# FYUGP Data Science

**Course Outline for Semester IV**

| S. No. | Category | Course Code | Course Title | Credits | L | T | P | S | Hours per week |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Major | DS250MJ | Algorithms and Data Structures | 4 | 3 | 0 | 2 | 0 | 5 |
| | | DS251MJ | Data Engineering with Python | 4 | 3 | 0 | 2 | 0 | 5 |
| | | DS252MJ | Theory of Statistics | 4 | 4 | 0 | 0 | 0 | 4 |
| | | DS253MJ | Linear Regression and ANOVA | 4 | 4 | 0 | 0 | 0 | 4 |
| 2 | Minor 4 | MTHS254MN | R Programming for Data Analytics | 4 | 2 | 0 | 4 | 0 | 6 |

**FYUGP Data Science**

| **Course Title:** Algorithms and Data Structures | **L** | **T** | **P** | **S** | **Semester: 4th** |
|---|---|---|---|---|---|
| **Course Code:** DS250MJ | **4** | **x** | **1** | **x** | **Max Marks**: 100 |
| **Credits:** 4 | | | | | |

**Course Objectives:** The course aims to introduce the fundamental principles of algorithms and data structures, develop the ability to analyze algorithmic efficiency in terms of time and space complexity, provide practical skills for implementing algorithms using C and Python, and prepare students to apply data structures and algorithms in solving real-world computational problems.

**Course Outcomes:** By the end of this course, students will be able to:

1. Explain the principles of algorithm design and complexity analysis.

2. Implement and use fundamental data structures such as arrays, linked lists, stacks, queues, trees, and graphs.

3. Apply sorting, searching, and traversal algorithms efficiently in different contexts.

4. Design solutions to computational problems by integrating suitable algorithms and data structures.

**Unit I** – Introduction to Algorithms Definition and characteristics of algorithms, performance analysis (time and space complexity), asymptotic notations (Big-O, Big-Ω, Big-Θ), and iterative and recursive algorithms.

**Unit II** – Linear Data Structures Arrays and linked lists: representation, operations, and applications; Stacks and queues: implementation using arrays and linked lists; circular queues, priority queues, and dequeues.

**Unit III** – Non-Linear Data Structures Trees: binary trees, binary search trees, AVL trees, tree traversals (inorder, preorder, postorder); Heaps and priority queues; Graphs: representation (adjacency matrix and list), graph traversal algorithms (BFS, DFS).

**Unit IV** – Searching and Sorting Algorithms Searching methods: linear search and binary search. Sorting techniques: bubble sort, selection sort, insertion sort, merge sort, quick sort, and heap sort. Graph algorithms: shortest path (Dijkstra's and Bellman-Ford), and minimum spanning tree (Prim's and Kruskal's).

**Unit V** – Advanced Algorithm Concepts Algorithm design strategies: greedy method, divide and conquer, dynamic programming, backtracking, branch and bound. Introduction to complexity classes: P, NP, NP-Hard, and NP-Complete problems.

**Suggested Readings / References**

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, MIT Press.

2. Ellis Horowitz, Sartaj Sahni, Fundamentals of Data Structures.

3. Mark Allen Weiss, Data Structures and Algorithm Analysis in C/Python.

4. Official documentation of Python data structures and libraries.

**FYUGP Data Science**

| **Course Title:** Data Engineering with Python | **L** | **T** | **P** | **S** | **Semester: 4th** |
|---|---|---|---|---|---|
| **Course Code:** DS251MJ | **4** | **x** | **1** | **x** | **Max Marks**: 100 |
| **Credits:** 4 | | | | | |

**Course Objectives:** To introduce the concepts, architecture, and components of data engineering; develop skills in data collection, processing, storage, and management using Python; and prepare students to build data pipelines supporting analytics and machine learning workflows.

**Course Outcomes:** By the end of this course, students will be able to:

1. Explain the concepts, workflows, and tools of data engineering, including ETL/ELT processes, data sources, and storage systems.
2. Handle text and file-based data (CSV, JSON, XML) using Python modules, regular expressions, APIs, and web scraping techniques.
3. Use NumPy and pandas to process, analyze, and transform numeric and tabular data, including cleaning, filtering, merging, and handling missing values.
4. Design and implement end-to-end data engineering pipelines in Python that integrate file handling, numeric computation, database operations, and data quality practices.

**Unit I:** Role of data engineering in the data lifecycle, Data sources: structured, semi-structured, and unstructured data, ETL and ELT processes, Batch processing vs. stream processing, Data storage systems, Apache Airflow.

**Unit:** File handling (text, CSV, JSON, XML), OS, pathlib, and Glob modules, Regular Expressions, Relational databases, CRUD operations using Python, Connecting to APIs, web scraping (requests, BeautifulSoup)

**Unit III:** Understanding ndarrays, Data Types and Attributes, Array Creation and Properties, Indexing and Slicing, Array Mathematics (Addition, Subtraction, Scalar Multiplication)

**Unit IV:** Pandas: Series and DataFrames, Data Importing and Exporting, Data Cleaning and Preparation, Data Manipulation (Indexing, Selection, Filtering), Working with Missing Data, Merging and Joining DataFrames.

**Unit V:** Practicals which cover file handling, text and numeric data processing, relational database operations, data cleaning and manipulation using pandas, and designing simple ETL workflows that integrate Python libraries for data engineering tasks.

**TextBooks / References**


1. Joe Reis, Matt Housley, *Fundamentals of Data Engineering*, O'Reilly.
2. Jacqueline Kazil, Katharine Jarmul, *Data Wrangling with Python*, O'Reilly.
3. Wes McKinney, *Python for Data Analysis*, O'Reilly.
4. Tyler Akidau et al., *Streaming Systems*, O'Reilly.
5. Official documentation of Pandas, NumPy, SQLAlchemy, Airflow, and PySpark.

**FYUGP Data Science**

| **Course Title:** Theory of Statistics | **L** | **T** | **P** | **S** | **Semester: 4th** |
|---|---|---|---|---|---|
| **Course Code: DS252MJ** | **4** | **x** | **x** | **x** | **Max Marks**: 100 |
| **Credits:** 4 | | | | | |

**Course Objective:** This course covers concepts in various aspects of classical theoretical statistics. Students will be introduced to descriptive statistics, visualizations, classical estimation, hypothesis testing. Focus will be on the theoretical foundations of concepts.

**Course Outcomes:** After completion of this course student will able to

1. Understand the basic concepts of statistical data and its representation through various diagrammatic and graphical tools.
2. Apply appropriate statistical measures to summarize and describe data effectively.
3. Assess relationships and associations between two or more variables using suitable statistical methods.
4. Apply and interpret different statistical tests for hypothesis testing in real-world situations.

**Unit-I:** Statistics a conceptual frame work, statistical enquiry, various types of data. Collection of data, methods of sampling: Simple random sampling with (and without) replacement, stratified sampling, cluster sampling & systematic sampling, classification and tabulation of data. Visualizing (univariate and multivariate): boxplots, histogram, stem and leaf plot, bar graph, ogive, scatterplot, side-by-side boxplot. Measures of Central Tendency: Means (arithmetic, geometric, harmonic), median, mode. Measures of dispersion-range, mean deviation, quartile deviation Standard deviation and variance.

**Unit-II:** Measure of skewness- Karl-Pearson's and Bowley's methods. Measure of Kurtosis. Sample $r^{th}$ moment (raw and center moments). Measures of association: consistency and independence of data with special reference to attributes. Karl Pearsons's correlation coefficient: simple, partial and multiple correlations. Covariance matrix, correlation matrix. Kendall's $\tau$, Spearman rank correlation. Concurrent deviation methods. Probable error (ungrouped data), coefficient of determination.

**Unit-III**: Population and sample; parameter and statistics; Sampling distributions, Sampling distribution of mean and proportions. Chi-square, t and F distributions. The concept of estimation, estimator and estimate. Criteria of a good estimator: unbiasedness, consistency, efficiency and sufficiency. Methods of estimation: maximum likelihood estimator, method of moments

**Unit-IV:** Hypothesis testing, general procedure and errors in hypothesis testing, hypothesis testing for population parameters with large and small samples, Hypothesis testing based on Z, F and t-distribution. Chi-Square test for goodness of fit, chi-square test for population variances, chi-square test for association.
Non-parametric tests. One-sample problem: Sign test, signed rank test, Kolmogrov-Smirnov test, Test of independence (run test). Two sample problem: Sign test, Wilcoxon Mann-Whitney test, Median test, Kolmogrov-Smirnov test, run test.

**Text Books/ References:**

1. Hogg, R. V., McKean, J., Craig, A. T. (2005). Introduction to Mathematical Statistics. Pearson Education.
2. Rohatgi, V. K., Saleh, A. M. E. (2015). An introduction to probability and statistics. John Wiley and Sons.
3. Kale, B.K. (2005) :A First Course on Parametric Inference, Alpha Science
4. Deshpande, J. V, Naik-Nimbalkar, U., Dewan, I. (2017): Nonparametric statistics: theory and methods. New Jersey : World Scientific.
5. Casella, G., Berger, R. L. (2021). Statistical inference. Cengage Learning.

**FYUGP Data Science**

| **Course Title:** Linear Regression and ANOVA | **L** | **T** | **P** | **S** | **Semester: 4th** |
|---|---|---|---|---|---|
| **Course Code:** DS253MJ | **4** | **x** | **x** | **x** | **Max Marks**: 100 |
| **Credits:** 4 | | | | | |

**Course Objectives:** This course introduces the principles and methods of statistical modelling for applications across diverse fields, covering both theoretical foundations of regression and ANOVA, and techniques for analysing real-world datasets

**Course Outcomes:** By the end of this course, students will be able to:

1. Apply and interpret simple and multiple linear regression models using least squares and maximum likelihood methods.
2. Perform hypothesis testing, construct confidence intervals, and use ANOVA to assess regression models.
3. Diagnose and address multicollinearity, outliers, leverage points, and assumption violations, including design-based ANOVA methods.
4. Formulate and analyze generalized linear models such as logistic and Poisson regression for real-world data.

**Unit I:** Introduction to simple and multiple linear regression models; estimation of the parameters using least squares and maximum likelihood estimation methods and their properties; distribution of response variable and regression coefficient estimators, unbiased estimator for error variance, variance-covariance matrix of parameter vector; practical examples based on simple and multiple linear regression models.

**Unit II:** Testing of hypotheses and confidence intervals for regression coefficients, global significance of simple/multiple regression models using analysis of variance; residual analysis and regression diagnostics: detecting and dealing with outliers, hat matrix diagonals (in connection with leverage points); departures from underlying assumptions- diagnosis and remedies.

**Unit III:** Implication of multicollinearity; diagnostics for multicollinearity: VIF and variance decomposition methods, variable selection, under and over-fitting problems; introduction and principles of designs, CRD and RBD- methodology and development of analysis of variance, real life examples based on CRD and RBD.

**Unit IV:** Generalized Linear Models- Introduction to GLM: systematic and random components, link functions, maximum likelihood estimation: iteratively re-weighted least squares, logistic regression for binary data, Poisson regression for count data. Applications of binary logistic and count regression models.

**Text Books/ Reference Books:**


1. Montgomery, D. C., Peck, E. A. and Vining, G. G. (2012) Introduction to Linear Regression Analysis, Wiley Series in Probability and Statistics, Wiley, United States.
2. Seber G. A. F. and Lee, A. J. (2003) Linear Regression Analysis, WileySeries in Probability and Statistics, Wiley, United States.
3. Draper, N. R. and Smith, H. (1998) Applied Regression Analysis, WileySeries in Probability and Statistics, Wiley, United States.

4. Sengupta, D. and Jammalamadaka, S. R. (2003) Linear Models: An Integrated Approach, World Scientific, Singapore.
5. Vinod, H. D. and Ullah, A. (1981) Recent Advances in Regression Methods, M. Dekker, New York, United State

**FYUGP Data Science**

| **Course Title:** R Programming for Data Analytics | **L** | **T** | **P** | **S** | **Semester: 4th** |
|---|---|---|---|---|---|
| **Course Code:** MTHS254MN | **2** | **x** | **4** | **x** | **Max Marks**: 100 |
| **Credits:** 4 | | | | | |

**Course Objectives:** The objective of this course is to provide students with practical knowledge and skills in R programming for data analysis and statistical computing. It aims to develop the ability to write R scripts, manipulate data, perform descriptive and inferential statistical analyses, and create effective data visualizations. By the end of the course, students will be capable of using R for real-world data analysis and research applications.

**Course Outcomes:** After completing this course, students will be able to:

1. Understand the R environment and basic R syntax for statistical computing and data management.
2. Work with various data structures in R such as vectors, matrices, lists, and data frames, and perform data import/export operations.
3. Apply descriptive statistical methods and data manipulation techniques using built-in R functions and apply-family functions.
4. Develop reusable R scripts and functions for solving practical data analysis problems.

**Unit I:** Fundamentals of R programming; overview of R and R Studio environment; using R as a calculator, assignment operator, object name rules, basic operations on objects; vectors and data frames, subsetting vectors and data frames using indices, logical conditions, names, and other subset functions; applying basic mathematical functions on data frames for summarization and transformation.

**Unit II:** Working with matrices, arrays, and lists, indexing, and operations, subsetting using indices and logical conditions; applying basic functions for summarization and transformation; setting and managing the working directory; importing data from various file formats such as CSV and Excel; exporting processed data to external files; exploring and analyzing default datasets available in R for practice and demonstration.

**Unit III:** Writing and calling functions in R; creating basic graphics using base plotting system; descriptive statistics-mean, median, variance, standard deviation, frequency tables, cross-tabulations, proportion tables, correlation and simple linear regression analysis using R.

**Unit IV:** Data manipulation using apply(), lapply(), sapply(), and tapply(), random number generation from different distributions, data cleaning and transformation, checking normality, handling outliers, preparing data for statistical modeling and visualization, swirl() package for interactive learning.=

**Text Books / Reference Books:**


1. Matloff, N. (2016). The art of R programming: A tour of statistical software design. No Starch Press.
2. Gardener, M. (2017). Beginning R: The statistical programming language, Wiley.
3. Cotton, R., Learning R: a step by step function guide to data analysis. 1st edition. O'reilly Media Inc.
4. Lawrence, M., & Verzani, J. (2016). Programming Graphical User Interfaces in R. CRC press. (ebook