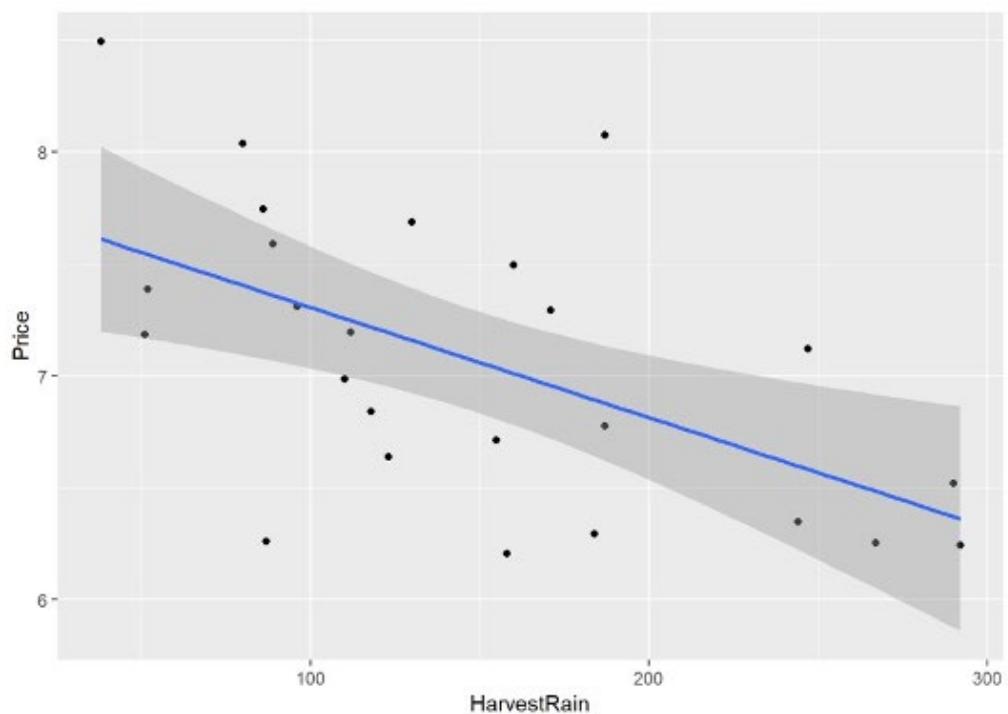
Step 3: Create Scatter Plots Using ggplot2 Library

Create a scatter plot using the library ggplot2 in R. This clearly shows that AGST and the Price of the wine are highly correlated. Similarly, the scatter plot between HarvestRain and the Price of wine also shows their correlation.

```
ggplot(wine,aes(x = AGST, y = Price)) + geom_point() +geom_smooth(method = "lm")
```



```
ggplot(wine,aes(x = HarvestRain, y = Price)) + geom_point() +geom_smooth(method = "lm")
```



Step 4: Create a Multilinear Regression Model

```
model1 <- lm(Price ~ AGST + HarvestRain,data = wine)
```

```
summary(model1)
```

Output:

```

## Call:
## lm(formula = Price ~ AGST + HarvestRain, data = wine)

## Residuals:
## Min 1Q Median 3Q Max
## -0.88321 -0.19600 0.06178 0.15379 0.59722

## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.20265 1.85443 -1.188 0.247585
## AGST 0.60262 0.11128 5.415 1.94e-05 ***
## HarvestRain -0.00457 0.00101 -4.525 0.000167 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 0.3674 on 22 degrees of freedom
## Multiple R-squared: 0.7074, Adjusted R-squared: 0.6808
## F-statistic: 26.59 on 2 and 22 DF, p-value: 1.347e-06

```

Interpretation of the Model

```

## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.20265 1.85443 -1.188 0.247585
## AGST 0.60262 0.11128 5.415 1.94e-05 ***
## HarvestRain -0.00457 0.00101 -4.525 0.000167 ***
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

B₀ = 98.7147 (Y- intercept)

B₁ = 0.9709 (Age coefficient)

Price = -2.20265 + 0.60262 AGST - 0.00457 HarvestRain

It means that a change in one unit in AGST will bring 0.60262 units to change in Price, and one

unit change in HarvestRain will bring 0.00457 units to change in Price.

Standard Error

The standard error is variability to expect in coefficient, which captures sampling variability, so the variation in intercept can be up to 1.85443, the variation in AGST will be 0.11128, and the variation in HarvestRain is 0.00101, not more.

In this case, the p-value of intercept, AGST, and HarvestRain are less than alpha (alpha = 0.05), which implies that all are statistically significant to our model.

```
## Residual standard error: 0.3674 on 22 degrees of freedom  
## Multiple R-squared: 0.7074, Adjusted R-squared: 0.6808  
## F-statistic: 26.59 on 2 and 22 DF, p-value: 1.347e-06
```

Residual Standard Error

The residual standard error or the standard error of the model is 0.3674 in our case, which means that our model can be off by an average of 0.3674 while predicting the Price of wines. The lesser the error, the better the model while predicting. We have also looked at the residuals, which need to follow a normal distribution.

Multiple R-squared is the ratio of (1-(sum of squared error/sum of squared total))

Two is the degree of freedom of the numerator of the F – statistic, and 22 is the degree of freedom of the errors.

Step 5: Predict the Values for Our Test Set

```
prediction <- predict(model1, newdata = wine_test)
```

Predicted values with the test data set

```
wine_test
```

```
## Year Price WinterRain AGST HarvestRain Age FrancePop  
## 1 1979 6.9541 717 16.1667 122 4 54835.83  
## 2 1980 6.4979 578 16.0000 74 3 55110.24
```

```
prediction
```

```
## 1 2  
## 6.982126 7.101033
```

Advantages of Simple Linear Regression in R

1. Simple to understand: Linear regression in R programming is easy to grasp, even for beginners, making it accessible to anyone interested in data analysis.

2. Easy interpretation: It's straightforward to interpret the relationship between two variables because linear regression provides coefficients that tell you how the dependent variable changes with a one-unit change in the independent variable.
3. Fast computations: Linear regression in R is computationally efficient, so you can analyze large datasets quickly, which is great for projects with tight deadlines.
4. Visualizations: You can easily create scatterplots and regression lines in R to visualize the relationship between variables, helping you understand your data better.

Disadvantages of Simple Linear Regression in R

- Assumes linearity: Linear regression assumes that the relationship between variables is linear. If this isn't true, your model may not be accurate.
- Assumes equal variance: Linear regression also assumes that the variability of the data (residuals) is the same across all values of the independent variable. If this assumption is violated, your predictions might not be reliable.
- Sensitive to outliers: Linear regression in R Programming is sensitive to outliers, which are data points that don't fit the pattern of the rest of the data. Outliers can skew your results and make your model less accurate.
- Limited to two variables: Linear regression can only analyze the relationship between two variables. If your data is more complex and involves multiple predictors, you might need to use more advanced techniques.
- Can't predict outside the data range: Linear regression shouldn't be used to make predictions outside the range of your data because it might not give you accurate results.

ANOVA (ANalysis Of VAriance) is a statistical test to determine whether two or more population means are different. In other words, it is used to compare two or more groups to see if they are significantly different.

In practice, however, the:

- Student t-test is used to compare 2 groups;
- ANOVA generalizes the t-test beyond 2 groups, so it is used to compare 3 or more groups.

Note that there are several versions of the ANOVA (e.g., one-way ANOVA, two-way ANOVA, mixed ANOVA, repeated measures ANOVA, etc.). In this article, we present the simplest form only—the one-way ANOVA¹—and we refer to it as ANOVA in the remaining of the article.

Although ANOVA is used to make inference about means of different groups, the method is called “analysis of variance”. It is called like this because it compares the “between” variance (the variance between the different groups) and the variance “within” (the variance within each group). If the between variance is significantly larger than the within variance, the group means are declared to be different. Otherwise, we cannot conclude one way or the other. The two variances are compared to each other by taking the ratio (variancebetweenvariancewithinvariancebetweenvariancewithin) and then by comparing this ratio to a threshold from the Fisher probability distribution (a threshold based on a specific significance level, usually 5%).

This is enough theory regarding the ANOVA method for now. In the remaining of this article, we discuss about it from a more practical point of view, and in particular we will cover the following points:

- the aim of the ANOVA, when it should be used and the null/alternative hypothesis
- the underlying assumptions of the ANOVA and how to check them
- how to perform the ANOVA in R
- how to interpret results of the ANOVA
- understand the notion of post-hoc test and interpret the results
- how to visualize results of ANOVA and post-hoc tests

Data

Data for the present article is the penguins dataset (an alternative to the well-known iris dataset), accessible via the {palmerpenguins} package:

```
# install.packages("palmerpenguins")
library(palmerpenguins)
```

The dataset contains data for 344 penguins of 3 different species (Adelie, Chinstrap and Gentoo). The dataset contains 8 variables, but we focus only on the flipper length and the species for this article, so we keep only those 2 variables:

```
library(tidyverse)
```

```
dat <- penguins %>%
  select(species, flipper_length_mm)
```

(If you are unfamiliar with the pipe operator (%>%), you can also select variables with `penguins[, c("species", "flipper_length_mm")]`. Learn more ways to select variables in the article about data manipulation.)

Below some basic descriptive statistics and a plot (made with the {ggplot2} package) of our dataset before we proceed to the goal of the ANOVA:

```
summary(dat)

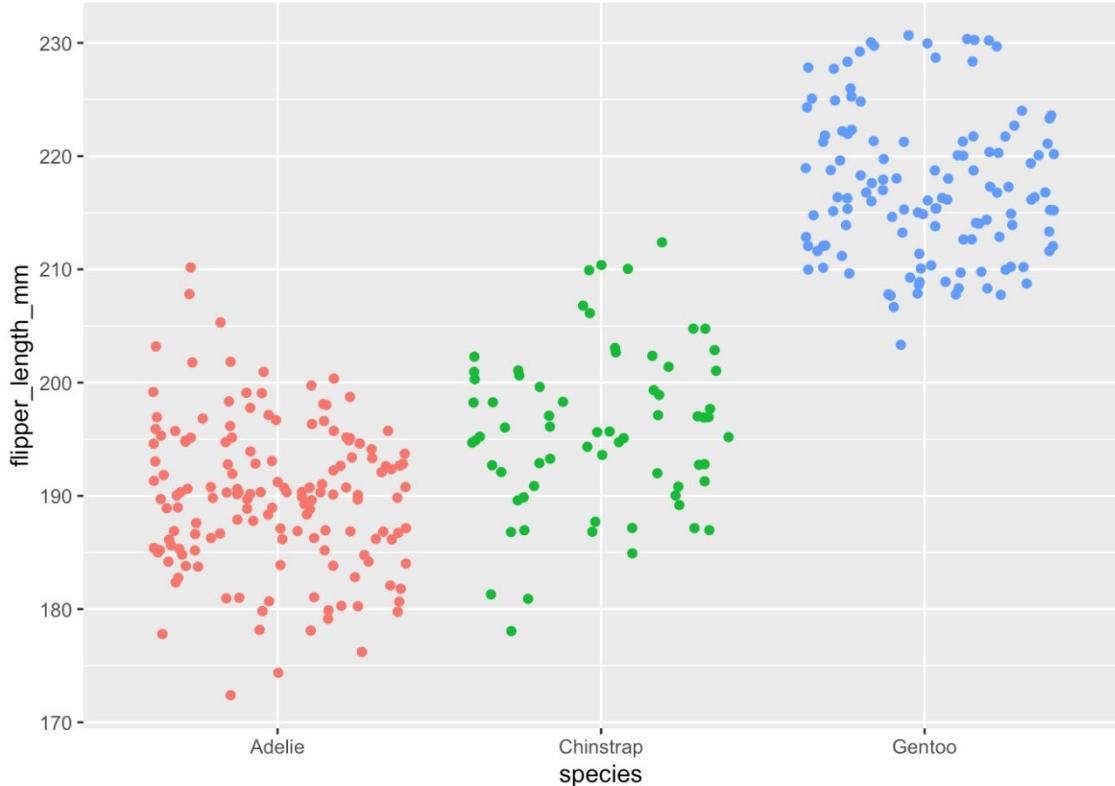
##   species   flipper_length_mm
## Adelie :152  Min. :172.0
## Chinstrap:68  1st Qu.:190.0
## Gentoo :124  Median:197.0
##                  Mean :200.9
##                  3rd Qu.:213.0
##                  Max. :231.0
##                  NA's :2
```

Flipper length varies from 172 to 231 mm, with a mean of 200.9 mm. There are respectively 152, 68 and 124 penguins of the species Adelie, Chinstrap and Gentoo.

```
library(ggplot2)
```

```
ggplot(dat) +
  aes(x = species, y = flipper_length_mm, color = species) +
```

```
geom_jitter() +  
theme(legend.position = "none")
```



Here, the factor is the species variable which contains 3 modalities or groups (Adelie, Chinstrap and Gentoo).

Aim and hypotheses of ANOVA

As mentioned in the introduction, the ANOVA is used to compare groups (in practice, 3 or more groups). More generally, it is used to:

- study whether measurements are similar across different modalities (also called levels or treatments in the context of ANOVA) of a categorical variable
- compare the impact of the different levels of a categorical variable on a quantitative variable
- explain a quantitative variable based on a qualitative variable

In this context and as an example, we are going to use an ANOVA to help us answer the question: “Is the length of the flippers different between the 3 species of penguins?”.

The null and alternative hypothesis of an ANOVA are:

- $H_0: \mu_{\text{Adelie}} = \mu_{\text{Chinstrap}} = \mu_{\text{Gentoo}}$ ($\Rightarrow \Rightarrow$ the 3 species are equal in terms of flipper length)
- $H_1: \text{at least one mean is different}$ ($\Rightarrow \Rightarrow$ at least one species is different from the other 2 species in terms of flipper length)

Be careful that the alternative hypothesis is *not* that all means are different. The opposite of all means being equal ($H_0 H_0$) is that *at least* one mean is different from the others ($H_1 H_1$).

In this sense, if the null hypothesis is rejected, it means that at least one species is different from the other 2, but not necessarily that all 3 species are different from each other. It could be that flipper length for the species Gentoo is different than for the species Chinstrap and Adelie, but flipper length is similar between Chinstrap and Adelie. Other types of test (known as post-hoc tests and covered in this [section](#)) must be performed to test whether all 3 species differ.

Underlying assumptions of ANOVA

As for many [statistical tests](#), there are some assumptions that need to be met in order to be able to interpret the results. When one or several assumptions are not met, although it is technically possible to perform these tests, it would be incorrect to interpret the results and trust the conclusions.

Below are the assumptions of the ANOVA, how to test them and which other tests exist if an assumption is not met:

- Variable type: ANOVA requires a mix of one continuous quantitative dependent variable (which corresponds to the measurements to which the question relates) and one qualitative independent variable (with at least 2 levels which will determine the groups to compare).
- Independence: the data, collected from a representative and randomly selected portion of the total population, should be independent between groups and within each group. The assumption of independence is most often verified based on the design of the experiment and on the good control of experimental conditions rather than via a formal test. If you are still unsure about independence based on the experiment design, ask yourself if one observation is related to another (if one observation has an impact on another) within each group or between the groups themselves. If not, it is most likely that you have independent samples. If observations between samples (forming the different groups to be compared) are dependent (for example, if three measurements have been collected on the same individuals as it is often the case in medical studies when measuring a metric (i) before, (ii) during and (iii) after a treatment), the repeated measures ANOVA should be preferred in order to take into account the dependency between the samples.
- Normality:
 - In case of small samples, residuals² should follow approximately a normal distribution. The normality assumption can be tested visually thanks to

a histogram and a QQ-plot, and/or formally via a normality test such as the Shapiro-Wilk or Kolmogorov-Smirnov test. If, even after a transformation of your data (e.g., logarithmic transformation, square root, Box-Cox, etc.), the residuals still do not follow approximately a normal distribution, the Kruskal-Wallis test can be applied (`kruskal.test(variable ~ group, data = dat in R)`). This non-parametric test, robust to non normal distributions, has the same goal than the ANOVA—compare 3 or more groups—but it uses sample medians instead of sample means to compare groups.

- In case of large samples, normality is not required (this is a common misconception!). By the central limit theorem, sample means of large samples are often well-approximated by a normal distribution even if the data are not normally distributed (Stevens 2013).³ It is therefore not required to test the normality assumption when the number of observations in each group/sample is large (usually $n \geq 30$).
- Equality of variances: the variances of the different groups should be equal in the populations (an assumption called homogeneity of the variances, or even sometimes referred as homoscedasticity, as opposed to heteroscedasticity if variances are different across groups). This assumption can be tested graphically (by comparing the dispersion in a boxplot or dotplot for instance), or more formally via the Levene's test (`leveneTest(variable ~ group)` from the `{car}` package) or Bartlett's test, among others. If the hypothesis of equal variances is rejected, another version of the ANOVA can be used: the Welch ANOVA (`oneway.test(variable ~ group, var.equal = FALSE)`). Note that the Welch ANOVA does not require homogeneity of the variances, but the distributions should still follow approximately a normal distribution. Note that the Kruskal-Wallis test does not require the assumptions of normality nor homoscedasticity of the variances.⁴
- Outliers: An outlier is a value or an observation that is distant from the other observations. There should be no significant outliers in the different groups, or the conclusions of your ANOVA may be flawed. There are several methods to detect outliers in your data but in order to deal with them, it is your choice to either:
 - use the non-parametric version (i.e., the Kruskal-Wallis test)
 - transform your data (logarithmic or Box-Cox transformation, among others)
 - or remove them (be careful)

Choosing the appropriate test depending on whether assumptions are met may be confusing so here is a brief summary:

1. Check that your observations are independent.
2. Sample sizes:
 - o In case of small samples, test the normality of residuals:
 - If normality is assumed, test the homogeneity of the variances:
 - If variances are equal, use ANOVA.
 - If variances are not equal, use the Welch ANOVA.
 - If normality is not assumed, use the Kruskal-Wallis test.
 - o In case of large samples normality is assumed, so test the homogeneity of the variances:
 - If variances are equal, use ANOVA.
 - If variances are not equal, use the Welch ANOVA.

Now that we have seen the underlying assumptions of the ANOVA, we review them specifically for our dataset before applying the appropriate version of the test.

Variable type

The dependent variable flipper_length_mm is a quantitative variable and the independent variable species is a qualitative one (with 3 levels corresponding to the 3 species). So we have a mix of the two types of variable and this assumption is met.

Independence

Independence of the observations is assumed as data have been collected from a randomly selected portion of the population and measurements within and between the 3 samples are not related.

The independence assumption is most often verified based on the design of the experiment and on the good control of experimental conditions, as it is the case here.

If you really want to test it more formally, you can, however, test it via a statistical test—the Durbin-Watson test (in R: `durbinWatsonTest(res_lm)` where `res_lm` is a linear model). The null hypothesis of this test specifies an autocorrelation coefficient = 0, while the alternative hypothesis specifies an autocorrelation coefficient $\neq 0$.

Normality

Since the smallest sample size per group (i.e., per species) is 68, we have large samples. Therefore, we do not need to check normality.

Usually, we would directly test the homogeneity of the variances without testing normality. However, for the sake of illustration, we act as if the sample sizes were small in order to illustrate what would need to be done in that case.

Remember that normality of residuals can be tested visually via a histogram and a QQ-plot, and/or formally via a normality test (Shapiro-Wilk test for instance).

Before checking the normality assumption, we first need to compute the ANOVA (more on that in this section). We then save the results in res_aov :

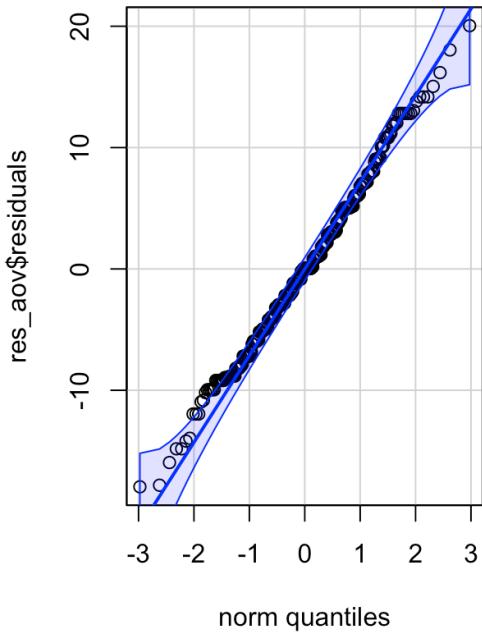
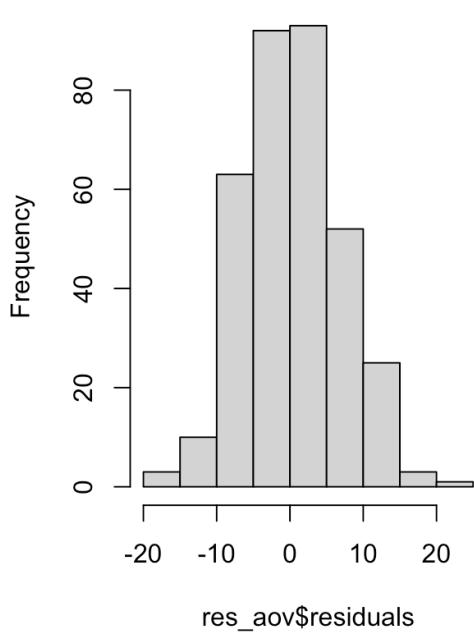
```
res_aov <- aov(flipper_length_mm ~ species,  
                 data = dat  
)
```

We can now check normality visually:

```
par(mfrow = c(1, 2)) # combine plots
```

```
# histogram  
hist(res_aov$residuals)  
  
# QQ-plot  
library(car)  
qqPlot(res_aov$residuals,  
       id = FALSE # id = FALSE to remove point identification  
)
```

Histogram of res_aov\$residuals



From the histogram and QQ-plot above, we can already see that the normality assumption seems to be met. Indeed, the histogram roughly form a bell curve, indicating that the residuals follow a normal distribution. Furthermore, points in the QQ-plots roughly follow the straight line and most of them are within the confidence bands, also indicating that residuals follow approximately a normal distribution.

Some researchers stop here and assume that normality is met, while others also test the assumption via a formal normality test. It is your choice to test it (i) only visually, (ii) only via a normality test, or (iii) both visually AND via a normality test. Bear in mind, however, the two following points:

1. ANOVA is quite robust to small deviations from normality. This means that it is not an issue (from the perspective of the interpretation of the ANOVA results) if a small number of points deviates slightly from the normality,
2. normality tests are sometimes quite conservative, meaning that the null hypothesis of normality may be rejected due to a limited deviation from normality. This is especially the case with large samples as power of the test increases with the sample size.

In practice, I tend to prefer the (i) visual approach only, but again, this is a matter of personal choice and also depends on the context of the analysis.

Still for the sake of illustration, we also now test the normality assumption via a normality test. You can use the Shapiro-Wilk test or the Kolmogorov-Smirnov test, among others.

Remember that the null and alternative hypothesis of these tests are:

- H0: data come from a normal distribution
- H1: data do *not* come from a normal distribution

In R, we can test normality of the residuals with the Shapiro-Wilk test thanks to the shapiro.test() function:

```
shapiro.test(res_aov$residuals)

##
##      Shapiro-Wilk normality test
##
## data: res_aov$residuals
## W = 0.99452, p-value = 0.2609
```

P-value of the Shapiro-Wilk test on the residuals is larger than the usual significance level of $\alpha=5\%$, so we do not reject the hypothesis that residuals follow a normal distribution (*p*-value = 0.261).

This result is in line with the visual approach. In our case, the normality assumption is thus met both visually and formally.

*Side note: Remind that the *p*-value is the probability of having observations as extreme as the ones we have observed in the sample(s) given that the null hypothesis is true. If the *p*-value $<\alpha<\alpha$ (indicating that it is not likely to observe the data we have in the sample given that the null hypothesis is true), the null hypothesis is rejected, otherwise the null hypothesis is not rejected. See more about *p*-value and significance level if you are unfamiliar with those important statistical concepts.*

Remember that if the normality assumption was not reached, some transformation(s) would need to be applied on the raw data in the hope that residuals would better fit a normal distribution, or you would need to use the non-parametric version of the ANOVA—the Kruskal-Wallis test.

As pointed out by a reader (see comments at the very end of the article), the normality assumption can also be tested on the “raw” data (i.e., the observations) instead of the residuals. However, if you test the normality assumption on the raw data, it must be tested for *each group separately* as the ANOVA requires normality in *each group*.

Testing normality on all residuals or on the observations per group is equivalent, and will give similar results. Indeed, saying “The distribution of Y within each group is normally distributed” is the same as saying “The residuals are normally distributed”.

Remember that residuals are the distance between the actual value of Y and the mean value of Y for a specific value of X, so the grouping variable is induced in the computation of the residuals.

So in summary, in ANOVA you actually have two options for testing normality:

1. Checking normality separately for each group on the “raw” data (Y values)
2. Checking normality on all residuals (but not per group)

In practice, you will see that it is often easier to just use the residuals and check them all together, especially if you have many groups or few observations per group.

If you are still not convinced: remember that an ANOVA is a special case of a linear model. Suppose your independent variable is a continuous variable (instead of a categorical variable), the only option you have left is to check normality on the residuals, which is precisely what is done for testing normality in linear regression models.

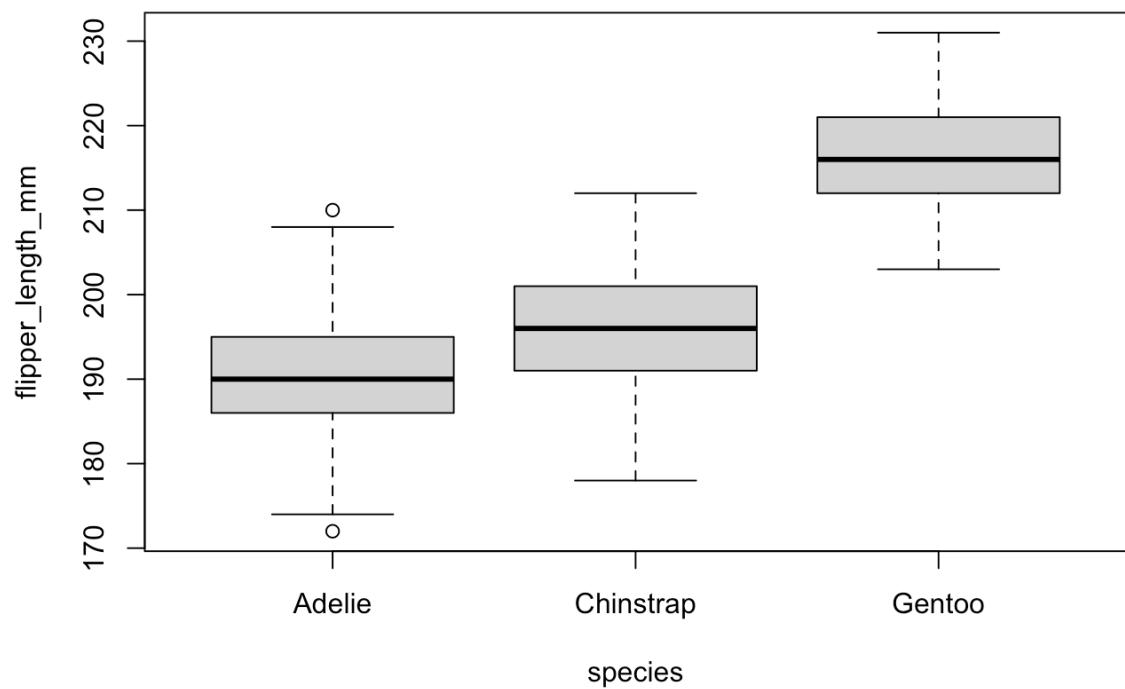
Equality of variances - homogeneity

Assuming residuals follow a normal distribution, it is now time to check whether the variances are equal across species or not. The result will have an impact on whether we use the ANOVA or the Welch ANOVA.

This can again be verified visually—via a boxplot or dotplot—or more formally via a statistical test (Levene’s test, among others).

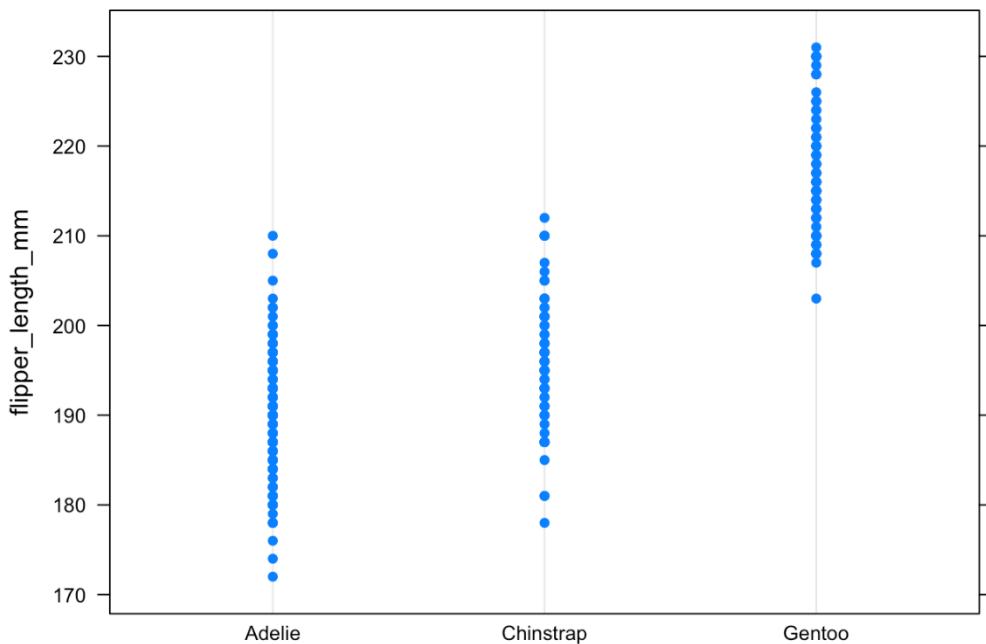
Visually, we have:

```
# Boxplot  
boxplot(flipper_length_mm ~ species,  
       data = dat  
)
```



```
# Dotplot
library("lattice")

dotplot(flipper_length_mm ~ species,
        data = dat
      )
```



Both the boxplot and the dotplot show a similar variance for the different species. In the boxplot, this can be seen by the fact that the boxes and the whiskers have a comparable size for all species.

There are a couple of outliers as shown by the points outside the whiskers, but this does not change the fact that the dispersion is more or less the same between the different species.

In the dotplot, this can be seen by the fact that points for all 3 species have more or less the same range, a sign of the dispersion and thus the variance being similar.

Like the normality assumption, if you feel that the visual approach is not sufficient, you can formally test for equality of the variances with a Levene's or Bartlett's test. Notice that the Levene's test is less sensitive to departures from normal distribution than the Bartlett's test.

The null and alternative hypothesis for both tests are:

- H₀: variances are equal
- H₁: at least one variance is different

In R, the Levene's test can be performed thanks to the `leveneTest()` function from the `{car}` package:

```
# Levene's test
```

```
library(car)
```

```
leveneTest(flipper_length_mm ~ species,
```

```
data = dat
)
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group  2  0.3306 0.7188
##      339
```

The p -value being larger than the significance level of 0.05, we do not reject the null hypothesis, so we cannot reject the hypothesis that variances are equal between species (p -value = 0.719).

This result is also in line with the visual approach, so the homogeneity of variances is met both visually and formally.

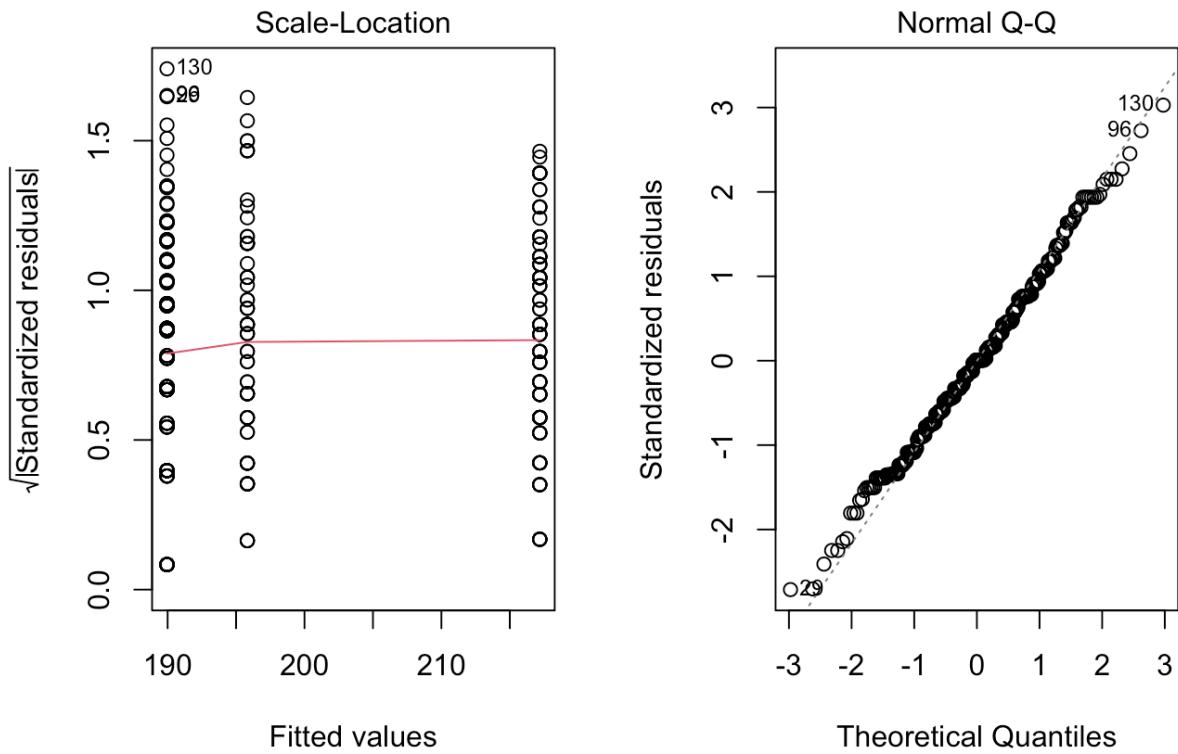
Another method to test normality and homogeneity

For your information, it is also possible to test the homogeneity of the variances and the normality of the residuals visually (and both at the same time) via the plot() function:

```
par(mfrow = c(1, 2)) # combine plots
```

```
# 1. Homogeneity of variances
plot(res_aov, which = 3)
```

```
# 2. Normality
plot(res_aov, which = 2)
```



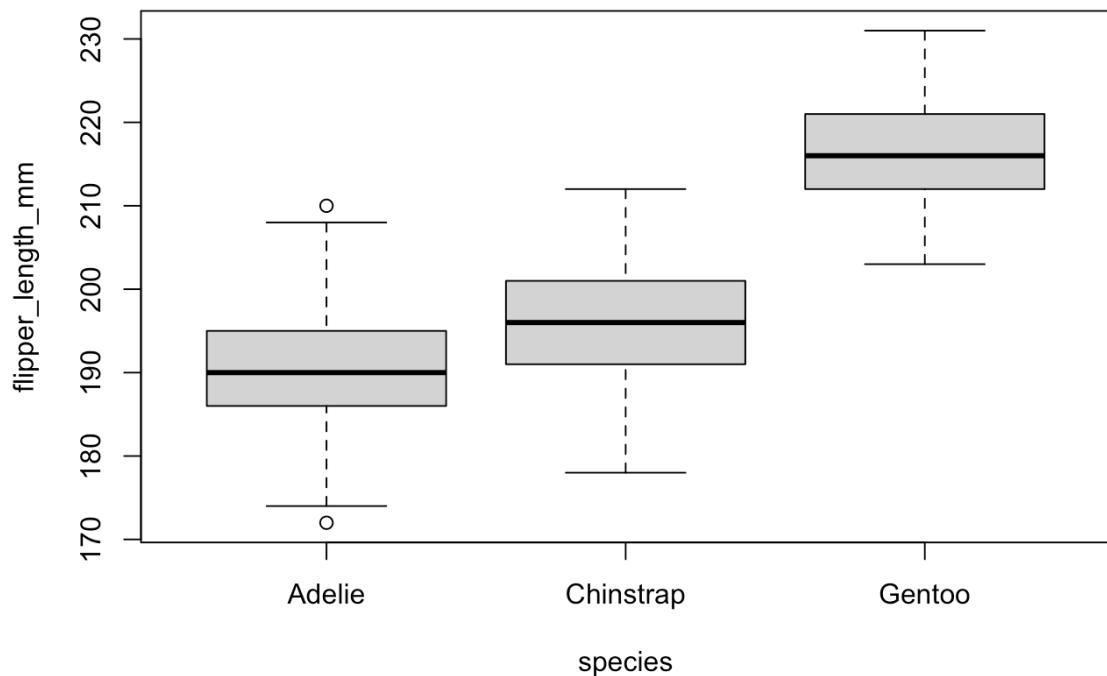
Plot on the left hand side shows that there is no evident relationships between residuals and fitted values (the mean of each group), so homogeneity of variances is assumed. If homogeneity of variances was violated, the red line would not be flat (horizontal).

Plot on the right hand side shows that residuals follow approximately a normal distribution, so normality is assumed. If normality was violated, points would consistently deviate from the dashed line.

Outliers

There are several techniques to detect outliers. In this article, we focus on the most simple one (yet very efficient)—the visual approach via a boxplot:

```
boxplot(flipper_length_mm ~ species,
        data = dat
    )
```



There is one outlier in the group Adelie, as defined by the interquartile range criterion. This point is, however, not seen as a significant outlier so we can assume that the assumption of no significant outliers is met.

ANOVA

We showed that all assumptions of the ANOVA are met.

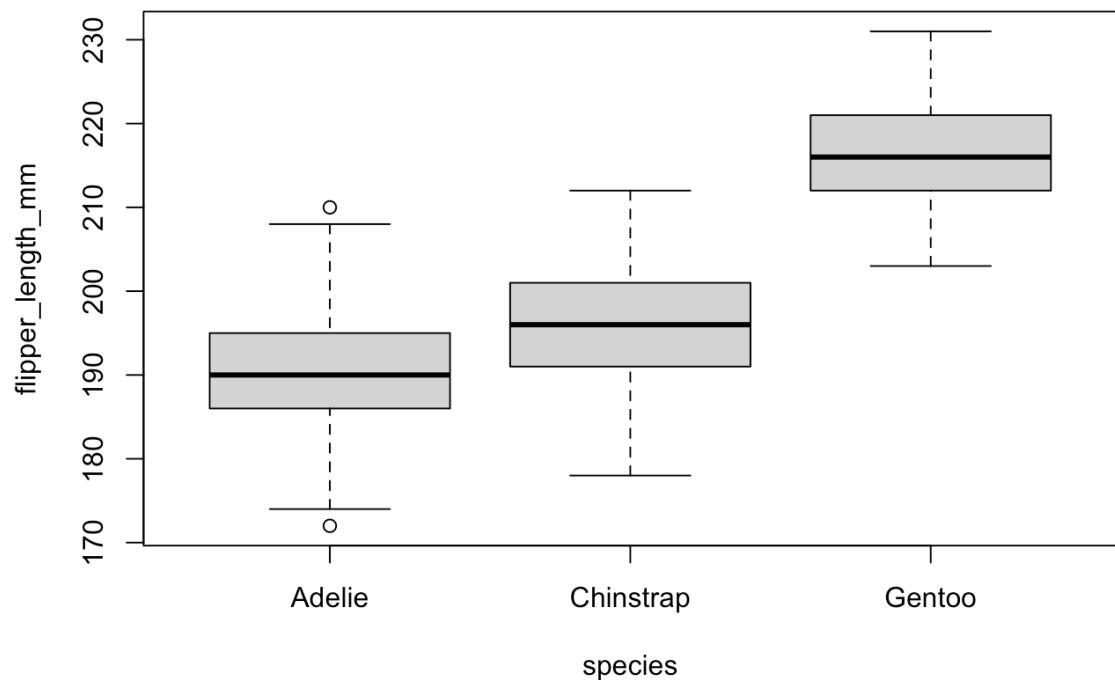
We can thus proceed to the implementation of the ANOVA in R, but first, let's do some preliminary analyses to better understand the research question.

Preliminary analyses

A good practice before actually performing the ANOVA in R is to visualize the data in relation to the research question. The best way to do so is to draw and compare boxplots of the quantitative variable `flipper_length_mm` for each species.

This can be done with the `boxplot()` function in base R (same code than the visual check of equal variances):

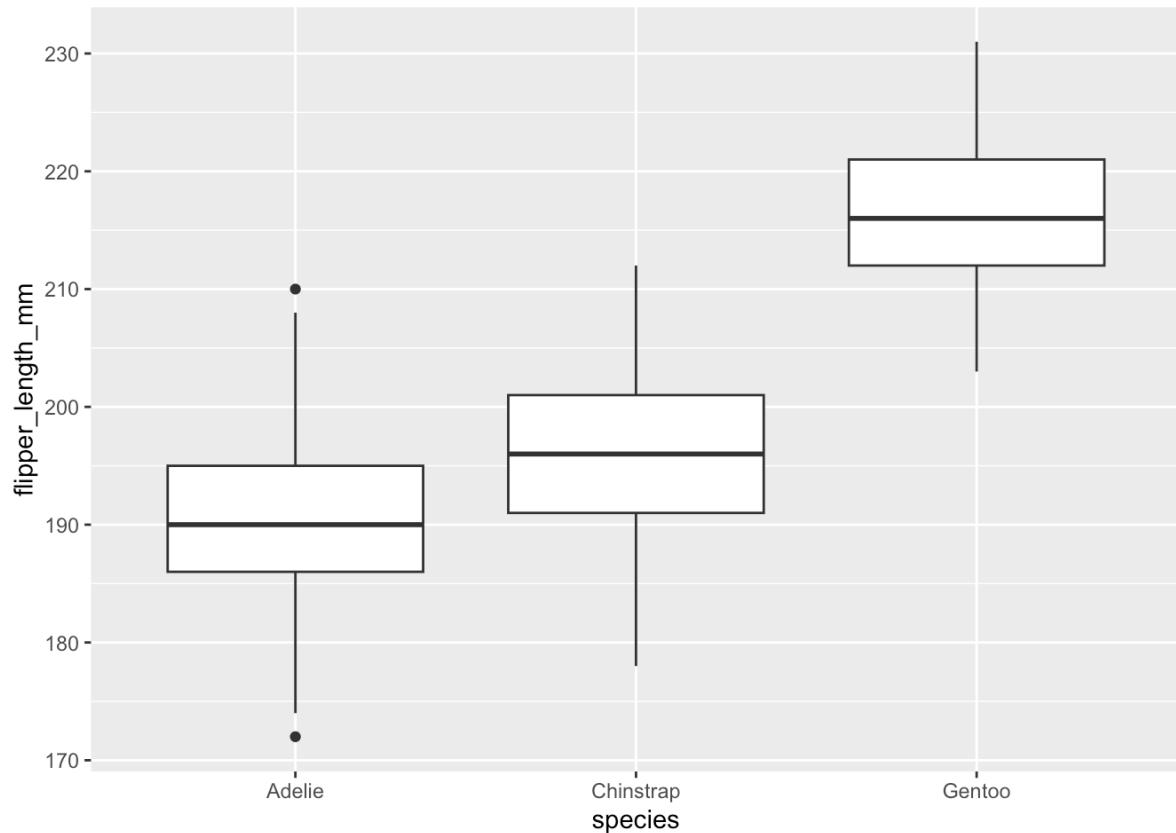
```
boxplot(flipper_length_mm ~ species,
       data = dat
      )
```



Or with the {ggplot2} package:

```
library(ggplot2)
```

```
ggplot(dat) +  
  aes(x = species, y = flipper_length_mm) +  
  geom_boxplot()
```



The boxplots above show that, at least for our sample, penguins of the species Gentoo seem to have the biggest flipper, and Adelie species the smallest flipper.

Besides a boxplot for each species, it is also a good practice to compute some descriptive statistics such as the mean and standard deviation by species.

This can be done, for instance, with the `aggregate()` function:

```
aggregate(flipper_length_mm ~ species,
  data = dat,
  function(x) round(c(mean = mean(x), sd = sd(x)), 2)
)

##   species flipper_length_mm.mean flipper_length_mm.sd
## 1  Adelie      189.95            6.54
## 2 Chinstrap     195.82            7.13
## 3  Gentoo      217.19            6.48
```

or with the `summarise()` and `group_by()` functions from the `{dplyr}` package:

```
library(dplyr)
```

```

group_by(dat, species) %>%
  summarise(
    mean = mean(flipper_length_mm, na.rm = TRUE),
    sd = sd(flipper_length_mm, na.rm = TRUE)
  )
## # A tibble: 3 × 3
##   species   mean    sd
##   <fct>     <dbl> <dbl>
## 1 Adelie    190.  6.54
## 2 Chinstrap 196.  7.13
## 3 Gentoo    217.  6.48

```

Mean is also the lowest for Adelie and highest for Gentoo. Boxplots and descriptive statistics are, however, not enough to conclude that flippers are significantly different in the 3 populations of penguins.

ANOVA in R

As you guessed by now, only the ANOVA can help us to make inference about the population given the sample at hand, and help us to answer the initial research question “Is the length of the flippers different between the 3 species of penguins?”.

ANOVA in R can be done in several ways, of which two are presented below:

1. With the `oneway.test()` function:

```

# 1st method:
oneway.test(flipper_length_mm ~ species,
            data = dat,
            var.equal = TRUE # assuming equal variances
)
## 
## One-way analysis of means
## 
## data: flipper_length_mm and species
## F = 594.8, num df = 2, denom df = 339, p-value < 2.2e-16

```

2. With the `summary()` and `aov()` functions:

```

# 2nd method:

res_aov <- aov(flipper_length_mm ~ species,
  data = dat
)

summary(res_aov)
##          Df Sum Sq Mean Sq F value Pr(>F)
## species     2  52473   26237   594.8 <2e-16 ***
## Residuals  339 14953      44
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## 2 observations deleted due to missingness

```

As you can see from the two outputs above, the test statistic (F = in the first method and F value in the second one) and the p -value (p-value in the first method and $\text{Pr}(>F)$ in the second one) are exactly the same for both methods, which means that in case of equal variances, results and conclusions will be unchanged.

The advantage of the first method is that it is easy to switch from the ANOVA (used when variances are equal) to the Welch ANOVA (used when variances are unequal). This can be done by replacing `var.equal = TRUE` by `var.equal = FALSE`, as presented below:

```

oneway.test(flipper_length_mm ~ species,
  data = dat,
  var.equal = FALSE # assuming unequal variances
)
## 

##      One-way analysis of means (not assuming equal variances)
## 

## data: flipper_length_mm and species
## F = 614.01, num df = 2.00, denom df = 172.76, p-value < 2.2e-16

```

The advantage of the second method, however, is that:

- the full ANOVA table (with degrees of freedom, mean squares, etc.) is printed, which may be of interest in some (theoretical) cases

- results of the ANOVA (res_aov) can be saved for later use (especially useful for post-hoc tests)

Interpretations of ANOVA results

Given that the p -value is smaller than 0.05, we reject the null hypothesis, so we reject the hypothesis that all means are equal. Therefore, we can conclude that at least one species is different than the others in terms of flippers length (p -value < 2.2e-16).

(*For the sake of illustration*, if the p -value was larger than 0.05: we cannot reject the null hypothesis that all means are equal, so we cannot reject the hypothesis that the 3 considered species of penguins are equal in terms of flippers length.)

A nice and easy way to report results of an ANOVA in R is with the report() function from the {report} package:

```
# install.packages("remotes")
# remotes::install_github("easystats/report") # You only need to do that once
library("report") # Load the package every time you start R
```

```
report(res_aov)

## The ANOVA (formula: flipper_length_mm ~ species) suggests that:

##
## - The main effect of species is statistically significant and large (F(2, 339)
## = 594.80, p < .001; Eta2 = 0.78, 95% CI [0.75, 1.00])
##
## Effect sizes were labelled following Field's (2013) recommendations.
```

As you can see, the function interprets the results for you and indicates a large and significant main effect of the species on the flipper length (p -value < .001).

Note that the report() function can be used for other analyses. See more tips and tricks in R if you find this one useful.

What's next?

If the null hypothesis is not rejected (p -value ≥ 0.05), it means that we do not reject the hypothesis that all groups are equal. The ANOVA more or less stops here.

Other types of analyses can be performed of course, but—given the data at hand—we could not prove that at least one group was different so we usually do not go further with the ANOVA.

On the contrary, if the null hypothesis is rejected (as it is our case since the p -value < 0.05), we proved that at least one group is different. We can decide to stop here if we are only interested to test whether all species are equal in terms of flippers length.

But most of the time, when we showed thanks to an ANOVA that at least one group is different, we are also interested in knowing which one(s) is(are) different. Results of an ANOVA, however, do *NOT* tell us which group(s) is(are) different from the others.

To test this, we need to use other types of test, referred as post-hoc tests (in Latin, “after this”, so after obtaining statistically significant ANOVA results) or multiple pairwise-comparison tests.⁵

This family of statistical tests is the topic of the following sections.

Post-hoc test

Issue of multiple testing

In order to see which group(s) is(are) different from the others, we need to compare groups 2 by 2.

2. In practice, since there are 3 species, we are going to compare species 2 by 2 as follows:

1. Chinstrap versus Adelie
2. Gentoo vs. Adelie
3. Gentoo vs. Chinstrap

In theory, we could compare species thanks to 3 Student's t-tests since we need to compare 2 groups and a t-test is used precisely in that case.

However, if several t-tests are performed, the issue of multiple testing (also referred as multiplicity) arises. In short, when several statistical tests are performed, some will have p -values less than α purely by chance, even if all null hypotheses are in fact true.

To demonstrate the problem, consider our case where we have 3 hypotheses to test and a desired significance level of 0.05.

The probability of observing at least one significant result (at least one p -value < 0.05) just due to chance is:

$$P(\text{at least 1 sig. result}) = 1 - P(\text{no sig. results}) = 1 - (1 - 0.05)^3 = 0.142625 \\ P(\text{at least 1 sig. result}) = 1 - P(\text{no sig. results}) = 1 - (1 - 0.05)^3 = 0.142625$$

So, with as few as 3 tests being considered, we already have a 14.26% chance of observing at least one significant result, even if all of the tests are actually not significant.

And as the number of groups increases, the number of comparisons increases as well, so the probability of having a significant result simply due to chance keeps increasing.

For example, with 10 groups we need to make 45 comparisons and the probability of having at

least one significant result by chance becomes $1 - (1 - 0.05)^{45} = 90\%$. So it is very likely to observe a significant result just by chance when comparing 10 groups, and when we have 14 groups or more we are almost certain (99%) to have a false positive!

Post-hoc tests take into account that multiple tests are done and deal with the problem by adjusting α in some way, so that the probability of observing at least one significant result due to chance remains below our desired significance level.⁶

Post-hoc tests in R and their interpretation

Post-hoc tests are a family of statistical tests so there are several of them. The most common ones are:

- Tukey HSD, used to compare all groups to each other (so all possible comparisons of 2 groups).
- Dunnett, used to make comparisons with a reference group. For example, consider 2 treatment groups and one control group. If you only want to compare the 2 treatment groups with respect to the control group, and you do not want to compare the 2 treatment groups to each other, the Dunnett's test is preferred.
- Bonferroni correction if one has a set of planned comparisons to do.

The Bonferroni correction is simple: you simply divide the desired global α level by the number of comparisons.

In our example, we have 3 comparisons so if we want to keep a global $\alpha=0.05$, we have $\alpha'=0.05/3=0.0167$. We can then simply perform a Student's t-test for each comparison, and compare the obtained p-values with this new α' .

The other two post-hoc tests are presented in the next sections.

Note that variances are assumed to be equal for all three methods (unless you use the Welch's t-test instead of the Student's t-test with the Bonferroni correction). If variances are not equal, you can use the Games-Howell test, among others.

Tukey HSD test

In our case, since there is no “reference” species and we are interested in comparing all species, we are going to use the Tukey HSD test.

In R, the Tukey HSD test is done as follows. This is where the second method to perform the ANOVA comes handy because the results (`res_aov`) are reused for the post-hoc test:

```

library(multcomp)

# Tukey HSD test:

post_test <- glht(res_aov,
  linfct = mcp(species = "Tukey")
)

summary(post_test)

##      Simultaneous Tests for General Linear Hypotheses

## Multiple Comparisons of Means: Tukey Contrasts

## Linear Hypotheses:

##           Estimate Std. Error t value Pr(>|t|)

## Chinstrap - Adelie == 0  5.8699   0.9699  6.052 1.03e-08 ***
## Gentoo - Adelie == 0  27.2333   0.8067 33.760 < 1e-08 ***
## Gentoo - Chinstrap == 0 21.3635   1.0036 21.286 < 1e-08 ***

## ---

## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## (Adjusted p values reported -- single-step method)

```

In the output of the Tukey HSD test, we are interested in the table displayed after Linear Hypotheses:, and more precisely, in the first and last column of the table. The first column shows the comparisons which have been made; the last column ($\text{Pr}(>|t|)$) shows the adjusted⁷ p -values for each comparison (with the null hypothesis being the two groups are equal and the alternative hypothesis being the two groups are different).

It is these adjusted p -values that are used to test whether two groups are significantly different or not, and we can be confident that the entire set of comparisons collectively has an error rate of 0.05.

In our example, we tested:

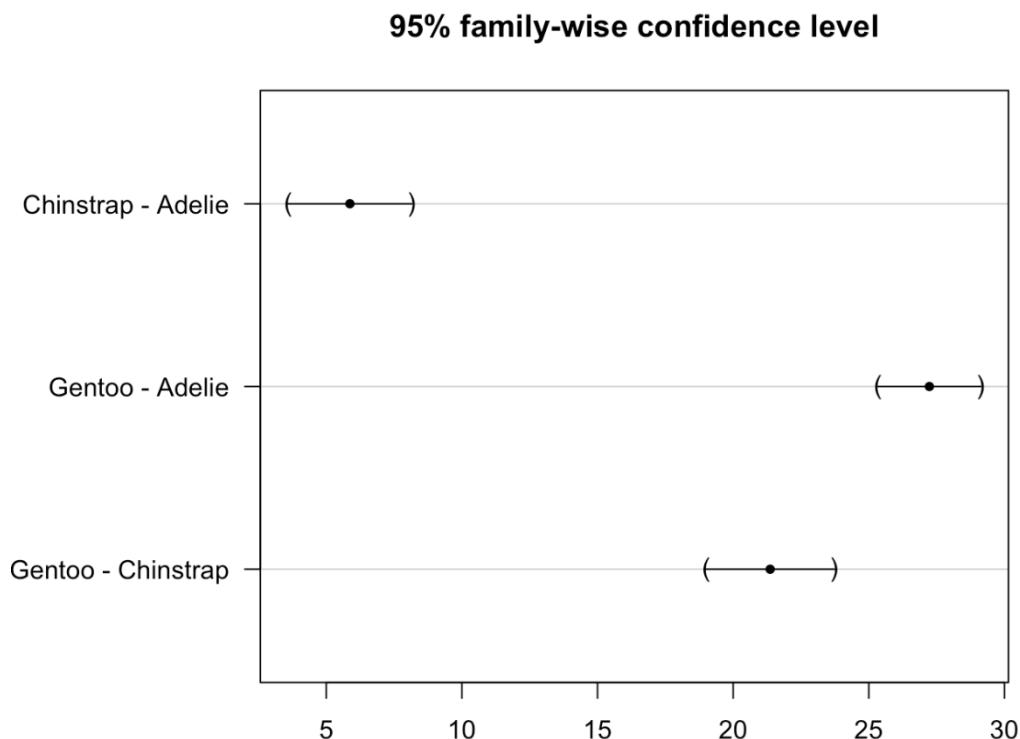
1. Chinstrap versus Adelie (line Chinstrap - Adelie == 0)
2. Gentoo vs. Adelie (line Gentoo - Adelie == 0)
3. Gentoo vs. Chinstrap (line Gentoo - Chinstrap == 0)

All three adjusted p -values are smaller than 0.05, so we reject the null hypothesis for all comparisons, which means that all species are significantly different in terms of flippers length.

The results of the post-hoc test can be visualized with the plot() function:

```
par(mar = c(3, 8, 3, 3))
```

```
plot(post_test)
```



We see that the confidence intervals do not cross the zero line, which indicate that all groups are significantly different.

Note that the Tukey HSD test can also be done in R with the TukeyHSD() function:

```
TukeyHSD(res_aov)

## Tukey multiple comparisons of means

## 95% family-wise confidence level

## 

## Fit: aov(formula = flipper_length_mm ~ species, data = dat)

## 

## $species
```

```

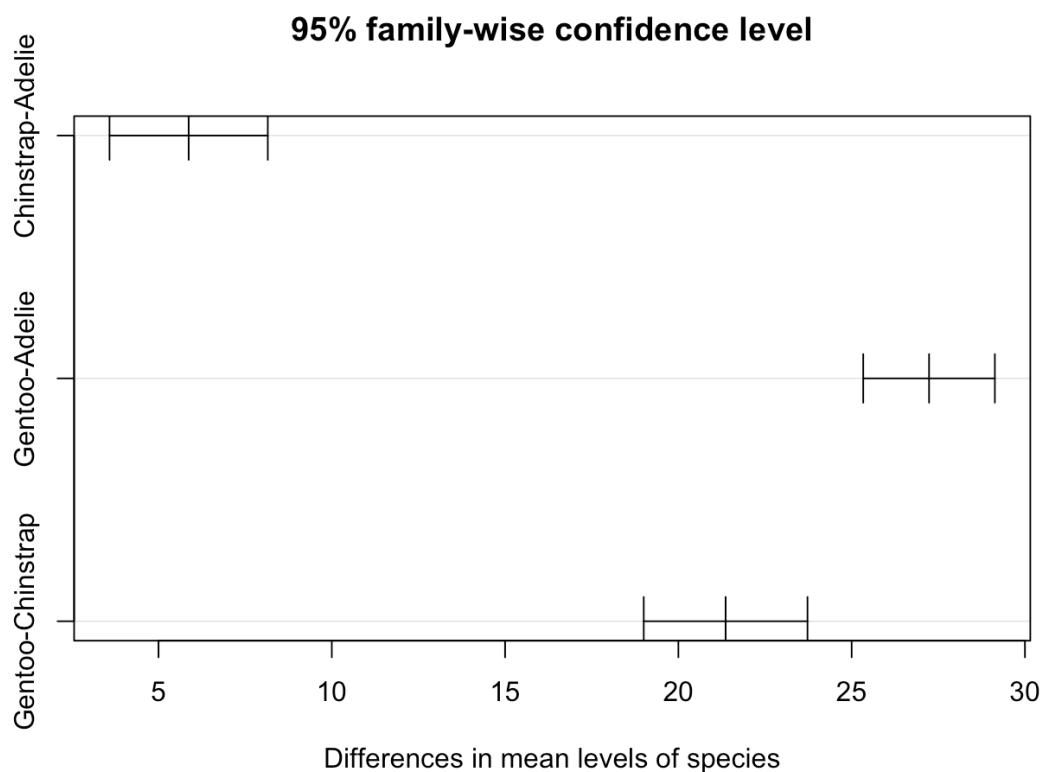
##          diff     lwr      upr p adj
## Chinstrap-Adelie 5.869887 3.586583 8.153191  0
## Gentoo-Adelie 27.233349 25.334376 29.132323  0
## Gentoo-Chinstrap 21.363462 19.000841 23.726084  0

```

With this code, it is the column `p adj` (also the last column) which is of interest. Notice that the conclusions are the same than above: all species are significantly different in terms of flippers length.

The results can also be visualized with the `plot()` function:

```
plot(TukeyHSD(res_aov))
```



Dunnett's test

We have seen in this [section](#) that as the number of groups increases, the number of comparisons also increases. And as the number of comparisons increases, the post-hoc analysis must lower the individual significance level even further, which leads to lower statistical power (so a difference between group means in the population is less likely to be detected).

One method to mitigate this and increase the statistical power is by reducing the number of comparisons. This reduction allows the post-hoc procedure to use a larger individual error rate to achieve the desired global error rate.

While comparing all possible groups with a Tukey HSD test is a common approach, many studies have a control group and several treatment groups. For these studies, you may need to compare the treatment groups only to the control group, which reduces the number of comparisons.

Dunnett's test does precisely this—it only compares a group taken as reference to all other groups, but it does not compare all groups to each others.

So to recap:

- the Tukey HSD test allows to compares all groups but at the cost of less power
- the Dunnett's test allows to only make comparisons with a reference group, but with the benefit of more power

Now, again for the sake of illustration, consider that the species Adelie is the reference species and we are only interested in comparing the reference species against the other 2 species. In that scenario, we would use the Dunnett's test.

In R, the Dunnett's test is done as follows (the only difference with the code for the Tukey HSD test is in the line linfct = mcp(species = "Dunnett")):

```
library(multcomp)
```

```
# Dunnett's test:
```

```
post_test <- glht(res_aov,
  linfct = mcp(species = "Dunnett")
)

summary(post_test)
##
##      Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Dunnett Contrasts
##
##
## Fit: aov(formula = flipper_length_mm ~ species, data = dat)
##
```

```

## Linear Hypotheses:
##                                Estimate Std. Error t value Pr(>|t|)
## Chinstrap - Adelie == 0    5.8699   0.9699  6.052 7.59e-09 ***
## Gentoo - Adelie == 0     27.2333   0.8067 33.760 < 1e-10 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)

```

The interpretation is the same as for the Tukey HSD test's except that in the Dunett's test we only compare:

1. Chinstrap versus Adelie (line Chinstrap - Adelie == 0)
2. Gentoo vs. Adelie (line Gentoo - Adelie == 0)

Both adjusted *p*-values (displayed in the last column) are below 0.05, so we reject the null hypothesis for both comparisons.

This means that both the species Chinstrap and Gentoo are significantly different from the reference species Adelie in terms of flippers length. (Nothing can be said about the comparison between Chinstrap and Gentoo though.)

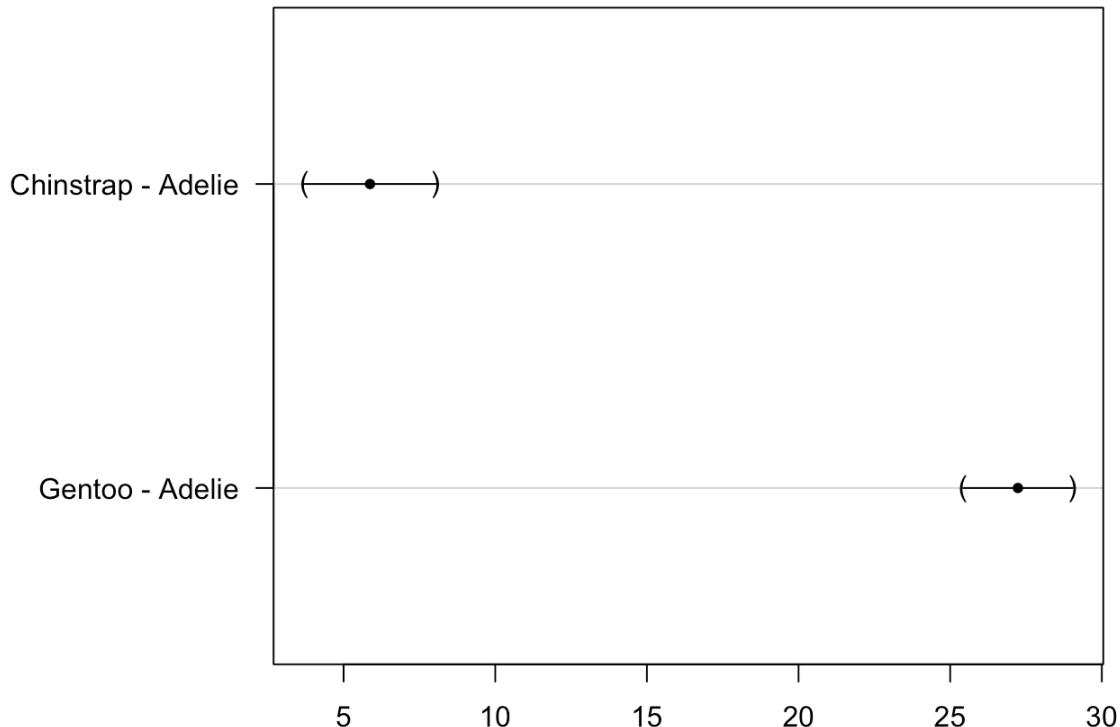
Again, the results of the post-hoc test can be visualized with the plot() function:

```

par(mar = c(3, 8, 3, 3))
plot(post_test)

```

95% family-wise confidence level



We see that the confidence intervals do not cross the zero line, which indicate that both the species Gentoo and Chinstrap are significantly different from the reference species Adelie.

Note that in R, by default, the reference category for a factor variable is the first category in alphabetical order. This is the reason that, by default, the reference species is Adelie.

The reference category can be changed with the `relevel()` function (or with the `{questionr} addin`). Considering that we want Gentoo as the reference category instead of Adelie:

```
# Change reference category:
```

```
dat$species <- relevel(dat$species, ref = "Gentoo")
```

```
# Check that Gentoo is the reference category:
```

```
levels(dat$species)
```

```
## [1] "Gentoo"  "Adelie"   "Chinstrap"
```

Gentoo now being the first category of the three, it is indeed considered as the reference level.

In order to perform the Dunnett's test with the new reference we first need to rerun the ANOVA to take into account the new reference:

```
res_aov2 <- aov(flipper_length_mm ~ species,
```

```

data = dat
)

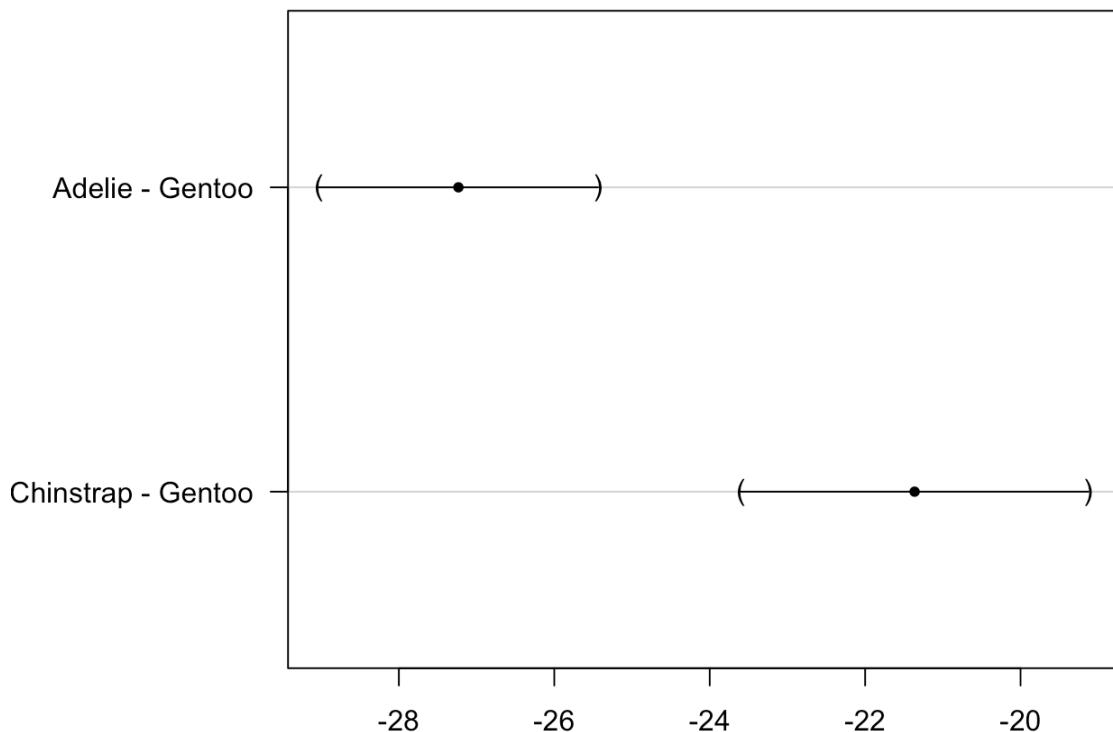
We can then run the Dunnett's test with the new results of the ANOVA:

# Dunnett's test:
post_test <- glht(res_aov2,
linfct = mcp(species = "Dunnett")
)

summary(post_test)
## Simultaneous Tests for General Linear Hypotheses
## Multiple Comparisons of Means: Dunnett Contrasts
## Fit: aov(formula = flipper_length_mm ~ species, data = dat)
## Linear Hypotheses:
##             Estimate Std. Error t value Pr(>|t|)
## Adelie - Gentoo == 0   -27.2333   0.8067 -33.76 <1e-10 ***
## Chinstrap - Gentoo == 0 -21.3635   1.0036 -21.29 <1e-10 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
par(mar = c(3, 8, 3, 3))
plot(post_test)

```

95% family-wise confidence level



From the results above we conclude that Adelie and Chinstrap species are significantly different from Gentoo species in terms of flippers length (adjusted p -values $< 1e-10$).

Note that even if your study does not have a reference group which you can compare to the other groups, it is still often better to do multiple comparisons determined by some research questions than to do all-pairwise tests. By reducing the number of post-hoc comparisons to what is necessary only, and no more, you maximize the statistical power.⁸

Other p -values adjustment methods

For the interested readers, note that you can use other p -values adjustment methods by using the `pairwise.t.test()` function:

```
pairwise.t.test(dat$flipper_length_mm, dat$species,  
  p.adjust.method = "holm"  
)  
##  
##      Pairwise comparisons using t tests with pooled SD  
##  
## data: dat$flipper_length_mm and dat$species  
##
```

```
##      Gentoo Adelie
## Adelie < 2e-16 -
## Chinstrap < 2e-16 3.8e-09
##
## P value adjustment method: holm
```

By default, the Holm method is applied but other methods exist. See `?p.adjust` for all available options.

Visualization of ANOVA and post-hoc tests on the same plot

If you are interested in including results of ANOVA and post-hoc tests on the same plot (directly on the boxplots), here are two pieces of code which may be of interest to you.

The first one is edited by me based on the code found in this [article](#):

```
# Edit from here
```

```
x <- which(names(dat) == "species") # name of grouping variable
y <- which(
  names(dat) == "flipper_length_mm" # names of variables to test
)
method1 <- "anova" # one of "anova" or "kruskal.test"
method2 <- "t.test" # one of "wilcox.test" or "t.test"
my_comparisons <- list(c("Chinstrap", "Adelie"), c("Gentoo", "Adelie"), c("Gentoo", "Chinstrap")) # comparisons for post-hoc tests
# Edit until here
```

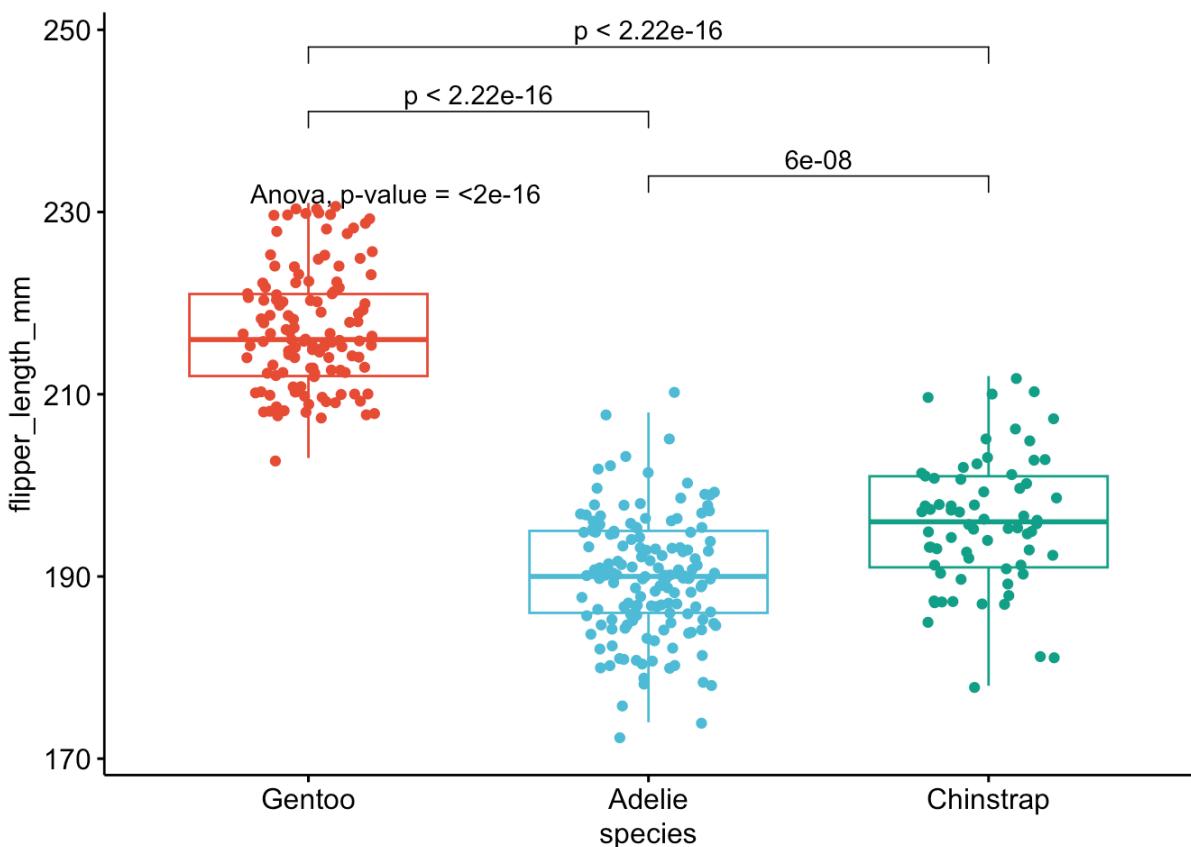
```
# Edit at your own risk
```

```
library(ggpubr)
for (i in y) {
  for (j in x) {
    p <- ggboxplot(dat,
      x = colnames(dat[j]), y = colnames(dat[i]),
      color = colnames(dat[j]),
      legend = "none",
```

```

palette = "npg",
add = "jitter"
)
print(
  p + stat_compare_means(aes(label = paste0(after_stat(method), ", p-value = ",
after_stat(p.format))),
method = method1, label.y = max(dat[, i], na.rm = TRUE)
)
+ stat_compare_means(comparisons = my_comparisons, method = method2, label =
"p.format") # remove if p-value of ANOVA or Kruskal-Wallis test >= alpha
)
}
}

```



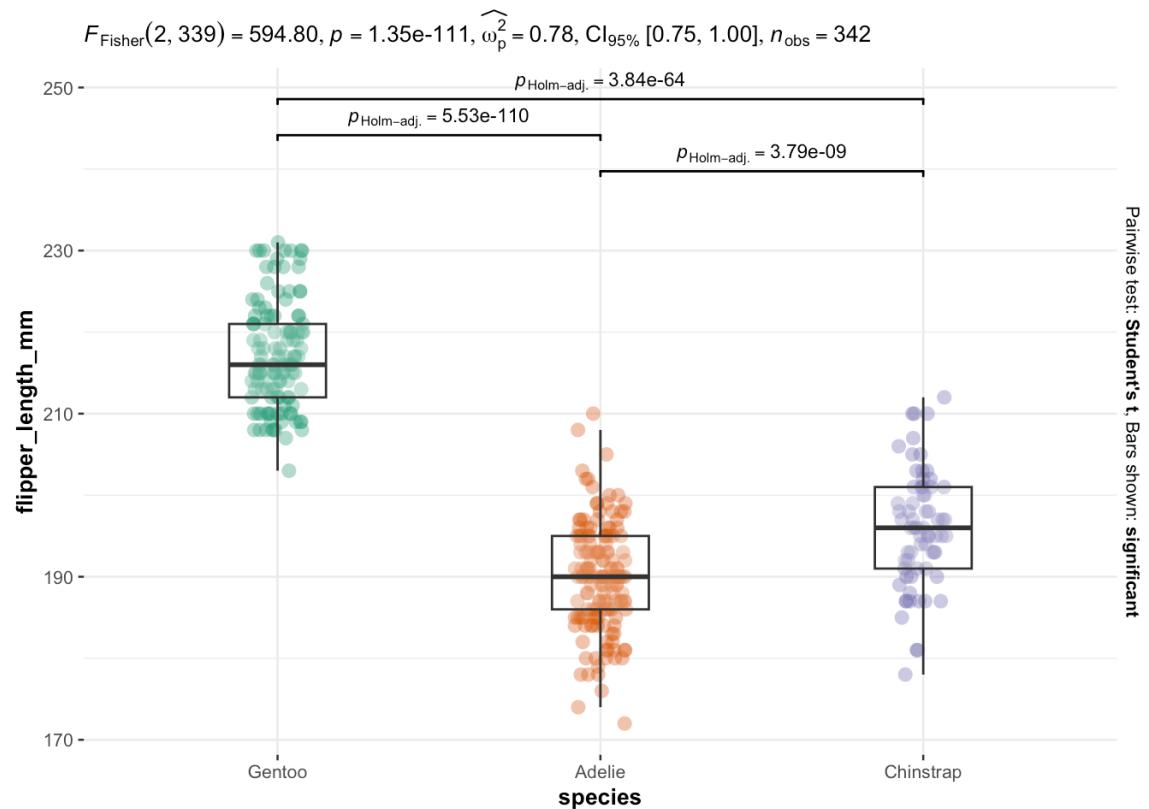
And the second method is from the `{ggstatsplot}` package:

```
library(ggstatsplot)
```

```

ggbetweenstats(
  data = dat,
  x = species,
  y = flipper_length_mm,
  type = "parametric", # ANOVA or Kruskal-Wallis
  var.equal = TRUE, # ANOVA or Welch ANOVA
  plot.type = "box",
  pairwise.comparisons = TRUE,
  pairwise.display = "significant",
  centrality.plotting = FALSE,
  bf.message = FALSE
)

```



As you can see on the above plot, boxplots by species are presented together with p -values of the ANOVA (after $p =$ in the subtitle of the plot) and p -values of the post-hoc tests (above each comparison).

Unit-wise

MCQs

Unit-1 – Introduction to Probability

1. What is the definition of Probability?

- A. The measure of the number of successful outcomes to total possible outcomes.
- B. The measure of the number of failures to total possible outcomes.
- C. The ratio of favorable outcomes to unfavorable outcomes.
- D. The ratio of unfavorable outcomes to favorable outcomes.

Answer: A

2. Which of the following is an example of Classical Probability?

- A. Tossing a biased coin
- B. Rolling a fair die
- C. Predicting the stock market
- D. Observing traffic patterns

Answer: B

3. How many ways can a committee of 4 members be selected from a group of 10 people?

- A. 210
- B. 5040
- C. 3024
- D. 1260

Answer: A

(Explanation: Combination formula: $10C4 = 10! / (4!(10-4)!) = 210$)

4. Which of the following is an example of a Permutation?

- A. Choosing 3 books out of 5 to arrange on a shelf
- B. Selecting 3 fruits from a basket of 10
- C. Dividing a group of 5 people into teams
- D. Choosing 2 letters from a word

Answer: A

(Explanation: Permutation involves arrangement/order: $5P3 = 5! / (5-3)!$)

5. What is the Addition Theorem of Probability?

- A. The probability that event A and event B both occur.

- B. The probability that event A or event B occurs, accounting for any overlap.
- C. The probability of the complement of event A.
- D. The probability that event A occurs given event B has occurred.

Answer: B

(Explanation: $P(A \cup B) = P(A) + P(B) - P(A \cap B)$)

- 6. Which formula defines the Multiplication Theorem of Probability?

- A. $P(A \text{ and } B) = P(A) + P(B)$
- B. $P(A \text{ or } B) = P(A) + P(B) - P(A \text{ and } B)$
- C. $P(A \text{ and } B) = P(A) * P(B|A)$
- D. $P(A \text{ given } B) = P(A) * P(B)$

Answer: C

(Explanation: The Multiplication Theorem: $P(A \cap B) = P(A) * P(B|A)$)

- 7. Conditional Probability is best described by which of the following?

- A. The probability that two independent events occur.
- B. The probability that an event occurs given that another event has already occurred.
- C. The probability of the union of two events.
- D. The probability that neither of the events occurs.

Answer: B

(Explanation: Conditional Probability $P(A|B) = P(A \cap B) / P(B)$)

- 8. What is the Conditional Probability, $P(A|B)$, when $P(A \text{ and } B) = 0.2$ and $P(B) = 0.5$?

- A. 0.5
- B. 0.4
- C. 0.1
- D. 0.2

Answer: B

(Explanation: $P(A|B) = P(A \cap B) / P(B) = 0.2 / 0.5 = 0.4$)

- 9. Bayes's Theorem is used to calculate:

- A. The conditional probability of an event based on prior knowledge.
- B. The probability of independent events.
- C. The likelihood of a mutually exclusive event.

D. The union of two events.

Answer: A

(Explanation: Bayes's Theorem is $P(A|B) = [P(B|A) * P(A)] / P(B)$)

10. Using Bayes's Theorem, if $P(B|A) = 0.7$, $P(A) = 0.5$, and $P(B) = 0.6$, what is $P(A|B)$?

A. 0.583

B. 0.233

C. 0.5833

D. 0.175

Answer: C

(Explanation: $P(A|B) = [P(B|A) * P(A)] / P(B) = (0.7 * 0.5) / 0.6 = 0.5833$)

Unit 2 – Random Variable

1. What is a Random Variable?

- A. A variable that can only take one specific value.
- B. A variable that takes on a deterministic value.
- C. A variable that takes on different values based on random outcomes.
- D. A variable that is only used in calculus.

Answer: C

2. Which of the following is a key difference between discrete and continuous random variables?

- A. Discrete random variables have only positive values.
- B. Continuous random variables are countable.
- C. Discrete random variables take distinct values, while continuous random variables take any value in a range.
- D. Continuous random variables are finite.

Answer: C

3. Which of the following is an example of a discrete random variable?

- A. The weight of an apple.
- B. The number of heads when flipping a coin three times.
- C. The height of a person.
- D. The temperature in a city on a given day.

Answer: B

4. The probability density function (PDF) is used for which type of random variable?

- A. Discrete random variables.
- B. Continuous random variables.
- C. Both discrete and continuous random variables.
- D. Neither discrete nor continuous random variables.

Answer: B

5. The sum of the probabilities for a discrete random variable must be equal to which of the following?

- A. 0
- B. 0.5
- C. 1
- D. It can be any value

Answer: C

6. For a continuous random variable, the area under the probability density function (PDF) over its entire range equals:

- A. 0
- B. 1
- C. Infinity
- D. A negative value

Answer: B

7. What is the mathematical expectation (or expected value) of a random variable XXX?

- A. The mode of XXX.
- B. The square root of XXX.
- C. The weighted average of all possible values of XXX.
- D. The highest possible value of XXX.

Answer: C

8. If XXX is a discrete random variable, how is its expected value $E[X]$ calculated?

- A. $E[X] = \int x f(x) dx$
- B. $E[X] = \sum (x \cdot P(X=x))$
- C. $E[X] = \max(x)$
- D. $E[X] = \sum (x^2)$

Answer: B

9. The expectation of a function of a random variable, $E[g(X)]$, is given by which theorem?

- A. Law of Total Probability
- B. Central Limit Theorem
- C. The Expectation Theorem

D. The Law of the Unconscious Statistician

Answer: D

10. Which of the following statements is true for the variance of a random variable XXX?

- A. Variance is always negative.
- B. Variance measures the spread of XXX around its mean.
- C. Variance is equal to the expected value.
- D. Variance is only defined for continuous random variables.

Answer: B

Unit 3- Data Distribution

1. Which of the following is a type of continuous probability distribution?

- a) Binomial distribution
- b) Poisson distribution
- c) Exponential distribution
- d) Bernoulli distribution

Answer: c) Exponential distribution

2. The sum of probabilities in any probability distribution is equal to:

- a) 1
- b) 0
- c) Infinity
- d) Depends on the type of distribution

Answer: a) 1

3. Which of the following distributions is best suited for modeling the number of occurrences of an event in a fixed interval of time or space?

- a) Normal distribution
- b) Poisson distribution
- c) Exponential distribution
- d) Binomial distribution

Answer: b) Poisson distribution

4. In a normal distribution, approximately how much data lies within one standard deviation from the mean?

- a) 50%
- b) 68%
- c) 95%
- d) 99.7%

Answer: b) 68%

5. Which distribution is used to model the time between events in a Poisson process?

- a) Normal distribution
- b) Exponential distribution
- c) Binomial distribution
- d) Uniform distribution

Answer: b) Exponential distribution

6. What is the key assumption of the Binomial distribution?

- a) Events are dependent
- b) Each trial has the same probability of success
- c) The number of trials is infinite
- d) Each trial has a different probability of success

Answer: b) Each trial has the same probability of success

7. Which of the following is NOT a characteristic of a normal distribution?

- a) Symmetry
- b) Mean = Median = Mode
- c) Skewness
- d) Bell-shaped curve

Answer: c) Skewness

8. In a Poisson distribution, the mean and variance are:

- a) Always equal
- b) Never equal
- c) Mean is twice the variance
- d) Mean is half the variance

Answer: a) Always equal

9. Which type of distribution is used in the Monte Carlo Simulation?

- a) Binomial distribution
- b) Any probability distribution
- c) Normal distribution only
- d) Poisson distribution only

Answer: b) Any probability distribution

10. Which method is commonly used for generating random numbers in simulations?

- a) Numerical integration
- b) Monte Carlo method
- c) Gaussian elimination
- d) Newton-Raphson method

Answer: b) Monte Carlo method

11. Which of the following describes the Binomial distribution?

- a) Continuous distribution
- b) Distribution with two possible outcomes
- c) Distribution of number of events per time period
- d) Distribution of waiting times between events

Answer: b) Distribution with two possible outcomes

12. Which distribution is used to model events that occur at a constant rate but randomly in time or space?

- a) Poisson distribution
- b) Normal distribution
- c) Exponential distribution
- d) Binomial distribution

Answer: a) Poisson distribution

13. The central limit theorem states that the sampling distribution of the sample mean approaches which distribution as the sample size increases?

- a) Binomial distribution
- b) Poisson distribution
- c) Exponential distribution
- d) Normal distribution

Answer: d) Normal distribution

14. Random number generation in computer simulations is typically based on which type of algorithms?

- a) Deterministic algorithms
- b) Stochastic algorithms

- c) Gaussian algorithms
- d) Recursion algorithms

Answer: a) Deterministic algorithms

15. Monte Carlo Simulation is mainly used for:

- a) Determining exact probabilities
- b) Sampling data from distributions
- c) Solving deterministic problems
- d) Generating random numbers

Answer: b) Sampling data from distributions

Unit 4- Testing of Hypothesis

Hypothesis Testing Procedure

1. Which of the following is the first step in the hypothesis testing procedure? a) Collecting data
b) Formulating a hypothesis
c) Choosing a level of significance
d) Drawing a conclusion

Answer: b) Formulating a hypothesis

2. The null hypothesis (H_0) usually represents which of the following?
a) A claim of effect or difference
b) A claim of no effect or no difference
c) A researcher's alternative hypothesis
d) A bias in data collection

Answer: b) A claim of no effect or no difference

Standard Error and Sampling Distribution

3. The standard error of the mean measures which of the following?
a) The spread of the entire population
b) The variability between sample means
c) The deviation of individual data points from the mean
d) The degree of correlation between variables

Answer: b) The variability between sample means

4. As the sample size increases, what happens to the standard error of the mean?
a) Increases
b) Decreases
c) Stays the same
d) Depends on the data

Answer: b) Decreases

Estimation

5. A point estimate is:
a) A single value estimate of a population parameter
b) The interval in which the population parameter is likely to fall
c) The same as the sample standard deviation
d) Always larger than the population mean

Answer: a) A single value estimate of a population parameter

6. Which of the following is a method for estimating parameters?

- a) Method of moment estimation
- b) Likelihood estimation
- c) Bayesian estimation
- d) All of the above

Answer: d) All of the above

Student's t-Distribution

7. The Student's t-distribution is most appropriate when:

- a) The population standard deviation is known
- b) The population standard deviation is unknown and the sample size is small
- c) The sample size is large
- d) The data is non-normal

Answer: b) The population standard deviation is unknown and the sample size is small

8. As the degrees of freedom increase, the t-distribution approaches which distribution?

- a) Binomial distribution
- b) Normal distribution
- c) Chi-square distribution
- d) F-distribution

Answer: b) Normal distribution

Chi-Square Test and Goodness of Fit

9. The Chi-square test is used to test for:

- a) Equality of two means
- b) Independence between two categorical variables
- c) Equality of variances
- d) Correlation between variables

Answer: b) Independence between two categorical variables

10. A goodness of fit test is used to determine:

- a) Whether sample data matches a population distribution
- b) Whether two populations have the same variance
- c) The mean difference between two independent samples
- d) The strength of correlation between two variables

Answer: a) Whether sample data matches a population distribution

F-test and Analysis of Variance (ANOVA)

11. An F-test is used to compare:

- a) The means of two groups
- b) The variances of two or more groups
- c) The correlation coefficients of two variables
- d) The proportions in a population

Answer: b) The variances of two or more groups

12. In ANOVA, a significant F-ratio indicates that:

- a) All group means are equal
- b) At least one group mean is different
- c) All variances are equal
- d) All groups have the same number of observations

Answer: b) At least one group mean is different

Factor Analysis

13. Factor analysis is primarily used to:

- a) Test hypotheses
- b) Reduce the dimensionality of data
- c) Compare two population means
- d) Estimate the population proportion

Answer: b) Reduce the dimensionality of data

14. In factor analysis, the factors are typically: a) Observable variables

- b) Latent (unobservable) variables
- c) Test statistics
- d) Confidence intervals

Answer: b) Latent (unobservable) variables

15. Which of the following is an assumption of factor analysis?

- a) Data is binary
- b) Variables are highly correlated
- c) Samples are normally distributed
- d) Variances are unequal

Answer: b) Variables are highly correlated

Unit 5- Introduction to R Programming Language

1. Which of the following is true about R programming language?
 - A) R is only used for statistical analysis.
 - B) R is a programming language and software environment for statistical computing.
 - C) R can only be run on Unix systems.
 - D) R does not support data manipulation.

Answer: B) R is a programming language and software environment for statistical computing.

2. Which function is used to install packages in R?

- A) load()
- B) install()
- C) install.packages()
- D) library()

Answer: C) install.packages()

3. How can you read a CSV file into an R data frame?

- A) readFile("data.csv")
- B) read.csv("data.csv")
- C) import.csv("data.csv")
- D) load("data.csv")

Answer: B) read.csv("data.csv")

4. Which of the following is a valid way to create a matrix in R?

- A) matrix(1:9, nrow=3, ncol=3)
- B) mat(1:9, rows=3, cols=3)
- C) mat(1,2,3,4)
- D) createMatrix(1:9, 3, 3)

Answer: A) matrix(1:9, nrow=3, ncol=3)

5. Which control structure is used to execute a set of statements repeatedly in R?

- A) if
- B) switch
- C) while

D) repeat

Answer: C) while

6. What is the correct way to define a function in R?

A) `function(x) { x * 2 }`

B) `def(x): return x * 2`

C) `func(x) { return x * 2 }`

D) `x <- function { x * 2 }`

Answer: A) `function(x) { x * 2 }`

7. Which of the following is an object in R?

A) Data frames

B) Vectors

C) Lists

D) All of the above

Answer: D) All of the above

8. Which function is used to create a data frame in R?

A) `matrix()`

B) `list()`

C) `data.frame()`

D) `create.data()`

Answer: C) `data.frame()`

9. In R, which operator is used for element-wise multiplication of matrices?

A) `*`

B) `%*%`

C) `/`

D) `%%`

Answer: A) `*`

10. What is the output of `c(1,2,3) + c(4,5,6)` in R?

A) 5 7 9

B) 1 2 3

C) 4 5 6

D) Error

Answer: A) 5 7 9

11. Which of the following functions in R is used to remove objects?

A) remove()

B) rm()

C) delete()

D) del()

Answer: B) rm()

12. What will the following command do: str(object)?

A) Prints the structure of an object.

B) Deletes the object.

C) Converts the object to a string.

D) None of the above.

Answer: A) Prints the structure of an object.

13. Which function is used to read data from a file in R?

A) scan()

B) read()

C) read.file()

D) file()

Answer: A) scan()

14. What type of control structure is if-else in R?

A) Looping structure

B) Conditional structure

C) Function definition

D) Data manipulation

Answer: B) Conditional structure

15. Which function is used to view the first few rows of a data frame in R?

A) head()

B) view()

C) summary()

D) show()

Answer: A) head()

Unit 6- Graphical Analysis using R

1. What function is used to create a basic scatter plot in R?

- a. barplot()
- b. boxplot()
- c. plot()
- d. hist()

Answer: c) plot()

2. Which argument in the plot() function defines the title of the plot?

- a. xlab
- b. ylab
- c. main
- d. title

Answer: c) main

3. What does a boxplot display in a dataset?

- a. Frequency distribution
- b. Mean, variance, standard deviation
- c. Minimum, lower quartile, median, upper quartile, maximum
- d. Correlation between two variables

Answer: c) Minimum, lower quartile, median, upper quartile, maximum

4. Which function is used to create a box-whisker plot in R?

- a. boxplot()
- b. plot()
- c. barplot()
- d. hist()

Answer: a) boxplot()

5. How can the layout of multiple plots be adjusted in R?

- a. par(mfrow)
- b. layout.plot()

- c. plot(multi)
- d. mfrow()

Answer: a) par(mfrow)

6. In a pie chart, which function argument defines the labels of each slice?

- a. labels
- b. xlab
- c. legend
- d. slices

Answer: a) labels

7. Which function is used to create a pie chart in R?

- a. pie()
- b. barplot()
- c. plot()
- d. boxplot()

Answer: a) pie()

8. In R, what is the correct way to create a bar chart for categorical data?

- a. plot()
- b. barplot()
- c. hist()
- d. pie()

Answer: b) barplot()

9. Which argument in the boxplot() function helps you add horizontal lines to indicate the median?

- a. notch
- b. median
- c. hline
- d. horiz

Answer: a) notch

10. How can you modify the size of plot windows in R?

- a. plot.window()
- b. resize.window()
- c. par()
- d. window.size()

Answer: c) par()

11. Which function creates a matrix of scatter plots in R?

- a. matrix()
- b. pairs()
- c. scatter()
- d. matplot()

Answer: b) pairs()

12. Which argument in the par() function helps control the margins of the plot?

- a. mar
- b. xlab
- c. xlim
- d. grid

Answer: a) mar

13. What is the default method for handling overlapping text labels in R plots?

- a. Text wrapping
- b. Clipping
- c. Overplotting
- d. Plot resizing

Answer: c) Overplotting

14. Which argument in the barplot() function allows you to specify whether the bars should be horizontal?

- a. horizontal
- b. barh
- c. beside
- d. horiz

Answer: d) horiz

15. Which argument in the pie() function allows you to set colors for the slices?

- a. col
- b. slice
- c. color
- d. fill

Answer: a) col

Unit 7- Advance R

1. What does the apply() function in R do?

- a) Apply a function over a vector
- b) Apply a function over the margins of an array
- c) Apply a function to a dataframe
- d) Apply a function over multiple vectors

Answer: b) Apply a function over the margins of an array

2. Which of the following is used for memory management in R?

- a) gc()
- b) rm()
- c) save()
- d) summary()

Answer: a) gc()

3. In R, the function to create a copy of an object without making it reference-based is:

- a) clone()
- b) deepcopy()
- c) copy.deepcopy()
- d) No such function; R uses copy-on-modify by default

Answer: d) No such function; R uses copy-on-modify by default

Correlation and Regression Analysis

4. Which of the following is true for Pearson's correlation coefficient?

- a) It is always positive
- b) It only works for non-linear relationships
- c) It ranges from -1 to 1
- d) It is unit dependent

Answer: c) It ranges from -1 to 1

5. What is multicollinearity in regression analysis?

- a) A scenario where independent variables are highly correlated
- b) A scenario where dependent variables are highly correlated
- c) When the regression model does not fit the data
- d) A case where residuals are autocorrelated

Answer: a) A scenario where independent variables are highly correlated

6. In linear regression, the R-squared value represents:

- a) The number of independent variables
- b) The goodness-of-fit of the model
- c) The intercept of the regression line
- d) The p-value of the regression

Answer: b) The goodness-of-fit of the model

Analysis of Variance (ANOVA)

7. What is the main purpose of conducting an ANOVA test?

- a) To compare means between two groups
- b) To compare means between more than two groups
- c) To test for correlation between variables
- d) To test for linear regression

Answer: b) To compare means between more than two groups

8. The F-statistic in ANOVA is used to:

- a) Compare variances within groups
- b) Test the null hypothesis that all group means are equal
- c) Measure the correlation between dependent variables
- d) Compare regression coefficients

Answer: b) Test the null hypothesis that all group means are equal

9. Which assumption is crucial for ANOVA to be valid?

- a) Homogeneity of variance
- b) Multicollinearity
- c) Independence of errors
- d) Both a and c

Answer: d) Both a and c

Creating Data for Complex Analysis

10. Which function in R can be used to simulate random normal data for analysis?

- a) rnorm()
- b) rpois()
- c) runif()
- d) rbinom()

Answer: a) rnorm()

11. To create a data frame in R from existing vectors, which function should be used?

- a) cbind()
- b) rbind()
- c) data.frame()
- d) list()

Answer: c) data.frame()

12. What does the sample() function in R do?

- a) Sample a subset of rows from a dataframe
- b) Generate random samples from a given vector
- c) Create a summary of a vector
- d) Perform a bootstrap analysis

Answer: b) Generate random samples from a given vector

Summarizing Data

13. Which function is commonly used in R for generating summary statistics for each variable in a data frame?

- a) summary()
- b) describe()
- c) summarize()
- d) overview()

Answer: a) summary()

14. What does the aggregate() function in R do?

- a) Summarizes multiple data sets into one
- b) Splits data into groups and computes summary statistics
- c) Creates frequency distributions
- d) Combines columns from multiple data frames

Answer: b) Splits data into groups and computes summary statistics

Case Studies

15. In a real-world regression case study, which of the following would indicate overfitting?

- a) High accuracy on both training and test sets
- b) High accuracy on training data but low accuracy on test data
- c) Low R-squared value
- d) High p-value for most predictors

Answer: b) High accuracy on training data but low accuracy on test data

Practice Questions

Questions on Binomial Distribution

Q1 What is meant by binomial distribution?

The binomial distribution is the discrete probability distribution that gives only two possible results in an experiment, either success or failure.

Q2 Mention the formula for the binomial distribution.

The formula for binomial distribution is:

$$P(x) = {}_nC^x p^x (q)^{n-x}$$

Where p is the probability of success, q is the probability of failure, n= number of trials

Q3 What is the formula for the mean and variance of the binomial distribution?

The mean and variance of the binomial distribution are:

$$\text{Mean} = np$$

$$\text{Variance} = npq$$

Q4 What are the criteria for the binomial distribution?

The number of trials should be fixed.

Each trial should be independent.

The probability of success is exactly the same from one trial to the other trial.

Q5 What is the difference between a binomial distribution and normal distribution?

The binomial distribution is discrete, whereas the normal distribution is continuous.

Practice Problems

Solve the following problems based on binomial distribution:

1. The mean and variance of the binomial variate X are 8 and 4 respectively. Find $P(X < 3)$.
2. The binomial variate X lies within the range $\{0, 1, 2, 3, 4, 5, 6\}$, provided that $P(X=2) = 4P(x=4)$. Find the parameter “p” of the binomial variate X.
3. In binomial distribution, X is a binomial variate with $n= 100$, $p= \frac{1}{3}$, and $P(x=r)$ is maximum. Find the value of r.
4. The probability that a mountain-bike rider travelling along a certain track will have a tyre burst is 0.05. Find the probability that among 17 riders: (a) exactly one has a burst tyre (b) at most three have a burst tyre (c) two or more have burst tyres.
5. (a) A transmission channel transmits zeros and ones in strings of length 8, called ‘words’. Possible distortion may change a one to a zero or vice versa; assume this distortion occurs with probability .01 for each digit, independently. An error-correcting code is employed in the construction of the word such that the receiver can deduce the word correctly if at most one digit is in error. What is the probability the word is decoded incorrectly?

- (b) Assume that a word is a sequence of 10 zeros or ones and, as before, the probability of incorrect transmission of a digit is .01. If the error-correcting code allows correct decoding of the word if no more than two digits are incorrect, compute the probability that the word is decoded correctly.
6. An examination consists of 10 multi-choice questions, in each of which a candidate has to deduce which one of five suggested answers is correct. A completely unprepared student guesses each answer completely randomly. What is the probability that this student gets 8 or more questions correct? Draw the appropriate moral!
 7. The probability that a machine will produce all bolts in a production run within specification is 0.998. A sample of 8 machines is taken at random. Calculate the probability that (a) all 8 machines, (b) 7 or 8 machines, (c) at least 6 machines will produce all bolts within specification
 8. The probability that a machine develops a fault within the first 3 years of use is 0.003. If 40 machines are selected at random, calculate the probability that 38 or more will not develop any faults within the first 3 years of use.

Questions on Normal Distribution

Q1 What is a normal distribution in statistics?

A probability function that specifies how the values of a variable are distributed is called the normal distribution. It is symmetric since most of the observations assemble around the central peak of the curve. The probabilities for values of the distribution are distant from the mean narrow off evenly in both directions.

Q2 What does normal distribution mean?

In statistics (and in probability theory), the Normal Distribution, also called the Gaussian Distribution, is the most important continuous probability distribution. Sometimes it is also called a bell curve.

Q3 What is a normal distribution used for?

A normal distribution is significant in statistics and is often used in the natural sciences and social arts to describe real-valued random variables whose distributions are unknown.

Q4 What are the characteristics of a normal distribution?

The essential characteristics of a normal distribution are:

It is symmetric, unimodal (i.e., one mode), and asymptotic.

The values of mean, median, and mode are all equal.

A normal distribution is quite symmetrical about its center. That means the left side of the center of the peak is a mirror image of the right side. There is also only one peak (i.e., one mode) in a normal distribution.

Q5 How do you know if data is normally distributed?

A histogram presents a useful graphical representation of the given data. When a histogram of distribution is superimposed with its normal curve, then the distribution is known as the normal distribution.

Q6 How do you use a normal distribution table?

As we know, the label for rows contains the integer part and the first decimal place of z. In contrast, the title for columns comprises the second decimal place of z. The values within the table are the probabilities corresponding to the table type. Hence, to get the value of 0.56 from the z-table, identify the probability value corresponding to the 0.5 row and 0.06 column (=0.2123).

Questions on Poisson Distribution

Q1 What is a Poisson distribution?

A Poisson distribution is defined as a discrete frequency distribution that gives the probability of the number of independent events that occur in the fixed time.

Q2 When do we use Poisson distribution?

Poisson distribution is used when the independent events occurring at a constant rate within the given interval of time are provided.

Q3 What is the difference between the Poisson distribution and normal distribution?

The major difference between the Poisson distribution and the normal distribution is that the Poisson distribution is discrete whereas the normal distribution is continuous. If the mean of the Poisson distribution becomes larger, then the Poisson distribution is similar to the normal distribution.

Q4 Are the mean and variance of the Poisson distribution the same?

The mean and the variance of the Poisson distribution are the same, which is equal to the average number of successes that occur in the given interval of time.

Q5 Mention the three important constraints in Poisson distribution.

The three important constraints used in Poisson distribution are:
The number of trials (n) tends to infinity
The probability of success (p) tends to zero
 $np=1$, which is finite.

1. Large sheets of metal have faults in random positions but on average have 1 fault per 10 m². What is the probability that a sheet 5 m × 8 m will have at most one fault?
2. If 250 litres of water are known to be polluted with 106 bacteria what is the probability that a sample of 1 cc of the water contains no bacteria?
3. Suppose vehicles arrive at a signalised road intersection at an average rate of 360 per hour and the cycle of the traffic lights is set at 40 seconds. In what percentage of cycles will the number of vehicles arriving be (a) exactly 5, (b) less than 5? If, after the lights change to green, there is time to clear only 5 vehicles before the signal changes to red again, what is the probability that waiting vehicles are not cleared in one cycle?
4. Previous results indicate that 1 in 1000 transistors are defective on average.
 - (a) Find the probability that there are 4 defective transistors in a batch of 2000.
 - (b) What is the largest number, N, of transistors that can be put in a box so that the probability of no defectives is at least 1/2?
5. A manufacturer sells a certain article in batches of 5000. By agreement with a customer the following method of inspection is adopted: A sample of 100 items is drawn at random from each batch and inspected.

If the sample contains 4 or fewer defective items, then the batch is accepted by the customer. If more than 4 defectives are found, every item in the batch is inspected. If inspection costs are 75 p per hundred articles, and the manufacturer normally produces 2% of defective articles, find the average inspection costs per batch.

6. A book containing 150 pages has 100 misprints. Find the probability that a particular page contains

- (a) no misprints,
- (b) 5 misprints,
- (c) at least 2 misprints,
- (d) more than 1 misprint.

7. For a particular machine, the probability that it will break down within a week is 0.009. The manufacturer has installed 800 machines over a wide area. Calculate the probability that

- (a) 5, (b) 9, (c) less than 5, (d) more than 4 machines breakdown in a week.

8. At a given university, the probability that a member of staff is absent on any one day is 0.001. If there are 800 members of staff, calculate the probabilities that the number absent on any one day is

- (a) 6,
- (b) 4,
- (c) 2,
- (d) 0,
- (e) less than 3,
- (f) more than 1.

9. The number of failures occurring in a machine of a certain type in a year has a Poisson distribution with mean 0.4. In a factory there are ten of these machines. What is

- (a) the expected total number of failures in the factory in a year?
- (b) the probability that there are fewer than two failures in the factory in a year?

Case Studies

Case Study – Exponential Distribution



The time in minutes, X , between the arrival of successive customers at a post office is exponentially distributed with pdf

$$f(x) = 0.2e^{-0.2x}.$$

(A) What is the expected time between arrivals?

(B) A customer walks into the post office at 12.30 p.m. What is the probability the next customer arrives:

(i) on or before 12:32p.m.?

(ii) after 12:35p.m.?

Solution

(A) Here $\lambda=0.2$ and so the mean time between arrivals is $1/0.2=5$ minutes.

(B) (i) If the next customer arrives on or before 12:32 p.m., it means that the time between their arrival and the previous arrival is at most 22 minutes. So, we require $P(X \leq 2)$.

Using the formula above we have:

$$\begin{aligned} P(X \leq 2) &= 1 - e^{-0.2 \times 2} \\ &= 1 - 0.67032 \\ &= 0.330 \text{ (to 3 d.p.)} \end{aligned}$$

(ii) If the next customer arrives after 12:35 p.m. then the time between the two customers is more than 55 minutes. We now require $P(X>5)$. To calculate $P(X>x)$.

This is equivalent to $1-P(X \leq x)$ and so:

$$\begin{aligned} P(X > 5) &= 1 - P(X \leq 5) \\ &= 1 - (1 - e^{-0.2 \times 5}) \\ &= e^{-1} \\ &= 0.368 \text{ (to 3 d.p.)}. \end{aligned}$$

Poisson Processes

The Exponential distribution is often used as a model for the times between events. We have looked at events occurring randomly in time in association with the Poisson distribution. The Poisson distribution gives the probabilities for the *number* of events taking place in the given time period whereas the exponential distribution gives the probabilities for *times between* the events. Both of these concern events occurring randomly in time at a constant average rate, $\lambda\lambda$. This is known as a Poisson process.

For example, consider a series of randomly occurring events, such as customers entering a bank. The times of arrivals might look like this:



There are two ways we can view the data.

1. The number of arrivals in each minute (1,1,0,3,1).
2. The times between successive arrivals.

For the Poisson process we have,

1. the number of arrivals follows a Poisson distribution with parameter $\lambda\lambda$, and
- the time between successive calls has an exponential distribution with parameter $\lambda\lambda$.

Other Continuous Probability Distributions

There are other important continuous probability distributions which you will meet in practical business problems and decision making. In particular, you will meet the *Student's*

t-distribution or *t distribution* and the *Chi-squared (χ^2) distribution*. The *t*-distribution is used when testing a hypothesis about a mean or a difference between two means. The Chi-square distribution is used when analysing categorical data.

Caselet

On the average, a certain computer part lasts ten years. The length of time the computer part lasts is exponentially distributed.

- a) What is the probability that a computer part lasts more than 7 years?

Solution: Let x = the amount of time (in years) a computer part lasts.

$$\mu = 10 \text{ so } m = \frac{1}{\mu} = \frac{1}{10} = 0.10$$

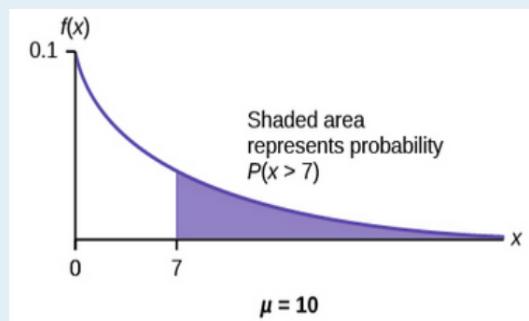
$P(x > 7)$. Draw the graph.

$$P(x > 7) = 1 - P(x < 7).$$

Since $P(X < x) = 1 - e^{-mx}$ then $P(X > x) = 1 - (1 - e^{-mx}) = e^{-mx}$

$P(x > 7) = e^{(-0.1)(7)} = 0.4966$. The probability that a computer part lasts more than seven years is 0.4966.

On the home screen, enter $e^{(-0.1)(7)}$.



- b) On the average, how long would five computer parts last if they are used one after another?

Solution:

Case – Binomial Distribution

A University Engineering Department has introduced a new software package called SOLVIT. To save money, the University's Purchasing Department has negotiated a bargain price for a 4-user license that allows only four students to use SOLVIT at any one time. It is estimated that this should allow 90% of students to use the package when they need it. The Students' Union has asked for more licenses to be bought since engineering students report having to queue excessively to use SOLVIT. As a result, the Computer Centre monitors the use of the software. Their findings show that on average 20 students are logged on at peak times and 4 of these want to use SOLVIT. Was the Purchasing Department's estimate correct?

Solution

$$P(\text{student wanted to use SOLVIT}) = \frac{4}{20} = 0.2$$

Let X be the number of students wanting to use SOLVIT at any one time, then

$$\begin{aligned} P(X = 0) &= {}^{20}C_0(0.2)^0(0.8)^{20} = 0.0115 \\ P(X = 1) &= {}^{20}C_1(0.2)^1(0.8)^{19} = 0.0576 \\ P(X = 2) &= {}^{20}C_2(0.2)^2(0.8)^{18} = 0.1369 \\ P(X = 3) &= {}^{20}C_3(0.2)^3(0.8)^{17} = 0.2054 \\ P(X = 4) &= {}^{20}C_4(0.2)^4(0.8)^{16} = 0.2182 \end{aligned}$$

Therefore

$$\begin{aligned} P(X \leq 4) &= P(X = 0) + P(X = 1) + P(X = 2) + P(X = 3) + P(X = 4) \\ &= 0.01152 + 0.0576 + 0.1369 + 0.2054 + 0.2182 \\ &= 0.61862 \end{aligned}$$

The probability that more than 4 students will want to use SOLVIT is

$$P(X > 4) = 1 - P(X \leq 4) = 0.38138$$

That is, 38% of the time there will be more than 4 students wanting to use the software. The Purchasing Department has grossly overestimated the availability of the software on the basis of a 4-user licence.

Case normal distribution

A Practical Example: Your company packages sugar in 1 kg bags.

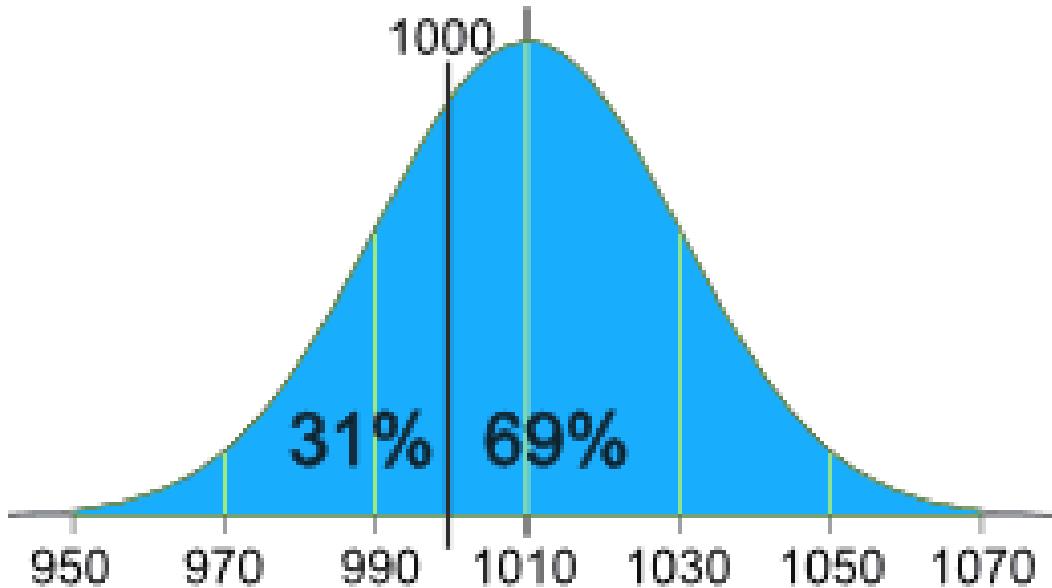
When you weigh a sample of bags you get these results:

- 1007g, 1032g, 1002g, 983g, 1004g, ... (a hundred measurements)
- Mean = 1010g

- Standard Deviation = 20g

Some values are less than 1000g ... can you fix that?

The normal distribution of your measurements looks like this:



31% of the bags are less than 1000g,

which is cheating the customer!

It is a random thing, so we can't stop bags having less than 1000g, but we can try to reduce it a lot.

Let's adjust the machine so that 1000g is:

- at -3 standard deviations:

From the big bell curve above we see that 0.1% are less. But maybe that is too small.

- at -2.5 standard deviations:

Below 3 is 0.1% and between 3 and 2.5 standard deviations is 0.5%, together that is $0.1\% + 0.5\% = 0.6\%$ (a good choice I think)

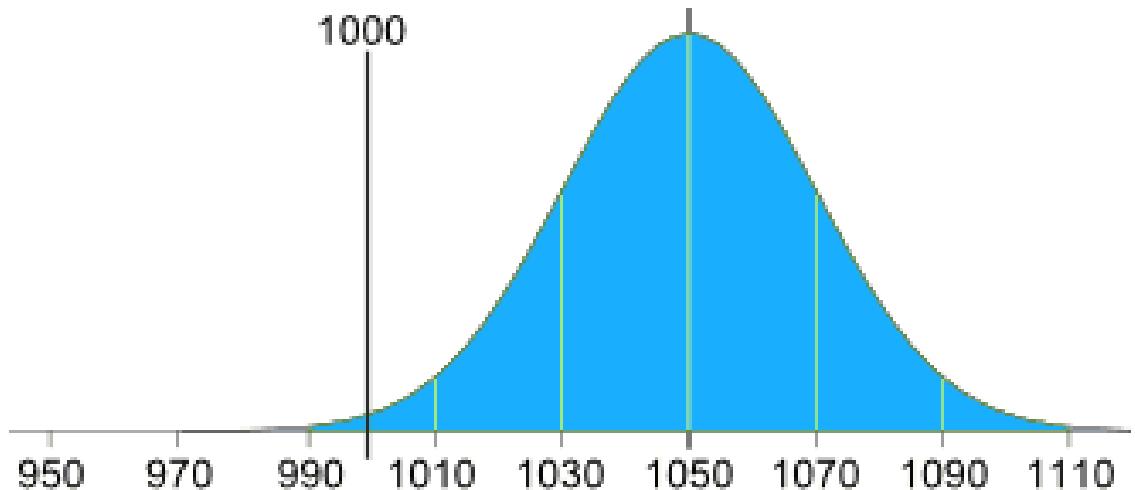
So let us adjust the machine to have 1000g at -2.5 standard deviations from the mean.

Now, we can adjust it to:

- increase the amount of sugar in each bag (which changes the mean), or
- make it more accurate (which reduces the standard deviation)

Let us try both.

Adjust the mean amount in each bag

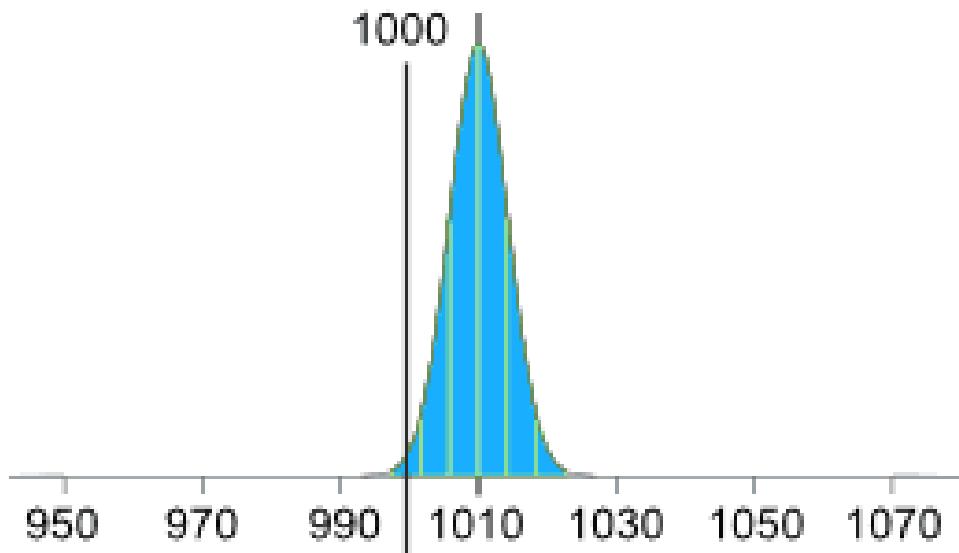


The standard deviation is 20g, and we need 2.5 of them:

$$2.5 \times 20\text{g} = 50\text{g}$$

So the machine should average 1050g, like this:

Adjust the accuracy of the machine



Or we can keep the same mean (of 1010g), but then we need 2.5 standard deviations to be equal to 10g:

$$10\text{g} / 2.5 = 4\text{g}$$

So the standard deviation should be 4g, like this:

(We hope the machine is that accurate!)

Or perhaps we could have some combination of better accuracy and slightly larger average size, I will leave that up to you!

Case questions on passion distribution

1. A manufacturer produces light-bulbs that are packed into boxes of 100. If quality control studies indicate that 0.5% of the light-bulbs produced are defective, what percentage of the boxes will contain: (a) no defective? (b) 2 or more defectives?

Solution

As n is large and p , the $P(\text{defective bulb})$, is small, use the Poisson approximation to the binomial probability distribution. If $X = \text{number of defective bulbs in a box}$, then

$$X \sim P(\mu) \text{ where } \mu = n \times p = 100 \times 0.005 = 0.5$$

$$(a) P(X = 0) = \frac{e^{-0.5}(0.5)^0}{0!} = \frac{e^{-0.5}(1)}{1} = 0.6065 \approx 61\%$$

$$(b) P(X = 2 \text{ or more}) = P(X = 2) + P(X = 3) + P(X = 4) + \dots \text{ but it is easier to consider:}$$

$$P(X \geq 2) = 1 - [P(X = 0) + P(X = 1)]$$

$$P(X = 1) = \frac{e^{-0.5}(0.5)^1}{1!} = \frac{e^{-0.5}(0.5)}{1} = 0.3033$$

$$\text{i.e. } P(X \geq 2) = 1 - [0.6065 + 0.3033] = 0.0902 \approx 9\%$$

2. A Council is considering whether to base a recovery vehicle on a stretch of road to help clear incidents as quickly as possible. The road concerned carries over 5000 vehicles during the peak rush hour period. Records show that, on average, the number of incidents during the morning rush hour is 5. The Council won't base a vehicle on the road if the probability of having more than 5 incidents in any one morning is less than 30%. Based on this information should the Council provide a vehicle?

Answer

We need to calculate the probability that more than 5 incidents occur i.e. $P(X > 5)$. To find this we use the fact that $P(X > 5) = 1 - P(X \leq 5)$. Now, for this problem:

$$P(X = r) = e^{-5} \frac{5^r}{r!}$$

Writing answers to 5 d.p. gives:

$$P(X = 0) = e^{-5} \frac{5^0}{0!} = 0.00674$$

$$P(X = 1) = 5 \times P(X = 0) = 0.03369$$

$$P(X = 2) = \frac{5}{2} \times P(X = 1) = 0.08422$$

$$P(X = 3) = \frac{5}{3} \times P(X = 2) = 0.14037$$

$$P(X = 4) = \frac{5}{4} \times P(X = 3) = 0.17547$$

$$P(X = 5) = \frac{5}{5} \times P(X = 4) = 0.17547$$

$$\begin{aligned} P(X \leq 5) &= P(X = 0) + P(X = 1) + P(X = 2) + P(X = 3) + P(X = 4) + P(X = 5) \\ &= 0.61596 \end{aligned}$$

The probability of more than 5 incidents is $P(X > 5) = 1 - P(X \leq 5) = 0.38403$, which is 38.4% (to 3 s.f.) so the Council should provide a vehicle.

A Business Planning Example using Monte-Carlo Simulation

Imagine you are the marketing manager for a firm that is planning to introduce a new product. You need to estimate the first year net profit from this product, which will depend on:

- Sales volume in units
- Price per unit
- Unit cost
- Fixed costs

Net profit will be calculated as Net Profit = Sales Volume* (Selling Price - Unit cost) - Fixed costs. Fixed costs (for overhead, advertising, etc.) are known to be \$120,000. But the other factors all involve some *uncertainty*. Sales volume (in units) can cover quite a range, and the selling price per unit will depend on competitor actions. Unit costs will also vary depending on vendor prices and production experience.

Uncertain Variables

To build a risk analysis model, we must first identify the uncertain variables -- also called *random variables*. While there's *some* uncertainty in almost *all* variables in a business model, we want to focus on variables where the range of values is significant.

Sales and Price

Based on your market research, you believe that there are equal chances that the market will be Slow, OK, or Hot.

- In the "Slow market" scenario, you expect to sell 50,000 units at an average selling price of \$11.00 per unit.
- In the "OK market" scenario, you expect to sell 75,000 units, but you'll likely realize a lower average selling price of \$10.00 per unit.
- In the "Hot market" scenario, you expect to sell 100,000 units, but this will bring in competitors who will drive down the average selling price to \$8.00 per unit.

As a result, you *expect* to sell 75,000 units (*i.e.*, $(50,000+75,000+100,000)/3 = 75,000$) at an average selling price of \$9.67 per unit (*i.e.*, $(\$11+\$10+\$8)/3 = \9.67).

Unit Cost

Another uncertain variable is Unit Cost. Your firm's production manager advises you that unit costs may be anywhere from \$5.50 to \$7.50, with a most likely cost of \$6.50. In this case, the most likely cost is also the average cost.

Uncertain Functions

Net Profit

Our next step is to identify uncertain functions -- also called *functions of a random variable*. Recall that Net Profit is calculated as Net Profit = Sales Volume * (Selling Price - Unit cost) - Fixed costs. However, Sales Volume, Selling Price and Unit Cost are all uncertain variables, so Net Profit is an uncertain function.

The Flawed Average Model

Before we explore how to use simulation to analyze this problem, consider the Excel model pictured below, which calculates Net Profit based on average sales volume, average selling price, and average unit cost.

Financial Forecast					
Sales Scenarios		Volume	Price	Sales & Cost Data	
1	1-Hot Market	100,000	\$8.00	Sales Scenario	Average
2	2-OK Market	75,000	\$10.00	Sales Volume	75,000
3	3-Slow Market	50,000	\$11.00	Selling Price	\$9.67
4				Unit Cost	\$6.50
Cost Scenarios		Unit Cost		Profit Forecast	
9	1-Minimum Cost	\$5.50		Net Profit	
10	2-Most Likely Cost	\$6.50		\$117,500	
11	3-Maximum Cost	\$7.50			
12					
13	Fixed Costs	\$120,000			
14					

Intuition might suggest that plugging the average value of our uncertain inputs (Sales Volume, Selling Price, and Unit Cost) into our model should produce the average value of the output (Net Profit). However, as we'll see in a moment, the Net Profit figure of \$117,750 calculated by this model, based on average values for the uncertain factors, is quite misleading. The true average Net Profit is closer to \$93,000! As Dr. Sam Savage warns, "Plans based on average assumptions will be *wrong* on average."

Internal Sample paper

Internal Examination (Sep -2024)

Course: BCA

Subject: Statistical Analysis using R

Max. Marks: 40

Semester: V

Course Code: 504

Max. Time: 1:30 hrs.

Instructions: Use of calculator for subjects like Financial Management, operation etc. allowed if required. (Scientific calculators not allowed).

Use of unfair means will lead to cancellation of paper followed by disciplinary action.

Attempt any two questions from section-I and Attempt any two questions from section-II.

Section-I

(Theoretical Concept and Practical/Application oriented)

Answer in 500 words. Each question carries 08 marks.

Q1.

Q2.

Q3.

Q4. Write short note on any two. Answer in 300 words. **Each carries 04 marks.**

a)

b)

c)

Section-II

(Analytical Question / Case Study / Essay Type Question to test analytical and Comprehensive Skills)

Answer in 700 words. Attempt any 2 questions. Each question carries 12 marks

Q5.

Q6.

Q7.

Checklist for Coursepacks

- Title page should be standardized bearing title of subject, course, course code, semester, year of batch (see sample attached)
 - Name of the instructors teaching the course
 - Name of course leader
- Forwarding by HOD bearing his/her signature for approval by Director
- Logo of BVIMR, name of the institution, address
- Warning "strictly for internal use" must be printed on the front title page.
- Table of content bearing
 - Serial no.
 - Contents
 - Page no.
- Copy of latest syllabus of course as specified by university
- Lesson plan bearing
 - Introduction to course
 - Course objectives
 - Learning outcomes
- List of topics/ modules with content
- Evaluation criteria
 - CES evaluation description
 - Recommended text books & reference books
 - Internet resources
 - Swayan courses
- Session plan bearing
 - Session number
 - Topic
 - Readings/case required
 - Pedagogy followed
 - Learning outcome
- Contact details of instructors along with profile
- Main body of course pack having reading material, exercises, case studies, pages for notes
- University question papers (preferably last five years including latest university paper)
- Internal question papers (internal-I-05 papers), (Internal-II-05 papers with latest last year papers)

Note: Include question paper of same subject of old syllabus if required to cover up five years papers.

Declaration by Faculty:

I, Dr. Rakhee Chhibber, Visiting faculty Teaching Statistical Analysis using R in BCA course V Sem have incorporated all the necessary pages section/quotations papers mentioned in this check list above except 5 years Papers because this course is introduced first time in University subjects.

Signature