International Year of Cooperatives 2025
Cooperatives Build a Better World

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/2025-26/322 Date: 12.08.2025

CIRCULAR

The Academic Council & Executive Council of the University has approved Ordinance OA-35A relating to PG Programmes offered at the University campus and its affiliated Colleges based on UGC 'Curriculum and Credit Framework for Postgraduate Programmes'. Accordingly, the University has proposed introduction of Ordinance OA-35A from the Academic year 2025-2026 onwards.

The Programme structure and syllabus of Semester I and II of the **Master of Science in Artificial Intelligence** Programme approved by the Academic Council in its meeting held on 13th & 14th June 2025 is attached.

The Dean & Vice-Dean (Academic) 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: 2025.08.12 13:10:25 +05'30'

(Ashwin V. Lawande)
Deputy Registrar – Academic

To,
1. The Dean, Goa Business School, Goa University.
2. The Vice-Dean (Academic), Goa Business School, Goa University.

Copy to:
1. Chairperson, BoS in Data Science and Artificial Intelligence, Goa University.
2. Programme Director, M.Sc. Artificial Intelligence, Goa University.
3. Controller of Examinations, Goa University.
4. Assistant Registrar Examinations (PG), Goa University.
5. Director, Directorate of Internal Quality Assurance, Goa University for uploading the Syllabus on the University website.

GOA UNIVERSITY

MASTER OF SCIENCE IN ARTIFICIAL INTELLIGENCE

(Effective from the Academic Year 2025-2026)

ABOUT THE PROGRAMME

The M.Sc. in Artificial Intelligence is a two-year postgraduate program designed to provide in-depth expertise in core AI areas such as machine learning, deep learning, natural language processing, computer vision, and data science. The curriculum blends theoretical foundations with hands-on lab work, covering both foundational and advanced topics like generative AI, agentic AI, and AI engineering. It also includes interdisciplinary applications, ethical considerations, and industry-relevant skills through electives, research components, and internships. The program prepares graduates for roles such as AI engineers, data scientists, machine learning researchers, and innovators in AI-driven domains.

OBJECTIVES OF THE PROGRAMME

1. M.Sc. in AI Programme provides in-depth knowledge of Machine Learning, Deep Learning, and Data Science while offering hands-on experience in building and deploying AI solutions.
2. It emphasizes ethical AI practices, problem-solving, and interdisciplinary applications in industries like healthcare and finance.
3. The program prepares students for careers as AI Engineers, Data Scientists, and Machine Learning Researchers while fostering innovation through research and industry collaboration.

PROGRAMME SPECIFIC OUTCOMES (PSO)	
PSO 1.	**Core AI Competencies** – Develop a strong foundation in key AI concepts, including machine learning, deep learning, neural networks, and data science.
PSO 2.	**Practical Implementation** – Gain hands-on experience in designing, implementing, and optimizing AI models for real-world applications.
PSO 3.	**Interdisciplinary Applications** – Apply AI techniques across various domains, such as engineering, healthcare, finance, and business analytics.
PSO 4.	**Ethical and Responsible AI** – Understand ethical considerations, bias mitigation, AI governance, and societal impact for responsible AI development.
PSO 5.	**Problem-Solving with AI** – Apply AI methodologies to solve complex challenges across multiple sectors and innovate new solutions.
PSO 6.	**AI System Deployment** – Learn to integrate AI solutions in industry and optimize performance for scalability and efficiency.
PSO 7.	**Career Readiness** – Develop the skills needed for roles like AI Engineer, Data Scientist, Machine Learning Researcher, and Robotics Engineer. It also encourages AI start-ups and entrepreneurship
PSO 8.	**Research and Innovation** – Conduct original research in AI, contribute to advancements in the field, and publish academic papers.

PROGRAMME STRUCTURE
Master of Science in Artificial Intelligence
Effective from Academic Year 2025-26

Bridge Course			
Sr. No.	**Course Code**	**Title of the Course**	**Credits**
1	CSI-1000	Fundamentals of Python Programming	2

SEMESTER I				
Discipline Specific Core (DSC) Courses (16 credits)				
Sr. No.	**Course Code**	**Title of the Course**	**Credits**	**Level**
1	CSI-5000	Fundamentals of Artificial Intelligence	2	400
2	CSI-5001	Mathematical Foundation for AI	2	400
3	CSI-5002	Pattern Recognition and Machine Learning	2	400
4	CSI-5003	Algorithm Design and Data Structures	2	400
5	CSI-5004	Fundamentals of Artificial Intelligence Lab	2	400
6	CSI-5005	Mathematical Foundation for AI Lab	2	400
7	CSI-5006	Pattern Recognition and Machine Learning Lab	2	400
8	CSI-5007	Algorithm Design and Data Structures Lab	2	400
Total Credits for DSC Courses in Semester I			**16**	
Discipline Specific Elective (DSE) Course (4 credits) (any 1)				
Sr. No.	**Course Code**	**Title of the Course**	**Credits**	**Level**
1	CSI-5201	Fundamentals of Natural Language Processing	4	400
2	CSI-5202	Fundamentals of Robotics	4	400
3	CSI-5203	Computer Vision	4	400
4	CSI-5204	Speech Processing	4	400
5	CSI-5205	Data Science and Data Engineering	4	400
Total Credits for DSE Courses in Semester I			**4**	
Total Credits in Semester I			**20**	

SEMESTER II				
Discipline Specific Core (DSC) Courses				
Sr. No.	**Course Code**	**Title of the Course**	**Credits**	**Level**
1	CSI-5008	Fundamentals of Deep Learning and Generative AI Techniques	2	500
2	CSI-5009	Reinforcement Learning	2	500
3	CSI-5010	Big Data Frameworks	2	500
4	CSI-5011	MLOp	2	500
5	CSI-5012	Fundamentals of Deep Learning and Generative AI Techniques Lab	2	500
6	CSI-5013	Reinforcement Learning Lab	2	500
7	CSI-5014	Big Data Frameworks Lab	2	500
8	CSI-5015	MLOp Lab	2	500
		Total Credits for DSC Courses in Semester II	**16**	
Discipline Specific Elective (DSE) Courses (4 credits)				
Sr. No.	**Course Code**	**Title of the Course**	**Credits**	**Level**
1	CSI-5206	Machine Translation	4	400
2	CSI-5207	Robotic motion planning and control	4	400
3	CSI-5208	Interpretable Machine learning	4	400
4	CSI-5209	Explainable AI	4	400
		Total Credits for DSE Courses in Semester II	**4**	
		Total Credits in Semester II	**20**	

BRIDGE COURSE

Title of the Course	Fundamentals of Python Programming
Course Code	CSI-1000
Number of Credits	1T+1P
Theory/Practical	Theory/Practical
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	Yes (Bridge Course)
Course for advanced learners	No

Pre-requisites for the Course:	Nil			
Course Objectives:	The objective of the course is to equip students with a foundational understanding of Python programming, enabling them to write and execute basic Python programs.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember Python programming constructs.		PSO1, PSO5, PSO7	
	CO 2. Apply variables, control structures, and functions in programs.		PSO2, PSO7	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Introduction to Problem Solving & Python Basics** ● **Problem-Solving Approach**: Understanding the problem, Algorithm design	**15**	CO1	K1, K2

	(pseudocode& flowcharts) ● **Python Fundamentals**: Variables, expressions, statements, data types, operators, Input/output operations, Basic syntax & indentation ● **Control Structures:** Boolean values and operators, Conditional statements ● **Iteration**: Loops, Nested loops & pattern printing exercises **Functions** ● function and its use, pass keyword, flow of execution, parameters and arguments **Strings** ● Strings, String manipulation methods, String formatting			
Module 2:	**Practical Work** ● Assignments to practice input/output and use of basic data types. ● Assignments to practice arithmetic operations and expressions. ● Assignments to practice control structures, branch and loops ● Assignments to practice writing modular code ● Assignments to practice strings and string manipulation functions	**30**	CO2	K3, K4, K6
Pedagogy:	Mentoring/ Assignments/ Flipped Classroom			
Texts:	1. Downey, A., Meyer, C., &Elkner, J. (2016). How to think like a computer scientist: learning with Python. Green Tea Press. 2. Barry, P. (2016). Head first Python: a brain-friendly guide. O'reilly.			
References/ Readings:	1. Hill, C. (2020). *Learning scientific programming with Python*. Cambridge University Press. 2. Lee, K. D. (2014). Python programming fundamentals. Springer.			
Web Resources:	1. W3Schools.com.(n.d.). https://www.w3schools.com/python/ 2. GeeksforGeeks. (2025, May 3). Python tutorial | Learn Python programming language. Retrieved May 16, 2025, from https://www.geeksforgeeks.org/python-programming-language-tutorial/			

SEMESTER I

Discipline Specific Core Courses

Title of the Course	Fundamentals of Artificial Intelligence
Course Code	CSI-5000
Number of Credits	2T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil	
Course Objectives:	This course is aimed at developing a foundational understanding of AI concepts and techniques, including search algorithms, intelligent agents, machine learning, and probabilistic reasoning, while gaining the skills to design ethical and effective AI solutions for real-world problems.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Apply foundational AI concepts and techniques, including intelligent agents, search strategies, and problem formulation, to select appropriate search algorithms for goal-oriented tasks.	PSO1, PSO5, PSO7

	CO 2. Apply knowledge representation and reasoning techniques such as propositional logic, first-order logic, and constraint satisfaction methods to model and solve structured problems.		PSO1, PSO5, PSO6	
	CO 3. Create basic machine learning models using supervised and unsupervised learning approaches, including decision trees, k-NN, and neural networks, incorporating understanding of training mechanisms and evaluation metrics.		PSO1, PSO2, PSO6, PSO7	
	CO 4. Evaluate probabilistic reasoning models and decision-making frameworks including Bayesian networks and Markov decision processes, and assess the ethical implications of AI systems with focus on explainability and responsible AI.		PSO1, PSO4, PSO5, PSO8	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Foundations of Artificial Intelligence and Intelligent Agents** • **AI Basics**: Definitions, history, and goals • **AI Tribes**: Symbolic, Connectionist, Bayesian, Evolutionary, Analogizer • **Intelligent Agents**: Definitions, environments, PEAS (Performance, Environment, Actuators, Sensors) • **Agent Types**: Reactive, Model-based, Goal-based, Utility-based • **Problem-Solving by SearchFormulation**: States, actions, goals, **Uninformed Search**: BFS, DFS **Knowledge and Logical Reasoning in AI** • **Knowledge Representation**: Concepts and structures • **Logic in AI**: Propositional & First-order Logic • **Ontological Engineering**: Basics and applications • **Inference Techniques**: Forward & Backward Chaining, Resolution, Unification • **Constraint Satisfaction Problems (CSPs)**: Problem formulation,Solving via Backtracking & Constraint Propagation.	**15**	CO1, CO2	K1, K2, K3, K4, K5

Module 2:	**Learning and Decision-Making in AI** • **Intro to Machine Learning**: Concepts, features, labels, train/test split • **Types of Learning**: Supervised, Unsupervised, Reinforcement • **Algorithms**: Decision Trees, k-NN, Neural Networks (Perceptrons, MLPs) • **Training Basics**: Loss functions, Gradient Descent • **Evaluation Metrics**: Accuracy, Precision, Recall, F1-score **Probabilistic Reasoning** • **Probability Basics**: Conditional probability, Bayes' Theorem • **Bayesian Networks**: Structure and inference • **Utility Theory**: Preferences and rational decision-making • **Markov Decision Processes (MDPs)**: States, actions, rewards, policies **AI Applications, Explainability, and Ethics** **Natural Language Processing (NLP)** • Tokenization, Stemming, Lemmatization • N-gram Models, Information Retrieval & Extraction • Machine Translation, Speech Recognition **Explainable AI (XAI)** • Black-box Models, Transparency, Interpretability • Rule-based Explanations, Feature Importance **Ethics in AI** • Bias, Fairness, Accountability • Ethical Dilemmas, Case Studies • Social Implications of AI	**15**	CO3, CO4	K2, K3, K4, K5
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Russell, S. J., &Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Pearson Education. 2. Rich, E., & Knight, K. (2017). *Artificial intelligence* (3rd ed.). McGraw-Hill Education			

References/ Readings:	1. Nilsson, N. J. (1997). *Artificial intelligence: A new synthesis*. Elsevier. 2. Luger, G. F. (2002). *Artificial intelligence: Structures and strategies for complex problem solving* (4th ed.). Pearson Education. *(Note: Title and edition inferred based on standard editions; please adjust if your copy is different.)* 3. Padhy, N. P. (2005). *Artificial intelligence*. Oxford University Press.
Web Resources:	1. Stanford University. (2024). CS221: Artificial intelligence: Principles and techniques. Retrieved May 16, 2025, from https://cs221.stanford.edu 2. Massachusetts Institute of Technology. (2010). 6.034: Artificial intelligence. Retrieved May 16, 2025, from https://ocw.mit.edu/courses/6-034-artificial-intelligence-fall-2010/

Title of the Course	Mathematical Foundations for AI
Course Code	CSI-5001
Number of Credits	2T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil			
Course Objectives:	The aim of the course is to introduce the fundamental concepts of probability, statistics, linear algebra, and calculus, and to emphasize their importance in solving problems and making decisions in Artificial Intelligence.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember basic probability principles.		PSO1, PSO5	
	CO 2. Understand statistical data analysis techniques.		PSO1, PSO2, PSO5	
	CO 3. Apply linear algebra for solving mathematical problems.		PSO1, PSO5	
	CO 4. Analyze multivariable functions using calculus.		PSO1, PSO2	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**

Module 1:	**Probability:** • Basic concepts of probability, conditional probability, law of total probability, independence of events, Bayes' theorem, random variables (discrete and continuous), expectation, moments, moment generating functions, commonly used probability distributions, joint and conditional distributions, transformation of random variables, covariance and correlation. **Statistics:** • Sampling techniques, sampling distributions, parameter estimation, hypothesis testing, mixture models, expectation-maximization (EM) algorithm.	**15**	CO1, CO2	K2, K3, K4
Module 2	**Basics of Linear Algebra:** • Representation of vectors and matrices, linear dependence and independence, vector spaces and subspaces (definition, examples, and basis), linear transformations, range and null space, special types of matrices, eigenvalues and eigenvectors, diagonalization, singular value decomposition (SVD), least squares and minimum norm solutions, applications to data analysis. **Gradient Calculus:** • Basic concepts of calculus, partial derivatives, gradient, directional derivatives, Jacobian, Hessian. **Optimization:** • Introduction to Optimization, Convex Sets and Convex Functions, Unconstrained Optimization, Derivative-Free Methods (Golden Section Method, Fibonacci Search), Gradient-Based Methods (Steepest Descent Method, Newton's Method), Constrained Optimization, and Penalty Function Methods.	**15**	CO3 CO4	K2, K4, K5
Pedagogy	Lectures/ Assignments/ Flipped Classroom			
Texts:	Deisenroth, M. P., Faisal, A. A., &Ong, C. S. (2020). Mathematics for machine learning. Cambridge University Press.			
References/ Readings:	1. Axler, S. (2024). Linear algebra done right (p. 390). Springer Nature. 2. Johnson, R. A., Miller, I., & Freund, J. E. (2000). Probability and statistics for engineers. *Proc. Miller Freund's,*			

	546-554. 3. Kreyszig, E., Stroud, K., & Stephenson, G. (2008). Advanced engineering mathematics. *Integration*, *9*(4), 1014. 4. C. Mohan and K. Deep: "Optimization Techniques", New Age Publishers, New Delhi.
Web Resources:	1. Massachusetts Institute of Technology. (2018). RES.6-012: Introduction to probability. MIT OpenCourseWare. Retrieved May 16, 2025, from https://ocw.mit.edu/courses/res-6-012-introduction-to-probability-spring-2018/ 2. Harvard University. (2024). Statistics and R. Harvard Professional and Lifelong Learning. Retrieved May 16, 2025, from https://pll.harvard.edu/course/statistics-and-r

Title of the Course	Pattern Recognition and Machine Learning
Course Code	CSI-5002
Number of Credits	2T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil			
Course Objectives:	To introduce students to the fundamental principles of machine learning, including decision-making models and ensemble techniques, and to develop their ability to apply regression, classification, and advanced methods for data analysis and optimization.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember foundational ML and decision models.		PSO1, PSO2, PSO5	
	CO 2. Understand regression and classification techniques.		PSO1, PSO2, PSO5	
	CO 3. Apply machine learning models to datasets.		PSO1, PSO2, PSO5, PSO3	
	CO 4. Analyze clustering and model accuracy.		PSO1, PSO2, PSO5, PSO6	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**

Module 1:	**Introduction to Concept Learning**: • Version Space, Decision Tree, Random Forest Algorithm **Linear Models for Regression:** • Linear Basis Function Models, Maximum likelihood and least squares, Geometry of least squares, Sequential learning, Regularized least squares, Multiple outputs, The Bias-Variance Decomposition, Bayesian Linear Regression. **Linear Models for Classification:** • Discriminant Functions, Two classes, Multiple classes, Least squares for classification, Fisher's linear discriminant, Relation to least squares, Fisher's discriminant for multiple classes, Binary class Logistic Regression, Multiclass Logistic Regression. **Neural Networks:** • The perceptron algorithm, Feed-forward Network Functions, Weight-space symmetries, Network Training, Parameter optimization, Local quadratic approximation, Use of gradient information, Gradient descent optimization, Error Backpropagation, Evaluation of error-function derivatives, A simple example, Efficiency of backpropagation, The Jacobian matrix, The Hessian Matrix,	**15**	CO1, CO2	K2, K3, K4
Unit/ Module 2:	**Sparse Kernel Machines:** • Maximum Margin Classifiers, Overlapping class distributions, Relation to logistic regression, Multiclass SVMs, SVMs for regression. **Mixture Models for EM:** • K-means Clustering, Image segmentation and compression, Mixtures of Gaussians, Maximum likelihood, EM for Gaussian mixtures, EM algorithm. **Continuous Latent Variable:** • Principal Component Analysis, Maximum variance formulation, Minimum-error formulation, Applications of PCA, PCA for high-	**15**	CO3, CO4	K3, K4, K5

	dimensional data. **Sequential Data:** • Markov Models, Hidden Markov Models, Maximum likelihood for the HMM, The forward-backward algorithm, The sum-product algorithm for the HMM, Scaling factors, The Viterbi algorithm. **Ensemble Learning:** • Voting classifier, bagging and boosting			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	Bishop, C. M., &Nasrabadi, N. M. (2006). Pattern recognition and machine learning (Vol. 4, No. 4, p. 738). New York: springer.			
References/ Readings:	1. Starmer, J. (2022). The Statquest illustrated guide to machine learning. 2. Mitchell, T. M. (1997). Machine learning (Vol. 1). McGraw-hill New York. 3. Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. Cambridge university press. 4. Geron A. (2022) Hands-on Machine Leaning with Sci-Learn, Keras&TensorFlow. Shroff/O'Reilly.			
Web Resources:	1. Kaggle. (n.d.). Kaggle Learn. Retrieved May 16, 2025, from https://www.kaggle.com/learn 2. Google. (n.d.). Google AI: For developers. Retrieved May 16, 2025, from https://ai.google/get-started/for-developers/			

Title of the Course	Algorithm Design and Data structures
Course Code	CSI-5003
Number of Credits	2T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-1000			
Course Objectives:	The aim of the course is to introduce the fundamental concepts of data structures and to emphasize the importance of data structures in developing and implementing efficient algorithms.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember various data structures.		PSO1	
	CO 2. Understand their functional differences and uses.		PSO1, PSO2	
	CO 3. Apply them in solving problems.		PSO2	
	CO 4. Analyze algorithms and computing their complexity.		PSO2	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**

Module 1:	• **Introduction:** Three-Level Approach : Application/User level, Abstract/ Logical level, Physical/ Implementation Level. • **Abstract Data Types (ADTs):** Concept of ADTs, Data Structure definition, Data type v/s data structure, Applications of data structures. • **Algorithms analysis and its complexity**: Best case, worst case, and Average case performance, time-space tradeoff, Asymptotic Analysis, Big-o notation. • **Linear Data Structures**: Array and its application: Polynomials, Sparse matrices, String-pattern Matching. Linked Lists, Doubly linked list, Circular linked list, Stack and Queues.	**15**	CO1, CO2, CO3	K1, K2, K3, K4
Module 2	**Nonlinear Data Structures**: • **Trees**: Binary tree representation, Binary Search Trees, AVL Trees, M-way Search Trees, B-trees, B tree algorithms, Heap Structures. • **Graphs**: Graph representations; Graph Traversals **Algorithms:** • **Complexity of Searching & Sorting algorithms**: Bubble sort, Quick sort, Selection sort, Insertion sort, Merge sort and Heap sort. An Empirical Comparison of Sorting Algorithms, Lower Bounds for Sorting. Linear search, binary search. • **Dynamic programming and Greedy algorithms**: Assembly line scheduling, Matrix-chain multiplication, Prim's Algorithm, Kruskal's Algorithm	**15**	CO2, CO3, CO 4	K1, K2, K4, K5, K6
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Dr. BasantAgarwal, &Baka, B. (2018). Hands-On Data Structures and Algorithms with Python. Packt Publishing Ltd. 2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms. The Mit Press.			
References/ Readings:	1. Dasgupta, S., Papadimitriou, C. H., &Vazirani, U. V. (2006). Algorithms. McGraw-Hill Publishing. 2. Mark Allen Weiss. (2003). Data structures & algorithm analysis in C++. Pearson Education.			

	3. Horowitz, E., &Sahni, S. (1976). Fundamentals of Data Structures. Computer Science Press, Incorporated.
Web Resources:	1. Massachusetts Institute of Technology. (2011). 6.006 Introduction to algorithms. MIT OpenCourseWare. Retrieved May 16, 2025, from https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/ 2. GeeksforGeeks. (2025, April 25). DSA tutorial: Learn data structures and algorithms. Retrieved May 16, 2025, from https://www.geeksforgeeks.org/dsa-tutorial-learn-data-structures-and-algorithms/

Title of the Course	Fundamentals of Artificial Intelligence Lab
Course Code	CSI-5004
Number of Credits	2P
Theory/Practical	Practical
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil	
Course Objectives:	This course is aimed at imparting students with hands-on experience in implementing, evaluating, and designing AI systems, enabling them to solve real-world problems using algorithms, machine learning, probabilistic reasoning, and ethical AI practices.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Apply uninformed and informed search algorithms, logic-based reasoning, and constraint satisfaction techniques to solve structured AI problems.	PSO1, PSO2, PSO5
	CO 2. Evaluate machine learning models—including decision trees, k-NN, and neural networks—for classification and regression tasks.	PSO1, PSO2, PSO5, PSO7
	CO 3. Create probabilistic models (e.g., Bayesian networks) and perform inference and decision-making under uncertainty.	PSO1, PSO3, PSO5
	CO 4. Synthesize ethical principles and explainability techniques into AI systems using	PSO2, PSO6, PSO7,

	model interpretation and fairness metrics.		PSO8	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Lab 1: Problem-solving by Searching** ● Implement uninformed search algorithms (BFS, DFS). ● Solve a pathfinding problem (e.g., maze navigation) using A* algorithm. **Lab 2: Knowledge Representation and Reasoning** ● Build a simple expert system using propositional logic. ● Solve a constraint satisfaction problem (e.g., Sudoku or map coloring) using backtracking. **Lab 3: Machine Learning Basics** ● Implement and evaluate decision trees and k-nearest neighbors (k-NN) on a dataset (e.g., Iris or Titanic dataset). ● Train a multi-layer perceptron (MLP) using TensorFlow/Keras for a classification task. **Lab 4: Probabilistic Reasoning** ● Build a Bayesian network for a real-world scenario (e.g., medical diagnosis). ● Perform inference on the network using a library like PyMC3 or pgmpy.	**30**	CO1, CO2, CO3	K3, K4
Module 2:	**Lab 5: Natural Language Processing** ● Perform text preprocessing (tokenization, stemming, lemmatization) using NLTK. ● Build a simple language model (n-grams) for text generation or classification. **Lab 6: Explainable AI** ● Use SHAP or LIME to explain predictions of a machine learning model. ● Analyze feature importance in a decision tree or neural network. **Lab 7: Ethics in AI**	**30**	CO1, CO3 CO3, CO4	K3, K5, K6

	● Analyze bias in a dataset (e.g., gender or racial bias in hiring data). ● Implement fairness metrics (e.g., demographic parity, equal opportunity) using AI Fairness 360 or Fairlearn. **Mini Project:** The capstone project integrates concepts from the course into a comprehensive AI application. Students work in teams to solve a real-world problem, demonstrating their ability to design, implement, and evaluate an AI system. The project involves Problem definition, Data collection and preprocessing, Model Development, Evaluation etc			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	1. Russell, S. J., &Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Pearson Education. 2. Rich, E., & Knight, K. (2017). *Artificial intelligence* (3rd ed.). McGraw-Hill Education.			
References/ Readings:	1. Raschka, S., &Mirjalili, V. (2019). *Python machine learning* (2nd ed.). Packt Publishing. 2. Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media. 3. Bird, S., Klein, E., &Loper, E. (2009). *Natural language processing with Python*. O'Reilly Media.			
Web Resources:	1. GitHub. (n.d.). Artificial intelligence projects. Retrieved May 16, 2025, from https://github.com/topics/artificial-intelligence-projects 2. DataCamp. (2023, July 14). 7 AI projects for all levels. Retrieved May 16, 2025, from https://www.datacamp.com/blog/7-ai-projects-for-all-levels			

Title of the Course	Mathematical Foundations for AI (Lab)
Course Code	CSI-5005
Number of Credits	2P
Theory/Practical	Practical
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil			
Course Objectives:	To provide students with hands-on experience in applying mathematical concepts, including linear algebra, statistics, and optimization, using Python to solve real-world problems in AI and Data Science.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember Python libraries used in data science.		PSO1, PSO2, PSO5	
	CO 2. Understand visualizations like bar plots and histograms.		PSO1, PSO2, PSO5	
	CO 3. Apply statistical methods to datasets.		PSO1, PSO5	
	CO 4. Analyze linear algebra and optimization techniques.		PSO1, PSO2, PSO3, PSO5, PSO6	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**

Module 1:	**Assignment 1:** Introduction the following Python libraries and their core functionalities: • **NumPy**: Array-based numerical computing using ndarray, including statistical routines. • **SciPy**: Scientific computing built on NumPy, with modules like scipy.stats for statistical analysis. • **Pandas**: Data handling using Series (1D) and DataFrame (2D) structures, built on NumPy. • **Matplotlib**: Data visualization library compatible with NumPy, SciPy, and Pandas. **Assignment 2** – Understanding Exploratory Data Analysis (EDA) concepts using Python Libraries. **Assignment 3** – Sampling, Variables in Statistics, Frequency Distributions. Generate frequency distribution tables, generate grouped frequency distribution tables and visualize frequency distributions. Generate bar plots, pie charts, and histograms. Employ bar plots, pie charts and histograms. (6 hours) **Assignment 4** – Comparing Frequency Distributions: grouped bar plots, step-type histogram, kernel density estimate plots, strip plots and box plots. (6 hours) **Assignment 5** – Multidimensional image operations, solving differential equations and the Fourier transform using SciPy. **Assignment 6** – Optimization algorithms using SciPy. **Assignment 7** – Linear algebra using SciPy.	**30**	CO1, CO2	K1, K2, K3, K4
Module 2:	**Assignment 8** – 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. **Assignment 9** – Implement Eigen value decomposition in Python. **Assignment 10** – Implement SVD using Python. **Assignment 11** – Implement some optimization algorithms using Python libraries	**30**	CO3, CO4	K1, K2, K3, K6

	(e.g., SciPy, TensorFlow, PyTorch). **Assignment 12:** Mini Project			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	Deisenroth, M. P., Faisal, A. A., &Ong, C. S. (2020). Mathematics for machine learning. Cambridge University Press.			
References/ Readings:	1. Axler, S. (2024). Linear algebra done right (p. 390). Springer Nature. 2. Johnson, R. A., Miller, I., & Freund, J. E. (2000). Probability and statistics for engineers. *Proc. Miller Freund's*, 546-554. 3. Kreyszig, E., Stroud, K., & Stephenson, G. (2008). Advanced engineering mathematics. *Integration*, *9*(4), 1014.			
Web Resources:	1. Massachusetts Institute of Technology. (2018). RES.6-012: Introduction to probability. MIT OpenCourseWare. Retrieved May 16, 2025, from https://ocw.mit.edu/courses/res-6-012-introduction-to-probability-spring-2018/ 2. Harvard University. (2024). Statistics and R. Harvard Professional and Lifelong Learning. Retrieved May 16, 2025, from https://pll.harvard.edu/course/statistics-and-r			

Title of the Course	Pattern Recognition and Machine Learning Lab
Course Code	CSI-5006
Number of Credits	2P
Theory/Practical	Practical
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil	
Course Objectives:	To provide hands-on experience in implementing and applying machine learning techniques, including decision trees, neural networks, and advanced models for classification, clustering, and dimensionality reduction.	
Course Outcomes:	At the end of the course, the students will be able to:	**Mapped to PSO**
	CO 1. Remember core ML algorithms like decision trees and regression.	PSO1, PSO2, PSO5
	CO 2. Understand classification models and neural networks.	PSO1, PSO2, PSO5
	CO 3. Apply SVM and EM clustering methods.	PSO1, PSO2, PSO3, PSO5
	CO 4. Evaluate dimensionality reduction and ensemble models.	PSO1, PSO2, PSO5, PSO6

Content:		No of hours	Mapped to CO	Cognitive Level
Module 1:	**Experiment 1: EDA and Data Visualization** Analyze and visualize datasets (e.g., Iris, Titanic) using tools like Pandas and Matplotlib. **Experiment 2: k-Nearest Neighbors (k-NN)** Implement k-NN classification on datasets like Iris, tuning k and evaluating accuracy. **Experiment 3: Linear Regression** Predict continuous targets (e.g., Boston Housing) and explore feature selection techniques. **Experiment 4: Logistic Regression** Perform binary classification (e.g., Breast Cancer) and evaluate using precision, recall, and ROC-AUC. **Experiment 5: Decision Trees and Random Forests** Build and compare tree-based models on datasets like Titanic, analyzing feature importance. **Experiment 6: Support Vector Machines (SVM)** Classify data using SVM with different kernels (e.g., Iris, MNIST) and tune hyperparameters. **Experiment 7: Clustering (k-Means and Hierarchical)** Cluster data (e.g., Mall Customer Segmentation) and evaluate using silhouette score.	**30**	CO1, CO2	K3 K4, K5
Module 2:	**Experiment 8: Principal Component Analysis(PCA)** Reduce dimensions of high-dimensional data (e.g., MNIST) and visualize results. **Experiment 9: Neural Networks** Build a basic neural network for image classification (e.g., MNIST, CIFAR-10).	30	CO3, CO4	K3, K4, K6

	Experiment 10: End-to-End Machine Learning Pipeline Develop a complete pipeline (data preprocessing, modeling, evaluation) on a real-world dataset**.**			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	Bishop, C. M., &Nasrabadi, N. M. (2006). Pattern recognition and machine learning (Vol. 4, No. 4, p. 738). New York: springer.			
References/ Readings:	1. Starmer, J. (2022). The Statquest illustrated guide to machine learning. 2. Mitchell, T. M. (1997). Machine learning (Vol. 1). McGraw-hill New York. 3. Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. Cambridge university press. 4. Geron A. (2022) Hands-on Machine Learning with Sci-Learn, Keras&TensorFlow. Shroff/O'Reilly.			
Web Resources:	1. Google. (n.d.). Machine learning crash course. Retrieved May 16, 2025, from https://developers.google.com/machine-learning/crash-course 2. Kaggle. (n.d.). Kaggle learn. Retrieved May 16, 2025, from https://www.kaggle.com/learn			

Title of the Course	Algorithm Design and Data Structures Lab
Course Code	CSI-5007
Number of Credits	2P
Theory/Practical	Practical
Level	400
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-1000			
Course Objectives:	The course objective is to provide hands-on exposure to various data structures and algorithm analysis, including lists, stacks, queues, trees, and various sorting and searching algorithms.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember standard data structures such as stacks and queues.		PSO2, PSO5	
	CO 2. Understand complex structures like AVL and B-trees.		PSO2, PSO5	
	CO 3. Apply appropriate structures to solve given problems.		PSO3	
	CO 4. Evaluate data structure choices in software development.		PSO7	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**

Module 1:	**List of suggested assignments:** 1. Object-Oriented Design Goals, Object-Oriented Design Principles. The programming assignment should introduce and enforce the concepts of encapsulation, polymorphism and Inheritance. 2. Implement Singly Linked Linear Lists and circular linked lists 3. Implement Doubly Linked Linear Lists and Circular linked List 4. Implement Stack using linked list 5. Implement Queue using linked list 6. Implement Binary Trees 7. Implement Binary Search Trees 8. Implement AVL Trees 9. Implement B-Trees and its variants	**30**	CO1, CO2	K3
Unit/ Module 2	1. Program to convert the given infix expression to postfix expression using stack 2. Program to evaluate a postfix expression using stack 3. Program to traverse a binary tree in the following way: Pre- order, In-order, Post-order 4. Write a program to implement Huffman encoding using Binary tree. 5. Write a program to create a binary tree for the given infix expression. 6. Write a program that reads a list of names and telephone number from a textfile and inserts them into an AVL tree. Write afunction to allow the user to search the tree. Searching and sorting 7. Program to implement Binary search technique using Iterative method and Recursive methods. 8. Programs to implement following sorting algorithm-Bubble sort, Selection sort, Insertionsort, Quicksort, Mergesort and Heap sort	**30**	CO3, CO4	K3, K5, K6

	9. Implement assembly line scheduling 10. Implement Matrix-chain multiplication 11. Implement Prim's Algorithm and Kruskal's Algorithm			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	1. Dr. BasantAgarwal, &Baka, B. (2018). Hands-On Data Structures and Algorithms with Python. Packt Publishing Ltd. 2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms. The Mit Press.			
References/ Readings:	1. Dasgupta, S., Papadimitriou, C. H., &Vazirani, U. V. (2006). Algorithms. McGraw-Hill Publishing. 2. Mark Allen Weiss. (2003). Data structures & algorithm analysis in C++. Pearson Education. 3. Horowitz, E., &Sahni, S. (1976). Fundamentals of Data Structures. Computer Science Press, Incorporated.			
Web Resources:	1. Massachusetts Institute of Technology. (2011). 6.006 Introduction to algorithms. MIT OpenCourseWare. Retrieved May 16, 2025, from https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/ 2. GeeksforGeeks. (2025, April 25). DSA tutorial: Learn data structures and algorithms. Retrieved May 16, 2025, from https://www.geeksforgeeks.org/dsa-tutorial-learn-data-structures-and-algorithms/			

Discipline Specific Elective Courses

Title of the Course	Fundamentals of Natural Language Processing
Course Code	CSI-5201
Number of Credits	4T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	Yes

Pre-requisites for the Course:	Nil	
Course Objectives:	To understand the fundamentals of Natural Language Processing (NLP).	
Course Outcomes:	After the completion of this course, the students will be able to	**Mapped to PSO**
	CO 1. Remember core NLP terminologies.	PSO1, PSO2, PSO3
	CO 2. Understand NLP tasks and processing steps.	PSO1, PSO3, PSO5
	CO 3. Apply NLP techniques in real-world applications.	PSO2, PSO5, PSO6
	CO 4. Evaluate outcomes of NLP-based systems.	PSO2, PSO3, PSO4, PSO7, PSO8

Content:		No of hours	Mapped to CO	Cognitive Level
Module 2:1	● **Introduction:** Definition, Natural Language Understanding, Natural Language Generation, Three generations of NLP, NLP trinity, Corpora and their construction, concordance, collocation, regular expressions, Issues and Challenges, NLP applications. ● **Word Sense Disambiguation:** Lexical knowledge networks, Princeton WordNet, Indian language wordnet, WordNet relations, WordNet applications, Idioms and Metaphors. ● **Computational Morphology:** Definition, Agglutination, Types of Morphology.	**15**	CO1, CO2	K1, K2,
Module 2	● **Shallow Parsing:** POS tagging, Chunking, Multi-word expressions, Named entity recognition – techniques, challenges, and applications. ● **Deep parsing:** Constituency parsing, Statistical parsing, Dependency parsing, Scope ambiguity, Attachment ambiguity, rule-based parsing, and statistical parsing.	**15**	CO1, CO2	K1, K2, K3
Module 3:	● **Sentiment Analysis:** Ambiguity – lexical, syntactic, semantic, discourse, pragmatic; Lexicons – manual creation, automatic creation; Rule-based – word level, sentence level, document level; Statistical – Naïve Bayes, Support Vector Machine. ● **Neural networks for NLP:** Review of neural networks basics (Perceptron, Feed forward networks, Back-propagation algorithm). ● **Word embeddings:** Word2vec, Glove, FastText.	**15**	CO2, CO3	K1, K2, K3, K4
Module 4:	**Tutorials and Mini-Projects:** **Suggested Tutorials:** ● Tokenization – word, sentence, character, sub-word, using stop words as delimiter ● Stop word removal, Punctuation removal	**15**	CO4	K1, K2, K3, K4, K5, K6

	● Use of Stemmer and Lemmatizer ● Extracting all nouns in a text ● Finding cosine similarity between two texts **Suggested Mini projects:** ● Develop a POS tagger using a statistical technique. ● Implement a morphological analyzer. ● Implement a model to analyze the sentiment of a given text. ● Generate a summary for a given document. ● Implement a Language Detection system for any 4 languages of your choice. ● Implement a Named Entity Recognition system to identify the named entities from a given text. ● Implement a model to identify the multi-word expressions in a given text. ● Implement a model to identify if the given phrase is used in an idiomatic sense or a regular sense.			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	Bhattacharyya, Pushpak and Joshi, Aditya, Natural Language Processing, 2023.			
References/ Readings:	1. Allen, James, Natural Language Understanding, Second Edition, Benjamin/Cumming, 1995. 2. Charniack, Eugene, Statistical Language Learning, MIT Press, 1993. 3. Jurafsky, Dan, and Martin, James, Speech and Language Processing, Second Edition, Prentice Hall, 2008. 4. Manning, Christopher, and Heinrich, Schutze, Foundations of Statistical 5. Natural Language Processing, MIT Press, 1999.			
Web Resources:	1. Bhattacharyya, P. (n.d.). Natural language processing [Course]. NPTEL, IIT Bombay. Retrieved May 16, 2025, from https://nptel.ac.in/courses/106101007 2. IIT Madras. (n.d.). Introduction to natural language processing (i-NLP) [Course]. Retrieved May 16, 2025, from https://study.iitm.ac.in/ds/course_pages/BSCS5002.html 3. Goayal, P. (n.d.). Natural language processing [Course]. NPTEL, IIT Kharagpur. Retrieved May 16, 2025, from			

	https://onlinecourses.nptel.ac.in/noc19_cs56/preview

Title of the Course	Fundamentals of Robotics
Course Code	CSI-5202
Number of Credits	4T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	NIL	
Course Objectives:	This course introduces robotics fundamentals and applications, covering robotic hardware, motion control systems, communication protocols, and hands-on development of robotics applications using microcontrollers.	
Course Outcomes:		**Mapped to PSO**
	CO 1. Understand the fundamental concepts and evolution of robotics.	PSO1, PSO3, PSO7
	CO 2. Analyze robotic hardware and their drive mechanisms	PSO2, PSO3, PSO6
	CO 3. Work with different sensors and actuators in robotics applications.	PSO2, PSO3, PSO5, PSO6, PSO7
	CO 4. Design and develop robotics applications using embedded systems and AI.	PSO1, PSO2, PSO3, PSO4, PSO5, PSO6, PSO7, PSO 8

Content:		**No of**	**Mapped**	**Cognitive**

		hours	to CO	Level
Module 1	**Fundamentals of Robotics** ○ Definition and Evolution of Robotics ○ Types of Robots (Industrial, Mobile, Humanoid, Swarm, Soft Robots) ○ Applications of Robotics (Healthcare, Manufacturing, Defense, Agriculture) ○ Overview of Robot Kinematics and Dynamics ○ Degrees of Freedom, Joints, and Configurations ○ Actuators and Motion Mechanisms	**15**	CO1	K1, K2, K4
Module 2	**Hardware Components in Robotics** ○ Battery Types & Power Systems ○ Li-ion, LiPo, NiMH, Lead-Acid Batteries ○ Charging Circuits for 2S, 3S, 4S Batteries ○ Balance Charging, Battery Management Systems (BMS) ○ Motors & Drive Systems ○ DC Motors, DC Geared Motors, Servo Motors, Stepper Motors ○ Different Drive Mechanisms: Differential, Ackermann, Mecanum, Omni-Wheel Drive ○ Planetary Gear Systems, Reduction Gear Mechanisms ○ Motor Drivers: L298N, DRV8825, TB6612FNG, ESC (Electronic Speed Controller) ○ Voltage Levels & Converters ○ Step-up (Boost) and Step-down (Buck) Converters ○ Voltage Regulators, Linear & Switching Regulators ○ Logic Level Shifters	**15**	CO2, CO3	K2, K3,K4

Module 3:	**Sensors and Communication in Robotics** • Types of Sensors • Proximity Sensors: Ultrasonic, IR, Lidar, Time-of-Flight Sensors • Human Detection: PIR Sensors, Thermal Cameras • Positioning & Navigation: IMU (Inertial Measurement Unit), Accelerometers, Gyroscopes, GPS • Environmental Sensors: Temperature, Humidity, Air Quality, Water Level Sensors • Line Following & Light Sensors • Communication Protocols for Robotics • Wired Communication: UART, I2C, SPI • Wireless Communication: Bluetooth, NRF24L01, Wi-Fi, 2.4 GHz RF Modules • LoRa and Zigbee-based Communication	**15**	CO3	K2,K3, K4
Module 4:	**Robotics Lab - Prototyping & Application Development** • Experiments and Projects: • Comparative study of Arduino Microcontroller and Raspberry Pi SoC and its application in Smart Robotics. • Line follower robot. • Develop a system for a smart dustbin. • Develop a system for a smart solar panel. • Develop a system for a firefighting robot. • Develop a system for a smart irrigation system. • Study of AI-based Virtual Reality Robotic Gadgets. • Case study of Smart Industry and its applications. • Case study of Robotic Defense applications.	**15**	CO3,CO4	k3,k4,K6

Approved by: Academic Council on 13th & 14th June 2025

	• Application with MQTT. • Develop a system for a surveillance robot. • Develop a mobile app-controlled home cleaning robot. • Develop a system for a gesture-based smart robotic arm. • To develop an obstacle avoidance robot. • To develop an edge detection robot. • To develop a pathfinding robot.			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	**Dr. Robotics. (n.d.).** *Smart robots: Fundamentals, technologies, and applications*.			
References/ Readings:	1. Hunt, V. D. (2011). *Smart robots: A handbook of intelligent robotic systems* (1st ed.). Springer-Verlag New York Inc. 2. Correll, N., Hayes, B., & Wingate, D. (n.d.). *Introduction to autonomous robots*. 3. Siciliano, B., Sciavicco, L., Villani, L., &Oriolo, G. (2010). *Robotics: Modelling, planning and control*. Springer.			
Web Resources:	1. IEEE Spectrum. (n.d.). Robotics. IEEE Spectrum. https://spectrum.ieee.org/topic/robotics/ 2. Asada, H., & Leonard, J. (2005). 2.12 Introduction to Robotics. MIT OpenCourseWare. https://ocw.mit.edu/courses/2-12-introduction-to-robotics-fall-2005/			

Title of the Course	Computer Vision
Course Code	CSI-5203
Number of Credits	4T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	Yes

Pre-requisites for the Course:	Linear Algebra, Calculus, Probability			
Course Objectives:	To understand the basics of computer vision to enable computers to interpret and extract meaningful information from images or videos.			
Course Outcomes:	At the end of the course, the student will be able to		**Mapped to PSO**	
	CO 1. Understand the foundational concepts of Computer Vision.		PSO1	
	CO 2. Apply the basics of image formation, processing, and analysis.		PSO1, PSO2	
	CO 3. Analyze key concepts of different domains and models.		PSO1, PSO2, PSO3, PSO5, PSO8	
	CO 4. Create computer vision applications, including mining of visual content, image rendering, camera surveillance, etc.		PSO3, PSO4, PSO5, PSO6, PSO7, PSO8	
Content:		**No of**	**Mapped**	**Cognitive**

		hours	to CO	Level
Module 1:	● Introduction to computer vision, Image formation fundamentals, Radiometry — measuring light, Sources, shadows and shading, Color.	**15**	CO1, CO2, CO3	K1, K2, K3
Module 2:	● Image Models, Geometric and Analytical Image Features ● Linear filters and convolution, Edge detection. ● Segmentation by clustering: Human vision, applications, segmentation by graph-theoretic clustering. Segmentation by fitting a model, Hough transform, fitting lines, and fitting curves;	**15**	CO2, CO3	K1, K2, K3, K4
Module 3:	● Tracking and Motion ● The Basics of Tracking, Corner Finding, Subpixel Corners, Invariant Features, Optical Flow, Mean-Shift &Camshift Tracking, Motion Templates, Estimators, Lucas-Kanade ● algorithm for optical flow, Multi-scale Lucas-Kanade algorithm, Comparison of Horn-Shunck and Lucas-Kanade algorithms, Applications of optical flow	**15**	CO2, CO3, CO4	K1, K2, K3, K4, K5
Module 4:	● Camera Models and Calibration ● Developing Camera Model, Calibration -Concept of camera calibration and the basic aim of ● Camera calibration, Motivation for camera calibration - implications for 3D ● reconstruction using two calibrated cameras, Un-distortion, Putting Calibration Together, Rodrigues Transform	**15**	CO3, CO4	K1, K2, K3, K4, K5, K6
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	David A Forsynth and Jean Ponce, "Computer vision- A modern approach", Pearson education series, 2003.			
References/ Readings:	1. Milan Sonka, Vaclav Hlavac and Roger Boyle , "Digital image processing and computer vision", Cengage Learning, 2008. 2. Schalkoff R. J., "Digital image processing and computer vision", John Wiley, 2004.			

	3. Sonka M., Hlavac V., Boyle R., "Image processing analysis and machine design". PWS Publishers 4. Ballard D., Brown C., "Computer vision", Prentice Hall
Web Resources:	1. GeeksforGeeks. (n.d.). *Computer vision*. Retrieved May 16, 2025, from https://www.geeksforgeeks.org/computer-vision/ 2. IIT Hyderabad. (n.d.). *Deep learning for computer vision* [Course]. NPTEL. Retrieved May 16, 2025, from https://onlinecourses.nptel.ac.in/noc20_cs88/preview

Title of the Course	Speech Processing
Course Code	CSI-5204
Number of Credits	4T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	Yes

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.	
Course Outcomes:	After completion of this course, students will be able to:	**Mapped to PSO**
	CO 1. Remember foundational speech processing concepts.	PSO1, PSO8
	CO 2. Understand signal processing techniques for feature extraction.	PSO2, PSO5, PSO6
	CO 3. Apply speech models like HMM and DNN.	PSO2, PSO3, PSO5, PSO7
	CO 4. Evaluate ASR systems using performance metrics.	PSO2, PSO4, PSO5, PSO6, PSO7

Content:		No of hours	Mapped to CO	Cognitive Level
Module 1:	● 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**	CO1, CO2	K1, K2, K3
Module 2	● 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**	CO2, CO3	K1, K2, K3, K4
Module 3:	● Hidden Markov Model (HMM) for speech recognition, Viterbi algorithm, Training and testing using HMMs, Adapting to variability in speech (DTW), Language models.	**15**	CO2, CO3, CO4	K1, K2, K3, K4, K5

Approved by: Academic Council on 13th & 14th June 2025

	• 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.			
Module 4:	**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	**15**	CO3, CO4	K1, K2, K3, K4, K5, K6
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Quatieri, T. F. (2002). Discrete-time speech signal processing: principles and practice. Pearson Education India. 2. Martin, J. H., &Jurafsky, D. (2009). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition (Vol. 23). Upper Saddle River: Pearson/Prentice Hall.			
References/ Readings:	1. Rabiner, L. R. (2003). Digital processing of speech signals. Pearson Education India. 2. O'shaughnessy, D. (1999). Speech communications: Human and machine (IEEE). Universities press. 3. Rabiner, L. R., &Juang, B. H. (1999). *Fundamentals of speech recognition*. Tsinghua University Press..			
Web Resources:	1. IIT Madras. (n.d.). Speech signal processing [Course]. NPTEL. Retrieved May 16, 2025, from https://onlinecourses.nptel.ac.in/noc22_ee117/preview 2. Massachusetts Institute of Technology. (2003). 6.345 Automatic speech recognition [Course]. MIT OpenCourseWare. Retrieved May 16, 2025, from https://ocw.mit.edu/courses/6-345-automatic-speech-recognition-spring-2003/			

Title of the Course	Data Science and Data Engineering
Course Code	CSI-5205
Number of Credits	4T
Theory/Practical	Theory
Level	400
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	NIL	
Course Objectives:	This course provides a solid foundation in data science and engineering, covering essential techniques such as data collection, cleaning, processing, and scalable pipeline development. It also emphasizes practical applications, equipping learners with hands-on experience in data visualization, and industry-standard tools to tackle real-world challenges.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Analyze the complete data lifecycle and differentiate between structured, semi-structured, and unstructured data.	PSO1, PSO3, PSO 5
	CO 2. Apply SQL, NoSQL, Hadoop, and Spark for data processing, and configure cloud storage solutions like AWS S3 and Google BigQuery.	PSO1, PSO2, PSO3, PSO5, PSO6, PSO7
	CO 3. Apply data cleaning, transformation, and visualization techniques using Python libraries like Pandas, NumPy, Matplotlib, and Seaborn to uncover	PSO1, PSO2, PSO3, PSO5, PSO7

	insights from real-world datasets.			
	CO 4. Create ETL/ELT data pipelines, orchestrate workflows with Apache Airflow, and deploy models using cloud platforms, Docker, and Kubernetes.		PSO1, PSO2, PSO3, PSO5, PSO6, PSO7	
	CO 5. Evaluate machine learning models using Scikit-learn and process large-scale datasets using distributed computing frameworks like Hadoop and Spark.		PSO1, PSO2, PSO3, PSO5, PSO6, PSO7	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Foundations of Data Science and Data Engineering** ● Introduction to Data Science and Data Engineering: Definitions, scope, and organizational impact ● Overview of the data lifecycle: Data collection, cleaning, processing, analysis, and visualization ● Types of data: Structured, semi-structured, and unstructured data ● Essential tools and technologies: Python, SQL, Hadoop, and Spark ● Database systems: Relational databases (SQL) vs. NoSQL databases (MongoDB, Cassandra) ● Modern data storage solutions: Data lakes, data warehouses, and cloud platforms (AWS S3, Google BigQuery) ● Data quality issues: Missing data, outliers, inconsistencies ● Data transformation techniques: Normalization, scaling, encoding ● Specialized data types: Handling time-series and text data ● Exploratory Data Analysis (EDA): Descriptive statistics, summary metrics ● Data visualization tools and techniques: Matplotlib, Seaborn, Plotly, Tableau	**15**	CO1, CO2, CO3	K2, K3,K4, K6
Module 2	**Unit 2: Machine Learning, Big Data Processing, and Deployment** ● Introduction to Machine Learning: Supervised, unsupervised, and	**15**	CO4, CO5	K2, K3, K4, K5

	reinforcement learning ● Common ML algorithms: Linear regression, decision trees, k-means clustering ● Model evaluation metrics: Accuracy, precision, recall, F1-score, ROC-AUC ● Big Data fundamentals: The four Vs – Volume, Velocity, Variety, Veracity ● Distributed computing frameworks: Hadoop and Spark ● Scalable data pipeline architecture: ETL and ELT processes ● Workflow orchestration tools: Apache Airflow, Luigi ● Stream processing systems: Kafka, Apache Flink ● Cloud platforms for data engineering: AWS, Azure, Google Cloud Platform ● Containerization and orchestration: Docker, Kubernetes ● Infrastructure as Code (IaC): Terraform ● Overview of end-to-end deployment strategies in data projects			
Module 3:	Lab Experiments (Data Engineering) 1. **Basic ETL Process** Create a simple ETL pipeline using Python to extract data from CSV files, transform it (clean missing values, normalize columns), and load it into a SQLite database. Analyze the transformation steps and their impact on data quality. 2. **Database Query Optimization** Compare the performance of optimized versus unoptimized SQL queries on a medium-sized dataset. Experiment with adding appropriate indexes and measure execution time improvements. 3. **Data Visualization Dashboard** Build a basic dashboard using Python libraries (Matplotlib, Plotly) to	**15**	CO4, CO5	K3, K4, K5, K6

	visualize insights from a dataset, incorporating interactive elements that allow filtering and drill-down capabilities. 4. **File Format Comparison** Analyze the same dataset stored in different formats (CSV, JSON, Parquet) and compare processing speed, storage efficiency, and query performance across each format. 5. **Simple Data Pipeline Scheduling** Implement a scheduled data pipeline using tools like cron or Airflow that automatically extracts data at regular intervals, performs basic transformations, and updates a target database with new information.			
Module 4:	**Data Science Practical Lab Experiments** 1. **Exploratory Data Analysis** Analyze a real-world dataset using Python (pandas, matplotlib) to identify patterns, outliers, and relationships. Create visualizations that highlight key insights and present a summary of findings. 2. **Classification Model Comparison** Build and compare multiple classification algorithms (Decision Tree, Logistic Regression, Random Forest) on the same dataset. Evaluate performance using metrics like accuracy, precision, and recall. 3. **Clustering for Customer Segmentation** Apply K-means clustering to a customer dataset to identify natural groupings. Visualize the clusters, interpret their characteristics, and suggest how these segments might guide business decisions. 4. **Time Series Forecasting** Use historical time series data to build a simple forecasting model (moving averages, ARIMA) that predicts future values. Evaluate forecast accuracy and visualize predictions against actual values.	**15**	,CO3, CO4	K3,K4.K5
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			

Texts:	1. Reis, J., &Housley, M. (2022). *Fundamentals of data engineering: Plan and build robust data systems* (Grayscale Indian Edition). 2. Grus, J. (2019). *Data science from scratch: First principles with Python*. O'Reilly Media.
References/ Readings:	1. McKinney, W. (2022). *Python for Data Analysis: Data Wrangling with pandas, NumPy, and IPython* (3rd ed.). O'Reilly Media. 2. Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (3rd ed.). O'Reilly Media. 3. Grus, J. (2019). *Data Science from Scratch: First Principles with Python* (2nd ed.). O'Reilly Media. 4. Bruce, P., Bruce, A., &Gedeck, P. (2020). *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python* (2nd ed.). O'Reilly Media. 5. Leskovec, J., Rajaraman, A., & Ullman, J. D. (2020). *Mining of Massive Datasets* (3rd ed.). Cambridge University Press.
Web Resources:	1. KDNuggets. (n.d.). KDNuggets: News. Retrieved May 19, 2025, from https://www.kdnuggets.com/news/index.html 2. Towards Data Science. (n.d.). Towards Data Science. Retrieved May 19, 2025, from https://www.towardsdatascience.com

SEMESTER II

Discipline Specific Core Courses

Title of the Course	Fundamentals of Deep Learning and Generative AI Techniques
Course Code	CSI-5008
Number of Credits	2T
Theory/Practical	Theory
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	N0
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5000, CSI-5001,CSI-5002	
Course Objectives:	This course covers deep learning fundamentals including MLPs, CNNs, and RNNs, with focus on generative models like VAEs and GANs. Students learn optimization techniques, regularization methods, and advanced strategies such as WGAN, applying these skills to real-world image, text, and music generation tasks while exploring AI's creative potential.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Apply foundational deep learning techniques such as MLPs, CNNs, and RNNs to solve classification and sequence modeling problems.	PSO 1, PSO5

	CO 2. Create generative models including VariationalAutoencoders (VAEs) and Generative Adversarial Networks (GANs) for image, text, and music generation.		PSO2, PSO6	
	CO 3. Apply advanced optimization strategies like regularization, batch normalization, and learning rate scheduling to fine-tune deep learning models for improved performance.		PSO3, PSO5	
	CO 4. Evaluate generative models in creative applications by addressing challenges like mode collapse and applying models such as WGAN and CycleGAN.		PSO4, PSO7	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Foundations and Core Architectures** ● History of Deep Learning McCulloch-Pitts Neuron, Perceptron Algorithm MLPs, Sigmoid Neurons Gradient Descent, Backpropagation Optimization (GD, Momentum, Adam) CNN Architectures (LeNet, AlexNet, VGG, ResNet) RNNs: BPTT, LSTM, GRU **Model Optimization and Regularization** ● Feedforward NN training ● Batch Normalization Regularization (L1/L2, dropout) Learning Rate Scheduling Autoencoders (Basic, Denoising, Sparse) Comparison: Autoencodersvs PCA	**15**	CO1, CO3	K2, K3, K4, K5
Module 2	**Generative Deep Learning Models** ● Introduction to Generative Models Generative vs Discriminative VariationalAutoencoders (VAEs) Latent Space Arithmetic, Face Generation	**15**	CO2, CO4	K3, K4, K5

	GAN Architecture (Generator, Discriminator) GAN Training Challenges (Mode Collapse, etc.) **Advanced Generative Models and Applications** WGAN, WGAN-GP CycleGAN for Style Transfer Neural Style Transfer Text Generation with LSTM Music Generation with MuseGAN Transformer Architecture (Self-Attention, Positional Encoding) Applications: BERT, GPT, Vision Transformers (ViT)			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Goodfellow, I., Bengio, Y., &Courville, A. (2016). *Deep learning*. MIT Press. 2. Foster, D. (2020). *Generative deep learning: Teaching machines to paint, write, compose, and play* (1st ed.). O'Reilly Media.			
References/ Readings:	1. Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media. 2. Bishop, C. M. (2006). *Pattern recognition and machine learning* (1st ed.). Springer. 3. Chollet, F. (2021). *Deep learning with Python* (2nd ed.). Manning Publications.			
Web Resources:	1. DeepLearning.AI. (n.d.). DeepLearning.AI. Retrieved May 19, 2025, from https://www.deeplearning.ai 2. fast.ai. (n.d.). fast.ai. Retrieved May 19, 2025, from https://www.fast.ai			

Title of the Course	Reinforcement Learning
Course Code	CSI-5009
Number of Credits	2T
Theory/Practical	Theory
Level	500
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil			
Course Objectives:	To enable the student to understand core concepts of reinforcement learning.			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember RL principles and terminology.		PSO1, PSO4	
	CO 2. Understand Markov Decision Processes.		PSO1, PSO5	
	CO 3. Apply model-based and model-free techniques.		PSO1, PSO2, PSO5	
	CO 4. Analyze real-world tasks as RL problems.		PSO2, PSO5, PSO6	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Introduction**	15	CO1, CO2	K1, K2,

	● Background, Supervised, Unsupervised and Reinforcement Learning, RL framework, Limitations, Examples. **Multi-Arm Bandits** ● k-armed bandit problem, Exploration and Exploitation, Greedy, Epsilon greedy, Upper Confidence Bound, Gradient Bandit algorithms, Contextual Bandits. **Finite Markov Decision Processes** ● Agent, Environment, Goals, Rewards, Return, Episodic and Continuing tasks, Policies, Value functions, Optimal policies and value functions, Bellman Equations, Backup diagrams. **Dynamic Programming** ● Policy evaluation, Policy improvement, Policy iteration, Value iteration, Asynchronous Dynamic Programming. **Monte Carlo Methods** ● Monte Carlo Prediction, Action value estimation, Control, Off-policy prediction, Off-policy control. **Temporal Difference Learning** ● TD Prediction, Advantages, On-policy TD control – SARSA, Off-policy TD control – Q learning, Expected SARSA, Maximization bias and double learning, n-step TD prediction, n-step SARSA, n-step off-policy learning.			K3, K4, K5
Module 3:	**Planning and Learning** ● Models, Dyna, Prioritized sweeping, Expected vs. Sample updates, Trajectory Sampling, Real time DP, Heuristic search, Rollout algorithms, Monte Carlo tree search. **On-policy Prediction with Approximation** ● Value function approximation, prediction objective, Stochastic-gradient and Semi-gradient Methods, Linear Methods, Non-linear function approximation, Memory based function approximation, Kernel based	15	CO3, CO4	K3, K4, K5, K6

	function approximation. **On-policy Control with Approximation** ● Episodic Semi-gradient Control, Semi-gradient n-step Sarsa, Deprecating the Discounted Setting, Differential Semi-gradient n-step Sarsa. **Off-policy Methods with Approximation** ● Semi-gradient Methods, Off-policy Divergence, The Deadly Triad, Bellman Error, Gradient-TD Methods, Emphatic-TD Methods, Eligibility Traces. **Policy Gradient Methods** ● Policy Approximation, Advantages, Policy Gradient Theorem, REINFORCE, Actor–Critic Methods, Policy Parameterization.			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	Sutton, R. S., &Barto, A. G. (1998). *Reinforcement learning: An introduction* (Vol. 1, No. 1, pp. 9-11). Cambridge: MIT press.			
References/ Readings:	1. Szepesvári, C. (2022). Algorithms for reinforcement learning. Springer nature. 2. Lattimore, T., &Szepesvári, C. (2020). Bandit algorithms. Cambridge University Press. 3. Lapan, M. (2024). Deep Reinforcement Learning Hands-On. Packt Publishing Ltd.			
Web Resources:	Hugging Face. (n.d.). Introduction to Deep Reinforcement Learning [Online course]. Retrieved May 19, 2025, from https://huggingface.co/learn/deep-rl-course/unit0/introduction			

Title of the Course	Big Data Frameworks
Course Code	CSI-5010
Number of Credits	2T
Theory/Practical	Theory
Level	500
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-1000	
Course Objectives:	The course objective is to equip students with a comprehensive understanding of Big Data, the challenges faced in storing and analyzing it, and the workings of big data platforms, with a specific focus on Apache Hadoop and its ecosystem & Apache Spark.	
Course Outcomes:	At the end of the course, the students will be able to:	**Mapped to PSO**
	CO 1. Remember big data concepts and architectures.	PSO1
	CO 2. Understand big data analytics platforms.	PSO1
	CO 3. Apply distributed frameworks like Hadoop and Spark.	PSO2, PSO3, PSO5, PSO6
	CO 4. Evaluate real-world case studies using big data tools.	PSO5

Content:		No of hours	Mapped to CO	Cognitive Level
Module 1:	**Introduction to Big Data** ● Big Data and its characteristics, Big Data Analytics, Challenges faced in storage, querying and analysis of big data; need for big data frameworks **Hadoop Framework:** ● **Apache Hadoop Basics**: Introduction, Hadoop 1 v/s Hadoop 2, Use cases and anti-patterns, Main components of Hadoop, Hadoop Ecosystem ● **Hadoop Distributed Filesystem (HDFS):** HDFS Architecture, HDFS daemons and their roles, data blocks, replication policy, Handling node & disk failures, Namenode startup operation, checkpointing process, HDFS file read & write process, Error handling in read/write, NameNode resilience ● **Mapreduce:** Introduction to MapReduce framework, Map, Shuffle-Sort and Reduce Phases, Input Splits, Word Count problem, Data Flow & Daemons in MapReduce, Partitions, Combiner Functions. ● **Apache YARN:** YARN applications, YARN daemons, Anatomy of a YARN application run	**15**	CO1, CO2, CO3	K1, K2, K3, K4
Module 2	**MapReduce Programming** ● **Mapreduce Programming in Java:** Hadoop Data Types, Input & Output formats, Record Reader, Record Writer, Sample Mapreduce programs ● **Advanced Mapreduce:** Chaining jobs, Joining data (reduce-side join, replicated joins, semi-join), Secondary sorting. **Hadoop Ecosystem:** ● **Apache PIG**: PIG and its use, Execution Modes, Grunt Shell and Grunt commands, Data Model (Relations, Bags, Tuples, Fields), Pig Latin	**15**	CO 2, CO 3, CO 4	K1, K2, K4, K5, K6

Approved by: Academic Council on 13th & 14th June 2025

	Basics. ● **Apache Hive**: Introduction and need, Hive Architecture, Metastore, Schema on Read, Hive Tables (Managed and External), Partitions, Buckets. ● **Apache HBase:** Introduction and need, Data Model, Architecture, Metadata, API (Get, Put, Scan) **Apache Spark:** ● **Apache Spark Basics**: Introduction and Need, Spark v/s Hadoop, Use Cases and antipatterns, Spark Components, Spark Program Flow, Resilient Distributed Dataset (RDD), Actions & Transformations on Basic and Pair RDD. ● **SparkQL:** Spark SQL vs. Traditional SQL Databases, DataFrames and Datasets, Spark SQL Functions and Queries			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. White, T. (2015). Hadoop: The Definitive Guide (4th ed.). O'Reilly Media 2. Jean Georges Perrin. (2020). Spark in Action. Manning Publications Company.			
References/ Readings:	1. Lam, C. (2011). Hadoop in Action. Manning Publications. 2. Holmes, A. (n.d.). Hadoop in Practice. 3. Deroos, D., Zikopoulos, P., Brown, B., Coss, R., &Melnyk, R. B. (2014). Hadoop for Dummies. John Wiley & Sons, Inc.			
Web Resources:	1. Overview. (2017). Apache.org. https://pig.apache.org/docs/latest/index.html 2. RDD Programming Guide - Spark 3.0.0 Documentation. (n.d.). Spark.apache.org. https://spark.apache.org/docs/latest/rdd-programming-guide.html			

Title of the Course	MLOp
Course Code	CSI 5011
Number of Credits	2T
Theory/Practical	Theory
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5002, CSI-5006	
Course Objectives:	1. To equip students with the knowledge and skills to design, implement, and manage end-to-end machine learning operations (MLOps) pipelines 2. It also enables them to deploy, monitor, and maintain scalable, reliable, and ethical ML systems in production environments.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Create end-to-end MLOps pipelines and ML workflows.	PSO1, PSO2, PSO5, PSO6
	CO 2. Apply containers, cloud platforms, and CI/CD tools to deploy ML models.	PSO2, PSO6, PSO7
	CO 3. Analyze ML systems through logging, drift detection, and performance tracking.	PSO2, PSO4, PSO6,

	CO 4. Evaluate ethical challenges including bias mitigation, fairness, and operational excellence in ML systems.		PSO4, PSO5,PSO8	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Introduction to MLOps :** ● Evolution and Importance, Definition, Goals, and Key Components, MLOps vs DevOps, MLOps Hierarchy of Needs (Data, Model, Code, Infrastructure) ● MLOps Foundations: Bash& Linux Command Line (Basic commands, scripting),Cloud Computing Basics (AWS, GCP, Azure),Minimal Python for MLOps, Descriptive Statistics and Optimization ● Introduction to MLOps Pipelines: Data ingestion, Model training, Model deployment **MLOps in Production** ● Containers & Edge Devices: Docker (Basics, Best Practices),Model Serving (APIs over HTTP),Edge AI (Coral, Azure Percept, TFHub),Porting models to Edge ● Continuous Delivery for ML: Packaging Models,Infrastructure as Code (Terraform, CloudFormation),Controlled Rollout (Canary, A/B Testing),Testing (Unit, Integration, Model Validation)	**15**	CO1, CO2	K2, K3, K4, K5
Module 2	**Advanced MLOps Tools and Practices :** ● AutoML and Continuous Improvement:AutoML Tools (Google AutoML, Azure AutoML, Ludwig, FLAML). KaizenML ● Feature Stores and Explainability:Versioning and managing features,Interpretable ML Techniques ● Monitoring and Logging:Observability in MLOps,Logging in Python.Model Monitoring (Data drift, Performance degradation).Drift Monitoring Tools (SageMaker, Azure ML)	**15**	CO3,CO 4	K2,K3 ,K4,K6

	● MLOps Applications and Interoperability : ● MLOps on AWS:SageMaker, Lambda, EC2,Serverless MLOps (Flask, AWS SAM) ● Interoperability and Microservices:ONNX (Conversion, Deployment),Python Packaging,CLI Tools for MLOps,Microservices and Authentication ● Case Studies and Challenges:Real-WorldMLOps Case Studies,Ethics in MLOps (Bias, Fairness),Operational Challenges			
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Treveil, M., Omont, N., Stenac, C., Lefevre, K., Phan, D., &Zentici, J. (2020). Introducing MLOps: How to Scale Machine Learning in the Enterprise. O'Reilly Media. 2. Burkov, A. (2020). Machine Learning Engineering. True Positive Inc.3.			
References/ Readings:	1. Treveil, M., Omont, N., Stenac, C., Lefevre, K., Phan, D., &Zentici, J. (2020). Introducing MLOps: How to Scale Machine Learning in the Enterprise. O'Reilly Media. 2. Burkov, A. (2020). Machine Learning Engineering. True Positive Inc. 3. Hapke, H., & Nelson, C. (2020). Building Machine Learning Pipelines: Automating Model Life Cycles with TensorFlow. O'Reilly Media.			
Web Resources:	1. Neptune.ai. (n.d.). How to learn MLOps. Retrieved May 19, 2025, from https://neptune.ai/blog/how-to-learn-mlops 2. DataCamp. (n.d.). 10 awesome resources for learning MLOps. Retrieved May 19, 2025, from https://www.datacamp.com/blog/10-awesome-resources-for-learning-mlops			

Title of the Course	Fundamentals of Deep Learning and Generative AI Techniques Lab
Course Code	CSI-5012
Number of Credits	2P
Theory/Practical	Practical
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5000,CSI-5001, CSI-5002	
Course Objectives:	This course provides practical experience in building, training, and evaluating deep learning and generative models, enabling students to translate theoretical knowledge into real-world AI applications involving image, text, and audio data.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Apply core deep learning architectures such as MLPs, CNNs, and RNNs.	PSO1, PSO2
	CO 2. Apply optimization, regularization, and training strategies effectively.	PSO1, PSO6
	CO 3. Evaluate generative models like VAEs and GANs.	PSO1, PSO5, PSO7
	CO 4. Create creative AI applications in image, text, and audio domains.	PSO3, PSO4, PSO5

Content:		No of hours	Mapped to CO	Cognitive Level
Module 1:	**Lab 1: Implementing Perceptrons and MLPs** ● Train and test perceptrons on binary classification tasks. ● Implement MLPs with different hidden layers for XOR problem. **Lab 2: Training Deep Networks** ● Implement gradient descent and backpropagation manually. ● Compare optimizers (SGD, Adam, RMSProp) on MNIST. **Lab 3: Autoencoders** ● Build and train a basic autoencoder. ● Apply denoising and sparse variants on image datasets. **Lab 4: CNN Architectures and Visualization** ● Train CNNs (LeNet, VGG) on CIFAR-10. ● Visualize filters, feature maps, and use DeepDream.	**30**	CO1, CO2, CO3, CO4	K3, K4, K6
Module 2:	**Lab 5: Sequence Modeling with RNNs** ● Implement LSTM for sentiment classification. ● Build encoder-decoder model for sequence translation. **Lab 6: VariationalAutoencoders** ● Build and train a VAE on face dataset. ● Visualize latent space interpolation and morphing. **Lab 7: Generative Adversarial Networks** ● Implement a basic GAN for MNIST. ● Experiment with WGAN and WGAN-GP. **Lab 8: Creative Applications with Generative Models** ● Train CycleGAN for image style transfer.	**30**	CO2, CO3, CO4	K3, K4,K6

	● Use LSTM for poem generation. ● Compose polyphonic music using MuseGAN.			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	1. Goodfellow, I., Bengio, Y., &Courville, A. (2016). *Deep learning*. MIT Press. 2. Foster, D. (2020). *Generative deep learning: Teaching machines to paint, write, compose, and play* (1st ed.). O'Reilly Media.			
References/ Readings:	1. Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media. 2. Bishop, C. M. (2006). *Pattern recognition and machine learning* (1st ed.). Springer. 3. Chollet, F. (2021). *Deep learning with Python* (2nd ed.). Manning Publications.			
Web Resources:	1. DeepLearning.AI. (n.d.). DeepLearning.AI. Retrieved May 19, 2025, from https://www.deeplearning.ai/ 2. Keras. (n.d.). Keras: The Python deep learning API. Retrieved May 19, 2025, from https://keras.io/ 3. PyTorch. (n.d.). PyTorch tutorials. Retrieved May 19, 2025, from https://pytorch.org/tutorials/ 4. Sourcell, S. (n.d.). llSourcell'sGitHub repositories. GitHub. Retrieved May 19, 2025, from https://github.com/llSourcell			

Title of the Course	Reinforcement Learning Lab
Course Code	CSI-5013
Number of Credits	2P
Theory/Practical	Practical
Level	500
Effective from AY	2025-26
New Course	No
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	Nil	
Course Objectives:	To enable the student to implement the core concepts of reinforcement learning.	
Course Outcomes:	At the end of the course, the students will be able to:	**Mapped to PSO**
	CO 1. Remember libraries used in RL implementations.	PSO2, PSO5
	CO 2. Understand the application flow of RL models.	PSO1, PSO2, PSO5
	CO 3. Apply RL techniques to complex tasks.	PSO2, PSO5
	CO 4. Evaluate performance metrics of RL algorithms.	PSO2, PSO5, PSO6

Content:		No of hours	Mapped to CO	Cognitive level
Module 1:	**RL Task Formulation:** Designing a real-world problem into the RL framework by defining action space, state space, agent, environment, rewards, and other components. **Multi-Arm Bandits:** Implementing MAB algorithms for balancing exploration and exploitation in RL problems. **Dynamic Programming:** Implementing dynamic programming algorithms for policy optimization in RL problems. **Monte Carlo Methods:** Implementing Monte Carlo algorithms for policy evaluation in RL problems. **Temporal Difference Learning:** Implementing TD learning methods for policy evaluation and improvement in RL problems.	**30**	CO1, CO2	K3, K4, K5, K6
Module 2:	**Off-policy Model-Free Algorithm:** Implement Q-learning for solving RL problems. **On-Policy Model-Free Algorithm:** Implement SARSA for solving RL problems. **Deep Q-Learning:** Implement DQN to solve a simple environment in OpenAI Gym. **Policy Gradient Methods:** Implement REINFORCE algorithm for direct policy optimization in RL. **Actor-Critic Methods:** Implement Actor-Critic methods and compare performance with Q-learning and SARSA.	**30**	CO3, CO4	K3, K4, K5, K6
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	Sutton, R. S., &Barto, A. G. (1998). *Reinforcement learning: An introduction* (Vol. 1, No. 1, pp. 9-11). Cambridge: MIT press.			

References/ Readings:	1. Szepesvári, C. (2022). Algorithms for reinforcement learning. Springer nature. 2. Lattimore, T., &Szepesvári, C. (2020). Bandit algorithms. Cambridge University Press. 3. Lapan, M. (2024). Deep Reinforcement Learning Hands-On. Packt Publishing Ltd.
Web Resources:	Hugging Face. (n.d.). Introduction to Deep Reinforcement Learning [Online course]. Retrieved May 19, 2025, from https://huggingface.co/learn/deep-rl-course/unit0/introduction

Title of the Course	Big Data Frameworks Lab
Course Code	CSI-5014
Number of Credits	2P
Theory/Practical	Practical
Level	500
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-1000			
Course Objectives:	The course objective is to provide hands-on experience in the storage and processing of big data with the help of tools like Apache Hadoop, Pig, Hive and Apache Spark			
Course Outcomes:	At the end of the course, the students will be able to:		**Mapped to PSO**	
	CO 1. Remember file storage and access methods in Hadoop/Spark.		PSO2	
	CO 2. Understand data flow in distributed systems.		PSO2, PSO5	
	CO 3. Apply big data analytics to large datasets.		PSO2, PSO5	
	CO 4. Create complete big data solutions using Spark or Hadoop ecosystem.		PSO2, PSO5, PSO7	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**

Module 1:	**List of suggested assignments:** 1. Install Hadoop and configure it to run in the pseudo-distributed mode. Understand the different startup scripts and configuration files. 2. Using HDFS commands, implement file management tasks in Hadoop such as adding files and directories, retrieving files, deleting files, copying files, moving files, merging files, and appending content to files. 3. Run a basic Word Count MapReduce program to understand the MapReduce paradigm. 4. Write MapReduce programs to a. Find the average rating of movies. b. Find the number of times each user of age > 25 has rated a movie. c. Implement Matrix Multiplication d. Mine weather data. 5. Install Pig and write a Pig Latin script to sort, group, join, project, and filter your data. 6. Install and run Hive, then use Hive to create, alter, and drop databases, tables, views, functions, and indexes. Using HiveQL, sort, group, join, project, and filter your data.	**30**	CO1, CO2, CO3, CO4	K3, K4, K5
	7. Install pySpark (can be done using Docker container) and create RDDs, perform actions and transformations on the RDDs 8. Using pySpark, load a text file as a RDD and perform word count. 9. Convert a JSON file into a DataFrame. Run SQL queries to to sort, group, join, project, and filter your data. **10. Mini project** Solve some real-life big data problems. ● Traffic control using big data ● Medical insurance fraud detection ● Recommendation system	**30**		

	● Anomaly detection in cloud servers ● Tourist behavior analysis ● Web server log analysis			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	1. White, T. (2015). Hadoop: The definitive guide (4th ed.). O'Reilly Media 2. Jean Georges Perrin. (2020). Spark in Action. Manning Publications Company.			
Web Resources:	1. Apache Hadoop 3.3.1 – Hadoop: Setting up a Single Node Cluster. (n.d.). Hadoop.apache.org. https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html 2. Overview. (2017). Apache.org. https://pig.apache.org/docs/latest/index.html 3. RDD Programming Guide - Spark 3.0.0 Documentation. (n.d.). Spark.apache.org. https://spark.apache.org/docs/latest/rdd-programming-guide.html			

Title of the Course	MLOP Lab
Course Code	CSI-5015
Number of Credits	2P
Theory/Practical	Practical
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5002, CSI-5006	
Course Objectives:	• Enable students to build complete MLOps pipelines covering data ingestion, model training, deployment, and automation. • Provide experience in deploying scalable and reliable ML models using cloud platforms, containers, and monitoring tools. • Instil practices for maintaining ethical, fair, and responsible AI systems in real-world production environments.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Create MLOps pipelines with end-to-end workflows for data ingestion, model training, and deployment.	PSO1, PSO2, PSO5, PSO6
	CO 2. Apply containers, cloud platforms, and CI/CD tools to deploy ML models in production for scalable deployments.	PSO2, PSO6, PSO7
	CO 3. Apply logging, monitoring, and drift detection techniques to monitor and maintain	PSO2, PSO6, PSO4

	ML systems.			
	CO 4. Evaluate ethical and operational challenges by detecting bias, ensuring fairness, and balancing accuracy with operational excellence.		PSO4, PSO2, PSO8	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	**Lab 1: Introduction to MLOps Tools (4 hours)** ● Set up Python environment (Anaconda, Jupyter Notebook). ● Explore MLOps tools (MLflow, DVC, Kubeflow). ● Version control for ML projects using Git and DVC. **Lab 2: Data Pipeline Development (6 hours)** ● Ingest and preprocess data using Pandas and PySpark. ● Build a data pipeline with Apache Airflow or Prefect. ● Version datasets using DVC. **Lab 3: Model Training and Experiment Tracking (6 hours)** ● Train a machine learning model (e.g., Scikit-learn, TensorFlow). ● Track experiments using MLflow. ● Log metrics, parameters, and artifacts. **Lab 4: Containerization with Docker (4 hours)** ● Docker basics: Create and run containers. ● Containerize a trained ML model. ● Push Docker images to a container registry (Docker Hub, AWS ECR).	**30**	CO1, CO2	K2, K3, K4
Module 2:	**Lab 5: CI/CD for ML Pipelines (6 hours)** ● Set up CI/CD pipelines using GitHub Actions or GitLab CI. ● Automate testing and deployment of ML models. ● Integrate with cloud platforms (AWS/GCP/Azure). **Lab 6: Model Deployment and Serving (6 hours)**	**30**	CO1, CO2, CO3 CO4,	K3, K4, K5, K6

	● Deploy models as REST APIs using Flask/FastAPI. ● Serve models using Kubernetes(Minikube or cloud-managed Kubernetes). ● Monitor API performance with Prometheus and Grafana. **Lab 7: Monitoring and Logging (4 hours)** ● Set up logging for ML models using Python's logging module. ● Monitor model performance and data drift using Evidently or WhyLabs. ● Visualize metrics with dashboards (Grafana, TensorBoard). **Lab 8: Ethical AI and Fairness (4 hours)** ● Detect bias in datasets using AI Fairness 360 or Fairlearn. ● Mitigate bias using reweighting or adversarial debiasing. ● Evaluate fairness metrics (demographic parity, equal opportunity). Mini Capstone Project The capstone project integrates concepts from the labs into a comprehensive MLOps workflow. Students work in teams to solve a real-world problem.			
Pedagogy:	Hands-on/ Tutorials/ Presentation			
Texts:	1. **Treveil, M., Omont, N., Stenac, C., Lefevre, K., Phan, D., &Zentici, J.** (2020). *Introducing MLOps: How to Scale Machine Learning in the Enterprise*. O'Reilly Media. 2. **Burkov, A.** (2020). *Machine Learning Engineering*. True Positive Inc.			
References/ Readings:	1. **Hapke, H., & Nelson, C.** (2020). *Building Machine Learning Pipelines: Automating Model Life Cycles with TensorFlow*. O'Reilly Media. 2. **Gift, N., &Deza, A.** (2021). *Practical MLOps: Operationalizing Machine Learning Models*. O'Reilly Media.			
Web Resources:	1. MLOps Community. (n.d.). MLOps. Retrieved May 19, 2025, from https://mlops.community/MLOps 2. AlmaBetter. (n.d.). MLOps tutorials. Retrieved May 19, 2025, from https://www.almabetter.com/bytes/tutorials/mlops			

Discipline Specific Elective Courses

Title of the Course	Machine Translation
Course Code	CSI-5206
Number of Credits	4T
Theory/Practical	Theory
Level	500
Effective from AY	2025-26
New Course	Yes
Bridge Course/ Value-added Course	No
Course for advanced learners	Yes

Pre-requisites for the Course:	Basics of NLP, Basics of Machine Learning and Deep Learning	
Course Objectives:	To provide students with a thorough understanding of machine translation paradigms, models, and evaluation techniques, enabling them to analyze and implement various MT approaches including rule-based, statistical, example-based, and neural machine translation.	
Course Outcomes:	After the completion of this course, the students will be able to	**Mapped to PSO**
	CO 1. Remember machine translation approaches and architectures.	PSO1, PSO5, PSO6
	CO 2. Understand evaluation methods for MT systems.	PSO1, PSO2, PSO3, PSO4, PSO5
	CO 3. Apply tools to build translation models.	PSO1, PSO2, PSO3, PSO6, PSO8

	CO 4. Create components for MT or preprocessing pipelines.		PSO2, PSO3, PSO5, PSO6	
Content:		**No of hours**	**Mapped to CO**	**Cognitive Level**
Module 1:	● **Introduction**: History of MT, MT Applications, Data-driven MT, MT Approaches, Language divergence, three major paradigms of MT. ● **Word-Based Models**: Translating Words, Lexical Translation Models, Higher IBM Models, Word Alignment. ● **Phrase-Based Models**: Standard Model, Phrase Translation Table, Translation Model extensions, Reordering Model extensions, EM Training of Phrase-Based Models.	**15**	CO1 K1, K2	CO1 K1, K2
Module 2:	● **Decoding**: Translation Process, Beam Search, Future Cost Estimation. ● **Rule-Based Machine Translation (RBMT)**: Kinds, UNL, Interlingua and Word Knowledge, UNL conversion, Transfer-based MT.	**15**	CO2, CO3, CO4	K1, K2, K3, K6
Module 3:	● **Example-Based Machine Translation (EBMT)**: Essential steps of EBMT, Text similarity computation, Translation memory, Statistical Machine Translation ● **MT Evaluation**: Manual Evaluation, Automatic Evaluation, Hypothesis Testing, Task-Oriented Evaluation	**15**	CO2, CO3, CO4	K1, K2, K3, K5, K6
Module 4:	● **Introduction to NMT**: History of NMT, Challenges in NMT ● **Neural Language Models**: Feed-Forward Neural Language Models, Word Embedding, RNN, GRU, LSTM ● **Neural Translation Models**: Encoder-Decoder Approach, Alignment Model, Beam Search, CNN, CNN with attention	**15**	CO2, CO3, CO4	K1, K2, K3, K4, K5
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	Machine Translation by Pushpak Bhattacharyya, Chapman and Hall/CRC, February 2015			

References/ Readings:	1. Machine Translation on Coursera by Prof. Alexander Waibel and Jan Niehueshttps: //www.coursera.org/learn/machinetranslation 2. An Open Source Neural Machine Translation System https://opennmt.net/ 3. Bhashini Project – https://bhashini.gov.in/bhashadaan/en/likho-india
Web Resources:	1. Bhattacharyya, P. (n.d.). Introduction to machine translation and its evaluation [Video]. YouTube. Retrieved May 19, 2025, from https://www.youtube.com/watch?v=8BTk9ERyEKI 2. Coursera. (n.d.). Machine translation [Online course]. Retrieved May 19, 2025, from https://www.coursera.org/learn/machinetranslation 3. GeeksForGeeks. (n.d.). Machine translation in AI. Retrieved May 19, 2025, from https://www.geeksforgeeks.org/machine-translation-of-languages-in-artificial-intelligence/

Title of the Course	Robotic motion planning, control and Programming
Course Code	CSI-5207
Number of Credits	4
Theory/Practical	theory
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5202, CSI-5001	
Course Objectives:	To equip students with foundational knowledge of robotics fundamentals, design, programming, sensors, and hardware integration using ROS, Arduino, and Raspberry Pi.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Understand robotics design, kinematics, dynamics, and motion control.	PSO1, PSO3,PSO5
	CO 2. Apply robotic programming and ROS fundamentals.	PSO1, PSO2, PSO 6
	CO 3. Analyze appropriate sensors and actuators for robotic applications.	PSO 1, PSO 3, PSO 5
	CO 4. Design robotic systems using ROS integrated with Arduino and Raspberry Pi.	PSO 2, PSO 6, PSO7, PSO 8

Content:		No of hours	Mapped to CO	Cognitive Level
Module 1:	**Fundamentals** ● Introduction – Components, Degrees of Freedom, Joints, Coordinates, Mechanisms, Controller. **Kinematics** ● Position and Orientation of Objects, Coordinate Transformation, Joint Variables and Position of End ● Effector, Inverse Kinematics Problem, Jacobian Matrix, Statics and Jacobian Matrices. **Dynamics** ● Lagrangian and Newton-Euler Formulations, Derivation of Dynamics Equations Based on Lagrangian ● Formulation, Derivation of Dynamic Equations Based on Newton-Euler, Formulation, Use of Dynamics ● Equations and Computational Load, Identification of Manipulator Dynamics. **Manipulability** ● Manipulability Ellipsoid and Manipulability Measure, Best Configurations of Robotic Mechanisms from ● Manipulability Viewpoint, Various Indices of Manipulability, Dynamic Manipulability.	**15**	CO1	K2, K3, K4, K5

Module 2:	**Position Control** ● Generating a Desired Trajectory, Linear Feedback Control, Two-Stage Control by Linearization and Servo ● Compensation, Design and Evaluation of Servo Compensation, Decoupling Control, Adaptive Control. **Force Control** ● Impedance Control - Passive-Impedance Method, Active-Impedance Method-One- Degree-of- Freedom ● Case, Active-Impedance Method-General Case. **Hybrid Control** ● Hybrid Control - Hybrid Control via Feedback Compensation, Dynamic Hybrid Control.	**15**	CO1, CO4	K3, K4, K5
Module 3:	**Programming Assignments 1** 1. Understanding programmable robot simulator in Hill, C. (2020). Learning scientific programming with Python. Cambridge University Press. 2. Programming different capabilities control concerns of the robot like moving around free space. 3. Write a program to control inputs from sensors. 4. Write a program to control outputs of the robot. 5. Understanding and using API for robot simulator	**15**	CO2, CO3, CO4	K2, K3, K4

Module 4:	**Programming Assignments 2** 1. Write a program to apply physics rules to robot movements. 2. Write a program to implement collision with obstacles. 3. Implementing a simple model of a robot. Assumptions: Terrain is always flat and even, obstacles are never round, Wheels never slip, Sensors never fails or give false reviews, The wheels always turn when they are told to. • Write a program to program the behavior inside the robot. • Write a program to implement the control aspects of a robot i.e : Apply control signal, message the results, generate new control signals to bring to our goal.	**15**	CO1, CO2, CO4	K2, K3, K4, K5, K6
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	**Text Book(s)** 1. Tsuneo Yoshikawa, "Foundations of Robotics Analysis and Control", The MIT Press Cambridge,1990. 2. Saeed B Niku, "Introduction to Robotics Analysis, Control, Applications", 3rd Edition, Wiley, 2020. 3. Lentin Joseph, Robot Operating System (ROS) for Absolute Beginners: Robotics Programming Made Easy, 1 st Edition, APress, 2018. 4. Jonathan Cacace; Lentin Joseph, Mastering ROS for Robotics Programming: Design, build, and simulate complex robots using the Robot Operating System, 2nd Edition, Packt Publishing, 2018.			
References/ Readings:	1. Robert J. Schilling, "Fundamentals of Robotics, Analysis and Control", Prentice Hall India, 2003. 2. John J. Craig, "Introduction to Robotics, Mechanics and Control", 3rd Edition, Pearson Prentice Hall,2005. 3. Hughes, C. and Hughes, T., Robot programming: a guide to controlling autonomous robots. Que Publishing, 2016 4. Quigley, M., Gerkey, B. and Smart, W.D., Programming Robots with ROS: a practical introduction to the Robot Operating System. " O'Reilly Media, Inc.", 2015 5. Anil Mahtani, Luis Sanchez, Enrique Fernandez, Aaron Martinez, Lentin Joseph. ROS Programming: Building Powerful Robots. Packt Publishing, 2018.			
Web Resources:	Toptal. (n.d.). Programming a robot: An introductory tutorial. Retrieved May 19, 2025, from https://www.toptal.com/robotics/programming-a-robot-an-introductory-tutorial			

Title of the Course	Interpretable Machine Learning
Course Code	CSI-5208
Number of Credits	4T
Theory/Practical	theory
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5002, CSI-5006	
Course Objectives:	This course is aimed at developing foundational knowledge and practical skills in model interpretability, ethical assessment, and explainability in machine learning, with a focus on emerging trends and MLOps integration.	
Course Outcomes:	At the end of the course, the student will be able to:	**Mapped to PSO**
	CO 1. Understand key concepts of interpretation, interpretability, and explainability.	PSO1, PSO2, PSO3
	CO 2. Apply interpretation methods to tabular, image, and text data.	PSO1, PSO2, PSO3
	CO 3. Evaluate bias, fairness, and robustness in machine learning models.	PSO1, PSO2, PSO3
	CO 4. Analyze model interpretability for real-world deployment.	PSO1, PSO2

Content:		No of hours	Mapped to CO	Cognitive Level
Module 1:	**Foundations of Interpretability:** ● Interpretation vs. Interpretability vs. Explainability, White-box vs. Black-box, Business relevance, Interpreation, model agnostic interpretation	**15**	CO1	K2, K3
Module 2:	**Interpretation Methods and Challenges** ● Model-agnostic interpretation (global/local), glass-box models (EBM, GAMI-Net), trade-offs	**15**	CO4	K4,K5
Module 3:	**Domain-Specific Interpretability Techniques**: ● CNNs, Transformers, NLP, Time Series, Feature Selection, Visualization techniques	**15**	CO2, CO4	K3, K4,K5
Module 4:	**Ethics, Bias, Robustness, and Future of Interpretability:** ● Bias mitigation, causal inference, fairness constraints, adversarial robustness, future outlook	**15**	,CO3, CO4	k4.K5.K6
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Christoph Molnar, *Interpretable Machine Learning*, 2nd Edition. 2. Mark Molloy et al., *Machine Learning Interpretability in Action*, Manning Publications			
References/ Readings:	1. Patrick Hall, Navdeep Gill, *Machine Learning Interpretability*, O'Reilly. 2. Been Kim et al., *Interpretability and Explainability in AI Systems*, Springer. 3. Finale Doshi-Velez et al., *Accountability in AI: Interpretable Machine Learning*, MIT Press.			
Web Resources:	1. Ribeiro, M. T. (n.d.). LIME: Local Interpretable Model-agnostic Explanations [GitHub repository]. GitHub. Retrieved May 19, 2025, from https://github.com/marcotcr/lime 2. Lundberg, S. M. (n.d.). SHAP: SHapley Additive exPlanations [GitHub repository]. GitHub. Retrieved May 19, 2025, from https://github.com/shap/shap			

Title of the Course	Explainable Artificial Intelligence (XAI)
Course Code	CSI-5209
Number of Credits	4T
Theory/Practical	Theory
Level	500
Effective from AY	2025-26
New Course	yes
Bridge Course/ Value added Course	No
Course for advanced learners	No

Pre-requisites for the Course:	CSI-5000			
Course Objectives:	This course is aimed at learning to apply foundational XAI techniques to interpret model behavior across data types and communicate explainability insights effectively to diverse stakeholders.			
Course Outcomes:	At the end of the course, the student will be able to:		**Mapped to PSO**	
	CO 1. Understand the fundamental concepts, types, and stakeholders of Explainable AI.		PSO1, PSO4	
	CO 2. Apply explainability techniques to various machine learning models and data types.		PSO2, PSO3, PSO6	
	CO 3. Compare explainability methods using appropriate metrics.		PSO8, PSO4	
	CO 4. Evaluate the communication, regulatory, ethical, and usability aspects of XAI.		PSO4, PSO7	
Content:		**No of hours**	**Mapped to CO**	

Module 1:	**Foundations of Explainable AI:** ● Introduction to XAI, importance, stakeholders, types of explanations (premodeling, local/global, post hoc), interpretability, taxonomy	**15**	CO1	K1, K2, K3, K4
Module 2:	**Explainability for Structured and Visual Data:** ● Explainability for tabular and image data using SHAP, PDP, ICE, Grad-CAM, LIME, Guided Backprop, XRAI, etc.	**15**	CO2	K3, K4, K5
Module 3:	**Explainability for Text and Advanced Topics**: ● LIME for text, embeddings, attention, LRP, alternative and multimodal explainability, time series, evaluation methods	**15**	CO2, CO3	K3, K4, K5, K6
Module 4:	**Deployment, Interaction, and Future Directions:** ● Presenting explanations, interacting with stakeholders, pitfalls, regulatory aspects, ML lifecycle, and future of explainability	**15**	CO3, CO4	K4, K5, K6
Pedagogy:	Lectures/ Assignments/ Flipped Classroom			
Texts:	1. Munn, M., & Pitman, D. (2022). *Explainable AI for practitioners: Designing and implementing explainable ML solutions* (Grayscale Indian ed.). O'Reilly Media. 2. Dan Becker, Margaret Mitchell – *Interpretable Machine Learning with Python* (O'Reilly Media, 2023)			
References/ Readings:	1. Sameer Singh et al. – *Explainable AI in Practice* (Springer, 2021) 2. Patrick Hall et al. – *Machine Learning for High-Stakes Decisions* (O'Reilly, 2024) 3. WojciechSamek – *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (Springer, 2019)			
Web Resources:	1. IBM. (n.d.). Explainable AI. Retrieved May 19, 2025, from https://www.ibm.com/think/topics/explainable-ai 2. TDAN. (n.d.). Explainable AI: 5 open-source tools you should know. Retrieved May 19, 2025, from https://tdan.com/explainable-ai-5-open-source-tools-you-should-know/31589			