SYLLABUS FOR COMPUTER SCIENCE MAJOR WITH HONOURS

SEM. VII

4th Year [Four Year Under Graduate Program with Honours] (FYUGPHONS)

Under Single Major Single Minor

(To be implemented from Session 2023-24)

Proposed Syllabus for four years COMPUTER SCIENCE (Major) Programme							
Year	**Semester**	**Paper Code**	**Paper**	**Credits**	**Periods/Week**	**Exam. Marks**	**Total Marks**
3rd Year	**VII**	UCMSMAJ47017	Machine Learning	3	3	60	60
		UCMSMAJ47017 T	Machine Learning ((Tutorial))	1	1	00	
		UCMSMAJ47018	LaTeX Programming	3	3	40	60
		UCMSMAJ47018 L	LaTeX Programming (Lab)	1	1	20	
		UCMSMAJ47019	R Programming	3	3	40	60
		UCMSMAJ47019 L	R Programming (Lab)	1	1	20	
	VIII	UCMSMAJ48020	**Project / Dissertation**	4	4		
		UCMSMAJ48021	**Research Methodology**	4	4		
		UCMSMAJ48022	**Field Word / Case Studies**	4	4		
		UCMSMAJ48023	**Seminar Presentation / Group Discussion**	4	4		

NOTE: Tutorials should involve problem solving session/activity related to the subject taught.

SYLLABUS FOR COMPUTER SCIENCE MAJOR WITH HONOURS

Discipline: Science [√] Arts, Humanities & Social Science ☐

Commerce ☐ BBA ☐ BCA ☐

Subject Name: COMPUTER SCIENCE

Subject Code: **UCMSMAJ47017** (Will be provided by the University)

Semester: Semester I ☐ Semester II ☐ Semester III ☐ Semester IV ☐

Semester V ☐ Semester VI ☐ Semester VII [√] Semester VIII ☐

Course Name: **MACHINE LEARNING**

Course Code: (Will be provided by the University)

Course Credit: Theoretical 3 Practical/Tutorial 1

Marks Allotted: Theoretical 60 Practical/Tutorial 0

Continuing Evaluation Attendance

Course Type (tick the correct alternatives):

Major Core	[√]	AEC	☐
Interdisciplinary/ DSE	☐	SEC	☐
Minor / Generic Elective	☐	VAC	☐
Research Project/Dissertation	☐	Vocational	☐

Is the course focused on employability / entrepreneurship? YES [√] NO ☐

Is the course focused on imparting life skill? YES [√] NO ☐

Is the course based on Activity ? YES [√] NO ☐

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

Course Code: **UCMSMAJ47017**

Course Name: **MACHINE LEARNING**

Brief Course Description

This course provides a comprehensive introduction to Machine Learning, covering fundamental concepts, statistical learning methods, regression techniques, classification and clustering algorithms, regularization, and neural networks. Students will learn how intelligent systems can learn from data and apply machine learning techniques to solve real-world problems across diverse domains.

Prerequisite(s) and/or Note(s)

Students should have basic knowledge of programming, mathematics, statistics, and data structures. Practical sessions using Python, R, MATLAB, or WEKA will accompany theoretical concepts, and students are encouraged to undertake mini-projects and case studies for hands-on learning.

Course Objectives

The course aims to introduce students to the principles and methodologies of machine learning, develop their ability to preprocess and analyze data, apply machine learning algorithms for prediction and classification tasks, evaluate model performance, and understand the role of neural networks in intelligent systems.

Course Outcomes

Upon successful completion of this course, students will be able to understand and apply fundamental machine learning concepts and techniques. They will preprocess and analyze data, develop predictive models using regression, classification, clustering, and neural networks, and evaluate model performance effectively. Students will gain proficiency in using machine learning tools and software for intelligent system development. They will also be able to solve real-world problems through data-driven decision-making and predictive analytics.

Knowledge Acquired

Students will acquire knowledge of machine learning fundamentals, statistical learning techniques, regression models, classification and clustering methods, dimensionality reduction, regularization techniques, neural network architectures, and their applications in solving real-world problems.

Skills Gained

Students will develop skills in data analysis, machine learning model development, predictive analytics, feature engineering, data visualization, algorithm implementation, model evaluation, and the use of machine learning tools and software platforms.

Competency Developed

Upon successful completion of the course, students will be able to design, implement, evaluate, and optimize machine learning models for various applications. They will demonstrate the ability to analyze complex datasets, develop intelligent predictive systems, apply neural network techniques, and use machine learning methodologies in research, industry, and interdisciplinary problem-solving environments.

Detailed Syllabus		
4th Year: Semester 7		
UCMSMAJ47017	**MACHINE LEARNING**	**[Credits: 3, Lectures 45]**

Unit 1: Introduction **(7 Lectures)**

Concept of Machine Learning, Applications of Machine Learning, Key elements of Machine Learning, Supervised vs. Unsupervised Learning - Characteristics, Applications, Challenges, Statistical Learning: Bayesian Method, The Naive Bayes Classifier.

Unit 2: Machine Learning and Linear Algebra **(5 Lectures)**

Plotting of Data, Vectorization, Matrices and Vectors: Addition, Multiplication, Transpose and Inverse using available tool such as Python, R Programming, MATLAB, and WEKA.

Unit 3: Linear Regression **(5 Lectures)**

Concept of linear regression- Prediction using Linear Regression, Gradient Descent, Linear Regression with one variable and with multiple variables, Polynomial Regression.

Unit 4: Logistic Regression **(4 Lectures)**

Concept of logistic regression - Classification using Logistic Regression, Logistic Regression vs Linear Regression, Logistic Regression with one variable and with multiple variables.

Unit 5: Regularization **(4 Lectures)**

Concept of Regularization, Regularization and its utility- The problem of Overfitting, Application of Regularization.

Unit 6: Classification and Clustering Techniques **(10 Lectures)**

Concept of Classification and Clustering- Classification Algorithms - K-Nearest Neighbour (KNN), Decision Trees, Naïve Bayes Classifier, Support Vector Machine (SVM) - Clustering Techniques- K-Means Clustering, Hierarchical Clustering, and Density based clustering- Dimensionality Reduction - Principal Component Analysis (PCA), Feature Reduction Techniques

Unit 7: Neural Networks **(10 Lectures)**

Concept of Artificial Neural Networks (ANN), Biological Neuron vs Artificial Neuron, Structure of Neural Networks: Input Layer, Hidden Layer, and Output Layer - Activation Functions: Sigmoid, ReLU, and Softmax, Perceptron and Multilayer Perceptron (MLP)- Feedforward and Backpropagation Neural Networks - Training of Neural Networks: Weights, Biases , Applications of Neural Networks in Image Recognition, Speech Processing, Prediction Systems and various other fields.

Suggested Readings:

1. Brett Lantz, Machine Learning with R: Expert techniques for predictive modeling, 3rd Edition, Packt Publishing, 2019

2. K. P. Murphy, Machine Learning: A Probabilistic Perspective. MIT Press, 2012.

3. P. Harrington, Machine Learning in Action. Manning Publications, 2012

4. C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2016.

5. S. Marsland, Machine Learning: An Algorithmic Perspective. 1st Ed. Chapman and Hall, 2009

Web Resources:

1. www.towardsdatascience.com
2. https://machinelearningmastery.com

UCMSMAJ47017L MACHINE LEARNING [Credits: 1, Lecture Hours:15]

Machine Learning tutorial as assigned and advised by teacher(s).

SYLLABUS FOR COMPUTER SCIENCE MAJOR WITH HONOURS

Discipline: Science [√] Arts, Humanities & Social Science ☐

Commerce ☐ BBA ☐ BCA ☐

Subject Name: COMPUTER SCIENCE

Subject Code: **UCMSMAJ47018** (Will be provided by the University)

Semester: Semester I ☐ Semester II ☐ Semester III ☐ Semester IV ☐

Semester V ☐ Semester VI ☐ Semester VII [√] Semester VIII ☐

Course Name: **LaTeX PROGRAMMING**

Course Code: (Will be provided by the University)

Course Credit: Theoretical 3 Practical/Tutorial 1

Marks Allotted: Theoretical 40 Practical/Tutorial 20

Continuing Evaluation Attendance

Course Type (tick the correct alternatives):

Major Core	[√]	AEC	☐
Interdisciplinary/ DSE	☐	SEC	☐
Minor / Generic Elective	☐	VAC	☐
Research Project/Dissertation	☐	Vocational	☐

Is the course focused on employability / entrepreneurship? YES [√] NO ☐

Is the course focused on imparting life skill? YES [√] NO ☐

Is the course based on Activity ? YES [√] NO ☐

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

Course Code: **UCMSMAJ47018**

Course Name: **LaTeX PROGRAMMING**

Brief Course Description

This course introduces LaTeX, a powerful document preparation system widely used for creating scientific, technical, academic, and professional documents. Students will learn to create well-structured documents, format text, insert tables, figures, equations, references, and prepare reports, articles, and research papers using LaTeX.

Prerequisite(s) and/or Note(s):

Basic computer literacy and familiarity with document editing are recommended. No prior programming experience is required.

Course Objectives

The course aims to provide students with the knowledge and practical skills required to create professional-quality documents using LaTeX, including the preparation of academic reports, research papers, presentations, and technical documents.

Course Outcomes

Upon successful completion of this course, students will be able to create and organize LaTeX documents using appropriate document classes, sections, formatting styles, and structured content. They will gain proficiency in typesetting mathematical expressions, equations, matrices, theorems, and proofs, as well as creating professional tables, figures, cross-references, and bibliographies using BibTeX. Students will also be able to design academic presentations with Beamer, manage large documents using multiple files, and develop complete research reports, dissertations, and thesis documents that meet professional and academic publishing standards.

Knowledge Acquired

Students will gain knowledge of LaTeX syntax, document structure, text formatting, mathematical typesetting, table and figure creation, bibliography management, and document customization techniques.

Skills Gained

Students will develop skills in creating, editing, formatting, and managing professional documents, mathematical equations, references, and presentations efficiently using LaTeX tools and packages.

Competency Developed

Upon completion of the course, students will be able to independently design and produce high-quality academic and technical documents, research papers, project reports, and presentations using LaTeX while following professional publishing standards.

Detailed Syllabus

4th Year: Semester 7

UCMSMAJ47018	LaTeX PROGRAMMING	[Credits: 3, Lectures 45]

UNIT 1 – INTRODUCTION TO LATEX AND DOCUMENT (10 Lectures)

History of LaTeX and difference between TeX and LaTeX, advantages for scientific writing, LaTeX distributions such as TeX Live and MiKTeX and online editors such as Overleaf and TeXstudio, basic document anatomy using documentclass, preamble, begin document, and end document, standard document classes article, report, and book, essential packages geometry, graphicx, hyperref, amsmath, and amssymb, title page creation using title, author, date, maketitle, and the abstract environment, sectioning commands section, subsection, subsubsection, paragraph, and subparagraph, text formatting using textbf, textit, underline, font size commands from tiny to Huge, and textcolor, lists using itemize, enumerate, and description environments including nested lists, and page layout basics using geometry for margins, pagestyle, fancyhdr for headers and footers, and pagenumbering.

UNIT 2 – MATHEMATICAL TYPESETTING (10 Lectures)

Math modes for inline math using dollar signs and parentheses and displayed math using square brackets, double dollar signs, and the equation environment, Greek letters, mathematical symbols, operators such as sum, int, frac, and sqrt, subscripts and superscripts, limits, stacked expressions, delimiters including parentheses, brackets, braces, and automatic sizing using left and right, arrays and matrices using matrix, pmatrix, and bmatrix environments from amsmath, multiline equations using align, gather, and multline environments with alignment points, numbered versus unnumbered equations using tag and notag, and theorem-like environments including theorem, lemma, proof, and definition via the amsthm package.

UNIT 3 – FIGURES, TABLES, AND CROSS-REFERENCING (9 Lectures)

External images using include graphics, the figure environment, caption, and label, figure placement specifiers htbp and the concept of floating environments, creating tables using the tabular environment with column specifiers l, c, r, and p width, horizontal lines using hline, toprule, midrule, and bottomrule via booktabs and vertical lines, advanced tables using multicolumn and multirow, cross-referencing using label, ref, pageref, and eqref, automatic lists using tableofcontents, listoffigures, and listoftables, hyperlinks using the hyperref package for clickable links, URLs using url, and footnotes using footnote.

UNIT 4 – BIBLIOGRAPHY AND CITATION MANAGEMENT (8 Lectures)

Manual bibliography using the bibliography versus automated BibTeX workflow, BibTeX fundamentals for creating and structuring a bib file with common entry types article, book, in proceedings, misc, and tech report, bibliography fields including author, title, journal, year, volume, pages, publisher, doi, and url, citation commands cite, citep, and citet with the natbib package, bibliography styles plain, unsrt, alpha, abbrv, and IEEE-style formatting, the compiling workflow latex to bibtex to latex to latex, managing bib files with reference managers such as JabRef or directly in Overleaf, and avoiding common errors with special characters in BibTeX.

UNIT 5 – BEAMER PRESENTATIONS AND STRUCTURING LARGE DOCUMENTS

(8 Lectures)

Beamer class for presentations with frames and title frames, frame content using frametitle, block environments such as theorem, alert, and example, columns environment, and overlays using pause, uncover, only, and item overlay specifications, beamer themes including outer, inner, color, and font themes with professional themes such as Madrid, Berlin, and Singapore, including figures, tables, and mathematics inside Beamer frames, the handout option in documentclass, structuring large documents such as theses and project reports, modular writing using input versus include and creating a master root file, selective compilation using includeonly, standard thesis front matter including title page, certificate, declaration, acknowledgments, abstract, table of contents, list of figures and tables, and abbreviations, chapter-wise organization with separate files and consistent preamble packages, and appendices using the appendix command.

Suggested Readings:

1. **Tobias Oetiker, Hubert Partl, Irene Hyna, and Elisabeth Schlegl,***The Not So Short Introduction to LaTeX2e* (also known as *lshort*), Free online edition.
2. **Stefan Kottwitz,***LaTeX Beginner's Guide*, Packt Publishing.
3. **Leslie Lamport,***LaTeX: A Document Preparation System*, 2nd Edition, Addison-Wesley.

UCMSMAJ47018L **LaTeX PROGRAMMING** **[Credits: 1, Lecture Hours:15]**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Create a basic LaTeX article. Add a title, your name as author, and today's date using \title, \author, \date, and \maketitle. Write an abstract. Create two sections and one subsection under each. Make one sentence bold and one sentence italic.
2. Write a LaTeX document using the report class. Add a title page. Create three sections. Inside the first section, add a bulleted list with three items. Inside the second section, add a numbered list with three items. Set page margins to 1 inch on all sides using the geometry package.
3. Make a document with a description list containing three items labeled "Introduction," "Method," and "Conclusion." Change the font size of the heading to \Large using \Large. Add page numbers at the bottom using \pagestyle{plain}.
4. Write a document containing four simple inline math expressions (e.g., $a+b=c$) and two displayed equations using the equation environment. Use at least one fraction (\frac) and one square root (\sqrt).
5. Create a document with a 2×2 matrix using the pmatrix environment. Write one multiline equation using the align environment with two lines aligned at the = sign. Use Greek letters \alpha and \beta somewhere in the document.
6. Define a theorem environment using the amsthm package. Write one theorem statement and one proof environment below it. Include one numbered equation inside the theorem.
7. Insert one image using \includegraphics inside a figure environment. Add a caption ("Figure 1: Sample Image") and a label. In the text, refer to the figure using \ref.
8. Create a simple table with 3 columns and 4 rows using the tabular environment. Use |c|c|c| for centered columns with vertical lines. Add \hline after the header row and at the end. Give it a caption.
9. Make a document with one figure and one table. Add an automatic Table of Contents using \table of contents. Use the hyperref package to make the table of contents entries clickable.
10. Create a .bib file with exactly three entries: one @article, one @book, and one @misc. Write a short paragraph in LaTeX citing all three sources using \cite. Compile using latex → bibtex → latex → latex to generate the bibliography. Use the plain style.
11. Write a document using the manual the bibliography environment. Add three entries with \bibitem and cite them in the text using \cite. Keep the formatting simple.
12. Create a Beamer presentation with exactly 5 slides: a title slide, three slides with bullet points, and one slide with a simple equation ($E=mc^2$). Use any one theme, e.g., \usetheme{Madrid}.
13. Make a 4-slide Beamer presentation. On the second slide, use the \pause command so that bullet points appear one by one. On the third slide, add an example block using \begin{example}.
14. Create a main.tex file that uses \input to include two separate files: intro.tex and conclusion.tex. The main file should contain the preamble and a title page. Each included file should contain one section with one paragraph.
15. Create a one-page mini project report containing: (a) a title page with title and author, (b) two sections, (c) one simple equation, (d) one figure with caption, (e) one table with 2 columns and 3 rows, and (f) one bibliography entry (either manual or BibTeX).

SYLLABUS FOR COMPUTER SCIENCE MAJOR WITH HONOURS

Discipline: Science [√] Arts, Humanities & Social Science ☐

Commerce ☐ BBA ☐ BCA ☐

Subject Name: COMPUTER SCIENCE

Subject Code: **UCMSMAJ47019** (Will be provided by the University)

Semester: Semester I ☐ Semester II ☐ Semester III ☐ Semester IV ☐

Semester V ☐ Semester VI ☐ Semester VII [√] Semester VIII ☐

Course Name: **R PROGRAMMING**

Course Code: (Will be provided by the University)

Course Credit: Theoretical 3 Practical/Tutorial 1

Marks Allotted: Theoretical 40 Practical/Tutorial 20

Continuing Evaluation Attendance

Course Type (tick the correct alternatives):

Major Core	[√]	AEC	☐
Interdisciplinary/ DSE	☐	SEC	☐
Minor / Generic Elective	☐	VAC	☐
Research Project/Dissertation	☐	Vocational	☐

Is the course focused on employability / entrepreneurship? YES [√] NO ☐

Is the course focused on imparting life skill? YES [√] NO ☐

Is the course based on Activity ? YES [√] NO ☐

Remarks by Chairman, UG BOS, if any

UG BOS Meeting Reference Number: Date:

Course Code: **UCMSMAJ47019**

Course Name: **R PROGRAMMING**

Brief Course Description

This course introduces the fundamentals of R programming for data analysis, statistical computing, and visualization. Students will learn R syntax, data structures, control statements, data manipulation, graphical representation of data, and basic statistical analysis techniques using R. The course emphasizes practical applications of R in data science and analytics.

Prerequisite(s) and/or Note(s)

Basic knowledge of computer fundamentals and elementary mathematics/statistics is desirable. No prior programming experience is required.

Course Objectives

The course aims to provide students with a strong foundation in R programming, data handling, statistical analysis, and data visualization. It enables learners to develop analytical skills and apply R tools for solving real-world data-driven problems.

Course Outcomes

Upon successful completion of this course, students will be able to write and execute R programs, manage and analyze datasets, create meaningful data visualizations, perform statistical and probabilistic analyses, and apply R programming techniques to solve real-world data-driven problems. They will develop the ability to use R as an effective tool for data analysis, decision-making, research, and advanced studies in data science and analytics.

Knowledge Acquired

Students will gain knowledge of R programming concepts, data structures, control structures, data manipulation techniques, visualization methods, statistical analysis, probability distributions, and hypothesis testing using R.

Skills Gained

Students will develop skills in writing R programs, managing and analyzing datasets, creating informative visualizations, performing statistical computations, and importing/exporting data from various sources.

Competency Developed

Upon completion of the course, students will be able to use R effectively for data analysis, statistical modeling, visualization, and decision-making tasks, preparing them for advanced studies and careers in data science, analytics, and research.

Detailed Syllabus		
4th Year: Semester 7		
UCMSMAJ47019	R PROGRAMMING	[Credits: 3, Lectures 45]

Unit 1: Introduction to R Programming **(3 Lectures)**

Overview of R programming: Evolution of R language, features of R programming language, what is R? , Why R? , Advantages of R over Other Programming Languages, Downloading and installing of R language, setting up the R environment - Interactivity and Code Execution: R basic syntax - R Studio - R command Prompt, R script file, comments

Unit 2: Data Structures in R **(8 Lectures)**

Vectors: Creating vectors using c(), Indexing and subsetting vectors in R, Operations on Vectors, Understanding vectorized operations and efficiency – *Matrices* : Creating matrices using matrix(), *Basic matrix operations* : addition, subtraction, multiplication, Transposing matrices - *Lists* : Creating lists with list(), Nested lists and their components, Indexing and extracting values from lists, Manipulating list components – *Arrays* : Creating arrays using array(), Manipulating and transforming array data - *Data Frames:* Introduction to Data Frames, Manipulating Data Frames, Indexing, filtering, and subsetting data frames, Applying functions to columns, Aggregating and summarizing data- *Factors:* Categorical data representation with factors, Modifying and Working with Ordinal Factors in R, Merging and Simplifying Factor Levels in R, Converting factors to other data types.

Unit 3: Control Structures in R **(7 Lectures)**

Conditional Statements: Basic if and else statements, else if and multiple conditions, Vectorized conditions with ifelse() - Loops in R : For Loop, Nested For Loops, break and next statements, While Loop, Repeat Loop.

Unit 4: Variables, Data Types and operators **(5 Lectures)**

Concept of Variables: Variable assignment, Finding Variable, Deleting Variables – Atomic Data Types in R (Numeric, Integer, Logical, Character, Complex and Raw) - Operators in R: Arithmetic Operators, Logical Operators, Comparison Operator, Assignment Operators.

Unit 5: Data Manipulation and Cleaning **(7 Lectures)**

Working with Strings : String manipulation - Data Cleaning: Handling missing values with is.na() and na.omit() - Reshaping Data : Reshaping data frames with reshape(), melt(), cast() - Data Import and Export: Reading data from CSV, Excel, and other formats.

Unit 6: Data Visualization in R **(7 Lectures)**

Concept of Plotting, Base R plotting functions: plot(), hist(), boxplot(), Customizing plots (titles, labels, colors), Saving plots to file (PDF, PNG) - Visualizations with ggplot2: ggplot2 package, Creating scatter plots, bar plots, histograms, and more, Faceting in ggplot2: Splitting Data into Subplots, Themes in ggplot2 - 3D Visualizations: concept of 3D visualization, Generating 3D scatter plots, surface plots.

Unit 7: R in Statistical Analysis and Resampling Techniques **(8 Lectures)**

Concept of R in Statistics: R as a tool for statistical analysis, calculation: mean, median, mode, variance, standard deviation, creating frequency tables and cross-tabulations - Probability Distributions in R: Normal Distribution in R, Plotting of Normal Distribution over Histogram in R - Hypothesis Testing: hypothesis testing concepts, T-tests, Chi-squared tests.

Suggested Readings:

1. The Book of R by Tilman M. Davies, no starch press (San Francisco)

2. The Art of R programming by Norman Matloff, no starch press(San Francisco)

3 R for Beginners by Emmanuel Paradis

UCMSMAJ47019L R PROGRAMMING [Credits: 1, Lecture Hours:15]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. Download and install R-Programming environment and install basic packages using install. Packages() command in R.
2. Learn all the basics of R-Programming (Data types, Variables, Operators etc.)
3. Implement R-Loops with different examples.
4. Learn the basics of functions in R and implement with examples.
5. Implement data frames in R. Write a program to join columns and rows in a data frame using c bind () and r bind () in R.
6. Implement different String Manipulation functions in R.
7. Implement different data structures in R(Vectors ,Lists ,Data Frames)
8. Create a data set and do statistical analysis on the data using R.
9. Write an R program to demonstrate variable declaration, data types, arithmetic operations, and output display.
10. Create two vectors and perform addition, subtraction, multiplication, division, and indexing operations.
11. Create two matrices and perform matrix addition, subtraction, multiplication, and transpose operations.
12. Create a list containing student information (Name, Roll No., Marks, Grade) and demonstrate accessing and modifying list elements.
13. Create a three-dimensional array and display specific elements, rows, and columns.
14. Create a student data frame and perform sorting, filtering, subsetting, and column-wise calculations.
15. Create nominal and ordinal factors, modify factor levels, and convert factors into character and numeric formats.
16. Write an R program using if-else and nested if-else statements to determine grades based on student marks.
17. Write R programs using for, while, and repeat loops to generate multiplication tables and factorials.
18. Perform string concatenation, substring extraction, case conversion, and string length calculation.
19. Create a dataset with missing values and use is.na() and na.omit() to identify and remove missing entries.
20. Read data from a CSV file, perform basic analysis, and export the processed data to a new CSV file.
21. Create a histogram, bar chart, boxplot, and pie chart using sample datasets.
22. Create scatter plots, bar plots, and histograms with appropriate labels and themes using ggplot2.
23. Calculate mean, median, mode, variance, and standard deviation. Perform a t-test and Chi-Square test on a given dataset and interpret the results.
24. Write an R Program for "Hello World ".
25. Write an R Program to Add Two Vectors
26. Find the Sum, Mean and Product of the Vector in R Programming
27. Create an R Program to Take Input From the User
28. Create an R Program to Find the Minimum and Maximum
29. How to create R Multiplication Table
30. Write a R Program to Check Prime Number
31. Write a R Program to check Armstrong Number
32. Write a R Program to Print the Fibonacci Sequence
33. Write a R Program to Check for Leap Year
34. Write a R Program to Check if a Number is Odd or Even in R Programming
35. Write a R Program to print the elements of a vector.