ATMANIRBHAR BHARAT
SWAYAMPURNA GOA

गोंय विद्यापीठ
ताळगांव पठार,
गोंय –४०३ २०६
फोन : +९१–८६६९६०९०४८

(Accredited by NAAC)

Goa University
Taleigao Plateau, Goa-403 206
Tel : +91-8669609048
Email : registrar@unigoa.ac.in
Website : www.unigoa.ac.in

GU/Acad –PG/BoS -NEP/2024/475 Date: 30.08.2024

CIRCULAR

Ref. No.: GU/Acad –PG/BoS -NEP/2023/184/3 dated 04.07.2023

In supersession to the above referred Circular, the approved Semester I to IV Syllabus of the **Master of Science in Data Science** Programme is enclosed.

The Dean/ Vice-Deans of the Goa Business School are requested to take note of the above and bring the contents of the Circular to the notice of all concerned.

ASHWIN VYAS LAWANDE
Digitally signed by ASHWIN VYAS LAWANDE
Date: 2024.08.30 11:49:00 +05'30'

(Ashwin V. Lawande)
Deputy Registrar – Academic

To,
1. The Dean, Goa Business School, Goa University.
2. The Vice-Deans, Goa Business School, Goa University.

Copy to,
1. The Chairperson, BOS in Computer Science and Technology.
2. The Programme Director, Data Science, Goa University.
3. The Controller of Examinations, Goa University.
4. The Assistant Registrar, PG Examinations, Goa University.
5. Directorate of Internal Quality Assurance, Goa University for uploading the Syllabus on the University website.

M.Sc. in DATA SCIENCE

(To be effective from Academic Year 2023-24)

Programme Specific Outcomes:

PSO1: Know fundamental statistics, mathematics, and computer science concepts and be aware of various data analysis and data science software tools.

PSO2: Cultivate critical thinking skills to create data science-enabled solutions.

PSO3: Develop proficiency in implementing machine learning algorithms and models to address real-world challenges across diverse domains.

PSO4: Foster a research-oriented mindset, enabling the formulation of hypotheses, experimentation, and analysis to address emerging research challenges in data science.

PSO5: Apply data science expertise to contribute to addressing societal issues

M.Sc. IN DATA SCIENCE (TO BE EFFECTIVE FROM ACADEMIC YEAR 2023-24)		
SEMESTER I		
DISCIPLINE SPECIFIC CORE (DSC) COURSES		
Course Code	**Course Title**	**Credits**
CSD-500	Fundamentals of Data Science (Theory)	2
CSD-501	Fundamentals of Data Science (Practical)	2
CSD-502	Machine learning (Theory)	2
CSD-503	Machine learning (Practical)	2
CSD-504	Mathematical Foundations for Data Science (Theory)	2
CSD-505	Mathematical Foundations for Data Science (Practical)	2
CSD-506	Fundamentals of Artificial Intelligence (Theory)	2
CSD-507	Fundamentals of Artificial Intelligence (Practical)	2
	Total Credits	**16**
DISCIPLINE SPECIFIC ELECTIVE (DSE) COURSES – any one to be opted from the DSE list		
Course Code	**Course Title**	**Credits**
CSD-521	Domain-specific Predictive Analytics	4
CSD-522	Design Thinking for Data-Driven App Development*	4
	Total Credits	**4**

***offered as generic elective for other programs**

SEMESTER II		
DISCIPLINE SPECIFIC CORE (DSC) COURSES		
Course Code	**Course Title**	**Credits**
CSD-508	Reinforcement learning (Theory)	**2**
CSD-509	Reinforcement learning (Practical)	**2**
CSD-510	Optimization techniques	**4**
CSD-511	MLOps (Theory)	**2**
CSD-512	MLOps (Practical)	**2**
CSD-513	Software Engineering for AI Enabled systems (Theory)	**2**
CSD-514	Software Engineering for AI Enabled systems (Practical)	**2**
	Total Credits	**16**
DISCIPLINE SPECIFIC ELECTIVE (DSE) COURSES – any one to be opted from DSE list		
Course Code	**Course Title**	**Credits**
CSD-523	Signal processing	**4**
CSD-524	Regression Analytics and Predictive Models	**4**
CSD-525	Cloud Computing	**4**
CSD-526	Big Data Analytics	**4**
	Total Credits	**4**

SEMESTER III		
RESEARCH SPECIFIC ELECTIVE (RSE) COURSES – any two to be opted		
Course Code	**Course Title**	**Credits**
CSD-600	Research Methodology	**4**
CSD-601	Natural Language Processing	**4**
CSD-602	Deep Learning Models	**4**
CSD-603	Data Engineering	**4**
CSD-604	Programming Paradigm	**4**
	Total Credits	**8**
GENERIC ELECTIVE (GE) COURSES - total of 12 credits to be opted from GE list specified below		
List/Categories of Generic Elective (GE) Courses		
Course Code	**Course Title**	**Credits**
CSA-621	Corporate Skills offered by Computer Science	**4**
	Courses offered by other Disciplines from GBS during the respective Semester	**4**
	Courses offered by other Disciplines from other Schools during the respective Semester	**4**
	Courses offered under MOOC during the respective Semester and approved by DFC	**4**
	Total Credits	**12**

SEMESTER IV		
ONE RESEARCH SPECIFIC ELECTIVE (RSE) Course to be opted from the list in consultation with the Mentor/Research supervisor. It can be completed in Semester 3.		
Course Code	**Course Title**	**Credits**
CSD-605	Internet of Things	**4**
CSD-606	Speech Processing	**4**
CSD-607	Web Analytics	**4**
CSD-608	Financial Machine Learning	**4**
CSD-609	Recommender Systems	**4**
	Total Credits	**4**
Dissertation		**Credits**
Course Code	**Course Title**	**Credits**
CSD-651	Research Project on Data Science in Academic or Research Institutes or Industry	**16**
	Total Credits	**16**

SEMESTER I
DISCIPLINE SPECIFIC CORE (DSC) COURSES

Name of the Programme	**: MSc. in Data Science**
Course Code	**: CSD-500**
Title of the Course	**: Fundamentals of Data Science (Theory)**
Number of Credits	**: 2(2L-0T- 0P)**
Contact hours	**: 30 hours (30L-0T-0P)**
Effective from AY	**: 2023-24**

Pre-requisites for the course	Statistics and probability theory and python programming	
Objectives	The objective is to gain a comprehensive understanding of data science, covering fundamental concepts, tools, and techniques.	
Content	**Unit I:** Introduction: Typology of problems - Data science in a big data world: Benefits and uses of data science and big data-Facets of data-The data science process-The big data ecosystem and data science-The data science process: Overview of the data science process- Defining research goals and creating a project charter-Retrieving data-Cleansing, integrating, and transforming data-Exploratory data analysis-Build the models- Presenting findings and building applications on top of them. Mathematics for Data Science – A quick Review: Importance of linear algebra, statistics and optimization from a data science perspective; Structured thinking for solving data science problems. Linear Algebra: Matrices and their properties (determinants, traces, rank, nullity, etc.); Eigenvalues and eigenvectors; Matrix factorizations; Inner products; Distance measures; Projections; Notion of hyperplanes; half-planes. Probability, Statistics and Random Processes: Probability theory and axioms; Random variables; Probability distributions and density functions (univariate and multivariate); Expectations and moments; Covariance and correlation; Statistics and sampling distributions; Hypothesis testing of means, proportions, variances and correlations; Confidence (statistical) intervals; Correlation functions; White-noise process. Data clearing (EDA) Introduction to Data Science Methods: Linear regression as an exemplar function approximation problem; Linear classification problems-PCA	**15 hours**

	Unit II Handling large data on a single computer - The problems you face when handling large data-General techniques for handling large volumes of data-General programming tips for dealing with large data sets - Case study 1: Predicting malicious URLs - First steps in big data-Distributing data storage and processing with frameworks Introduction to NoSQL The rise of graph databases Introducing connected data and graph databases Introducing Neo4j: a graph database Data visualization to the end user Data visualization options Cross filter, the JavaScript MapReduce library Creating an interactive dashboard with dc.js Dashboard development tools	**15 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References / Readings	1. Baesens, B. (2014). Analytics in a big data world: The essential guide to data science and its applications. John Wiley & Sons. 2. Bruce, P., Bruce, A., & Gedeck, P. (2020). Practical statistics for data scientists: 50+ essential concepts using R and Python. O'Reilly Media. 3. Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer. 4. McKinney, W. (2022). Python for data analysis. " O'Reilly Media, Inc.". 5. Taddy, M. (2019). Business data science. 6. Wheelan, C. Naked Statistics: Stripping the Dread from the Data.	
Course Outcomes	1. Understanding of fundamental concepts and techniques in data science. 2. Proficiency in data manipulation, analysis, and visualization using tools like Python or R. 3. Introduction to machine learning algorithms and evaluation methods. 4. Awareness of ethical considerations and responsible practices in data science.	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course code : **CSD-501**
Title of the course : **Fundamentals of Data Science (Practical)**
Number of credits : **2(0L-0T-2P)**
Total contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Basic programming skills, Statistics	
Course Objectives	The course aims to provide an introduction to the fundamental processes of data science using Python and Jupyter notebooks, enabling participants to manipulate and analyze uncurated datasets, apply basic statistical analysis and machine learning methods, and effectively visualize the results.	
Content	1. Create a Jupyter notebook and import the numpy library. Generate a 2D numpy array of size 5x5 filled with random integers between 1 and 100. Perform the following operations: a. Calculate the mean and standard deviation of the array. b. Find the sum of all elements in the array. c. Reshape the array into a 1D array and compute the median. 2. Download a dataset from an online source (e.g., Kaggle or UCI Machine Learning Repository) and load it into a pandas DataFrame. Perform the following tasks: a. Display the first five rows of the dataset. b. Check for missing values and handle them appropriately. c. Calculate summary statistics for numerical columns. d. Plot a histogram of one of the numerical variables. 3. Download a messy dataset containing missing values, duplicates, and inconsistent formatting. Use pandas to clean and prepare the data by: a. Handling missing values through imputation or removal. b. Identifying and removing duplicate entries. c. Standardizing formatting across columns (e.g., converting strings to lowercase). 4. Choose a dataset of your interest and create visualizations to explore its characteristics. Tasks include: a. Plotting a line chart to visualize the trend of a numerical variable over time (if applicable). b. Creating a scatter plot to examine the relationship between two numerical variables. c. Generating a bar chart or pie chart to display categorical data. 5. Apply machine learning techniques to analyze a dataset and make predictions. Tasks include: a. Preprocessing the data using numpy and pandas (e.g., handling missing values, encoding categorical variables). b. Splitting the dataset into training and testing sets.	**60 hours**

	c. Building and training a machine learning model using scikit-learn. d. Evaluating the model's performance using appropriate metrics (e.g., accuracy, precision, recall). 6. Select a dataset suitable for a classification or regression task. Apply machine learning techniques using scikit-learn to build and evaluate a predictive model. Requirements: a. Preprocess the data, including feature scaling and handling categorical variables. b. Split the dataset into training and testing sets. c. Choose an appropriate algorithm (e.g., decision tree, logistic regression) and train the model. d. Evaluate the model's performance using relevant metrics (e.g., accuracy, precision, recall). 7. Access a text dataset (e.g., movie reviews, news articles) and perform basic text analysis using NLTK. Requirements: a. Preprocess the text data by tokenizing, removing stopwords, b. and stemming or lemmatizing. c. Analyze the frequency of words and visualize the most common terms using word clouds or bar charts. d. Apply sentiment analysis to categorize the text into positive, negative, or neutral sentiments. 8. Connect to a database (e.g., SQLite, MySQL) using Python and perform basic operations. Requirements: a. Establish a connection to the database and retrieve data from one or more tables. b. Execute CRUD operations (Create, Read, Update, Delete) on the database using SQL queries or Python libraries (e.g., SQLAlchemy). c. Perform simple data analysis or visualization on the retrieved data. 9. Choose a dataset of interest and perform an end-to-end data analysis project, showcasing all your skills.	
Pedagogy	Tutorials/ Lab assignments/ Project work	
References/ Readings	1. Baesens, B. (2014). Analytics in a big data world: The essential guide to data science and its applications. John Wiley & Sons 2. Bruce, P., Bruce, A., & Gedeck, P. (2020). Practical statistics for data scientists: 50+ essential concepts using R and Python. O'Reilly Media. 3. Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer. 4. McKinney, W. (2022). Python for data analysis. " O'Reilly Media, Inc.". 5. Taddy, M. (2019). Business data science.	

	6. Wheelan, C. Naked Statistics: Stripping the Dread from the Data.
Course Outcomes	1. Practical data analysis skills using data science tools. 2. Hands-on experience with real-world data projects. 3. Collaboration and teamwork in interdisciplinary settings. 4. Ethical considerations and responsible practices in data science 5. Experimentation and evaluation of data science techniques.

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-502**
Title of the Course : **Machine Learning (Theory)**
Number of Credits : **2(2L-0T-0P)**
Total Contact Hours : **30 hours (30L-0T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Familiarity with linear algebra, statistics & probability theory	
Course Objectives:	This course provides students with 1. In-depth introduction to three main areas of Machine Learning: supervised and unsupervised and reinforcement learning. 2. This course will cover some of the main models and algorithms for regression, classification, clustering and Markov decision processes. Topics will include linear and logistic regression, regularisation, SVMs and kernel methods, ANNs, clustering, and dimensionality reduction, sequential learning Like HMM and deep learning CNN and RNN	
Content:	**Unit 1:** **Introduction:** well posed learning problem, designing a learning system, perspectives and issues in machine learning- types of learning - supervised, unsupervised and reinforcement learning **Concept learning:** concept learning task, notation, inductive learning hypothesis, concept learning as search, version space and candidate elimination algorithm, decision tree, random forest. **Linear regression:** logistic regression-Support vector machine kernel, Model selection and feature selection-Ensemble methods: Bagging, boosting, Evaluating and debugging learning algorithms. **Continuous Latent Variables:** Principal Component Analysis, Maximum variance formulation, Minimum error formulation, Applications of PCA, PCA for high-dimensional data. **Neural Networks:** -Feed-forward Network, Functions, perceptron, -Weight-space symmetries, Network Training, Parameter optimization, Local quadratic approximation, Use of gradient information, Gradient descent optimization, Error Backpropagation, Evaluation of error-function derivatives, Efficiency of backpropagation.	**15 hours**
	Unit 2: **Deep learning:** Deep Feedforward Networks, Gradient-Based Learning, Hidden Units, -Architecture Design, CNN and RNN (simple RNN and LSTM). **Unsupervised learning;** Clustering, K-means, EM.Mixture of Gaussians. **Sequential Data:** Markov Models, Hidden Markov Models, Maximum likelihood for the HMM, The forward-backward algorithm, The sum-product algorithm for the HMM, Scaling	**15 hours**

	factors, -The Viterbi algorithm. **Reinforcement learning:** introduction- learning task-Q learning, non-deterministic rewards and actions-temporal difference learning.	
Pedagogy:	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	Main Reading:- 1. Alpaydin, E. (2020). Introduction to machine learning. MIT press. 2. Bishop, C. M. (2006). Pattern recognition and machine learning: springer New York.. 3. Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. Cambridge university press. 4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press. 5. Hart, Peter E., David G. Stork, and Richard O. Duda.(2000) Pattern classification. Hoboken: Wiley, 2000. 6. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.	
Course Outcomes	1. Develop an appreciation for what is involved in learning from data. 2. Understand a wide variety of learning algorithms. 3. Understand how to apply a variety of learning algorithms to data. 4. Understand how to perform evaluation of learning algorithms and model selection and Have a basic understanding of deep learning.	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-503**
Title of the Course : **Machine Learning (Practical)**
Number of Credits : **2(0L-0T-2P)**
Total Contact Hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Machine learning theory and programming in python	
Course Objective:	This course aimed at imparting implementation of machine learning algorithms using python and its APIs	
Content:	**Suggested Lab assignments/work with respect to the following using python (scikit /keras libraries) /amazon sage maker/matlab toolbox - each assignment with duration of 4 hrs. and 8 hrs. for project work**	
	1. Write a program to implement version space.	**5 hours**
	2. Write a program to implement a decision tree for given data.	**5 hours**
	3. Write a program to implement linear regression for given data.	**5 hours**
	4. Write a program to implement logistic regression.	**5 hours**
	5. Write a program to implement SVM.	**5 hours**
	6. Write a program to implement perceptron.	**5 hours**
	7. Write a program to implement a multilayer perceptron.	**5 hours**
	8. Write a program to implement RNN.	**5 hours**
	9. Write a program to implement CNN.	**5 hours**
	10. Write a program to implement HMM.	**5 hours**
	Capstone Mini Project work to assess the overall learning.	**10 hours**
Pedagogy:	Lab Assignments / Mini Project	

References/ Readings	Main Reading:- 1. Alpaydin, E. (2020). Introduction to machine learning. MIT press. 2. Bishop, C. M. (2006). Pattern recognition and machine learning: springer New York. 3. Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. Cambridge university press. 4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press. 5. Hart, Peter E., David G. Stork, and Richard O. Duda.(2000) Pattern classification. Hoboken: Wiley, 2000. 6. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.
Course Outcomes	1. Practical implementation skills of machine learning algorithms. 2. Model development, evaluation, and feature engineering techniques. 3. Interpretability and explainability of machine learning models. 4. Awareness of ethical considerations in machine learning.

(Back to Index)

Name of the Programme : **MSc. in Data Science**
Course code : **CSD-504**
Title of the course : **Mathematics Foundation for Data Science (Theory)**
Number of credits : **2 (2L-0T-0P)**
Total contact hours : **30 hours (30L-0T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Basic mathematics	
Course Objectives	1. To build a strong foundation in maths required for learning computer science/data science subjects. 2. To understand fundamental concepts and tools in calculus, linear algebra etc. with emphasis on their applications to computer science in particular data science/machine learning	
Content	**Unit1:** **Introduction** Importance of mathematics and their applications for computer science/machine learning/data science/deep learning Functions, variables, equations, graphs revision **Probability and Statistics:** Probability Rules & Axioms, Bayes' Theorem, Random Variables, Variance and Expectation, Conditional and Joint Distributions, Standard Distributions (Bernoulli, Binomial, Multinomial, Uniform and Gaussian), Moment Generating Functions, Maximum Likelihood Estimation (MLE), Prior and Posterior, Maximum a Posteriori Estimation (MAP) and Sampling Methods-confidence intervals, Hypothesis testing, p-values, A/B testing-ANOVA, t-test, Linear regression, regularization **Calculus** Overview of Differential and Integral Calculus, Partial Derivatives Product and chain rule-Taylor's series, infinite series summation/integration concepts-Fundamental and mean value-theorems of integral calculus, evaluation of definite and improper integrals-Beta and Gamma functions, Functions of multiple variables, limit, continuity, partial derivatives-Basics of ordinary and partial differential equations -Applications of Calculus	**15 hours**
	Unit 2: **Linear Algebra:** Systems of Linear Equations-Matrices-Solving Systems of Linear Equations-Vector Spaces-Linear Independence-Basis and Rank-Linear Mappings Affine Spaces **Analytic Geometry** Norms-(Inner Products-Lengths and Distances	**15 hours**

	Angles and Orthogonality-Orthonormal Basis Orthogonal Complement-Inner Product of Functions-Orthogonal Projections-Rotations) - Eigen value decomposition and SVD **Optimization** Differentiation of Univariate Functions-Partial Differentiation and Gradients-Gradients of Vector-Valued Functions-Gradients of Matrices Useful Identities for Computing Gradients-Backpropagation and Automatic Differentiation-Higher-Order Derivatives-Linearization and Multivariate Taylor Series-Gradient Descent-Constrained Optimization -Lagrange Multipliers-Convex Optimization,	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Gel'fand, I. M., Glagoleva, E. G., & Shnol, E. E. E. (1990). Functions and graphs (Vol. 1). Springer Science & Business Media. 2. Lay, D. C. (2003). Linear algebra and its applications. Pearson Education India. 3. McClave, J. T., Benson, P. G., & Sincich, T. (2008). Statistics for business and economics. Pearson Education. 4. Sternstein, M. (2017). Barron's AP statistics. Simon and Schuster. 5. Strang, G. (2022). Introduction to linear algebra. Wellesley-Cambridge Press. 6. Wheelan, C. (2013). Naked statistics: Stripping the dread from the data. WW Norton & Company. 7. Witte, R. S., & Witte, J. S. (2017). Statistics. John Wiley & Sons.	
Course Outcomes	1. Strong understanding of mathematical concepts relevant to data science, including linear algebra, calculus, probability theory, and statistics. 2. Ability to apply mathematical principles to solve data science problems, such as dimensionality reduction, optimization, and uncertainty modeling. 3. Proficiency in mathematical modeling techniques and algorithms used in data science, such as regression, clustering, and classification. 4. Development of mathematical reasoning and problem-solving skills for analyzing and interpreting data, formulating mathematical solutions, and communicating results.	

(Back to Index)

Name of the Programme : **MSc. in Data Science**
Course code : **CSD-505**
Title of the course : **Mathematical Foundation for Data Science (Practical)**
Number of credits : **2 (0L-0T-2P)**
Total contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Mathematical foundation theory and programming background	
Course Objectives	The lab assignments are aimed at demonstrating of the following regarding statistics	
Content	Recap of the following – A. NumPy is a third-party library for numerical computing, optimized for working with single- and multi-dimensional arrays. Its primary type is the array type called ndarray. This library contains many routines for statistical analysis. B. SciPy is a third-party library for scientific computing based on NumPy. It offers additional functionality compared to NumPy, including scipy.stats for statistical analysis. C. Pandas is a third-party library for numerical computing based on NumPy. It excels in handling labelled one-dimensional (1D) data with Series objects and two-dimensional (2D) data with Data Frame objects. D. Matplotlib is a third-party library for data visualization. It works well in combination with NumPy, SciPy, and Pandas.	**3 hours**
	Assignment 1 - Write program to implement the EDA concepts using python libraries -Numpy,Pandas, matplotlib, seaborn,scipy, scrapy and beautiful soup, and tensor flow ,keras and pytorch etc.	**3 hours**
	Assignment -2 - Sampling, Variables in Statistics, Frequency Distributions. Generate frequency distribution tables, Generate grouped frequency distribution tables and -Visualizing Frequency Distributions -Generate bar plots, pie charts, and histograms, Employ bar plots, pie charts and histograms.	**6 hours**
	Assignment-3-Comparing Frequency Distributions -grouped bar plots- step-type histogram-kernel density estimate plots- strip plots and box plots	**6 hours**
	Assignment-4 -Multidimensional image operations, Solving differential equations and the Fourier transform using scipy	**6 hours**
	Assignment-5 -Optimization algorithms using scipy.	**6 hours**
	Assignment -6 -Linear algebra using scipy	**6 hours**

	Assignment- 7-Program in python to implement the concepts such as Vector space, subspace, span, column space, row space, null space, left-null space, rank, basis, orthogonal matrix, symmetric matrix.	**6 hours**
	Assignment -8 – Implement Eigen value decomposition in python.	**6 hours**
	Assignment-9 – implement SVD using python.	**6 hours**
	Assignment -10 – implements some of optimization algorithm using the python library.	**6 hours**
Pedagogy	lab assignments /Project	
References/ Readings	1. Gel'fand, I. M., Glagoleva, E. G., & Shnol, E. E. E. (1990). Functions and graphs (Vol. 1). Springer Science & Business Media. 2. Lay, D. C. (2003). Linear algebra and its applications. Pearson Education India. 3. McClave, J. T., Benson, P. G., & Sincich, T. (2008). Statistics for business and economics. Pearson Education. 4. Sternstein, M. (2017). Barron's AP statistics. Simon and Schuster. 5. Strang, G. (2022). Introduction to linear algebra. Wellesley-Cambridge Press. 6. Wheelan, C. (2013). Naked statistics: Stripping the dread from the data. WW Norton & Company. 7. Witte, R. S., & Witte, J. S. (2017). Statistics. John Wiley & Sons.	
Course Outcomes	1. Practical application of mathematical concepts in data science. 2. Proficiency in using mathematical software and tools for data analysis. 3. Hands-on experience in data analysis and modeling using mathematical techniques. 4. Collaborative teamwork on data science projects involving mathematical foundations.	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course code : **CSD-506**
Title of the course : **Fundamentals of Artificial Intelligence (Theory)**
Number of credits : **2(2L-0T-0P)**
Total contact hours : **30 hours (30L-0T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Programming, probability and statistics and linear algebra	
Course Objectives	To develop a basic understanding of 1. Problem-solving 2. Knowledge representation 3. Reasoning and learning methods of AI.	
Content	**Unit 1:** Artificial Intelligence Introduction -Intelligent Agents, Problem-solving Solving Problems by Searching -Search in Complex Environments - Adversarial Search and Games- Constraint Satisfaction Problems Knowledge, reasoning, and planning Knowledge Representation-First-Order Predicate Logic - Unification Forward and Backward Chaining - Resolution - Ontological Engineering Categories and Objects - Events-Mental Events and Mental Objects - Reasoning Systems for Categories - Reasoning with Default Information Uncertain knowledge and reasoning Quantifying Uncertainty - Probabilistic Reasoning - Probabilistic Reasoning over Time Probabilistic Programming -Making Simple Decisions - Making Complex Decisions -Multiagent Decision Making Machine Learning, Learning from Examples - Learning Probabilistic Models - Deep Learning - Reinforcement Learning Communicating, perceiving, and acting	**15 hours**
	Unit 2: Natural Language Processing - Deep Learning for Natural Language Processing - Computer Vision - Robotics. Artificial Intelligence applications Language Models - Information Retrieval - Information Extraction Natural Language Processing - Machine Translation - Speech Recognition Robotics-Hardware and Software for Robots - Planning and Perception Explainable AI - Definitions and concepts such as black-box models, transparency, interpretable machine learning and explanations. - Decision-making and decision support Human-Computer Interaction (HCI) and AI. - Explainable AI. -	**15 hours**

	Methods for Explainable AI. - Applications and examples. - Trust and acceptance.-Evaluation methods and metrics. - Ethical, legal and social issues of explainable AI. Contemporary issues in AI-Philosophy, Ethics, and Safety of AI -The Future of AI	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. GF Luger, (2002). Artificial Intelligence, Pearson Education, 2002. 2. M.C. Trivedi, (2019). A Classical Approach to Artificial Intelligence, Khanna Book Publishing. 3. Nilsson, N. J. (1998). Artificial intelligence: a new synthesis. Morgan Kaufmann. 4. Padhy, N. P. (2005). Artificial intelligence and intelligent systems (Vol. 337). Oxford: Oxford University Press. 5. Russell, S. J., & Norvig, P. (2010). Artificial intelligence a modern approach. London. 6. V., Rich, E., Knight, K., & Nair, S. (2009). Artificial Intelligence. Tata McGraw Hill	
Course Outcomes	1. Understand the basic concepts and techniques of Artificial Intelligence. 2. Apply AI algorithms for solving practical problems. 3. Apply basics of Fuzzy logic and neural networks. 4. Explain Expert System and implementation.	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-507**
Title of the Course : **Fundamentals of Artificial Intelligence (Practical)**
Number of Credits : **2 (0L-0T-2P)**
Total Contact Hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Artificial Intelligence theory, probability and statistics, linear algebra, and Python programming	
Course Objectives:	To develop a basic understanding of 1. Problem solving 2. Knowledge representation 3. Reasoning and learning methods of AI 4. Implementing AI algorithms	
Content:	Assignment-1 -Real-world path planning for pedestrians. In the first part, students implement A* over a map that includes roads/paths as well as elevations. In the second part, students collect actual data through walking around the real world, and the cost model is then learned via regression techniques.	**10 hours**
	Assignment-2 -Solve maze via search -this assignment involves formulating maze-solving as a search problem, image processing (via OpenCV) as a step in maze-solving, as well as guided performance/quality analysis of representational parameters	**10 hours**
	Assignment 3-Within the context of an artificial intelligence course, students are taught to identify ethical issues within technical projects and to engage in moral problem solving with regard to such issues.	**10 hours**
	Assignment 4-Neural network for face recognition using tensor flow -build feedforward neural networks for face recognition using TensorFlow. Students then visualize the weights of the neural networks they train. The visualization allows students to understand feedforward one-hidden layer neural networks in terms of template matching, and allows students to explore overfitting.	**10 hours**
	Assignment -5 -Organic path finding -Students develop a “human-like” pathfinding technique by specializing a generic search algorithm with custom action cost and heuristic cost functions. Students apply classical search algorithms and reflect on example organic paths to achieve “human-like” pathfinding.	**10 hours**

	Assignment - 6 -Implement a genetic algorithm in Python to evolve strategies for Robby the Robot to collect empty soda cans that lie scattered around his rectangular grid world. And also Compare the performances of a brute-force search and a search employing the Minimum Remaining Values (MRV) heuristic in solving Sudoku puzzles.	**10 hours**
Pedagogy:	lectures/practical/ tutorials/assignments/self-study	
References /Readings:	1. GF Luger, (2002). Artificial Intelligence, Pearson Education, 2002. 2. M.C. Trivedi, (2019). A Classical Approach to Artificial Intelligence, Khanna Book Publishing. 3. Nilsson, N. J. (1998). Artificial intelligence: a new synthesis. Morgan Kaufmann. 4. Padhy, N. P. (2005). Artificial intelligence and intelligent systems (Vol. 337). Oxford: Oxford University Press. 5. Russell, S. J., & Norvig, P. (2010). Artificial intelligence a modern approach. London. 6. V., Rich, E., Knight, K., & Nair, S. (2009). Artificial Intelligence. Tata McGraw Hill.	
Course Outcomes:	1. Students will demonstrate a deep understanding of feedforward neural networks and the backpropagation algorithm. 2. Students will be able to extend an existing implementation of the backpropagation algorithm to recognize static hand gestures in images. 3. Students will learn digit recognition using the MNIST dataset, applying their knowledge of feedforward neural networks and backpropagation. 4. Implementation of Advanced Search Strategies in Game Playing.	

(Back to Index)

DISCIPLINE SPECIFIC ELECTIVE (DSE) COURSES

Name of the Programme	**: MSc in Data Science**
Course Code	**: CSD-521**
Title of the Course	**: Domain Specific Predictive Analytics**
Number of Credits	**: 4(2L-2T-0P)**
Contact hours	**: 60 hours (30L-30T-0P)**
Effective from AY	**: 2023-24**

Pre-requisites for the course	Data science fundamentals and programming background	
Course Objectives	The course introduces theoretical foundations, Algorithms, Methodologies for analysing data in various domains such Retail, Finance, Risk and Healthcare.	
Content for Theory	**Retail Analytics** Understanding Customer: Profiling and Segmentation, Modelling Churn. Modelling Lifetime Value, Modelling Risk, Market Basket Analysis. **Risk Analytics** Risk Management and Operational Hedging: An Overview, Supply Chain Risk Management, A Bayesian Framework for Supply Chain Risk Management, Credit Scoring and Bankruptcy Prediction **Financial Data Analytics** Financial News analytics: Framework, techniques, and metrics, News events impact market sentiment, Relating news analytics to stock returns **Financial Time Series Analytics** Financial Time Series and Their Characteristics, Common Financial Time Series models, Autoregressive models, Markov chain models, Time series models with leading indicators, Long term forecasting	**15 hours**
	Introduction Healthcare Analytics An Introduction to Healthcare Data Analytics, Electronic Health Records, Privacy-Preserving Data Publishing Methods in Healthcare, Clinical Decision Support Systems **Healthcare Data Analytics** Natural Language Processing and Data Mining for Clinical Text: Core NLP Components, Information Extraction and Named Entity Recognition, Social Media Analytics for Healthcare: Tracking of Infectious Disease Outbreaks, Readmission risk Prediction **Genomic Data Analytics** Microarray Data, Microarray Data Analysis , Genomic Data Analysis for Personalized Medicine , Patient Survival Prediction from Gene Expression Data , Genome Sequence Analysis	**15 hours**

Content for Tutorial Slots	Finance: a) Stock Market Prediction: Develop a predictive model to forecast stock prices based on historical data, using techniques such as time series analysis and machine learning algorithms.	**3 hours**
	b) Credit Risk Assessment: Build a model to predict the creditworthiness of individuals or businesses, incorporating relevant financial and non-financial factors to assess default probabilities.	**3 hours**
	c) Fraud Detection: Create an algorithm to identify fraudulent transactions or activities in financial systems by analysing patterns, anomalies, and historical data.	**3 hours**
	Medical Science: a) Disease Diagnosis: Develop a predictive model to diagnose diseases based on patient symptoms, medical history, and test results, using techniques like classification algorithms and medical data analysis.	**3 hours**
	b) Patient Readmission Prediction: Build a model to predict the likelihood of a patient being readmitted to the hospital within a certain time frame, considering factors such as demographics, medical conditions, and treatment history.	**3 hours**
	c) Drug Effectiveness Prediction: Create a model to predict the effectiveness of a particular drug for a specific patient or group of patients, utilizing genetic information, clinical data, and treatment outcomes.	**3 hours**
	Genomic Science: Predictive analytics in the domain of genomics can be highly beneficial for various applications, such as disease prediction, drug discovery, personalized medicine, and genetic engineering. Here are a few examples of predictive analytics techniques that can be applied in genomics a) Disease Risk Prediction: By analyzing an individual's genomic data, predictive analytics can be used to assess the risk of developing specific diseases. Machine learning algorithms can identify patterns and genetic markers associated with various diseases, allowing for early detection and preventive measures. For example, predictive models can be built to predict the risk of developing conditions like cancer, cardiovascular diseases, or genetic disorders.	**3 hours**

	b) Pharmacogenomics: Predictive analytics can aid in predicting an individual's response to specific drugs based on their genetic makeup. By analyzing genomic data along with clinical information, machine learning models can predict drug efficacy, potential side effects, and optimal dosage. This information can be used to develop personalized treatment plans and improve patient outcomes.	**3 hours**
	c) Genomic Variant Interpretation: Genomic variants play a crucial role in determining an individual's susceptibility to diseases. Predictive analytics can be used to interpret the functional consequences of these variants. Machine learning algorithms can predict the impact of genetic mutations on protein structure and function, helping researchers and clinicians understand the underlying mechanisms of diseases and develop targeted therapies.	**3 hours**
	d) Gene Expression Analysis: Predictive analytics can analyze gene expression data to identify patterns and correlations between genes and specific traits or diseases. By using machine learning algorithms, it is possible to predict gene expression levels based on genomic features and environmental factors. This can provide valuable insights into gene regulatory networks and help in understanding disease mechanisms and identifying potential therapeutic	**3 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Chapman, C., & Feit, E. M. (2015). R for marketing research and analytics (Vol. 67). New York, NY: Springer.. 2. Kouvelis, P., Dong, L., Boyabatli, O., & Li, R. (2011). Handbook of integrated risk management in global supply chains. John Wiley & Sons. 3. Reddy, C. K., & Aggarwal, C. C. (Eds.). (2015). Healthcare data analytics (Vol. 36). CRC Press.. 4. Rud, O. P. (2001). Data mining cookbook: modeling data for marketing, risk, and customer relationship management. John Wiley & Sons..	
Course Outcomes	1. Retail Analytics and Risk Analytics 2. Financial Data Analytics, Financial Time Series Analytics, 3. Healthcare Analytics, Healthcare Data Analytics and 4. Genomic Data Analytics.	

(Back to Index)

Name of the Programme : **MSc in Data Science**
Course Code : **CSD-522**
Title of the Course : **Design Thinking for Data-Driven App Development**
Number of Credits : **4(4L-0T-0P)**
Contact hours : **60 hours (60L-0T-0P)**
Effective from AY : **2023-24**

Pre-requisites of the Course	None	
Course Objectives	This course helps you learn 1. The basics of Design Thinking in an experiential way. 2. This course aims at an empathy-led data-driven app development approach for data scientists. 3. The learners will launch a fully functioning app in a real app store at the end of the course.	
Content	Introduction to Design Thinking – Course outline and projects, Intro to the Design of Everyday Things, Intro to Design Thinking in software apps, Project management. Empathize phase (Iteration #1) - Emotional and intellectual map of the user stories from interviews, User story creation and Customer Journey Mapping	**15 hours**
	Analyse phase (Iteration #1) - Stated needs and unsaid/latent needs, Root cause analysis, Multiple perspectives of customers and manufacturers, Frame conflicts from popular movies. Solve phase (Iteration #1) Structured and unstructured creativity, Dynamics of group thinking, Optimal conditions of creativity, Natural creativity, Concept creation via group activities, Silent brainstorming, inventive principles and concept consolidation	**15 hours**
	Test phase (Iteration #1)/ Empathize phase (Iteration #2) - Basics of prototyping, Assumptions in creation of new concepts, Features rather than ideas. Basics of Digital Marketing, User Experience Design, Website Development	**15 hours**
	Analyse phase (Iteration #2) Solve phase (Iteration #2) - Introduced problems via the solution from iteration #1, the subsequent ideation process in iteration #2, apply solutioning and analysis tools in iteration #2, subsequent testing and field trial skills required for iteration #3, analytical tools and data-oriented tools on iteration #3. Test (Iteration #2) / Empathize (Iteration #3) - Basics of obtaining insights from feedback from a live audience. Analyse (Iteration #3). Test phase (Iteration #3) - Launch of the App.	**15 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	

References/ Readings	1. Norman, D. A. (1988). Design of Everyday Things. *New York City, NY, USA: Doubleday*. 2. Marc, S. (2012). *This is service design thinking: Basics-tools-cases*. Bis Publishers.
Course Outcomes	1. Recall the basics of Design Thinking 2. Apply Agile method to developing software 3. Design an App using the principles of Design Thinking 4. Develop an App for Android and Collaborate with other developers using git version control method

(Back to Index)

SEMESTER II
DISCIPLINE SPECIFIC CORE (DSC) COURSES

Name of the Programme	**: M.Sc. in Data Science**
Course Code	**: CSD-508**
Title of the Course	**: Reinforcement Learning (Theory)**
Number of Credits	**: 2(2L-0T-0P)**
Contact hours	**: 30 hours (30L-0T-0P)**
Effective from AY	**: 2023-24**

Pre-requisites for the course	Linear algebra, multivariable calculus, Basic machine learning knowledge	
Course Objectives	To enable the student to understand 1. The reinforcement learning paradigm 2. Identify when an RL formulation is appropriate 3. Understand the basic solution approaches in RL 4. Implement and evaluate various RL algorithms.	
Content	**Unit1:** Review of ML fundamentals – Classification, Regression. Review of probability theory and optimization concepts. RL Framework; Supervised learning vs. RL; Explore-Exploit Dilemma; Examples. MAB: Definition, Uses, Algorithms, Contextual Bandits, Transition to full RL, Intro to full RL problem Intro to MDPs: Definitions, Returns, Value function, Q-function. Bellman Equation, DP, Value Iteration, Policy Iteration, Generalized Policy Iteration. Evaluation and Control: TD learning, SARSA, Q-learning, Monte Carlo, TD Lambda, Eligibility Traces. Maximization-Bias & Representations: Double Q learning, Tabular learning vs. Parameterized, Q-learning with NNs Function approximation: Semi-gradient methods, SGD, DQNs, Replay Buffer.	**15 hours**
	Unit 2: Policy Gradients: Introduction, Motivation, REINFORCE, PG theorem, Introduction to AC methods Actor-Critic Methods, Baselines, Advantage AC, A3C Advanced Value-Based Methods: Double DQN, Prioritized Experience Replay, Dueling Architectures, Expected SARSA.	**15 hours**

	Advanced PG/A-C methods: Deterministic PG and DDPG, Soft Actor-Critic (SAC) HRL: Introduction to hierarchies, types of optimality, SMDPs, Options, HRL algorithms POMDPS: Intro, Definitions, Belief states, Solution Methods; History-based methods, LSTMS, Q-MDPs, Direct Solutions, PSR. Model-Based RL: Introduction, Motivation, Connections to Planning, Types of MBRL, Benefits, RL with a Learnt Model, Dyna-style models, Latent variable models, Examples, Implicit MBRL. Case study on design of RL solution for real-world problems.	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Sutton, R. S., & Barto, A. G. (1999). Reinforcement learning: An introduction. Robotica, 17(2), 229-235. 2. Szepesvári, C. (2022). Algorithms for reinforcement learning. Springer Nature.	
Course Outcomes	1. Understanding of fundamental concepts and algorithms in reinforcement learning. 2. Proficiency in implementing and evaluating reinforcement learning algorithms. 3. Application of reinforcement learning to real-world problems. 4. Critical analysis and research skills in the field of reinforcement learning.	

(Back to Index)

Name of the Programme : **M.Sc. Data Science**
Course Code : **CSD-509**
Title of the Course : **Reinforcement Learning (Practical)**
Number of Credits : **2 (0L-0T-2P)**
Contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Linear algebra, multivariable calculus, Basic machine learning knowledge and programming background.	
Course Objectives	To understand the theory by carrying out the lab assignment based on the key ideas of reinforcement learning.	
Content	1. RL task formulation (action space, state space, environment definition)	**7 hours**
	2. Tabular based solutions (dynamic programming, Monte Carlo, temporal-difference)	**7 hours**
	3. Function approximation solutions (Deep Q-networks)	**7 hours**
	4. Policy gradient from basic (REINFORCE) towards advanced topics (proximal policy optimization, deep deterministic policy gradient, etc.)	**7 hours**
	5. Model-based reinforcement learning	**7 hours**
	6. Imitation learning (behavioral cloning, inverse RL, generative adversarial imitation learning)	**7 hours**
	7. Meta-learning	**8 hours**
	8. Multi-agent learning, partial observable environments	**10 hours**
Pedagogy	Lab assignments/ mini project	
References/ Readings	1. Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press. 2. Li, S. E. (2023). Deep reinforcement learning. In Reinforcement Learning for Sequential Decision and Optimal Control (pp. 365-402). Singapore: Springer Nature Singapore.). 3. Wiering, M. A., & Van Otterlo, M. (2012). Reinforcement learning. Adaptation, learning, and optimization, 12(3), 729. 4. Russell, S. J., & Norvig, P. (2010). Artificial intelligence a modern approach. London. 5. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press. 6. David Silver's course on Reinforcement Learning (link).	

Course Outcomes	1. Practical implementation of reinforcement learning algorithms in lab exercises. 2. Experimental evaluation and analysis of reinforcement learning algorithms. 3. Application of reinforcement learning techniques to real-world problems. 4. Systematic problem-solving approach in reinforcement learning.

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-510**
Title of the Course : **Optimization Techniques**
Number of Credits : **4(2L-2T-0P)**
Contact Hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	NIL	
Course Objectives	1. To familiarize the students with some basic concepts of optimization techniques and approaches. 2. To formulate a real-world problem as a mathematical programming model. 3. To develop the model formulation and applications are used in solving decision problems. 4. To solve specialized linear programming problems like the transportation and assignment problems.	
Content:	**Unit 1:** **Introduction to Operations Research :** Introduction-Mathematical models of Operation Research - Scope and applications of Operation Research - Phases of Operation Research study - Characteristics of Operation Research - Limitations of Operation Research. **Linear Programming:** Introduction –Properties of Linear Programming-Basic assumptions-Mathematical formulation of Linear Programming-Limitations or constraints-Methods for the solution of LP Problem-Graphical analysis of LP-Graphical LP Maximization problem-Graphical LP Minimization problem. **Linear Programming Models:** Simplex Method-Basics of Simplex Method - Formulating the Simplex Method-Simplex Method with two variables - Simplex Method with more than two variables - Big M Method. **Dual Linear Programming:** Introduction- Primal and Dual problem - Dual problem properties-Solution techniques of Dual problem - Dual Simplex method-Relations between direct and dual problem-Economic interpretation of Duality.	**15 hours**
	Unit2: **Transportation and Assignment Models:** Introduction: Transportation problem - Balanced - Unbalanced - Methods of basic feasible solution Optimal solution-MODI method. Assignment problem-Hungarian Method. **Network Analysis:** Basic concepts-Construction of Network-Rules and precautions-CPM and PERT Networks Obtaining of critical path. Probability and cost consideration. Advantages of Network.	**15 hours**

	Theory of Games : Introduction-Terminology-Two Person Zero-Sum game-Solution of games with saddle points and without saddle points-2X2 games-dominance principle – mX2 and 2Xn games-Graphical method.	
Tutorial Sessions	Case Studies and Mini Projects based on concepts covered during theory lectures	**2*15= 30 hours**
Pedagogy:	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	**Text Book(s)** 1. Gupta, P. K., & Hira, D. S. (2022). Introduction to Operations Research. S. Chand Publishing. 2. J K Sharma (2007), Operations Research Theory & Applications, 3e, Macmillan India Ltd. 3. Maurice Solient, Arthur Yaspen, Lawrence Fridman, OR methods and Problems (2003), New Age International Edition. 4. P. SankaraIyer, (2008), Operations Research, Tata McGraw-Hill. 5. Philips, D. T. (2007). Operations research: Principles and practice. John Wiley & Sons, Incorporated. 6. S.D. Sharma (2000). Operations Research. Nath& Co., Meerut.	
Course Outcomes	1. Apply operations research techniques like linear programming problem in industrial optimization problems. 2. Solve allocation problems using various OR methods. 3. Understand the characteristics of different types of decision making environment and the appropriate decision making approaches and tools to be used in each type. 4. Recognize competitive forces in the marketplace and develop appropriate reactions based on existing constraints and resources.	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-511**
Title of the Course : **MLOps (Theory)**
Number of Credits : **2(2L-0T-0P)**
Contact hours : **30 hours (30L-0T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Familiarity with linear algebra, probability theory, machine learning , familiarity with python.	
Course Objectives	This course is aimed at anyone who wishes to 1. Explore deep learning from scratch. 2. This course offers a practical hand on exploration of deep learning, avoiding mathematical notation, preferring instead to explain quantitative concepts through programming using python API	
Content	**Unit I:** Introduction to MLOps Rise of the Machine Learning Engineer and MLOps-What Is MLOps?-DevOps and MLOps-An MLOps Hierarchy of Needs-Implementing DevOps-Configuring-Continuous Integration with GitHub Actions-DataOps and Data Engineering-Platform Automation-MLOps MLOps Foundations-Bash and the Linux Command Line-Cloud Shell Development Environments-Bash Shell and Commands-List Files Run Commands Files and Navigation-Input/output-Configuration-Writing a Script-Cloud Computing Foundations and Building Blocks-Getting Started with Cloud Computing- minimalistic python revision-Descriptive Statistics and Normal Distributions-Optimization-Machine Learning Key Concepts-Doing Data Science-Build an MLOps Pipeline from Zero	**15 hours**
	MLOps for Containers and Edge Devices Containers-Container Runtime-Creating a Container Running a Container-Best Practices-Serving a Trained Model Over HTTP-Edge Devices-Coral Azure Percept-TFHub-Porting Over Non-TPU Models-Containers for Managed ML Systems-Containers in Monetizing MLOps-Build Once, Run Many MLOps Workflow Continuous Delivery for Machine Learning Models-Packaging for ML Models-Infrastructure as Code for Continuous Delivery of ML Models-Using Cloud Pipelines-Controlled Rollout of Models-Testing Techniques for Model Deployment AutoML and Kaizen ML-Auto ML-MLOps Industrial Revolution-Kaizen Versus Kaizen ML-Feature Stores-Apple's Ecosystem-Apple's AutoML: Create ML-Apple's Core ML Tools or Google's AutoML and Edge Computer Vision or Azure's AutoML or AWS AutoML-Open	

<table>
<tr><td rowspan="2"></td><td>Source AutoML Solutions-Ludwig-FLAML-Model Explainability</td><td></td></tr>
<tr><td>Unit II:
Monitoring and Logging-Observability for Cloud MLOps-Introduction to Logging-Logging in Python-Modifying log Levels-Logging Different Applications-Monitoring and
Observability-Basics of Model Monitoring-Monitoring Drift with AWS SageMaker-Monitoring Drift with Azure ML

MLOps for AWS-Introduction to AWS-Getting Started with AWS Services-MLOps on AWS-MLOps Cookbook on AWS-CLI Tools-Flask Microservice-AWS Lambda Recipes-AWS Lambda-SAM Local-AWS Lambda-SAM Containerized Deploy-Applying AWS Machine Learning to the Real World

Machine Learning Interoperability-Why Interoperability Is Critical-ONNX: Open Neural Network Exchange-ONNX Model Zoo-Convert PyTorch into ONNX -Convert TensorFlow into ONNX-Deploy ONNX to Azure-Apple Core ML-Edge Integration.
Building MLOps Command Line Tools and Microservices-Python Packaging-The Requirements File-Command Line Tools-Creating a Dataset Linter

Modularizing a Command Line Tool-Microservices-Creating a Serverless Function-Authenticating to Cloud Functions-Building a Cloud-Based CLI-Machine Learning CLI Workflows
Machine Learning Engineering and MLOps Case Studies</td><td>15 hours</td></tr>
<tr><td>Pedagogy</td><td colspan="2">Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom</td></tr>
<tr><td>References/ Readings</td><td colspan="2">1. Gift, N., & Deza, A. (2021). Practical MLOps. " O'Reilly Media, Inc.".
2. Gift, N., & Deza, A. (2021) Introduction to MLOps – O'Reilly Media, Inc.".</td></tr>
<tr><td>Course Outcomes</td><td colspan="2">1. Integration of machine learning and software engineering for production systems.
2. Automation of model development, training, and deployment processes.
3. Scalable and reliable infrastructure design for machine learning applications.
4. Monitoring and maintenance of deployed machine learning systems.</td></tr>
</table>

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-512**
Title of the Course : **MLOps (Practical)**
Number of Credits : **2(0L-0T-2P)**
Contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Machine Learning and programming	
Course Objectives	Aimed at imparting the knowledge required to deploy ML models	
Content	1. Perfect Project Structure – Cookiecutter& readme.so	**6 hours**
	2. Speed Exploratory Data Analysis to Minutes – Pandas Profiling, SweetViz	**6 hours**
	3. Track Data Science Projects with CI, CD, CT, CM –Data Version Control (DVC)	**6 hours**
	4. Explainable AI / XAI – SHAP, LIME, SHAPASH	**6 hours**
	5. Deploy ML Projects in minutes – Docker, FastAPI	**6 hours**
	6. End to End Machine Learning – MLflow	**6 hours**
	7. Building Production Ready ML Pipelines - Model Registry, Feature Store (Feast, ButterFlow)	**6 hours**
	8. Big Data using Python, instead of PySpark – DASK	**6 hours**
	9. Build a Chat bot and Deploy it (open-source)	**6 hours**
	10. FaaS Framework implementation – Apache OpenWhisk, OpenFaas	**6 hours**
Pedagogy	Lab Assignments / mini project	
References/ Readings	1. Alla, S., & Adari, S. K. (2020). Beginning MLOps with MLFlow: Deploy Models in AWS SageMaker. Google Cloud, and Microsoft Azure. 2. Burkov, A. (2020). Machine learning engineering (Vol. 1). Montreal, QC, Canada: True Positive Incorporated. 3. Gift, N., & Deza, A. (2021). Practical MLOps. " O'Reilly Media, Inc.". 4. Hapke, H., & Nelson, C. (2020). Building machine learning pipelines. O'Reilly Media. 5. Sweenor, D., Hillion, S., Rope, D., Kannabiran, D., Hill, T., & O'Connell, M. (2020). ML Ops: Operationalizing Data Science. O'Reilly Media, Incorporated. 6. Treveil, M., Omont, N., Stenac, C., Lefevre, K., Phan, D., Zentici, J., ... & Heidmann, L. (2020). Introducing MLOps. O'Reilly Media.	

Course Outcomes	1. Hands-on experience with MLOps tools and technologies. 2. Building end-to-end machine learning pipelines. 3. Deployment and management of infrastructure for machine learning models. 4. Collaboration and adoption of DevOps practices in MLOps.

(Back to Index)

Name of the Programme : M.Sc. in Data Science
Course code : CSD-513
Title of course : Software Engineering for AI Enabled Systems (Theory)
Number of credits : 2 (2L-0T-0P)
Contact hours : 30 hours (30L-0T-0P)
Effective from AY : 2023-24

Pre-requisites for the course	Programming & Data Structures, Python	
Course Objectives	1. Gain an in-depth understanding of Software Engineering including its importance. 2. Learn Scrum, Kanban, Agile, Waterfall, Prototyping, Incremental, RAD and Spiral Software Process Models. 3. Learn to perform systematic Software Requirement Engineering. 4. Applying SE approach to developing AI solutions	
Content	Software Engineering: Software Processes, SDLC, agile approaches to SE Requirements Engineering: elicitation techniques, specification. SCRUM and user stories. Test Driven Development: Refactoring and Unit testing	**15 hours**
	Use of frameworks and APIS and handling of big data Configuration management, continuous integration, and automated software engineering Cloud based software development, DevOps	**15 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Allbee, B. (2018). Hands-On Software Engineering with Python: Move beyond basic programming and construct reliable and efficient software with complex code. Packt Publishing Ltd. 2. Cohn, M. (2005). Agile estimating and planning. Pearson Education 3. Jalote, P. (2008). A concise introduction to software engineering. Springer Science & Business Media.	
Course Outcomes	1. Application of SE principles for AI and Data Science projects 2. How to work in self organizing teams 3. Use of tools and techniques for automating 4. Managing software development	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course code : **CSD-514**
Title of the course : **Software Engineering for AI Enabled Systems (Practical)**
Number of credits : **2 (0L-0T-2P)**
Contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Pre-requisites for the Course	Programming & Data Structures, Python	
Course Objectives	Applying SE approach to developing AI solutions Use of modern software engineering tools and frameworks	
Content	1 Version Control Tools- Git and Github	**12 hours**
	2 TDD –Unit testing and refactoring with Python	**12 hours**
	3 Working with Python libraries and frameworks	**12 hours**
	4 Use of testing tools- selenium, Jmeter	**12 hours**
	5 Cloud based software development &DevOps	**12 hours**
Pedagogy	Lab sessions and projects	
References/ Readings	1. Allbee, B. (2018). Hands-On Software Engineering with Python: Move beyond basic programming and construct reliable and efficient software with complex code. Packt Publishing Ltd. 2. Jalote, P. (2008). A concise introduction to software engineering. Springer Science & Business Media. 3. Cohn, M. (2005). Agile estimating and planning. Pearson Education.	
Course Outcomes	1. Application of SE principles for AI and Data Science projects 2. How to work in self-organizing teams 3. Use of tools and techniques for automating 4. Managing software development	

(Back to Index)

DISCIPLINE SPECIFIC ELECTIVE (DSE) COURSES

Name of the Programme	**: M.Sc. in Data Science**
Course Code	**: CSD-523**
Title of the Course	**: Signal Processing**
Number of Credits	**: 4(2L-2T-0P)**
Contact hours	**: 60 hours (30L-30T-0P)**
Effective from AY	**: 2023-24**

Pre-requisites for the Course	1. Linear algebra, 2. Calculus and multivariable calculus, 3. At least high school math on trigonometry, 4. Complex number 5. A little bit familiarity with programming, especially for numerical computation, such as GNU Octave.	
Course Objectives	1. To study various types of signals and its characteristics. 2. To study various operations on the signals. 3. To analyse the signals using Fourier transform and Laplace Transform. 4. To learn the fundamentals of robotics and sensor technology. 5. To understand the controlling applications of robotics using sensor responses.	
Content for Theory	**Unit1:** Introduction to Signals Continuous-time and Discrete-time Signals: Representation of signals, Signal classification, Types of Signals, Operations on signals - Scaling, Shifting. Fourier Analysis of Continuous-time Signals Introduction to Fourier series, Gibbs Phenomenon, and Continuous-time Fourier transform (CTFT), Existence, Magnitude and phase response, Parseval's theorem, Inverse Fourier transform. Relation between Laplace and Fourier transforms, Laplace Transform, Magnitude and phase response Signal conditioning Sensing - Pre-processing – Noise reduction, enhancement of details. Signal Conversion –Sampling, Quantization, Encoding Data Acquisition and sensing in Robotics Data Acquisition: Analogy and digital data acquisition, single channel and multi-channel data acquisition Image processing in Robotics: Vision sensor, Introduction to computer vision, Point operators, Linear Filters, More neighbourhood operators, Fourier transforms, Pyramids and wavelets, Geometric transformations.	**15 hours**

	Unit II Fundamentals of Robotics Basic components of robotic system. Basic terminology- Accuracy, Repeatability, Resolution, Degree of freedom. Mechanisms and transmission, End effectors, Grippers-different methods of gripping, Mechanical grippers-Slider crank mechanism, Screw type, Rotary actuators, Cam type gripper, Magnetic grippers, Vacuum grippers, Air operated grippers; Specifications of robot. Drive Systems and Sensors in Robotics Drive system- hydraulic, pneumatic and electric systems. Sensors in robot – Touch sensors, Tactile sensor, Proximity and range sensors, Robotic vision sensor, Force sensor, Light sensors, and Pressure sensors. Signal processing application in Robotics Robot applications: Application of robots in surgery, Manufacturing industries, space and underwater. Humanoid robots, Micro robots, Social issues and Future of robotics.	**15 hours**
Content for Tutorial:	1. To find Discrete Fourier Transform and Inverse Discrete Fourier Transform of given digital signal using MATLAB software.	**3 hours**
	2. To obtain Linear Convolution of two finite length sequences using MATLAB software.	**3 hours**
	3. To compute auto correlation between two sequences using MATLAB software.	**3 hours**
	4. AIM: To find frequency response of a given system in differential equation form using MATLAB software.	**3 hours**
	5. AIM: To find the FFT of a given sequence using MATLAB software.	**3 hours**
	6. Determination of Power Spectrum of a given signal using MATLAB software.	**3 hours**
	7. To implement LP FIR filter for a given sequence using MATLAB software.	**6 hours**
	8. To implement HP FIR filter for a given sequence using MATLAB software.	**6 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	

References/ Readings	**Text Book(s)** 1. Deb, S. R., & Deb, S. (2010). Robotics technology and flexible automation. McGraw-Hill Education. 2. Groover, M. P., Weiss, M., & Nagel, R. N. (1986). Industrial robotics: technology, programming and application. McGraw-Hill Higher Education. 3. Haykin, S., & Van Veen, B. (2007). Signals and systems. John Wiley & Sons. 4. Oppenheim, A. V., Willsky, A. S., Nawab, S. H., & Ding, J. J. (1997). Signals and systems (Vol. 2, pp. 74-102). Upper Saddle River, NJ: Prentice hall. 5. Pallas-Areny, R., & Webster, J. G. (2012). Sensors and signal conditioning. John Wiley & Sons.. 6. Rao R.K., Prakriya S. (2013). Signals and Systems. Mc-Graw Hill. 7. Saha,S. K. (2008). Introduction to Robotics. Tata McGraw-Hill Publishing Company Ltd.
Course Outcomes	1. To differentiate continuous and discrete time signals and to analyse the sensor response using Fourier transform 2. To analyse the trajectory of sensor signal using Laplace transform and to understand the signal conditioning and acquisition mechanism 3. To learn the fundamentals and peripherals of robots and to explore sensor responses in controlling robots 4. To explore various real-time application of sensor signal in robotics

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-524**
Title of the Course : **Regression Analytics and Predictive Models**
Number of Credits : **2 (2L-2T-0P)**
Contact hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the Course	Probability Theory and Distributions	
Course Objectives	1. Develop an understanding of regression analysis and model building. 2. Provide the ability to develop relationship between variables 3. Investigate possible diagnostics in regression techniques 4. Formulate feasible solutions using a regression model for real-life problems.	
Content (Theory)	**Unit 1:** **Simple Regression Analysis** Introduction to a linear and nonlinear model. Ordinary Least Square methods. Simple linear regression model, using simple regression to describe a linear relationship. Fitting a linear trend to time series data, validating simple regression model using t, F and p test. Developing confidence interval. Precautions in interpreting regression results. **Multiple Regression Analysis** Concept of Multiple regression model to describe a linear relationship, Assessing the fit of the regression line, inferences from multiple regression analysis, problem of over fitting of a model, comparing two regression model, prediction with multiple regression equation. **Fitting Curves and Model Adequacy Checking** Introduction, fitting curvilinear relationship, residual analysis, PRESS statistics, detection and treatment of outliers, lack of fit of the regression model, test of lack of fit, Problem of autocorrelation and heteroscedasticity. Estimation of pure errors from near neighbors. **Transformation techniques** Introduction, variance stabilizing transformations, transformations to linearize the model, Box Cox methods, transformations on the repressors variables, Generalized and weighted least squares, Some practical applications.	**15 hours**

	Unit 2: **Multicollinearity** Introduction, sources of multicollinearity, effects of multicollinearity. Multicollinearity diagnostics: examination of correlation matrix, variance Inflation factors (VIF), Eigen system analysis of X1X. Methods of dealing with Multicollinearity: collecting additional data, model re- specification, and ridge regression. **Generalized Linear Models** Generalized linear model: link functions and linear predictors, parameter estimation and inference in the GLM, prediction and estimation with the GLM, Residual Analysis, and concept of over dispersion. **Model building and Nonlinear Regression** Variable selection, model building, model misspecification. Model validation techniques: Analysis of model coefficients, and predicted values, data splitting method. Nonlinear regression model, nonlinear least squares, transformation to linear model, parameter estimation in nonlinear system, statistical inference in nonlinear regression.	**15 hours**
Content for Tutorial Slots:	1. Linear Regression	**2 hours**
	2. Minimum Least Square Method	**2 hours**
	3. Calculating coefficients values	**2 hours**
	4. Ascombe's Quartet	**2 hours**
	5. Regression Equations- x on y & y on x	**2 hours**
	6. Predicting mom's height based on daughter's height	**2 hours**
	7. Regression-Solved problem-2	**2 hours**
	8. Probable Error- Calculating correlation coefficient of POPULATION	**2 hours**
	9. Predictive modelling project for credit card fraud detection	**4 hours**
	Any two Projects from below -	
	10. Predictive modeling project for customer value prediction	**5 hours**
	11. Predictive modeling project for stock market forecasting	**5 hours**
	12. Predictive modeling project for corporate bankruptcy prediction	**5 hours**

Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom
References/ Readings	1. Draper, N. R., & Smith, H. (1998). Applied regression analysis (Vol. 326). John Wiley & Sons. 2. Johnson, R., & Wichern, D. (2007). Applied Multivariate Statistical Analysis, PHI Learning Pvt. 3. Montgomery, D. C., Peck, E. A., & Vining, G. G. (2021). Introduction to linear regression analysis. John Wiley & Sons. 4. Pardoe, I. (2020). Applied regression modeling. John Wiley & Sons.
Course Outcomes	1. Develop in-depth understanding of the linear and nonlinear regression model. 2. Demonstrate the knowledge of regression modelling and model selection techniques. 3. Examine the relationships between dependent and independent variables. 4. Estimate the parameters and fit a model.

(Back to Index)

Name of the Programme : **MSc Data Science**
Course Code : **CSD 525**
Title of the Course : **Cloud Computing**
Number of Credits : **4(4L-0T-0P)**
Contact hours : **60 hours(60L)**
Effective from AY : **2023-24**

<table>
<tr><td>Pre-requisites for the Course</td><td colspan="2">Web Development, Programming, Basics of Computer Networks</td></tr>
<tr><td>Course Objectives</td><td colspan="2">The course aims to equip students with an understanding of the fundamentals of Cloud Computing, enabling them to use and adopt cloud services and tools in real-life scenarios, explore major cloud platforms like Google Apps, Microsoft Azure, and Amazon Web Services, and gain knowledge in the practical applications of cloud computing.</td></tr>
<tr><td rowspan="4">Content</td><td>Unit I:
Introduction to Cloud Computing: Cloud Computing Overview: Characteristics – challenges, benefits, limitations, Evolution of Cloud Computing, Cloud computing architecture, Cloud Reference Model (NIST Architecture)
Infrastructure as a Service: Service Model, Characteristics, Benefits, Enabling Technologies Case Study: AWS, OpenStack</td><td>15 hours</td></tr>
<tr><td>Unit II
Platform as a Service: Service Model, Characteristics, Benefits, Enabling Technologies Case Studies : IBM Bluemix, GAE, Microsoft Azure
Software as a Service
Service Model, Characteristics, Benefits, Enabling Technologies Case Study: Salesforce.com, CRM, Online Collaboration Services</td><td>15 hours</td></tr>
<tr><td>Unit III:
Data Analytics as a Service: Hadoop as a service, MapReduce on Cloud, Chubby locking Service</td><td>15 hours</td></tr>
<tr><td>Unit IV:
Introduction to Public and Private Clouds
Shared Resources – Resource Pool – Usage and Administration Portal – Usage Monitor – Resource Management– Cloud Security – Workload
Distribution – Dynamic provisioning.
Storage as a service
Historical Perspective, Datacenter Components, Design Considerations,
Power Calculations, Evolution of Data Centers, Cloud data storage - CloudTM</td><td>15 hours</td></tr>
<tr><td>Pedagogy</td><td colspan="2">Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom</td></tr>
</table>

References/ Readings	1. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2010). Cloud computing: Principles and paradigms. John Wiley & Sons. 2. Hwang, K., Dongarra, J., & Fox, G. C. (2013). Distributed and cloud computing: from parallel processing to the internet of things. Morgan kaufmann. 3. Jamsa, K. (2013). Cloud Computing SaaS, PaaS, IaaS, Virturalization, Business Models, Mobile, Security, and More. 4. Khan, S. U., & Zomaya, A. Y. (Eds.). (2015). Handbook on data centers. 5. Manjunath, G., & Sitaram, D. (2011). Moving to the cloud: Developing apps in the new world of cloud computing. Elsevier. 6. Reese, G. (2009). Cloud application architectures: building applications and infrastructure in the cloud. " O'Reilly Media, Inc.". 7. Rhoton, J., & Haukioja, R. (2011). Cloud computing architected: solution design handbook. Recursive Press. 8. Shroff, G. (2010). Enterprise cloud computing: technology, architecture, applications. Cambridge university press.
Course Outcomes	Upon successful completion of the Cloud Computing course, students will be able to: 1. Understand Cloud Computing, including its characteristics, challenges, benefits, and limitations. 2. Apply Platform as a Service (PaaS) in depth, understanding its service model, characteristics, benefits, and the enabling technologies 3. Apply Data Analytics as a Service, including Hadoop as a service, MapReduce on Cloud, and Chubby locking Service. 4. Design, develop, and demonstrate real-world applications leveraging Cloud Computing technologies

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-526**
Title of the Course : **Big Data Analytics**
Number of Credits : **4(4L)**
Contact Hours : **60 hours (60L-0T-0P)**
Effective from AY : **2023-24**

<table>
<tr><td>Pre-requisites for the Course</td><td colspan="2">Programming Language, SQL queries, and exposure to Linux Environment.</td></tr>
<tr><td>Course Objectives:</td><td colspan="2">The course objective is to equip students with a comprehensive understanding of Big Data platforms, with a specific focus on Apache Hadoop and its ecosystem.</td></tr>
<tr><td rowspan="4">Content: Theory</td><td>UNIT I: INTRODUCTION TO BIG DATA AND HADOOP
Types of Digital Data, Introduction to Big Data, Big Data Analytics, History of Hadoop, Apache Hadoop, Analysing Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming, Hadoop Echo System, IBM Big Data Strategy, Introduction to Infosphere BigInsights and Big Sheets.</td><td>15 hours</td></tr>
<tr><td>UNIT II: HDFS(Hadoop Distributed File System)
The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures.</td><td>15 hours</td></tr>
<tr><td>UNIT III: Map Reduce
Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle, and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features.</td><td>15 hours</td></tr>
<tr><td>Unit IV: Hadoop Eco System
Pig: Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators.
Hive: Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data, and User-Defined Functions.
Hbase: HBasics, Concepts, Clients, Example, Hbase Versus RDBMS.
Big SQL: Introduction</td><td>15 hours</td></tr>
<tr><td>Pedagogy:</td><td colspan="2">Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom</td></tr>
<tr><td>References/Readings</td><td colspan="2">1. Franks, B. (2012). Taming the big data tidal wave: Finding opportunities in huge data streams with advanced analytics. John Wiley & Sons.
2. Liebowitz, J. (Ed.). (2013). Big data and business analytics. CRC press.
3. Warden, P. (2011). Big data glossary. O'Reilly Media, Inc..</td></tr>
</table>

Course Outcomes	Upon completion of the course, learners will be able to: 1. Develop an understanding of the principles, concepts, and technologies underlying big data analytics. 2. Acquire skills in processing and transforming large datasets using distributed computing frameworks like Apache Spark, enabling parallel and scalable data processing. 3. Apply machine learning algorithms to big data 4. Analyze case studies and real-world applications of big data analytic

(Back to Index)

SEMESTER III
RESEARCH SPECIFIC ELECTIVE (RSE) COURSES
Name of the Programme : M.Sc. in Data Science
Course Code : CSD-600
Title of the Course : Research Methodology
Number of Credits : 4 (4L-0T-0P)
Contact Hours : 60 hours (60L-0T-0P)
Effective from AY : 2023-24

Pre-requisites for the Course	None	
Course Objectives:	The objective of the course is to introduce the theoretical as well as practical aspects of Research	
Content:	Foundations of Research: Meaning, Objectives, Motivation, Utility. Concept of theory, empiricism, deductive and inductive theory. Characteristics of scientific method – Understanding the language of research – Concept, Construct, Definition, Variable. Research Process Problem Identification & Formulation – Research Question – Investigation Question – Measurement Issues – Hypothesis – Qualities of a good Hypothesis –Null Hypothesis & Alternative Hypothesis. Hypothesis Testing – Logic & Importance Research Design: Concept and Importance in Research – Features of a good research design – Exploratory Research Design – concept, types and uses, Descriptive Research Designs – concept, types and uses. Experimental Design: Concept of Independent & Dependent variables.	**15 hours**
	Qualitative and Quantitative Research: Qualitative research – Quantitative research – Concept of measurement, causality, generalization, replication. Merging the two approaches. Measurement: Concept of measurement– what is measured? Problems in measurement in research – Validity and Reliability. Levels of measurement – Nominal, Ordinal, Interval, Ratio. Sampling: Concepts of Statistical Population, Sample, Sampling Frame, Sampling Error, Sample Size, Non-Response. Characteristics of a good sample. Probability Sample – Simple Random Sample, Systematic Sample, Stratified Random Sample & Multi-stage sampling. Determining size of the sample – Practical considerations in sampling and sample size.	**15 hours**
	Data Analysis: Data Preparation – Univariate analysis (frequency tables, bar charts, pie charts, percentages), Bivariate analysis – Cross tabulations and Chi-square test including testing hypothesis of association. Interpretation of Data and results	**15 hours**

	Paper Writing – Layout of a Research Paper, Software for paper formatting like LaTeX/MS Office. Journals in Computer Science, Impact factor of Journals, When and where to publish? Ethical issues related to publishing, Plagiarism and Self-Plagiarism. Software for detection of Plagiarism . Use of Encyclopedias, Research Guides, Handbook etc., Academic Databases for Computer Science Discipline. Use of tools / techniques for Research: methods to search required information effectively, Reference Management Software like Zotero/Mendeley	**15 hours**
Pedagogy:	Lecture/Presentations/Assignments/Case Study/	
References/ Readings	1. Business Research Methods – Donald Cooper & Pamela Schindler, TMGH, 9th edition 2. Business Research Methods – Alan Bryman & Emma Bell, Sixth Edition, Oxford University Press. 3. Research Methodology: Methods and Techniques, C.R.Kothari, Second Revised Edition, New Age International Publishers	
Course Outcomes	After completion of this course, students will – 1. Understand how to formulate a research problem 2. Understand data collection and analysis techniques 3. Understand all aspects related to publishing research papers	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-601**
Title of Course : **Natural Language Processing**
Number of Credits : **4(3L+ 1T)**
Contact hours : **60 hours (45L-15T)**
Effective from AY : **2023-24**

Pre-requisites for the Course	Python Programming and Machine Learning	
Course Objectives	This course will provide a foundational understanding of NLP methods and strategies, evaluate strengths and weaknesses of various NLP technologies and frameworks, and gain practical experience in NLP toolkits.	
Content	Introduction, Machine Learning and NLP, ArgMax Computation, Word Sense Disambiguation: WordNet, Wordnet; Application in Query Expansion, Measures of WordNet Similarity. Resnick's work on WordNet Similarity, Parsing Algorithms, Evidence for Deeper Structure; Top-Down Parsing Algorithms, Noun Structure; Top-Down Parsing Algorithms, Non-noun Structure and Parsing Algorithms.	**15 hours**
	Probabilistic parsing; Sequence labelling, PCFG, Probabilistic parsing: Training issues, Arguments and Adjuncts, Probabilistic parsing; inside-outside probabilities. Speech: Phonetics, Hidden Markov Model, Morphology, Graphical Models for Sequence Labelling in NLP, Consonants (place and manner of articulation) and Vowels.	**15 hours**
	Forward Backward probability; Viterbi Algorithm, Phonology, Sentiment Analysis and Opinions on the Web, Machine Translation and MT Tools - GIZA++ and Moses, Text Alignment, POS Tagging. Phonology; ASR, Speech Synthesis, Hidden Markov Model and Viterbi, Precision, Recall, F-score, Map, Semantic Relations; UNL; Towards Dependency Parsing. Universal Networking Language, Semantic Role Extraction, Baum Welch Algorithm; HMM training.	**15 hours**

	Tutorial assignments: 1. Import nltk and download the 'stopwords' and 'punkt' packages and Import spacy and load the language model 2. Program to tokenize a given text, to get the sentences of a text document 3. program to tokenize a text using th`transformers` package, tokenize text with stopwords as delimiters, remove, stop words in a text, add custom stop words in spaCy remove punctuations, and perform stemming. 4. Program to lemmatize a given text, extract usernames from emails, find the most common words in the text excluding stopwords 5. Program to do spell correction in a given text, tokenize tweets, extract all the nouns in a text, extract all the pronouns in a text,find similarity between two words, find similarity between two documents, find the cosine similarity of two documents.	**3x5=15 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Allen, J. (1995). Natural language understanding. Benjamin-Cummings Publishing Co., Inc.. 2. Charniak, E. (1996). Statistical language learning. MIT press. 3. Jurafsky, D. (2008). Martin, and H. James. Speech and Language Processing (2nd Edition)(Prentice Hall Series in Artificial Intelligence). 4. Manning, C., & Schutze, H. (1999). Foundations of statistical natural language processing. MIT press.	
Course Outcomes	After completion of this course, students will be able to: 1. apply various NLP methods and strategies for tasks such as text representation, tokenization, part-of-speech tagging, and syntactic analysis. 2. Analyze sentence structure using syntactic analysis and parsing techniques, including constituency and dependency parsing. 3. Explore real-world applications of NLP	

(Back to Index)

Name of the Programme : **M.Sc. in Data Science**
Course Code : **CSD-602**
Title of the Course : **Deep Learning Models**
Number of Credits : **4(2L-2T-0P)**
Contact hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

<table>
<tr><td>Pre-requisites for the Course</td><td colspan="2">Python Programming and Machine Learning</td></tr>
<tr><td>Course Objectives</td><td colspan="2">The objective of the course is to explore the fundamentals of Neural Networks, including variants like Convolutional Neural Networks and Recurrent Neural Networks, and their diverse applications in problem-solving across domains such as Computer Vision, Speech, and NLP, while developing proficiency in handling extensive datasets through hands-on tools and techniques.</td></tr>
<tr><td>Content</td><td>Unit I
History of Deep Learning, McCulloch Pitts Neuron, Thresholding Logic, Perceptron Learning Algorithm and Convergence
Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Gradient Descent
Feedforward Neural Networks, Representation Power of Feedforward Neural Networks, Backpropagation
Gradient Descent(GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, Adagrad, AdaDelta, RMSProp, Adam, AdaMax, NAdam, learning rate schedulers
Autoencoders and relation to PCA , Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contractive autoencoders
Bias Variance Tradeoff, L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout
Greedy Layer Wise Pre-training, Better activation functions, Better weight initialization methods, Batch Normalization</td><td>15 hours</td></tr>
</table>

	Unit II Learning Vectorial Representations Of Words, Convolutional Neural Networks, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet Visualizing Convolutional Neural Networks, Guided Backpropagation, Deep Dream, Deep Art, Fooling Convolutional Neural Networks Recurrent Neural Networks, Backpropagation Through Time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT Gated Recurrent Units (GRUs), Long Short Term Memory (LSTM) Cells, Solving the vanishing gradient problem with LSTM Encoder Decoder Models, Attention Mechanism, Attention over images, Hierarchical Attention, Transformers.	**15 hours**
	Tutorial Topics **Tensorflow with Python** Introducing Tensorflow - Tensorflow as an Interface - Tensorflow as an environment - Tensors - Computation Graph - Installing Tensorflow - Tensorflow training - Prepare Data - Tensor types - Loss and Optimization - Running tensorflow programs. **Building Neural Networks using Tensorflow** Building Neural Networks using Tensorflow - Tensorflow data types - CPU vs GPU vs TPU - Tensorflow methods - Introduction to Neural Networks - Neural Network Architecture - Linear Regression example revisited - The Neuron - Neural Network Layers - The MNIST Dataset - Coding MNIST NN. **Deep Learning using Tensorflow** Deepening the network - Images and Pixels - How humans recognise images - Convolutional Neural Networks - ConvNet Architecture - Overfitting and Regularization - Max Pooling and ReLU activations - Dropout - Strides and Zero Padding - Coding Deep ConvNets demo - Debugging Neural Networks - Visualising NN using Tensorflow - Tensorboard. **Transfer Learning using Keras and TFLearn** Transfer Learning Introduction - Google Inception Model - Retraining Google Inception with our own data demo - Predicting new images - Transfer Learning Summary - Extending Tensorflow - Keras - TFLearn - Keras vs TFLearn Comparison.	**5 hours**

	Suggest assignment for tutorial Session (ANY FIVE) Assignment -1 Cat vs. Dog Image Classifier Assignment -2- Covid-19 Detection in Lungs Assignment -3- Digit Recognition System Assignment - 4- Facial Recognition Application Assignment -5- Face Mask Detection Assignment -6- Cyber-Attack Prediction Assignment -7- Automated Attendance System Assignment -8 Emotion Recognition Assignment -9- Object Detection System Assignment 10 - Recommender System	**5x5=25 hours**
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Aggarwal, C. C. (2018). Neural networks and deep learning. Springer, 10(978), 3. 2. Aston Z., Zachary C. L., Mu L., Alexander J. S. (2008). Dive into Deep Learning. Cambridge University Press. 3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press. 4. Skansi, S. (2018). Introduction to Deep Learning: from logical calculus to artificial intelligence. Springer.	
Course Outcomes	Upon completion of the course, students will be able to: 1. Develop a comprehensive understanding of fundamental concepts in deep learning, including neural network architecture, activation functions, and optimization algorithms. 2. Develop a comprehensive understanding of fundamental concepts in deep learning, including neural network architecture, activation functions, and optimization algorithms. 3. Understand and apply transfer learning methods to leverage pre-trained models for improved performance on specific tasks, saving computational resources and time. 4. Develop the ability to assess and evaluate the performance of deep learning models using appropriate metrics, ensuring effective analysis of their effectiveness in different applications.	

(Back to Index)

Name of the Programme : **MSc. Data Science**
Course Code : **CSD 603**
Title of the Course : **Data Engineering**
Number of Credits : **4(2L-2T-0P)**
Contact hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the Course	Data Base Fundamentals, Programming skills, mathematics and statistics	
Course Objectives	The objective is to acquire proficiency in data preparation, data integration, data storage, and management, support for analytics, scalability, and real-time processing, encompassing the comprehensive skills needed for the effective handling and utilization of data in various contexts.	
Content	**Unit-I:** Introduction to Data Engineering: Introduction, Evolution of Data Engineering, Data Engineering vs. Data Science, Skills and Activities of a Data Engineer The Data Engineering Lifecycle: Understanding the Data Engineering Lifecycle, Phases: Source Systems, Storage, Ingestion, Transformation, Serving Data, Major Undercurrents Across the Data Engineering Lifecycle Designing Good Data Architecture: Principles of Good Data Architecture, Major Architecture Concepts, Examples and Types of Data Architecture, Roles Involved in Designing Data Architecture Data Generation in Source Systems: Sources of Data and their creation, Types of Source Systems: Files, APIs, Databases, Logs, etc. Practical Details of Source System Handling Storage Fundamentals: Raw Ingredients of Data Storage, Types of Storage: Disk Drives, SSDs, Memory, etc. Storage Systems: File, Block, Object, etc. Storage Abstractions in Data Engineering	**15 hours**
	Unit II Ingestion: Introduction to Data Ingestion, Key Engineering Considerations, Patterns and Methods of Data Ingestion, Practical Issues and Solutions Queries, Modeling, and Transformation: Understanding Queries and Query Optimization, Data Modeling: Conceptual, Logical, and Physical, Batch and Streaming Transformations, Serving Data for Analytics and Machine Learning Security and Privacy: Importance of Security in Data Engineering, People, Processes, and Technology in Security, Best Practices for Ensuring Data Privacy, Security for Low-Level Data Engineering Tasks The Future of Data Engineering: Evolving Landscape of Data Engineering, Simplification of Data Tools, The Role of Cloud and Scalability, Future Trends and Predictions	**15 hours**

	Tutorial sessions (ANY SIX): Preliminaries required to be understood - Python data processing, csv, flat-file, parquet, json, etc, SQL database table design, Python + Postgres, data ingestion and retrieval, PySpark, Data cleansing / dirty data. (How to work on the problems. You will need two things to work effectively on most of these problems.) Docker, docker-compose (All the tools and technologies you need will be packaged into the docker file for each exercise. For each exercise you will need to cd into that folder and run the docker build command, that command will be listed in the README for each exercise, follow those instructions.) **Exercise 1 - Downloading files.** The first exercise tests your ability to download a number of files 2 from an HTTP source and unzip them, storing them locally with Python. cd Exercises/Exercise-1 and see README in that location for instructions. **Exercise 2 - Web Scraping + Downloading + Pandas** The second exercise tests your ability to perform web scraping, build uris, download files, and use Pandas to do some simple cumulative actions. cd Exercises/Exercise-2 and see README in that location for instructions. **Exercise 3 - Boto3 AWS + s3 + Python.** The third exercise tests a few skills. This time we will be using a popular aws package called boto3 to try to perform a multi-step action to download some open source s3 data files. cd Exercises/Exercise-3 and see README in that location for instructions **Exercise 4 - Convert JSON to CSV + Ragged Directories.** The fourth exercise focuses more on file types json and csv, and working with them in Python. You will have to traverse a ragged directory structure, finding any json files and converting them to csv. **Exercise 5 - Data Modeling for Postgres + Python.** The fifth exercise is going to be a little different than the rest. In this problem you will be given a number of csv files. You must create a data model / schema to hold these data sets, including indexes, then create all the tables inside Postgres by connecting to the database with Python. **Exercise 6 - Ingestion and Aggregation with PySpark.** The sixth exercise Is going to step it up a little and move onto more popular tools. In this exercise we are going to load some files using PySpark and then be asked to do some basic aggregation	**5x6=30 hours**

	Exercise 7 - Using Various PySpark Functions The seventh exercise Taking a page out of the previous exercise, this one is focus on using a few of the more common build in PySpark functions pyspark.sql.functions and applying their usage to real-life problems. Many times to solve simple problems we have to find and use multiple functions available from libraries. This will test your ability to do that. **Exercise 8: Project work (ANY ONE):** 1. Scrape Stock and Twitter Data Using Python, Kafka, and Spark 2. Scrape Real-Estate Properties With Python and Create a Dashboard With It 3. Focus on Analytics With Stack Overflow Data 4. Instead of Stocks, Predict Political and Financial Events With PredictIt 5. Scraping Inflation Data and Developing a Model With Data From CommonCrawl	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Burkov, A. (2020). Machine learning engineering (Vol. 1). Montreal, QC, Canada: True Positive Incorporated. 2. Chris, F., Antje B. (2021). Data Science on AWS: Implementing End-to-End, Continuous AI and Machine Learning Pipelines. O'Reilly Media, Inc, USA. 3. Densmore, J. (2021). Data pipelines pocket reference. O'Reilly Media. 4. Hamid, M.Q., Hammad, S. (2021). Snowflake Cookbook: Techniques for building modern cloud data warehousing solutions. Packt Publishing. 5. Macey, T. (2021). 97 Things Every Data Engineer Should Know. " O'Reilly Media, Inc.". 6. Martin, K. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems 7. Paul, C. (2020). Data Engineering with Python: Work with massive datasets to design data models and automate data pipelines using Python. Packt Publishing Limited 8. Ralph, K., Margy, R.(2013) The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. John Wiley & Sons 9. Walker, M. (2020). Python Data Cleaning Cookbook: Modern techniques and Python tools to detect and remove dirty data and extract key insights. Packt Publishing Ltd.	

Course Outcomes	After completion of this course, students will be able to: 1. Design and implement data engineering solutions, applying analytic algorithms to sample datasets for practical insights and problem-solving. 2. Acquire proficiency in developing machine-learning models tailored for real-world datasets, understanding the intricacies of model development and evaluation. 3. Evaluate the effectiveness of analytic algorithms on diverse datasets, fostering a nuanced understanding of their performance in various contexts. 4. Demonstrate the ability to apply machine-learning models to real-world scenarios, showcasing a practical grasp of deploying and assessing models in practical applications.

(Back to Index)

Name of the Programme : **M.Sc. Data Science**
Course Code : **CSD-604**
Title of the Course : **Programming Paradigms**
Number of Credits : **4 (4L-0T-0P)**
Contact hours : **60 hours**
Effective from AY : **2023-24**

<table>
<tr><td>Prerequisites for the course</td><td colspan="2">Knowledge of programming</td></tr>
<tr><td>Course Objectives</td><td colspan="2">To learn, understand and apply the various programming paradigms when writing programs.</td></tr>
<tr><td rowspan="4">Content</td><td>Understanding Programming Paradigm
1. Concept, motivation, types and classification
2. Factors affecting programming languages
Imperative Programming
1. Concepts, Constructs
2. Procedural (in Python/C)
3. Object Oriented (in Java/C++)</td><td>15 hours</td></tr>
<tr><td>Functional Programming (in Haskell/Clojure/Scala)
1. Mathematical functions
2. Side effects; Currying
3. Declare/define functions; composition
4. Recursion, Lazy evaluation
5. Lists; Higher order functions; Folds</td><td>15 hours</td></tr>
<tr><td>Logic Programming (in Prolog/ECLiPSe Constraint language)
1. Mathematical logic
2. Logic programming with facts, rules and goals
3. Constraint logic programming; constraints as relationship between variables; solving puzzles
Event-driven Programming (in Python/.NET)
1. Events; Handlers; Callback
2. Scheduler; Triggers
3. Reliable eventing; Asynchronous triggers</td><td>15 hours</td></tr>
<tr><td>Parallel Programming
1. Shared programming (in OpenMP)
2. Distributed programming (in MPI)
3. MPI with CUDA
Multi-Paradigms
1. Language support for multi paradigms
2. Reactive programming (in Elm/ReactiveX)
3. Meta programming (in Lisp)
4. Natural Language Programming (in SciLab/MATLAB)</td><td>15 hours</td></tr>
<tr><td>Pedagogy</td><td colspan="2">Lectures/ Tutorials/Hands-on assignments/ Self-study/ Flipped classroom</td></tr>
</table>

References/ Readings	1. Allen Tucker, Robert Noonan, “Programming Languages: Principles and Paradigms” 2. Bruce J. Mac Lennan, “Principles of Programming Languages: Design, Evaluation, and Implementation” 3. Graham Hutton, “Programming in Haskell” 4. Kenneth C. Louden, “Programming Languages: Principles and Practice” 5. Ravi Sethi, “Programming Languages Concepts & Constructs” 6. Robert L. Sebesta, “Concepts of Programming Languages” 7. Roland Kuhn, Brian Hanafee, Jamie Allen, “Reactive Design Patterns” 8. Slim Abdennadher, Thom Frühwirth, “Essentials of Constraint Programming” 9. Terrance W. Pratt, Marvin V. Zelkowitz, “Programming Languages - Design & Implementation” 10. W. Clocksin, “Programming in Prolog”
Course Outcomes	1. Learner will be able to distinguish between different programming paradigms, and expand the understand of popular paradigms 2. Learner will be able to decide and understand the need for functional programming based on sound mathematical principles 3. Learner will be able to write logic based decision programs, and also event-driven programs 4. Learner will be able to program on varied hardware/infrastructure platforms, and combine multiple paradigms to suit the requirements

(Back to Index)

GENERIC ELECTIVE (GE) COURSES

Name of the Programme : M.Sc. Data Science
Course Code : CSA-621
Title of the Course : Corporate Skills
Number of Credits : 4 (4L-0T-0P)
Total contact hours : 60 hours
Effective from AY : 2023-24

Prerequisites for the course	Programme prerequisites	
Course Objectives	The course is aimed at learners to gain practical and essential skills to work effectively in the industry.	
Content	Understanding the Industry and Companies ● Understanding the evolution of the industry and technology and methods used ● Understanding Innovation and how new Impactful ideas have evolved ● Types of companies and typical organization - Who does What ● Understanding companies - Domain, Offering, Customers, Strategy ● Company Culture & Professionalism ● Understanding companies financially	**8 hours**
	Understanding Execution and day to day work in organizations ● Product Solutioning and Development - Understanding beyond the theory ● Product Management - Understanding beyond the theory ● Quality - Understanding beyond the theory ● Solutioning and Design - A key step between requirements and delivery ● Site Reliability, Devops, Support - Understanding beyond the theory ● Common Metrics and Measurements ● Key Tools in a Product Life Cycle ● Issues Management and Lifecycle - A key aspect of customer Satisfaction ● Software delivery models and Release cycles - how they work in the real world ● Usability by end user - UI/UX and other key concepts and its importance ● Understanding Data engineering and Data science ● Writing good product or service specifications which can be translated to building a good product ● Understanding data from collection to modeling to usage ● How to do effective product, competition or technical research and use it effectively	**20 hours**

	● testing and testing automation - understand beyond the theory ● what is effective program management and scrum management ● Designing for performance, scalability and reliability in products ● Effective root cause analysis and building products which can allow quicker RCA ● Understanding dev ops and its importance and role in a company ● Understanding product architecture with respect to a monolith or modularity and its pros and cons ● Governance, alerts and monitoring and its importance	
	Useful skills to work effectively in an organization ● Continuous learning and improvement - An essential skill ● Ownership and Leadership ● Analyzing one's career path and making educated judgments ● Time management and multi-tasking model ● Being an effective Mentee and Mentor ● Being Inquisitive: Why is asking questions more difficult than giving answers? ● Effective Articulation and Communication ● Introducing yourself and making Effective Presentations ● Problem breakdown and resolving model ● Effective project management ● Mind Mapping - A powerful technique to learn ● Must have tips to succeed in any career	**20 hours**
	Mini-Project	**12 hours**
Pedagogy	Hands-on assignments / tutorials / peer-teaching / mini-project / case studies	
References/ Readings	All the course material is based on real life industry practices, experiences and case studies and focused on the application of skills and knowledge. The course is being imparted by experienced industry professionals who are still working in the industry and leading critical functions and teams and have the pedigree of building products, managing and delivering to customers, managing teams, and entrepreneurs or being part of core teams in software product or services organization.	
Course Outcomes	At the end of the course, the students will be able to 1. understand core concepts. (To measure this outcome, Question and Answers, Situations analysis, case studies would be used) 2. analyze the problem and apply the appropriate concept. (To measure this outcome, Projects and Case studies would be used) 3. give reasoning. (To measure this outcome, Problem analysis and solving techniques would be taught and used, Question and answers and use	

	cases would be utilized) 4. apply core concepts to new situations. (To measure this outcome, Group projects and Case studies based homework would be used)

(Back to Index)

SEMESTER IV
RESEARCH SPECIFIC ELECTIVE (RSE)

Name of the Programme	**: M.Sc. Data Science**
Course Code	**: CSD-605**
Title of the Course	**: Internet of Things**
Number of Credits	**: 4(4L)**
Contact Hours	**: 60 hours (60L-0T-0P)**
Effective from AY	**: 2023-24**

Prerequisites for the course	Programming knowledge	
Course Objectives:	The course objective is to identify sensor technologies for sensing real-world entities and understand the role of IoT in various domains of Industry.	
Content:	**UNIT I:** Fundamentals of IoT: Introduction, Definitions & Characteristics of IoT, IoT Architectures, Physical & Logical Design of IoT, Enabling Technologies in IoT, History of IoT, About Things in IoT, The Identifiers in IoT, About the Internet in IoT, IoT frameworks, IoT and M2M. Sensors Networks: Definition, Types of Sensors, Types of Actuators, Examples and Working, IoT Development Boards: Arduino IDE and Board Types, RaspberriPi Development Kit, RFID Principles and components, Wireless Sensor Networks: History and Context, The node, Connecting nodes, Networking Nodes, WSN and IoT.	**15 hours**
	UNIT II: Wireless Technologies for IoT: WPAN Technologies for IoT: IEEE 802.15.4, Zigbee, HART, NFC, Z-Wave, BLE, Bacnet, Modbus. IP Based Protocols for IoT IPv6, 6LowPAN, RPL, REST, AMPQ, CoAP, MQTT. Edge connectivity and protocols	**15 hours**
	UNIT III: Data Handling& Analytics: Introduction, Bigdata, Types of data, Characteristics of Big data, Data handling Technologies, Flow of data, Data acquisition, Data Storage, Introduction to Hadoop. Introduction to data Analytics, Types of Data analytics, Local Analytics, Cloud analytics and applications	**15 hours**
	Unit IV: Applications of IoT: Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection.	**15 hours**
Pedagogy:	lectures/ tutorials/lab assignments/self-study/ flipped classroom	
References/ Readings	1. Biron, J., & Follett, J. (2016). *Foundational elements of an iot solution*. O'Reilly Media, Incorporated. 2. Chaouchi, H. (Ed.). (2013). The internet of things: Connecting objects to the web. John Wiley & Sons. 3. Olivier Hersent, David Boswarthick, and Omar Elloumi, — “The Internet of Things: Key Applications and Protocols”	

Course Outcomes	Upon completion of the course, learners will be able to: 1. Gain in-depth knowledge of key concepts, terminology, and the overall architecture of IoT systems 2. Develop practical skills in utilizing a variety of sensors and actuators, essential components in IoT devices. 3. apply various protocols for the design of IoT systems 4. Understand various applications of IoT

(Back to Index)

Name of the Programme : **M.Sc. Data Science**
Course Code : **CSD-606**
Title of the Course : **Speech Processing**
Number of Credits : **4(3L+ 1T)**
Contact Hours : **60 hours (45L-15T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Mathematics for Computer Science and Machine Learning	
Course Objectives:	The objective of the course is to study fundamental concepts of automatic speech recognition.	
Content:	**Unit I:** Anatomy & Physiology of Speech Organs, The process of Speech Production, The Acoustic Theory of Speech Production, Digital models for speech signals. Introduction, Window considerations, Short time energy and average magnitude, Short time average zero crossing rate, Speech vs. silence discrimination using energy and zero crossing, Pitch period estimation using a parallel processing approach, The short time autocorrelation function, The short time average magnitude difference function, Pitch period estimation using the autocorrelation function. Basic principles of Linear Predictive Analysis: The Autocorrelation Method, The Covariance Method, Solution of LPC Equations: Cholesky Decomposition Solution for Covariance Method, Durbin's Recursive Solution for the Autocorrelation Equations, Pitch Detection and using LPC Parameters.	**15 hours**
	Unit II: Introduction, Homomorphic Systems for Convolution: Properties of the Complex Cepstrum, Computational Considerations, The Complex Cepstrum of Speech, Pitch Detection, Formant Estimation, Mel frequency cepstrum computation. Nature of interfering sounds, Speech enhancement techniques: spectral subtraction, Enhancement by resynthesis, Comb filter, Wiener filter. Basic pattern recognition approaches, Parametric representation of speech, Evaluating the similarity of speech patterns, Isolated digit Recognition System, Continuous digit Recognition System.	**15 hours**
	Unit III: Hidden Markov Model (HMM) for speech recognition, Viterbi algorithm, Training and testing using HMMs, Adapting to variability in speech (DTW), Language models. Issues in speaker recognition and speech synthesis of different speakers. Text to speech conversion, Calculating acoustic parameters, synthesized speech output performance and characteristics of text-to-speech, Voice processing hardware and software architectures.	**15 hours**

	Suggested tutorial assignments: Discuss the programs to implement the following: 1. Nature of Speech Signal 2. Time Domain Methods For Speech Processing 3. Frequency Domain Methods For Speech Processing 4. Linear Predictive Coding of Speech 5. Homomorphic Speech Analysis	**3x5= 15 Hours**
Pedagogy:	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/Readings	1. O'shaughnessy, D. (1999). Speech communications: Human and machine (IEEE). Universities press. 2. Rabiner, L. R. (2003). Digital processing of speech signals. Pearson Education India. 3. Rabiner, L. R., & Juang, B. H. (1999). *Fundamentals of speech recognition*. Tsinghua University Press..	
Course Outcomes	After completion of this course, students will be able to: 1. apply signal processing techniques to analyze and preprocess speech signals for feature extraction. 2. develop and implement acoustic models using Hidden Markov Models (HMMs) and deep neural networks to capture relationships between speech features and phonetic units. 3. evaluate ASR systems using appropriate metrics like Word Error Rate (WER) and phoneme error rate	

(Back to Index)

Name of the Programme : M.Sc. Data Science
Course Code : CSD-607
Title of the Course : Web Analytics
Number of Credits : 4 (4L-0T-0P)
Total contact hours : 60 hours
Effective from AY : 2023-24

<table>
<tr><td>Pre-requisites for the course</td><td colspan="2">Programme prerequisites</td></tr>
<tr><td>Course Objectives</td><td colspan="2">The course will help the learner to make strategic decisions based on customer interactions and business intelligence on the web.</td></tr>
<tr><td rowspan="4">Content</td><td>Introduction & Relevant Technologies
1. Definition, Process, Key terms & Key phrases
2. Building blocks of web analytics
3. Offsite web, On site web; Web analytics platform
4. Internet & TCP/IP, Client / Server Computing, HTTP, Server Log Files & Cookies, Web Bugs</td><td>15 hours</td></tr>
<tr><td>Data Collection & Qualitative Analysis
1. Data Collection via Clickstream Data, Outcomes Data, Research data & Competitive Data
2. Heuristic evaluations; site visits
3. Website & post-visit Surveys</td><td>15 hours</td></tr>
<tr><td>Analytic Fundamentals & Using Web Metrics
1. Capturing data via web logs, javascript tags, etc.
2. Separating data serve & data capture
3. Link coding issues
4. Common page metrics (page view, hits, unique visitors, average time on website)
5. Gauging optimization metrics (bounce rate, conversion rate, etc.)
6. Reports (real-time, average traffic, etc.)
7. KPI; perspectives</td><td>15 hours</td></tr>
<tr><td>Web Analytics 2.0 & Google Analytics
1. Overview of Web Analytics of 1.0 & 2.0
2. Competitive intelligence analysis
3. Website traffic analysis
4. Google Analytics; Adwords; benchmarking
5. Google website optimizer; Paid & Organic traffic; privacy concerns</td><td>15 hours</td></tr>
<tr><td>Pedagogy</td><td colspan="2">Lectures/ Tutorials/Hands-on assignments/ Self-study/ Flipped classroom</td></tr>
</table>

References/ Readings	1. Clifton, B. (2012). Advanced web metrics with Google Analytics. John Wiley & Sons. 2. Kaushik, A. (2009). Web analytics 2.0: The art of online accountability and science of customer centricity. John Wiley & Sons. 3. Sterne, J. (2003). Web metrics: Proven methods for measuring web site success. John Wiley & Sons.
Course Outcomes	1. Learner will understand in basic concept of web analysis & analytics, while also understanding the relevant web technologies 2. Learner will understand and apply the various methods & sources for web data collections 3. Learner will apply various methods qualitative analysis and quantitative measures 4. Learner will understand the various analytics aspects that will generate insights from web data collected, for the purpose of strategic decision making

(Back to Index)

Name of the Programme : **MSc. Data Science**
Course Code : **CSD-608**
Title of the Course : **Financial machine learning**
Number of Credits : **4(2L-2T-0P)**
Contact Hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Pre-requisites for the course	Machine learning and probability and statistics	
Course Objectives	The course aims to equip students to use machine learning for cost reduction through process automation, improve revenue generation with faster decision-making, enhance customer experiences by prioritizing critical issues automatically, and bolster security through expedited fraud detection in the financial domain.	
Content	**Unit I:** Financial Machine Learning as a distinct subject- DATA ANALYSIS-Financial Data Structure- Essential Types of Financial Data- Bars- Dealing with Multi-Product Series-Sampling Features LABELING -The Fixed-Time Horizon Method - Computing Dynamic Thresholds - The Triple-Barrier Method - Learning Side and Size - Meta-Labeling - How to Use Meta-Labeling - The Quantamental Way - Dropping Unnecessary Labels SAMPLE WEIGHTS - Overlapping Outcomes - Number of Concurrent Labels-Average Uniqueness of a Label-Bagging Classifiers and Uniqueness- Return Attribution-Time Decay- Class Weights Fractionally Differentiated Features: The Stationarity vs. Memory Dilemma- Literature Review - The Method - Implementation - Stationarity with Maximum Memory Preservation. Ensemble Methods - The Three Sources of Errors - Aggregation - Random Forest - Boosting - Bagging vs. Boosting in Finance - Bagging for Scalability	**15 hours**

	Unit II Cross-Validation in Finance - The Goal of Cross-Validation - Why K-Fold CV Fails in Finance - A Solution: Purged K-Fold CV - Bugs in Sklearn's Cross-Validation. Feature Importance - The Importance of Feature Importance - Feature Importance with Substitution Effects - Feature Importance without Substitution Effects - Parallelized vs. Stacked Feature Importance - Experiments with Synthetic Data Hyper-Parameter Tuning with Cross-Validation - Grid Search Cross-Validation - Randomized Search Cross-Validation - Scoring and Hyper-parameter Tuning HIGH-PERFORMANCE COMPUTING RECIPES Multiprocessing and Vectorization - Vectorization Example Single-Thread vs. Multithreading vs. Multiprocessing,Atoms and Molecules, Multiprocessing Engines, Multiprocessing Example	**15 hours**
	Suggested tutorial assignments (ANY SIX):	**6x5=30 hours**
	Assignment -1: - Process Automation In finance and insurance, employees spend more than half their time collecting and processing data. By implementing machine learning tools, companies can automate a large part of routine and time-consuming processes, increase productivity, save costs, and free up employees so they can focus on higher value-added tasks.	
	Assignment-2: - Document Analysis Text analysis tools use machine learning to make sense of unstructured data. These tools are helping companies in the finance industry gain value from their data in a fast and cost-effective way while reducing human error. Applications range from automatically classifying data in emails, contracts, and reports, to extracting relevant information from legal documents, statements, and bills.	
	Assignment-3: - Portfolio Management Robo-advisors are one of the most popular applications of machine learning in finance. A robo-advisor is an intelligent system that uses machine learning algorithms and statistics. Robo-advisors are often used to provide investment advice and portfolio management services to clients. By processing large amounts of data in a short space of time, robo-advisors can help customers stay ahead and make smart and well-informed investment decisions.	

	Assignment-4: - Algorithmic Trading Algorithmic trading helps businesses make fast and highly accurate trading decisions. Machine learning algorithms are trained to identify trading opportunities, by recognizing patterns and behaviors in historical data.	
	Assignment-5: - Digital Assistants The use of machine learning bots is gaining momentum in the banking industry, helping companies create better experiences in customer service while saving money on call centers. Chatbots, for instance, are equipped with machine learning algorithms and trained to handle common and non-critical customer queries around the clock, scaling support, and improving customer satisfaction.	
	Assignment -6: - Risk Management There is a huge amount of risk involved in the finance sector: market risk, credit risk, operational risk, regulatory risk, and so on. In the last few years, financial companies have increasingly been adopting AI and machine learning to improve risk management, helping them to detect and quantify risks, and make the right decisions. Machine learning algorithms can constantly monitor and analyze large sets of data, in order to spot trends and patterns and deliver critical information in real-time.	
	Assignment-7: - Fraud Detection & Money Laundering Prevention Machine learning is now a key player in the constant battle against fraudulent transactions and money laundering. This technology can detect anomalies in large sets of historical data, and monitor operations in real-time for suspicious behavior, alerting financial services to security threats and illegal activities in real time.	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study/Flipped classroom	
References/ Readings	1. Burkov, A. (2019). The hundred-page machine learning book (Vol. 1, p. 32). Quebec City, QC, Canada: Andriy Burkov. 2. Cartea, Á., Jaimungal, S., & Penalva, J. (2015). Algorithmic and high-frequency trading. Cambridge University Press. 3. De Prado, M. L. (2018). Advances in financial machine learning. John Wiley & Sons. 4. Ruppert, D., & Matteson, D. S. (2011). Statistics and data analysis for financial engineering (Vol. 13). New York: Springer.	

Course Outcomes	Upon completion of the course, students will be able to: 1. Understand Financial Machine Learning, covering data analysis, financial data structures, and methods for handling multi-product series and sampling features. 2. develop expertise in labeling techniques, along with mastering the application of sample weights and Fractionally Differentiated Features to enhance data analysis in the financial domain. 3. Apply Cross-Validation in Finance domain 4. Understand ensemble Methods such as Random Forest, Boosting, and the application of bagging in the financial domain for improved scalability.

(Back to Index)

Name of the Programme : **MSc. Data Science**
Course Code : **CSD-609**
Title of the Course : **Recommender systems**
Number of Credits : **4 (2L-2T-0P)**
Contact Hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

<table>
<tr><td>Pre-requisites for the course</td><td colspan="2">Machine learning and programming in Python</td></tr>
<tr><td>Course Objectives</td><td colspan="2">The course aims to train students to create advanced recommender systems for affordable, personalized, and high-quality recommendations, utilizing relevant tools and implementing algorithms tailored to specific application domains.</td></tr>
<tr><td rowspan="2">Content</td><td>Unit I:
Introduction: Recommender system functions, Linear Algebra notation: Matrix addition, Multiplication, transposition, and inverses; covariance matrices, Understanding ratings, Applications of recommendation systems, Issues with recommender systems.
Collaborative Filtering: User-based nearest neighbor recommendation, Item-based nearest neighbor recommendation, Model based and pre-processing based approaches, Attacks on collaborative recommender systems.
Content-based recommendation: High level architecture of content-based systems, Advantages and drawbacks of content based filtering, Item profiles, Discovering features of documents, Obtaining item features from tags, Representing item profiles, Methods for learning user profiles, Similarity based retrieval, Classification algorithms.</td><td>15 hours</td></tr>
<tr><td>Unit II
Knowledge based recommendation: Knowledge representation and reasoning, Constraint based recommenders, Case based recommenders.
Hybrid approaches: Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation, Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta-level, Limitations of hybridization strategies.
Evaluating Recommender System: Introduction, General properties of evaluation research, Evaluation designs, Evaluation on historical datasets, Error metrics, Decision-Support metrics, User-Centred metrics.
Recommender Systems and communities: Communities, collaboration and recommender systems in personalized web search, Social tagging recommender systems, Trust and recommendations, Group recommender systems.</td><td>15 hours</td></tr>
</table>

<table>
<tr><td></td><td>Suggested tutorial assignments:
1. Finding similarities among users and among content
• Write program to implement similarity functions.
• Write program to implement k means clustering algorithm
2. Collaborative filtering in the neighbourhood
• Amazon algorithm to recalculate item similarity
• Prediction with item-based filtering
3. Evaluating and testing your recommender
• verifying the algorithm
• regression testing.
4. Content-based filtering
• to extract information from descriptions using term fequency-inverse document frequency (TF-IDF) and latent Dirichlet allocation (LDA) to create content profiles.
• content-based filtering using descriptions of films in MovieGEEKs site.
5. Implementation of matrix factoring methods for recommender systems.</td><td>6x5= 30 hours</td></tr>
<tr><td>Pedagogy</td><td colspan="2">Lectures/ Tutorials/Hands-on assignments/ Self-study/ Flipped classroom</td></tr>
<tr><td>References/ Readings</td><td colspan="2">1. Jannach D., Zanker M., and FelFering A. (2011). Recommender Systems: An Introduction. Cambridge University Press.
2. Manouselis, N., Drachsler, H., Verbert, K., & Duval, E. (2012). Recommender systems for learning. Springer Science & Business Media.
3. Ricci F., Rokach L., Shapira D., Kantor B.P. (2011). Recommender Systems Handbook. Springer.</td></tr>
<tr><td>Course Outcomes</td><td colspan="2">Upon completion of the course, students will be able to:
1. Recognize common issues and challenges associated with recommender systems.
2. Explore model-based and pre-processing-based approaches in collaborative recommendation.
3. Explore methods for learning user profiles, similarity-based retrieval, and classification algorithms in content-based recommendation.
4. Explore various evaluation designs for recommender systems, including historical dataset evaluation.</td></tr>
</table>

(Back to Index)