ATMANIRBHAR BHARAT
SWAYAMPURNA GOA

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

(Accredited by NAAC)

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

GU/Acad –PG/BoS -NEP/2024/449 Date: 22.08.2024

CIRCULAR

Ref. No.: GU/Acad –PG/BoS -NEP/2023/184/2 dated 04.07.2023

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

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

ASHWIN VYAS LAWANDE
Digitally signed by ASHWIN VYAS LAWANDE
Date: 2024.08.22 16:59:52 +05'30'

(Ashwin V. Lawande)
Deputy Registrar – Academic

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

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

M.Sc. in Artificial Intelligence to be effective from Academic Year 2023-24

Program Specific Outcomes

PSO1: Develop knowledge of core programming skills and techniques for designing and developing AI-based applications

PSO2: Apply principles of code design and development in Artificial intelligence-based applications

PSO3: Build proficiency in using data science tools & techniques and handling big data

PSO4:Apply artificial intelligence tools and techniques such as problem-solving, knowledge representation, machine learning, computer vision, human-computer interaction for managing (mis)information diffusion in research and industry domains.

PSO5: Know ethical, legal, social, and professional frameworks to make responsible decisions in computing and data science

Pathway

- Fundamentals (Mathematics and Problem Solving, Programming)
- Core Courses (AI, Machine Learning, Deep Learning, etc.,)
- Specialization (Natural Language Processing, Computer Vision)
- Research and Dissertation (Core Research – Language Models, or Application Oriented Research or Product Based Research (MLOps, DevOps, Design Thinking, Pragmatic AI)

As per the above pathway vision, the structure for programme has been designed as follows

M.Sc. IN ARTIFICIAL INTELLIGENCE TO BE EFFECTIVE FROM ACADEMIC YEAR 2023-24		
SEMESTER I – Total 20 credits		
DISCIPLINE SPECIFIC CORE (DSC) COURSES		
Course Code	**Course Title**	**Credits**
*CSI-500	Fundamentals of Artificial Intelligence	**2**
CSI-501	Fundamentals of Artificial Intelligence Lab	**2**
CSI-502	Algorithms and Data structures	**2**
CSI-503	Algorithms and Data structures Lab	**2**
CSI-504	Mathematical Foundations for Artificial Intelligence	**2**
CSI-505	Mathematical Foundations for Artificial Intelligence Lab	**2**
*CSI-506	Data Science Fundamentals	**2**
CSI-507	Data Science Fundamentals Lab	**2**
	Total Credits	**16**
DISCIPLINE SPECIFIC ELECTIVE (DSE) COURSES – any one to be opted from DSC List		
Course Code	**Course Title**	**Credits**
CSI-521	Natural Language Processing	**4**
CSI-522	Computer Vision	**4**
CSI-523	Robotics	**4**
CSI-524	IoT Architecture and Protocol	**4**
	Total Credits	**4**

SEMESTER II – Total 20 credits		
DISCIPLINE SPECIFIC CORE (DSC) COURSES		
Course Code	**Course Title**	**Credits**
CSI-508	Deep Learning	**2**
CSI-509	Deep Learning Lab	**2**
CSI-510	Big Data Frameworks	**2**
CSI-511	Big Data Frameworks Lab	**2**
CSI-512	Reinforcement Learning	**2**
CSI-513	Reinforcement Learning Lab	**2**
CSI-514	Software Engineering for AI Enabled systems	**2**
CSI-515	Software Engineering for AI Enabled systems Lab	**2**
	Total Credits	**16**
DISCIPLINE SPECIFIC ELECTIVE (DSE) COURSES – any one to be opted from the DSE List		
Course Code	**Course Title**	**Credits**
CSI-525	Machine translation	**4**
CSI-526	Mathematics for computer vision and robotics	**4**
CSI-527	Soft computing	**4**
CSI-528	Regression and Predictive Analytics	**4**
CSI-529	Essential of Data Analytics	**4**
	Total Credits	**4**

SEMESTER III – Total 20 credits		
RESEARCH SPECIFIC ELECTIVE (RSE) COURSES – any two to be opted from the RSE List		
Course Code	**Course Title**	**Credits**
CSI-600	Research Methodology	**4**
CSI-601	Generative Deep Learning Models	**4**
CSI-602	MLOps	**4**
CSI-603	Cloud Computing	**4**
CSI-604	Design thinking	**4**
	Total Credits	**8**
GENERIC ELECTIVE (GE) COURSES - total 12 credits to be opted from GE list specified below		
List/Categories of Generic Elective (GE) Courses		
Course Code	**Course Title**	**Credits**
CSA-621	Corporate Skills offered by Computer Science	**4**
	Courses offered by other Disciplines from GBS during the respective Semester	**4**
	Courses offered by other Disciplines from other Schools during the respective Semester	**4**
	Courses offered under MOOC during the respective Semester and approved by DFC	**4**
	Course code marked with '*'be offered as Generic Electives for students of other discipline from M.Sc. AI Programme	
	Total Credits	**12**

SEMESTER IV – Total 20 credits		
One RESEARCH SPECIFIC ELECTIVE (RSE) Course to be opted from the RSE list given below in consultation with the Mentor. It can be completed in Semester 3.		
Course Code	**Course Title**	**Credits**
CSI-605	Speech Processing	**4**
CSI-606	Advanced Machine Translation	**4**
*CSI-607	Data Engineering	**4**
CSI-608	Financial Machine Learning	**4**
CSI-609	Recommender Systems	**4**
Total Credits for RSE for semester IV		**4**
Course Code	**Course Title**	**Credits**
Dissertation Type		**Credits**
CSI-651	Research Project in Academic or Research Institutes/ Industry	**16**
Total Credits for Dissertation		**16**

SEMESTER I
DISCIPLINE SPECIFIC CORE COURSES

Name of the Programme	**: M.Sc. Artificial Intelligence**
Course code	**: CSI-500**
Title of the course	**: Fundamentals of Artificial Intelligence**
Number of credits	**: 2(2L+0T+0P)**
Total contact hours	**: 30 hours(30L-0T-0P)**
Effective from AY	**: 2023-24**

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

	AI. - Applications and examples. - Trust and acceptance-Evaluation methods and metrics. - Ethical, legal and social issues of explainable AI. Contemporary issues in AI- Philosophy, Ethics, and Safety of AI -The Future of AI	
Pedagogy	Tutorials / Hands-on-assignments / Self-study	
References/ Reading	1. GF Luger, (2002). Artificial Intelligence, Pearson Education, 2002. 2. M.C. Trivedi, (2019). A Classical Approach to Artificial Intelligence, Khanna Book Publishing. 3. Nilsson, N. J. (1998). Artificial intelligence: a new synthesis. Morgan Kaufmann. 4. Padhy, N. P. (2005). Artificial intelligence and intelligent systems (Vol. 337). Oxford: Oxford University Press. 5. Russell, S. J., & Norvig, P. (2010). Artificial intelligence a modern approach. London. 6. V., Rich, E., Knight, K., & Nair, S. (2009). Artificial Intelligence. Tata McGraw Hill. 7. https://www.edx.org/course/artificial-intelligence-ai 8. https://www.udemy.com/course/artificial-intelligence-az/	
Course Outcomes	1. Understand the basic concepts and techniques of Artificial Intelligence. 2. Apply AI algorithms for solving practical problems. 3. Describe human intelligence and AI. 4. Explain Expert System and implementation, neural network and fuzzy logic	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-501**
Title of the Course : **Fundamentals of Artificial Intelligence Lab**
Number of Credits : **2 (0L+0T+2P)**
Total Contact Hours : **60 hours (0L+0T+2P)**
Effective from AY : **2023-24**

Prerequisites for the course:	Artificial Intelligence theory, probability and statistics, linear algebra and Python programming	
Course Objectives:	To develop a basic understanding of problem solving, knowledge representation, reasoning and learning methods of AI and implement AI algorithms	
Content:	Assignment-1 -Real-world path planning for pedestrians. In the first part, students implement A* over a map that includes roads/paths as well as elevations. In the second part, students collect actual data through walking around the real world, and the cost model is then learned via regression techniques.	**10 hours**
	Assignment-2 -Solve maze via search -this assignment involves formulating maze-solving as a search problem, image processing (via OpenCV) as a step in maze-solving, as well as guided performance/quality analysis of representational parameters.	**10 hours**
	Assignment 3-Within the context of an artificial intelligence course, students are taught to identify ethical issues within technical projects and to engage in moral problem solving with regard to such issues.	**10 hours**
	Assignment 4-Neural network for face recognition using tensor flow -build feedforward neural networks for face recognition using TensorFlow. Students then visualize the weights of the neural networks they train. The visualization allows students to understand feedforward one-hidden layer neural networks in terms of template matching, and allows students to explore overfitting.	**10 hours**
	Assignment -5 -Organic path finding -Students develop a "human-like" pathfinding technique by specializing a generic search algorithm with custom action cost and heuristic cost functions. Students apply classical search algorithms and reflect on example organic paths to achieve "human-like" pathfinding.	**10 hours**
	Assignment - 6 -Implement a genetic algorithm in Python to evolve strategies for Robby the Robot to collect empty soda cans that lie scattered around his rectangular grid world. And also Compare the performances of a brute-force search and a search employing the Minimum Remaining Values (MRV) heuristic in solving Sudoku	**10 hours**

	puzzles.	
Pedagogy:	lectures/practical/ tutorials/assignments/self-study	
References /Readings:	1. GF Luger, (2002). Artificial Intelligence, Pearson Education, 2002. 2. M.C. Trivedi, (2019). A Classical Approach to Artificial Intelligence, Khanna Book Publishing. 3. Nilsson, N. J. (1998). Artificial intelligence: a new synthesis. Morgan Kaufmann. 4. Padhy, N. P. (2005). Artificial intelligence and intelligent systems (Vol. 337). Oxford: Oxford University Press. 5. Russell, S. J., & Norvig, P. (2010). Artificial intelligence a modern approach. London. 6. V., Rich, E., Knight, K., & Nair, S. (2009). Artificial Intelligence. Tata McGraw Hill. 7. https://www.edx.org/course/artificial-intelligence-ai 8. https://www.udemy.com/course/artificial-intelligence-az/	
Course Outcomes:	1. The students need to understand existing implementation of algorithms 2. learn to extend an existing implementation of the back-propagation algorithm and use it to recognize static hand gestures in images. 3. Students learn about feedforward neural networks and the backpropagation algorithm by implementing a perceptron network for AND and XOR Boolean functions and, given an implementation of a feedforward network, learn digit recognition using the MNIST data set. 4. students extend a Tic Tac Toe program to Ultimate Tic Tac Toe and implement a different search strategy than the example code.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-502**
Title of the Course : **Algorithms and Data Structure**
Number of Credits : **2 (2L+0T+0P)**
Total Contact Hours : **30 hours(30L+0T+0P)**
Effective from AY : **2023-24**

Prerequisites for the course:	Programming in Python	
Course Objectives:	The aim of the course is to introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing efficient algorithms. It provides an exposure to various data structures and algorithm analysis including lists, stacks, queues, trees, and various sorting and searching algorithms.	
Content:	**Unit-I** Introduction: Three level Approach - Application/User level, Abstract/Logical level, Physical/Implementation level; Concept of Abstract Data Types (ADTs), Data Structure definition, Data type vs. 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 Hours**
	Unit-II: 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 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 hours**
Pedagogy:	Practical/ tutorials/assignments/self-study	
References /Readings:	1. Agarwal, B., & Baka, B. (2018). Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7. Packt Publishing Ltd. 2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms. MIT press. 3. Dasgupta, S., & Papadimitriou, C. H. (2006). Algorithms.	

	4. Horowitz, E., & Sahni, S. (1982). Fundamentals of data structures. 5. Weiss, M. A. (2011). Data Structures and Algorithm Analysis in C. Pearson Education India.
Course Outcomes:	1. Understanding of various data structures. 2. Proficiency in algorithmic problem-solving. 3. Practical implementation and application of data structures. 4. Enhanced critical thinking and problem analysis skills.

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-503**
Title of the Course : **Algorithms and Data Structure Lab**
Number of Credits : **2 (0L+0T+2P)**
Total Contact Hours : **60 hours (0L+0T+2P)**
Effective from AY : **2023-24**

Prerequisites for the course:	Programming in Python	
Course Objectives:	The aim of the course is to introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing efficient algorithms. It provides an exposure to various data structures and algorithm analysis including lists, stacks, queues, trees, and various sorting and searching algorithms.	
Content:	1. Object-Oriented Design Goals, Object-Oriented Design Principles. The programming assignment should introduce and enforce the concepts of encapsulation, polymorphism and Inheritance.	**20*3=60 Hours**
	2. Singly Linked Linear Lists and Circular Linked List	
	3. Doubly Linked Linear Lists and Circular linked List	
	4. Stack using linked list	
	5. Queue using linked list	
	6. Binary Trees	
	7. Binary Search Trees	
	8. AVL Trees	
	9. B-Trees and its variants	
	10. Program to convert the given infix expression to postfix expression using stack	
	11. Program to evaluate a postfix expression using stack.	
	12. Program to traverse a binary tree in the following way: Pre-order, In-order, Post-order	
	13. Write a program to implement Huffman encoding using Binary tree.	
	14. Write a program to create a binary tree for the given infix expression.	
	15. Write a program that reads a list of names and telephone	

	number from a text file and inserts them into an AVL tree. Write a function to allow the user to search the tree. Searching and sorting	
	16. Program to implement Binary search technique using Iterative method and Recursive methods.	
	17. Programs to implement following sorting algorithm- Bubble sort, Selection sort, Insertion sort, Quicksort, Merge sort and Heap sort	
	18. Assembly line scheduling	
	19. Matrix-chain multiplication	
	20. Prim‟s Algorithm and Kruskal‟s Algorithm	
Pedagogy:	Lectures/Practical/ tutorials/assignments/self-study	
References/Readings:	1. Agarwal, B., & Baka, B. (2018). Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7. Packt Publishing Ltd. 2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to algorithms. MIT press. 3. Dasgupta, S., & Papadimitriou, C. H. (2006). Algorithms. 4. Horowitz, E., & Sahni, S. (1982). Fundamentals of data structures. 5. Weiss, M. A. (2011). Data Structures and Algorithm Analysis in C. Pearson Education India.	
Course Outcomes:	1. Implement common data structures such as lists, stacks, queues, graphs, and binary trees for solving programming problems. 2. Identify and use appropriate data structures in the context of a solution to a given problem. 3. Learn to understand the implementation issues 4. Overall learn the foundation required for programming	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-504**
Title of course : **Mathematical Foundations for Artificial Intelligence**
Number of credits : **2 (2L-0T-0P)**
Total contact hours : **30 hours (30L-0T-0P)**
Effective from AY : **2023-24**

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

	Basics of ordinary and partial differential equations -Applications of Calculus **Optimization** Differentiation of Univariate Functions-Partial Differentiation and Gradients-Gradients of Vector-Valued Functions-Gradients of Matrices Useful Identities for Computing Gradients-Backpropagation and Automatic Differentiation-Higher-Order Derivatives-Linearization and Multivariate Taylor Series-Gradient Descent-Constrained Optimization -Lagrange Multipliers-Convex Optimization,	
Pedagogy	Problem solving approach and carrying out small project work using matlab tools	
References/ Readings	1. Gel'fand, I. M., Glagoleva, E. G., &Shnol, E. E. E. (1990). Functions and graphs (Vol. 1). Springer Science & Business Media. 2. Gonick, L. (2012). The cartoon guide to calculus. (No Title). 3. Lay, D. C. (2003). Linear algebra and its applications. Pearson Education India. 4. McClave, J. T., Benson, P. G., &Sincich, T. (2008). Statistics for business and economics. Pearson Education. 5. Optimization Methods in Business Analytics — edX, MIT 6. Savov, I. (2017). No bullshit guide to linear algebra. (No Title). 7. Sternstein, M. (2017). Barron's AP statistics. Simon and Schuster. 8. Strang, G. (2022). Introduction to linear algebra. Wellesley-Cambridge Press. 9. Wheelan, C. (2013). Naked statistics: Stripping the dread from the data. WW Norton & Company. 10. Witte, R. S., & Witte, J. S. (2017). Statistics. John Wiley & Sons.	
Course Outcomes	1. Strong understanding of mathematical concepts relevant to AI. 2. Application of mathematics in AI problem-solving. 3. Proficiency in quantitative analysis and data interpretation. 4. Development of algorithmic thinking skills for AI algorithms.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course code : **CSI-505**
Title of the course : **Mathematical Foundation for AI using Matlab**
Number of credits : **2 (0L-0T-2P)**
Total contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

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

	Assignment- 7-Program in python to implement the concepts such as Vector space, subspace, span, coumn space, row space, null space, left-null space, rank, basis, orthogonal matrix, symmetric matrix	**6 hours**
	Assignment -8 – Implement Eigen value decomposition in python.	**6 hours**
	Assignment-9 – implement SVD using python.	**6 hours**
	Assignment -10 – implement some of optimization algorithm using the python library	**6 hours**
Pedagogy	lab assignments /Project	
References/ Readings	1. McClave, J. T., Benson, P. G., &Sincich, T. (2008). Statistics for business and economics. Pearson Education. 2. Sternstein, M. (2017). Barron's AP statistics. Simon and Schuster. 3. Strang, G. (2022). Introduction to linear algebra. Wellesley-Cambridge Press. 4. Wheelan, C. (2013). Naked statistics: Stripping the dread from the data. WW Norton & Company. 5. Witte, R. S., & Witte, J. S. (2017). Statistics. John Wiley & Sons.	
Course Outcomes	1. Practical application of mathematical concepts in AI. 2. Proficiency in data manipulation, analysis, and visualization. 3. Implementation and experimentation with AI algorithms. 4. Development of critical thinking and problem-solving skills in AI.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-506**
Title of the Course : **Data Science Fundamentals**
Number of Credits : **2(2L+ 0T+ 0P)**
Contact hours : **30 hours (30L+0T+0P)**
Effective from AY : **2023-24**

Prerequisites for the course	Statistics and Probability theory and Python Programming	
Course Objectives	To get started with basics of Data Science and learn all aspects of Data Science in its entirety	
Content	**Unit-I** Introduction: Typology of problems - Data science in a big data world: Benefits and uses of data science and big data-Facets of data-The data science process-The big data ecosystem and data science-The data science process:Overview of the data science process- Defining research goals and creating a project charter-Retrieving data-Cleansing, integrating, and transforming data-Exploratory data analysis-Build the models- Presenting findings and building applications on top of them.	**15 Hours**
	Mathematics for Data science - Importance of linear algebra, statistics and optimization from a data science perspective; Structured thinking for solving data science problems. Linear Algebra: Matrices and their properties (determinants, traces, rank, nullity, etc.); Eigenvalues and eigenvectors; Matrix factorizations; Inner products; Distance measures; Projections; Notion of hyperplanes; half-planes. Probability, Statistics and Random Processes: Probability theory and axioms; Random variables; Probability distributions and density functions (univariate and multivariate); Expectations and moments; Covariance and correlation; Statistics and sampling distributions; Hypothesis testing of means, proportions, variances and correlations; Confidence (statistical) intervals; Correlation functions; White-noise process. Data clearing (EDA)	**3 hours**
	Introduction to Data Science Methods: Linear regression as an exemplar function approximation problem; Linear classification problems-PCA	**4 hours**
	Unit-II Introduction to NoSQL The rise of graph databases Introducing connected data and graph databases	**15 Hours**

	Introducing Neo4j: a graph database Data visualization to the end user Data visualization options Crossfilter, the JavaScript MapReduce library Creating an interactive dashboard with dc.js Dashboard development tools	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study	
References / Readings	1. Baesens, B. (2014). Analytics in a big data world: The essential guide to data science and its applications. John Wiley & Sons. 2. Bruce, P., Bruce, A., &Gedeck, P. (2020). Practical statistics for data scientists: 50+ essential concepts using R and Python. O'Reilly Media. 3. Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: Springer. 4. McKinney, W. (2022). Python for data analysis. O'Reilly Media, Inc. 5. Taddy, M. (2019). Business data science. 6. Wheelan, C. Naked Statistics: Stripping the Dread from the Data.	
Course Outcomes	1. Understanding of data science principles. 2. Proficiency in data manipulation and preprocessing. 3. Ability to visualize and communicate data insights. 4. Knowledge of statistical analysis and predictive modeling techniques.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course code : **CSI-507**
Title of the course : **Data Science Fundamentals Lab**
Number of credits : **2(0L+0T+2P)**
Total contact hours : **60 hours(0L-0T-60P)**
Effective from AY : **2023-24**

Prerequisites for the course	Basic programming skills, Statistics	
Course Objectives	1. To introduce Basic process of data science, Python and Jupyter notebooks. 2. To understanding how to manipulate and analyzeuncurated datasets 3. To learn basic statistical analysis and machine learning methods and effectively visualize results	
Content	Jupyter and Numpy: Jupyter notebooks are one of the most commonly used tools in data science as they allow you to combine your research notes with the code for the analysis. After getting started in Jupyter, we'll learn how to use numpy for data analysis. numpy offers many useful functions for processing data as well as data structures which are time and space efficient.	**10 hours**
	Pandas: Pandas, built on top of numpy, adds data frames which offer critical data analysis functionality and features.	**10 hours**
	Visualization: When working with large data sets you often need to visualize your data to gain a better understanding of it. Also, when you reach conclusions about the data, you'll often wish to use visualizations to present your results.	**10 hours**
	Mini Project: With the tools of Jupyter notebooks, numpy, pandas, and Visualization, you're ready to do sophisticated analysis on your own. You'll pick a dataset we've worked with already and perform an analysis for this first project.	**10 hours**
	Machine Learning: To take your data analysis skills one step further, we'll introduce you to the basics of machine learning and how to use sci-kit learn - a powerful library for machine learning.	**10 hours**
	Working with Text and Databases: You'll find yourself often working with text data or data from databases. This week will give you the skills to access that data. For text data, we'll also give you a preview of how to analyze text data using ideas from the field of Natural Language Processing and how to apply those ideas using the Natural Language Processing Toolkit (NLTK) library.	**5 hours**
	Mini-Project	**5 hours**

Pedagogy	Tutorials/ Lab assignments/ Project work
References/ Readings	1. Baesens, B. (2014). Analytics in a big data world: The essential guide to data science and its applications. John Wiley & Sons. 2. Bruce, P., Bruce, A., &Gedeck, P. (2020). Practical statistics for data scientists: 50+ essential concepts using R and Python. O'Reilly Media. 3. Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: Springer. 4. McKinney, W. (2022). Python for data analysis. O'Reilly Media, Inc. 5. Taddy, M. (2019). Business data science. 6. Wheelan, C. Naked Statistics: Stripping the Dread from the Data.
Course Outcomes	1. Application of data science techniques to real-world problems. 2. Proficiency in data acquisition and preprocessing. 3. Ability to perform exploratory data analysis. 4. Building and evaluating predictive models.

(Back to Index)

DISCIPLINE SPECIFIC ELECTIVE COURSES

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI-521
Title of the Course : Natural Language Processing
Number of Credits : 4 (2L-2T-0P)
Contact Hours : 60 hours (30L-30T-0P)
Effective from AY : 2023-24

<table>
<tr><td>Prerequisites for the course</td><td colspan="2">Fundamentals of Artificial Intelligence; Mathematical Foundations for Artificial Intelligence.
Machine Learning and Programming background. Introduction to NLP (Theory), Mathematical foundations for AI.</td></tr>
<tr><td>Course Objectives</td><td colspan="2">This course will focus on understanding the essentials of Natural Language Processing (NLP), areas in NLP, algorithms, and NLP tasks.
Students who complete this course will gain a foundational understanding in natural language processing methods and strategies. They will also learn how to evaluate the strengths and weaknesses of various NLP technologies and frameworks as they gain practical experience in the NLP toolkits available.</td></tr>
<tr><td rowspan="2">Content:</td><td>Unit-I
Part I: Foundations of Natural Language Processing
Introduction
● Natural Language Processing - Problems and perspectives
● Introduction/Recall to/of probability calculus
  ○ N-grams and Language Models
  ○ Markov Models
● Introduction to Machine Learning and Deep Learning
● Recurrent Neural Network Language Models
● The evaluation of NLP applications
Corpora
● Corpora and their construction: representativeness
● Concordances, collocations and measures of words association
● Methods for Text Retrieval
● Regular expressions</td><td>15 hours</td></tr>
<tr><td><u>Unit II</u>
Part II: Natural Language Processing
● Computational Phonetics and Speech Processing
  ○ Speech samples: properties and acoustic measures
  ○ Analysis in the frequency domain, Spectrograms
  ○ Applications in the acoustic-phonetic field.
  ○ Speech recognition with HMM and Deep Neural Networks
● Tokenisation and Sentence splitting
● Computational Morphology</td><td>15 hours</td></tr>
</table>

	- Morphological operations - Static lexica, Two-level morphology • Computational Syntax - Part-of-speech tagging - Grammars for natural language - Natural language Parsing - Supplementary worksheet: formal grammars for NL ▪ Formal languages and Natural languages. Natural language complexity ▪ Phrase structure grammars, Dependency Grammars ▪ Treebanks ▪ Modern formalisms for parsing natural languages • Computational Semantics - Lexical semantics: WordNet and FrameNet - Word Sense Disambiguation - Distributional Semantics & Word-Space models - Logical approaches to sentence semantics	
	Part III: Applications and Case studies: • Solving Downstream Tasks: Document classification, Sentiment Analysis, Named Entity Recognition, Semantic Textual Similarity • Prompting Pre-Trained Language Models • Network Embedding	

	Sample list of Assignments and a Mini Project using all these functionalities to be discussed and implemented during Tutorial Slots Assignment -1 -Import nltk and download the 'stopwords' and 'punkt' packages. Assignment-2 -Import spacy and load the language model. Assignment -3 -How to tokenize a given text? Assignment-4 -How to get the sentences of a text document? Assignment- 5-How to tokenize a text using the `transformers` package? Assignment -6 - How to tokenize text with stopwords as delimiters? Assignment- 7- How to remove stop words in a text? Assignment -8- How to add custom stop words in spaCy? Assignment- 9 -How to remove punctuations? Assignment-10 - How to perform stemming? Assignment -11 -How to lemmatize a given text? Assignment-12 -How to extract usernames from emails? Assignment -13-How to find the most common words in the text excluding stopwords Assignment -14- How to do spell correction in a given text? Assignment -15- How to tokenize tweets? Assignment -16- How to extract all the nouns in a text? Assignment -17- How to extract all the pronouns in a text? Assignment - 18 - How to find similarity between two words? Assignment -19- How to find similarity between two documents? Assignment -20 -How to find the cosine similarity of two documents?	**20 * 1 = 20 hours for Assignment Discussion + 10 hours for a Mini Project**
Pedagogy	Hands-on assignments/tutorials / peer-teaching / pair programming/presentations / mini-project. Lectures / Practical / tutorials / assignments / self-study / mini-project	
References/ Readings	1. Allen, James. Natural Language Understanding, Second Edition, Benjamin/Cumming, 1995. 2. Charniack, Eugene. Statistical Language Learning, MIT Press, 1993. 3. Deep Learning by Goodfellow, Bengio, and Courville (free online). 4. Jurafsky, Dan and Martin, James. Speech and Language Processing, Second Edition, Prentice Hall, 2008. 5. Machine Learning — A Probabilistic Perspective by Kevin Murphy (online). 6. Manning, Christopher and Heinrich, Schutze. Foundations of Statistical Natural Language Processing, MIT Press, 1999. 7. Natural Language Processing by Jacob Eisenstein (free online). 8. Speech and Language Processing by Dan Jurafsky and James H. Martin	

	(3rd ed. draft). 9. T. McEnery and A. Wilson. Corpus Linguistics, EUP. 2001. 10. Tamburini, F. Neural Models for the Automatic Processing of Italian, Bologna: Pàtron. 2022. 11. https://corpora.ficlit.unibo.it/NLP/ 12. https://www.machinelearningplus.com/nlp/nlp-exercises
Course Outcomes	1. Learners will learn about the concepts in natural language processing. 2. Learners will have a fair idea of different areas in NLP 3. Learners will appreciate the complexities involved in natural language processing. 4. Through lectures and practical assignments, students will learn the necessary tricks for making their models work on practical problems.

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-522**
Title of Course : **Computer Vision**
Number of Credits : **4 (2L-2T-0P)**
Contact hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Prerequisites for the course	Python programming, linear algebra and calculus , array manipulation	
Course Objectives	The aim of the course is to introduce the fundamental concept of computer vision and to emphasize the importance of computer vision in developing and implementing different projects	
Content	**Unit-I** Image Formation - Geometric Camera Models - Light and Shading - Color - Early Vision: Just One Image Linear Filters - Local Image Features - Texture - Early Vision: Multiple Images - Stereopsis - Structure from Motion - Mid-Level Vision Segmentation by Clustering - Grouping and Model Fitting-Tracking - High-Level Vision- Registration- Smooth Surfaces and Their Outlines - Range Data - Learning to Classify - Classifying Images	**15 Hours**
	Unit II: Detecting Objects in Images- Topics in Object Recognition Applications Image-Based Modeling and Rendering - Looking at People- Image Search and Retrieval - Optimization Techniques	**15 Hours**
Tutorial session/ Practical assignments:	1. Open CV setup and demo on getting started up. 2. Image representation and image manipulation using open CV 3. Image storage and manipulation. 4. Photographs and perspective projections 5. Gaussian smoothings 6. Canny edge detection 7. Corner detection 8. Gabor filters 9. Hough transformation for lines 10. Hough transformation for circles	**10*3 =30 Hours**
Pedagogy:	lectures/Practical/ tutorials/assignments/self-study	
References/Readings:	1. Szeliski, R. (2022). Computer vision: Algorithms and applications. Springer.	

	2. Ayyadevara, V. K., & Reddy, Y. (2020). Modern Computer Vision with PyTorch: Explore deep learning concepts and implement over 50 real-world image applications. Packt Publishing Ltd. 3. Howse, J., & Minichino, J. (2020). Learning OpenCV 4 Computer Vision with Python 3: Get to grips with tools, techniques, and algorithms for computer vision and machine learning. Packt Publishing Ltd.
Course Outcomes:	1. Acquire and process raw image data and Relate image data to 3D scene structures. 2. Know the concepts behind and how to use several model-based object representations, and critically compare them. 3. Know many of the most popularly used current computer vision techniques by carrying out suitable lab experiments listed above 4. Undertake computer vision work in MATLAB or python OpenCV

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-523**
Title of Course : **Robotics**
Number of Credits : **4 (2L-2T-0P)**
Contact Hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Prerequisites for the course	Linear Algebra, Set Theory, Complex Analysis, Matrices	
Course Objectives	1. To summarize and analyze the fundamentals of robotics. 2. To introduce students the kinematics and dynamics of robots. 3. To elucidate students the types of motion control. 4. To familiarize students with the basic techniques of designing the robots.	
Content	**Unit-I** **Module:1 Fundamentals** Introduction – Components, Degrees of Freedom, Joints, Coordinates, Mechanisms, Controller. **Module:2 Kinematics** Position and Orientation of Objects, Coordinate Transformation, Joint Variables and Position of End Effector, Inverse Kinematics Problem, Jacobian Matrix, Statics and Jacobian Matrices. **Module:3 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. **Module:4 Manipulability** Manipulability Ellipsoid and Manipulability Measure, Best Configurations of Robotic Mechanisms from Manipulability Viewpoint, Various Indices of Manipulability, Dynamic Manipulability.	**15 Hours**
	Module:5 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. **Module:6 Force Control** Impedance Control - Passive-Impedance Method, Active-Impedance Method-One- Degree-of- Freedom Case, Active-Impedance Method-General Case. **Module:7 Hybrid Control**	**15 Hours**

	Hybrid Control - Hybrid Control via Feedback Compensation, Dynamic Hybrid Control.	
Practicals to be discussed and implemented during the Tutorial Slots:	1. Assignment on introduction to Robot Configuration. 2. Demonstration of Robot with 2 dof, 3 dof, 4 dof etc. 3. Two assignments on programming the Robot for some simple real life applications. 4. Two assignments on programming the Robot for applications in Val II. 5. Two programming exercises for robots. 6. Two case studies of applications in industry.	**6*5=30 Hours**
Pedagogy	Lectures/Practical/ Tutorials/Assignments	
References/ Readings	**Text Book(s)** 1. Saeed B Niku, “Introduction to Robotics Analysis, Control, Applications”, 3rd Edition, Wiley, 2020. 2. Tsuneo Yoshikawa, "Foundations of Robotics Analysis and Control", The MIT Press Cambridge, 1990. **Reference Books** 1. John J. Craig, “Introduction to Robotics, Mechanics and Control”, 3rd Edition, Pearson Prentice Hall, 2005. 2. Robert J. Schilling, “Fundamentals of Robotics, Analysis and Control”, Prentice Hall India, 2003.	
Course Outcomes	After the completion of the course, student will be able to: 1. Comprehend, classify and analyze the fundamentals of robotics. 2. Analyze the kinematics in robots. 3. Gain knowledge about the dynamics of robots. 4. Elucidate the motion control in robotics.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-524**
Title of the Course : **IoT Architecture and Protocols**
Number of Credits : **4 (2L-2T-0P)**
Contact Hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Prerequisites for the course	Internet Technologies, Computer Organization and architecture, Operating Systems.	
Course Objectives	To understand the fundamentals of Internet of Things and the protocols and standards designed for IoT	
Theory	**Unit-I** Introduction to IoT: Introduction, IoT ecosystem, Applications, Challenges. Fundamentals: IoT Devices - Sensors, Actuators, and gateways, Basics of the wireless sensor network. IoT Architecture & Design: oneM2M, IoTWF, Additional Reference Models, Core functional stack, Data Management and compute stack. Communicating smart objects: Communication criteria, communication models, IoT access technologies – 3GPP MTC, IEEE 802.11, IEEE 802.15, WirelessHART, ZWave,	**15 Hours**
	Unit-II Bluetooth Low Energy, Zigbee Smart Energy, DASH7 IoT Network Layer: IP as IoT network layer, IPv6, 6LoWPAN, 6TiSCH, RPL, CORPL, CARP IoT Transport and Application protocols: Transport Layer: TCP, UDP, DCCP, SCTP, TLS, DTLS IoT application transport methods, HTTP, CoAP, XMPP, MQTT, AMQP, DDS Security in IoT: MAC802.15.4, 6LoWPAN, RPL, Application Layer security. IoT Application case study: Discuss any 3 applications of IoT	**15 Hours**

Any 15 Case Studies / Systems to be discussed during the Tutorial Slots:	1. Smart Agriculture System 2. Weather Reporting System 3. Home Automation System 4. Face Recognition Bot 5. Smart Garage Door 6. Smart Alarm Clock 7. Air Pollution Monitoring System 8. Smart Parking System 9. Smart Traffic Management System 10. Smart Cradle System 11. Smart Gas Leakage Detector Bot 12. Streetlight Monitoring System 13. Smart Anti-Theft System 14. Liquid Level Monitoring System 15. Night Patrol Robot 16. Health Monitoring System 17. Smart Irrigation System 18. Flood Detection System 19. Mining Worker Safety Helmet 20. Smart Energy Grid	**15 * 2 = 30 hours**
Pedagogy	lectures/ tutorials/Hands-on assignments/self-study	
References/ Readings	1. Buyya, R., &Dastjerdi, A. V. (Eds.). (2016). Internet of Things: Principles and Paradigms. Elsevier. 2. Hanes, D., Salgueiro, G., Grossetete, P., Barton, R., & Henry, J. (2017). IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things. CISCO Press. 3. Hersent, O., Boswarthick, D., &Elloumi, O. (2011). The Internet of Things: Key Applications and Protocols. John Wiley & Sons.	
Course Outcomes	1. Understanding and knowledge of various IoT protocols. 2. Ability to select and implement appropriate IoT protocols based on application requirements. 3. Awareness of security and privacy considerations in IoT protocols. 4. Familiarity with interoperability, performance optimization, and emerging trends in IoT protocols.	

(Back to Index)

SEMESTER II
DISCIPLINE SPECIFIC CORE COURSES

Name of the Programme	**: MSc Artificial Intelligence**
Course Code	**: CSI-508**
Title of the Course	**: Deep Learning**
Number of Credits	**: 2(2L-0T-0P)**
Contact hours	**: 30 hours (30L-0T-0P)**
Effective from AY	**: 2023-24**

Prerequisites for the course	Programme prerequisites	
Course Objectives	To study the basics of Neural Networks and their various variants such as the Convolutional Neural Networks and Recurrent Neural Networks, to study the different ways in which they can be used to solve problems in various domains such as Computer Vision, Speech and NLP.	
Content	**Unit-I** History of Deep Learning, McCulloch Pitts Neuron, Thresholding Logic, Perceptron Learning Algorithm and Convergence Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Gradient Descent Feedforward Neural Networks, Representation Power of Feedforward Neural Networks, Backpropagation Gradient Descent(GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, Adagrad, AdaDelta,RMSProp, Adam,AdaMax,NAdam, learning rate schedulers Autoencoders and relation to PCA, Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contractive autoencoders	**15 Hours**
	Unit-II Bias Variance Tradeoff, L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout Greedy Layer Wise Pre-training, Better activation functions, Better weight initialization methods, Batch Normalization Learning Vectorial Representations of Words, Convolutional Neural Networks, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet Visualizing Convolutional Neural Networks, Guided Backpropagation, Deep Dream, Deep Art, Fooling Convolutional Neural Networks Recurrent Neural Networks, Backpropagation Through Time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT Gated Recurrent Units (GRUs), Long Short Term Memory (LSTM) Cells, Solving the vanishing gradient problem with LSTM Encoder Decoder Models, Attention Mechanism, Attention over	**15 Hours**

	images, Hierarchical Attention, Transformers.	
Pedagogy	Lectures/ Tutorials/Hands-on assignments/Self-study	
References/ Readings	1. Charu, C. A. (2018). Neural networks and deep learning: a textbook. Springer. 2. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press. 3. Skansi, S. (2018). Introduction to Deep Learning: from logical calculus to artificial intelligence. Springer. 4. Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). Dive into deep learning. Cambridge University Press.	
Course Outcomes	1. Understanding of deep learning concepts and principles. 2. Implementation and training of deep learning models. 3. Practical application of deep learning in various domains. 4. Evaluation and interpretation of deep learning model performance.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-509**
Title of the Course : **Deep Learning Lab**
Number of Credits : **2(0L-0T-2P)**
Contact hours : **60 hours(0L-0T-60P)**
Effective from AY : **2023-24**

Prerequisites for the course	Programming, Machine Learning Skills. Statistics, Calculus, Linear Algebra. Probability.	
Course Objectives	1. To make students comfortable with tools and techniques required in handling large amounts of datasets. 2. They will also uncover various deep learning methods in NLP, Neural Networks etc.	
Content	**Tensorflow with Python** Introducing Tensorflow - Tensorflow as an Interface - Tensorflow as an environment - Tensors - Computation Graph - Installing Tensorflow - Tensorflow training - Prepare Data - Tensor types - Loss and Optimization - Running tensorflow programs.	**5 hours**
	Building Neural Networks using Tensorflow Building Neural Networks using Tensorflow - Tensorflow data types - CPU vs GPU vs TPU - Tensorflow methods - Introduction to Neural Networks - Neural Network Architecture - Linear Regression example revisited - The Neuron - Neural Network Layers - The MNIST Dataset - Coding MNIST NN.	**5 hours**
	Deep Learning using Tensorflow Deepening the network - Images and Pixels - How humans recognise images - Convolutional Neural Networks - ConvNet Architecture - Overfitting and Regularization - Max Pooling and ReLU activations - Dropout - Strides and Zero Padding - Coding Deep ConvNets demo - Debugging Neural Networks - Visualising NN using Tensorflow - Tensorboard.	**5 hours**
	Transfer Learning using Keras and TFLearn Transfer Learning Introduction - Google Inception Model - Retraining Google Inception with our own data demo - Predicting new images - Transfer Learning Summary - Extending Tensorflow - Keras - TFLearn - Keras vs TFLearn Comparison.	**5 hours**

	Suggest ideas for lab work Assignment -1 Cat vs. Dog Image Classifier Assignment -2- Covid-19 Detection in Lungs Assignment -3- Digit Recognition System Assignment - 4- Facial Recognition Application Assignment -5- Face Mask Detection Assignment -6- Cyber-Attack Prediction Assignment -7- Automated Attendance System Assignment -8 Emotion Recognition Assignment -9- Object Detection System Assignment 10 - Recommender System	**10*4= 40 Hours**
Pedagogy	Lab assignment/mini project	
References/ Readings	1. Charu, C. A. (2018). Neural networks and deep learning: a textbook. Springer. 2. Chollet, F. (2021). Deep learning with Python. Simon and Schuster. 3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press. 4. Hurbans, R. (2020). Grokking artificial intelligence algorithms. Manning Publications. 5. Skansi, S. (2018). Introduction to Deep Learning: from logical calculus to artificial intelligence. Springer. 6. Stevens, E., Antiga, L., & Viehmann, T. (2020). Deep learning with PyTorch. Manning Publications. 7. Weidman, S. (2019). Deep learning from scratch: Building with Python from first principles. O'Reilly Media. 8. Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2023). Dive into deep learning. Cambridge University Press.	
Course Outcomes	1. Practical application of deep learning techniques. 2. Implementation and training of deep learning models. 3. Data preprocessing and augmentation for deep learning. 4. Model evaluation and optimization.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-510**
Title of the Course : **Big Data Frameworks**
Number of Credits : **2(2L-0T-0P)**
Contact hours : **30 hours (30L-0T-0P)**
Effective from AY :**2023-24**

Prerequisites for the course	Probability and statistics and programming background	
Course Objectives	1. To understand the need of Big Data, challenges and different analytical architectures 2. Installation and understanding of Hadoop Architecture and its ecosystems 3. Processing of Big Data with Advanced architectures like Spark. 4. Describe graphs and streaming data in Spark	
Content	**Unit-I** **Introduction to big data** Data Storage and Analysis - Characteristics of Big Data – Big Data Analytics - Typical Analytical Architecture – Requirement for new analytical architecture – Challenges in Big Data Analytics – Need of big data frameworks **Hadoop framework** Hadoop – Requirement of Hadoop Framework - Design principle of Hadoop –Comparison with other system - Hadoop Components – Hadoop 1 vs Hadoop 2 – Hadoop Daemon's – HDFS Commands – Map Reduce Programming: I/O formats, Map side join, Reduce Side Join, Secondary sorting, Pipelining MapReduce jobs - **Hadoop Ecosystem** Introduction to Hadoop ecosystem technologies: Serialization: AVRO, Co-ordination: Zookeeper, Databases: HBase, Hive, Scripting language: Pig, Streaming: Flink, Storm	**15 Hours**
	Unit-II **Spark framework** Introduction to GPU Computing, CUDA Programming Model, CUDA API, Simple Matrix, Multiplication in CUDA, CUDA Memory Model, Shared Memory Matrix Multiplication, Additional CUDA API Features. **Data analysis with spark shell** Writing Spark Application - Spark Programming in Scala, Python, R, Java - Application Execution **Spark SQL and Graph X** SQL Context – Importing and Saving data – Data frames – using SQL – GraphX overview – Creating Graph – Graph Algorithms. **Spark streaming** Overview – Errors and Recovery – Streaming Source – Streaming live data with spark	**15 Hours**

	Recent trends in big data analytics	
Pedagogy	Assignment / Quiz / Project / Seminar	
References/ Readings	1. Guller, M. (2015). Big data analytics with Spark: A practitioner's guide to using Spark for large scale data analysis. Apress. 2. Kienzler, R. (2017). Mastering Apache Spark 2.x. Packt Publishing Ltd. 3. Miner, D., & Shook, A. (2012). MapReduce design patterns: building effective algorithms and analytics for Hadoop and other systems. O'Reilly Media, Inc. 4. Pentreath, N. (2015). Machine learning with Spark. Packt Publishing. 5. White, T. (2012). Hadoop: The definitive guide. O'Reilly Media, Inc.	
Course Outcomes	1. students would have a good understanding of Big Data 2. understand the basics of the frameworks like Hadoop and spark and 3. have a knowledge of Spark SQL 4. have understanding of Spark streaming.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-511**
Title of the Course : **Big Data Frameworks Lab**
Number of Credits : **2(0L -0T-2P)**
Contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

Prerequisites for the course	Should have knowledge of one Programming Language (Java preferably), Practice of SQL (queries and sub queries), exposure to Linux Environment.	
Course Objectives	Understand the Big Data Platform and its Use cases • Provide an overview of Apache Hadoop • Provide HDFS Concepts and Interfacing with HDFS • Understand Map Reduce Jobs • Provide hands on Hodoop Eco System • Apply analytics on Structured, Unstructured Data.	
Content	**List of Experiments:**	
	1. Implement the following Data structures in Java Linked Lists,Stacks,Queues,Set,Map	**5 hours**
	2. Perform setting up and Installing Hadoop in its three operating modes: Standalone, Pseudo Distributed, Fully distributed.	**5 hours**
	3. Implement the following file management tasks in Hadoop: Adding files and directories,Retrieving files (Deleting files Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.	**5 hours**
	4. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.	**5 hours**
	5. Write a Map Reduce program that mines weather data. Weather sensors collecting data everyhour at many locations across the globe gather a large volume of log data, which is a goodcandidate for analysis with MapReduce, since it is semi structured and record-oriented.	**5 hours**
	6. Implement Matrix Multiplication with Hadoop Map Reduce	**5 hours**
	7. Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.	**5 hours**
	8. Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes.	**5 hours**
	9. Solve some real life big data problems. - Traffic control using big data - Medical insurance fraud detection - Recommendation system	**20 hours**

	- Anomaly detection in cloud servers - Tourist behavior analysis - Web server log analysis	
Pedagogy	Lab assignments/mini project/ seminar	
References/ Readings	1. Berthold, M. R., & Hand, D. J. (Eds.). (2007). Intelligent data analysis: an introduction. Springer. 2. Franks, B. (2012). Taming the big data tidal wave: Finding opportunities in huge data streams with advanced analytics. John Wiley & Sons. 3. Liebowitz, J. (Ed.). (2013). Big data and business analytics. CRC Press. 4. Myatt, G. J. (2007). Making sense of data: a practical guide to exploratory data analysis and data mining. John Wiley & Sons. 5. Rajaraman, A., & Ullman, J. D. (2011). Mining of massive datasets. Cambridge University Press. 6. Russom, P. (2011). Big data analytics. TDWI Best Practices Report, fourth quarter, 19(4), 1-34. 7. White, T. (2012). Hadoop: The definitive guide. O'Reilly Media, Inc.	
Course Outcomes	1. Identify Big Data and its Business Implications. 2. List the components of Hadoop and Hadoop Eco-System 3. Access and Process Data on Distributed File System and Manage Job Execution in Hadoop Environment 4. Develop Big Data Solutions using Hadoop Eco System 5. Analyze Infosphere BigInsights Big Data Recommendations.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-512**
Title of the Course : **Reinforcement Learning**
Number of Credits : **2(2L-0T-0P)**
Contact hours : **30 hours(30L-0T-0P)**
Effective from AY : **2023-24**

Prerequisites for the course	Linear algebra, multivariable calculus Basic machine learning knowledge	
Course Objectives	To enable the student to understand the reinforcement learning paradigm, to be able to identify when an RL formulation is appropriate, to understand the basic solution approaches in RL, to implement and evaluate various RL algorithms.	
Content	**Unit-I** Review of ML fundamentals – Classification, Regression. Review of probability theory and optimization concepts. RL Framework; Supervised learning vs. RL; Explore-Exploit Dilemma; Examples. MAB: Definition, Uses, Algorithms, Contextual Bandits, Transition to full RL, Intro to full RL problem Intro to MDPs: Definitions , Returns, Value function, Q-function. Bellman Equation, DP, Value Iteration, Policy Iteration, Generalized Policy Iteration.	**15 Hours**
	Unit II Evaluation and Control: TD learning, SARSA, Q-learning, Monte Carlo, TD Lambda, Eligibility Traces. Maximization-Bias & Representations: Double Q learning, Tabular learning vs. Parameterized, Q-learning with NNs Function approximation: Semi-gradient methods, SGD, DQNs, Replay Buffer. Policy Gradients: Introduction, Motivation, REINFORCE, PG theorem, Introduction to AC methods Actor-Critic Methods, Baselines, Advantage AC, A3C Advanced Value-Based Methods: Double DQN, Prioritized Experience Replay, Dueling Architectures, Expected SARSA. Advanced PG/A-C methods: Deterministic PG and DDPG, Soft Actor-Critic (SAC) HRL: Introduction to hierarchies, types of optimality, SMDPs, Options, HRL algorithms POMDPS: Intro, Definitions, Belief states, Solution Methods; History-based methods, LSTMS, Q-MDPs, Direct Solutions, PSR. Model-Based RL: Introduction, Motivation, Connections to Planning, Types of MBRL, Benefits, RL with a Learnt Model, Dyna-style models, Latent variable models, Examples, Implicit MBRL. Case study on design of RL solution for real-world problems.	**15 Hours**

Pedagogy	Hands-on assignments / tutorials / peer-teaching / flip classroom/ presentations.
References/ Readings	1. Richard S., & Andrew B. (2018). Reinforcement learning -Introduction, MIT press. 2. Szepesvári, C. (2022). Algorithms for reinforcement learning. Springer Nature.
Course Outcomes	1. Solid understanding of reinforcement learning concepts, theories, and algorithms. 2. Ability to implement and apply reinforcement learning algorithms to real-world problems. 3. Evaluation and analysis of reinforcement learning systems. 4. Critical thinking skills, staying updated with current research and trends.

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-513**
Title of the Course : **Reinforcement Learning Lab**
Number of Credits : **2(0L-0T-2P)**
Contact hours : **60 hours(0L-0T-60P)**
Effective from AY : **2023-24**

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

	learning systems. 3. Contribution to the field through novel research or innovative applications. 4. Improved collaboration and communication skills within a research lab setting.

(Back to Index)

Name of the Programme : **M.Sc. in Artificial Intelligence**
Course code : **CSI-514**
Title of the course : **Software Engineering for AI Enabled systems**
Number of credits : **2 (2L-0T-0P)**
Contact hours : **30 hours (30L-0T-0P)**
Effective from AY :**2023-24**

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

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course code : **CSI-515**
Title of course : **Software Engineering for AI Enabled Systems Lab**
Number of credits : **2 (0L-0T-2P)**
Contact hours : **60 hours (0L-0T-60P)**
Effective from AY : **2023-24**

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

(Back to Index)

DISCIPLINE SPECIFIC ELECTIVE COURSES

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI-525
Title of the Course : Machine Translation
Number of Credits : 4 (2L-2T-0P)
Contact hours : 60 hours (30L-30T-0P)
Effective from AY : 2023-24

<table>
<tr><td>Prerequisites for the course</td><td colspan="2">Knowledge of Mathematics for Computer Science and Machine Learning will prove beneficial, A previous course on Artificial Intelligence and Natural Language Processing will help; Exposure to Linguistics is useful, though not mandatory</td></tr>
<tr><td>Course Objectives:</td><td colspan="2">The objective of the course is to understand and get an insight into the different approaches used for Machine Translation (MT).</td></tr>
<tr><td rowspan="9">Content:</td><td>Introduction: Data-driven MT, MT Approaches, Language divergence, three major paradigms of MT, MT Evaluation,</td><td rowspan="2">15 Hours</td></tr>
<tr><td>Bilingual Word Mappings: Combinatorial Argument, One-to-One Alignment, Heuristic and Iterative bases computation, Mathematics of Alignment, Expectation Maximization, IBM models of Alignment</td></tr>
<tr><td>Phrase-Based Machine Translation (PBMT): Need, Examples, Phrase Table, Mathematics of Phrase-Based SMT, Decoding.</td><td rowspan="3">15 Hours</td></tr>
<tr><td>Rule-Based Machine Translation (RBMT): Kinds, UNL, Interlingua and Word Knowledge, UNL conversion, Transfer-based MT.</td></tr>
<tr><td>Example-Based Machine Translation (EBMT): Essential steps of EBMT, Text similarity computation, Translation memory, Statistical Machine Translation</td></tr>
<tr><td>Assignments to be discussed during the Tutorial Slots -</td><td>30 hours</td></tr>
<tr><td>Assignment 1:
Data-driven MT, MT Approaches, Language divergence, three major paradigms of MT, MT Evaluation,</td><td rowspan="3">5*6 =30 Hours</td></tr>
<tr><td>Assignment 2:
Bilingual Word Mappings: Combinatorial Argument, One-to-One Alignment, Heuristic and Iterative bases computation, Mathematics of Alignment, Expectation Maximization, IBM models of Alignment</td></tr>
<tr><td>Assignment 3:
Phrase-Based Machine Translation (PBMT): Need, Examples, Phrase Table, Mathematics of Phrase-Based SMT, Decoding.</td></tr>
</table>

	Assignment 4: Rule-Based Machine Translation (RBMT): Kinds, UNL, Interlingua and Word Knowledge, UNL conversion, Transfer-based MT.	
	Assignment 5: Example-Based Machine Translation (EBMT): Essential steps of EBMT, Text similarity computation, Translation memory, Statistical Machine Translation	
Pedagogy:	lectures/ tutorials/assignments/self-learning/ flipped classroom	
References/ Readings	1. An Open Source Neural Machine Translation System. (n.d.). from https://opennmt.net/ 2. Bhattacharyya, P. (2015). Machine Translation. Chapman and Hall/CRC. 3. Bhashini Project. (n.d.). Retrieved from https://bhashini.gov.in/bhashadaan/en/likho-india 4. Waibel, A., & Niehues, J. (n.d.). Machine Translation on Coursera. Retrieved from https://www.coursera.org/learn/machinetranslation	
Course Outcomes	1. Understand the Machine Translation Approaches and Understand the differences between Phrase-Based, Rule-Based, and Example-Based Machine Translation 2. explain, apply, and assess evaluation methods for machine translation; describe and critically discuss the architecture of machine translation systems; 3. build their own translation model using existing tools for machine translation and evaluate and analyse the translation results; compare different types of machine translation strategies, such as rule-based, statistical, and neural machine translation; 4. implement components of machine translation systems or components used in evaluation or pre-processing	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-526**
Title of the Course : **Mathematics for Computer Vision and Robotics**
Number of Credits : **4(2L+2T+P)**
Contact hours : **60 hours(30L-30T-0P)**
Effective from AY : **2023-24**

<table>
<tr><td>Prerequisites for the course</td><td colspan="2">Linear Algebra, Probability and Statistics, Signal Processing</td></tr>
<tr><td>Course Objectives</td><td colspan="2">1. To understand basic concepts of linear algebra and to illustrate its power and utility through applications to computer vision.
2. To apply the concepts of vector spaces, linear transformations, matrices and inner product spaces in engineering.
3. To understand the concepts of curves and surfaces and solving linear programming problems that arise in engineering.</td></tr>
<tr><td rowspan="2">Theory:</td><td>Unit-I
Vectors and Matrices Points, vectors, vector spaces(Rn only), lines and planes as subspaces -Matrices and four fundamental spaces- Gaussian elimination.
Factorization of Matrices LU factorizations-Cholesky decomposition –eigenvalues and eigenvectors-–SVD - Applications of the SVD .- Solving Linear Systems and the Pseudoinverse -Principal Components Analysis (PCA)
Linear transformations Linear transformations(R^n only) – Basic properties-invertible linear transformation - matrices of linear transformations.</td><td>15 Hours</td></tr>
<tr><td>Unit-II
Geometry in Linear Transformation Projections, Rotations and reflection and applications
Orthogonality Dot products and inner products(R^n only) – lengths and angles of vectors –orthogonal matrices- Gram Schmidt orthogonalizations - QR factorization- orthogonal projections–Least Square solutions
Differential geometry Introduction to differential geometry - curves-curvature-torsion-osculating plane –surfaces
Linear programming Linear programming – Formulation of LPP-Graphical method - Simplex method</td><td>15 Hours</td></tr>
<tr><td>Assignments to be discussed during Tutorial Slots:</td><td>Assignment 1- Getting to Know the Python math Module, Constants of the math Module:Pi, Tau, Euler’s Number, Infinity, Not a Number (NaN) and Arithmetic Functions,Find Factorials With Python factorial(), Find the Ceiling Value With ceil(), Find the Floor Value With floor(), Truncate Numbers With trunc(), Find the Closeness of Numbers With Python isclose()

Assignment-2 - Power Functions, Calculate the Power of a</td><td>7 * 3 = 21 hours

+ 9 hours for a Mini</td></tr>
</table>

	Number With pow(),Find the Natural Exponent With exp(),Practical Example With exp(),Logarithmic Functions, Python Natural Log With log(),Understand log2() and log10(), Practical Example With Natural Log Assignment-3 -Other Important math Module Functions, Calculate the Greatest Common Divisor, Calculate the Sum of Iterables, Calculate the Square Root, Convert Angle Values, Calculate Trigonometric Values Assignment -4 -New Additions to the math Module in Python 3.8.cmath vs math, NumPy vs math, Assignment -5 -Calculating combinations and permutations using factorials, Calculating the height of a pole using trigonometric functions, Calculating radioactive decay using the exponential function, Calculating the curve of a suspension bridge using hyperbolic functions, Solving quadratic equations Assignment - 6 -Simulating periodic functions, such as sound and light waves, using trigonometric functions, Assignment -7 -Vector algebra in python, Physical Quantities, Vector and Scalars, Representation of vectors, Types of Vectors, Operations on Vectors, Section Formula, Concept of Euclidean Distance between two vectors,	**Project**
Pedagogy	Lectures/ Lab Assignments/ Seminar Presentations /Project Work	
References/ Readings	1. Basics of Matrix Algebra for Statistics with R, Nick Fieller, CRC Press, 2016. 2. Computer Vision: A Modern Approach, Forsyth and Ponce, 2nd Edition Pearson, 2012. 3. Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition, Manfredo P. do Carmo, Dover Publications, 2016. 4. Introduction to Linear Algebra, Gilbert Strang, 5th Edition, Cengage Learning, 2015. 5. Linear Algebra, Jin Ho Kwak and Sungpyo Hong, Second Edition, Springer, 2004. 6. Linear Algebra and Optimization with Applications to Machine Learning - Volume I. 7. Linear Algebra for Computer Vision, Robotics, and Machine Learning, Jean H. Gallier, Jocelyn Quaintance, World Scientific Publishing Company, 2020. 8. Mathematics for Machine Learning, Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, Cambridge University Press, 2020. 9. Modern Mathematics And Applications In Computer Graphics And	

	Vision, Hongyu Guo, World Scientific Publishing Company, 2014. 10. Operations Research Principles and Applications, G. Srinivasan, 3rd Edition, PHI Learning, 2017.
Course Outcomes	At the end of this course the students are expected to learn 1. The abstract concepts of matrices and system of linear equations using decomposition methods and applications in engineering 2. Understand the geometry behind linear transforms which is used in computer graphics,Understand the concepts of orthogonality through linear algebra 3. Understating properties curves and surfaces and Solving linear programming problems arise in engineering 4. Solving problems in Linear algebra, linear programming and differential geometry using matplotlib or Python.

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-527**
Title of the Course : **Soft Computing**
Number of Credits : **4 (2L+2T+0P)**
Contact hours : **60 hours (30L-30T-0P)**
Effective from AY : **2023-24**

Prerequisites for the course	Machine Learning, Statistices	
Course Objectives	The objective of this course is to introduce methods for handling imprecise and uncertain data using Rough sets, Neuro Fuzzy Systems and foster their abilities in designing and implementing optimal solutions for real-world and engineering problems using derivative free optimization techniques.	
Content	**Unit I** **Introduction to Soft Computing** Soft Computing Overview – Uncertainty in data, Hard vs Soft Computing **Neural Networks** Introduction, RBF Networks, Self-Organizing Map, Boltzmann Machines, Convolutional Neural Networks. **Fuzzy Systems** Fuzzy Sets, Fuzzy Relations, and Membership functions, Properties of Membership functions, Fuzzification and Defuzzification. **Fuzzy logic** Fuzzy Rule based systems, Fuzzy Decision making, Fuzzy Classification, Fuzzy CMeans Clustering.	**15 Hours**
	Unit-II: **Rough Sets** Rough Sets – Definition, Upper and Lower Approximations, Boundary Region, Decision Tables and Decision Algorithms. Properties of Rough Sets. Rough K-means clustering, Rough **Optimization Techniques** Introduction, Genetic Algorithm, Memetic Algorithms, Particle Swarm Optimization, Ant Colony Optimization, Frog-Leaping. **Hybrid Systems** GA Based Back Propagation Networks, Fuzzy Back Propagation Networks, Evolutionary Ensembles	**15 Hours**
Assignments and Mini Project Discussions during the Tutorial Slots:	**List of Assignments:** 1. To demonstrate the working of Hebbian learning rule 2. To demonstrate the working of perceptron learning rule 3. To demonstrate the working of Delta learning rule 4. To demonstrate the working of Widrow-Hoff learning rule 5. To demonstrate the working of Radial basis function network 6. To demonstrate the working of Learning vector quantization	**12 * 2 = 24 hours** **+ 6 hours**

	7. To demonstrate the working of Self-Organizing maps 8. To demonstrate the working of Recurrent neural networks 9. To demonstrate the working of Fuzzy inference system 10. To demonstrate the working of Genetic algorithm 11. To demonstrate the working of Particle Swarm Optimization 12. To demonstrate the working of Ant Colony Optimizations and TSP	**for a Mini Project**
Pedagogy	Lectures / Assignments / Quiz / Mini Project / Seminar Presentations	
References/ Readings	**Main Readings** 1. Andries P. Engelbrecht. (2007). Computational Intelligence: An Introduction. John Wiley & Sons. 2. Fausett, L. V. (1993). Fundamentals of Neural Networks: Architectures, Algorithms And Applications. Pearson. 3. Haykin, S. (2008). Neural Networks and Learning Machines. Prentice Hall. 4. Ross, T. (n.d.). Fuzzy Logic with Engineering Applications (3rd ed.). Wiley. 5. Sivanandham, S. N., & Deepa, S. N. (n.d.). Principles of Soft Computing (2nd ed.). Wiley Publications.	
Course Outcomes	1. Have a general understanding of soft computing methodologies, to deal with imprecise and uncertain data 2. Develop computational neural network models for some simple biological systems; 3. Develop fuzzy models for engineering systems, particularly for control systems; 4. Apply derivative free optimization methods to solve real world problems 5. Demonstrate some applications of computational intelligence.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI-528**
Title of the Course : **Regression and Predictive Analytics**
Number of Credits : **4 (2L+2T+0P)**
Contact Hours : **60 hours(30L+30T+0P)**
Effective from AY : **2023-24**

<table>
<tr><td>Prerequisites for the course</td><td colspan="2">Probability Theory and Distributions</td></tr>
<tr><td>Course Objectives</td><td colspan="2">1. Develop an understanding of regression analysis and model building.
2. Provide the ability to develop relationship between variables
3. Investigate possible diagnostics in regression techniques
4. Formulate feasible solutions using a regression model for real-life problems.</td></tr>
<tr><td rowspan="2">Theory:</td><td>Unit-I
Simple Regression Analysis
Introduction to a linear and nonlinear model. Ordinary Least Square methods. Simple linear regression model, using simple regression to describe a linear relationship. Fitting a linear trend to time series data, validating simple regression model using t, F and p test. Developing confidence interval. Precautions in interpreting regression results.
Multiple Regression Analysis
Concept of Multiple regression model to describe a linear relationship, Assessing the fit of the regression line, inferences from multiple regression analysis, problem of overfitting of a model, comparing two regression model, prediction with multiple regression equation.
Fitting Curves and Model Adequacy Checking
Introduction, fitting curvilinear relationship, residual analysis, PRESS statistics, detection and treatment of outliers, lack of fit of the regression model, test of lack of fit, Problem of autocorrelation and heteroscedasticity. Estimation of pure errors from near neighbors.</td><td>15 Hours</td></tr>
<tr><td>Unit II
Transformation techniques
Introduction, variance stabilizing transformations, transformations to linearize the model, Box Cox methods, transformations on the repressors variables, Generalized and weighted least squares, Some practical applications.
Multicollinearity
Introduction, sources of multicollinearity, effects of multicollinearity. Multicollinearity diagnostics: examination of correlation matrix, variance Inflation factors (VIF), Eigen system analysis of X1X. Methods of dealing with Multicollinearity:</td><td>15 Hours</td></tr>
</table>

	collecting additional data, model re-specification, and ridge regression. **Generalized Linear Models** Generalized linear model: link functions and linear predictors, parameter estimation and inference in the GLM, prediction and estimation with the GLM, Residual Analysis, and concept of over dispersion. **Model building and Nonlinear Regression** Variable selection, model building, model misspecification. Model validation techniques: Analysis of model coefficients, and predicted values, data splitting method. Nonlinear regression model, nonlinear least squares, transformation to linear model, parameter estimation in nonlinear system, statistical inference in nonlinear regression.	
Assignments and a Mini Project to be discussed during the Tutorial slots:	1. Linear Regression 2. Minimum Least Square Method 3. Calculating coefficients values 4. Ascombe's Quartet 5. Regression Equations- x on y & y on x 6. Predicting mom's height based on daughter's height 7. Regression-Solved problem-2 8. Probable Error- Calculating correlation coefficient of POPULATION 9. Predictive modelling project for credit card fraud detection 10. Predictive modelling project for customer value prediction 11. Predictive modelling project for stock market forecasting 12. Predictive modelling project for corporate bankruptcy prediction	**12 * 2 = 24 hours** **+ 6 hours for Mini Project discussions**
Pedagogy	Lectures/ tutorials/assignments/self-study	
References/ Readings	1. Draper, N. R., & Smith, H. (2015). Applied Regression Analysis (3rd ed.). Wiley India Pvt. Ltd. 2. Johnson, R. A., & Wichern, D. W. (2013). Applied Multivariate Statistical Analysis (6th ed.). PHI Learning Pvt. Ltd. 3. Montgomery, D. C., Peck, E. A., & Vining, G. G. (2016). Introduction to Linear Regression Analysis (3rd ed.). Wiley India Pvt. Ltd. 4. Pardoe, I. (2012). Applied Regression Modeling. John Wiley & Sons, Inc.	
Course Outcomes	1. Develop in-depth understanding of the linear and nonlinear regression model. 2. Demonstrate the knowledge of regression modeling and model selection techniques. 3. Examine the relationships between dependent and independent variables. 4. Estimate the parameters and fit a model.	

(Back to Index)

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI-529
Title of the Course : Essentials of Data Analytics
Number of Credits : 4(2L+2T+0P)
Contact hours : 60 hours(30L-30T-0P)
Effective from AY : 2023-24

Prerequisites for the course	Probability and Statistics	
Course Objectives	1. To understand the concepts of analytics using various machine learning models. 2. To appreciate supervised and unsupervised learning for predictive analysis. 3. To understand data analytics as the next wave for businesses looking for competitive advantage. 4. Carry out rule-based analysis of the data in line with the analysis plan. 5. Validate the results of their analysis according to statistical guidelines. 6. Validate and review data accurately and identify anomalies. 7. To learn aspects of computational learning theory. 8. Apply statistical models to perform Regression Analysis, Clustering and Classification.	
Theory:	**Unit-I** **Module:1** Regression Analysis Linear regression: simple linear regression - Regression Modelling - Correlation, ANOVA, Forecasting, Autocorrelation **Module:2** Classification Logistic Regression, Decision Trees, Naïve Bayes-conditional probability - Random Forest - SVM Classifier **Module:3** Clustering K-means, K-medoids, Hierarchical clustering	**15 hours**
	Unit II **Module:4** Optimization Gradient descent - Variants of gradient descent - Momentum - Adagrad - RMSprop - Adam - AMSGrad **Module:5** case study -Managing Health and Safety Comply with organization's current health, safety and security policies and procedures - Report any identified breaches in health, safety, and security policies and procedures to the designated person - Identify and correct any hazards that they can deal with safely, competently and within the limits of their authority - Report any hazards that they are not competent to deal with to the relevant person in line with organizational procedures and warn other people who may be affected. **Module:6- requirement analysis -** Data and Information Management Establish and agree with appropriate people the data/information they need to provide, the formats in which they need to provide it, and when they need to provide it - Obtain the data/information from reliable sources - Check that the	**15 Hours**

	data/information is accurate, complete and up-to-date **Module:7** Learning and Self Development Obtain advice and guidance from appropriate people to develop their knowledge, skills and competence - Identify accurately the knowledge and skills they need for their job role - Identify accurately their current level of knowledge, skills and competence and any learning and development needs - Agree with appropriate people a plan of learning and development activities to address their learning needs	
Practicals to be discussed during Tutorial Slots:	1. Web Scraping: - a. While you'll find no shortage of excellent (and free) public data sets on the internet, you might want to show prospective employers that you're able to find and scrape your own data as well. Plus, knowing how to scrape web data means you can find and use data sets that match your interests, regardless of whether or not they've already been compiled. b. If you know some Python, you can use tools like Beautiful Soup or Scrapy to crawl the web for interesting data. If you don't know how to code, don't worry. You'll also find several tools that automate the process (many offer a free trial), like Octoparse or ParseHub. c. If you're unsure where to start, here are some websites with interesting data options to inspire your project: d. Reddit, Wikipedia, Job portals 2. Data Cleaning a. A significant part of your role as a data analyst is cleaning data to make it ready to analyze. Data cleaning (also called data scrubbing) is the process of removing incorrect and duplicate data, managing any holes in the data, and making sure the formatting of data is consistent. b. As you look for a data set to practice cleaning, look for one that includes multiple files gathered from multiple sources without much curation. Some sites where you can find "dirty" data sets to work with include: c. CDC Wonder, Data.gov, World Bank, Data.world/r/datasets 3. Exploratory data analysis (EDA) a. Data analysis is all about answering questions with data. Exploratory data analysis, or EDA for short, helps you explore what questions to ask. This could be done separate from or in conjunction with data cleaning. Either way, you'll want to accomplish the following during these early investigations. b. Ask lots of questions about the data. c. Discover the underlying structure of the data. d. Look for trends, patterns, and anomalies in the data.	**5 * 6 = 30 hours**

	e. Test hypotheses and validate assumptions about the data. f. Think about what problems you could potentially solve with the data. 4. Sentiment analysis a. Sentiment analysis, typically performed on textual data, is a technique in natural language processing (NLP) for determining whether data is neutral, positive, or negative. It may also be used to detect a particular emotion based on a list of words and their corresponding emotions (known as a lexicon). b. This type of analysis works well with public review sites and social media platforms, where people are likely to offer public opinions on various subjects. c. To get started exploring what people feel about a certain topic, you can start with sites like: Amazon (product reviews), Rotten Tomato (movie reviews), Facebook witter, News sites 5. Data visualization a. Humans are visual creatures. This makes data visualization a powerful tool for transforming data into a compelling story to encourage action. Great visualizations are not only fun to create, they also have the power to make your portfolio look beautiful.	
Pedagogy	Lectures/Assignments/Seminar Presentations/Mini-Project	
References/ Readings	1. Berthold, M., & Hand, D. J. (2007). Intelligent Data Analysis. Springer. 2. Gupta, G. K. (2006). Introduction to Data Mining with Case Studies (Easter Economy Edition). Prentice Hall of India. 3. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning (2nd ed.). Springer. 4. McCue, C. (2007). Data Mining and Predictive Analysis: Intelligence Gathering and Crime Analysis. Elsevier. 5. Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective (1st ed.). MIT Press. 6. Myatt, G. J. (2014). Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining (2nd ed.). John Wiley & Sons. 7. O'Neil, C., & Schutt, R. (2014). Doing Data Science: Straight Talk from the Frontline. O'Reilly. 8. Prasad, R. N., & Acharya, S. (2016). Fundamentals of Business Analytics (2nd ed.). Wiley. 9. Toomey, D. (2014). R for Data Science. Packt Publishing. 10. SSCNASSCOM. (n.d.). Qualification Pack - Data Analyst. from https://www.sscnasscom.com/qualification-pack/SSC/Q2101/ 11. Mode of Evaluation: ISA/Assignment / Quiz / Project / Seminar	

Course Outcomes	1. Identify and apply the appropriate supervised learning techniques to solve real world problems with labeled data. 2. Choose and implement typical unsupervised algorithms for different types of applications with unlabelled data. 3. Implement statistical analysis techniques for solving practical problems. 4. Understand different techniques to optimize the learning algorithms.

(Back to Index)

SEMESTER III
RESEARCH SPECIFIC ELECTIVE COURSES

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI-600
Title of the Course : Research methodology
Number of Credits : 4(4L+0P)
Contact hours : 60 hours
Effective from AY : 2023-24

Prerequisites for the course	None	
Course Objectives	This course aims to impart a comprehensive understanding of research methodology, covering the techniques for defining research problems and delving into the meaning of interpretation along with associated techniques and precautions in the research process.	
Content	**Unit I** Introduction to research, Definitions and characteristics of research, Types of Research, Research Process, Problem definition, Objectives of Research, Research Questions, Research design, Quantitative vs. Qualitative Approach, Building and Validating Theoretical Models, Exploratory vs. Confirmatory Research, Experimental vs. Theoretical Research, Importance of reasoning in research.	**15 hours**
	Unit II Problem Formulation, Understanding Modeling& Simulation, Literature Review, Referencing, Information Sources, Information Retrieval, Indexing and abstracting services, Citation indexes, Development of Hypothesis, Measurement Systems Analysis, Error Propagation, Validity of experiments, Statistical Design of Experiments, Data/Variable Types & Classification, Data collection, Numerical and Graphical Data Analysis: Sampling, Observation, Interpretation of Results.	**15 hours**
	Unit III Statistics: Probability & Sampling distribution, Estimation, Measures of central Tendency, Arithmetic mean, Median, Mode, Standard deviation, Co efficient of variation (Discrete serious and continuous serious), Hypothesis testing & application, Correlation & regression analysis, Orthogonal array, ANOVA, Standard error, Concept of point and interval estimation, Level of significance, Degree of freedom, Analysis of variance, One way and two way classified data, 'F' test.	**15 hours**

	Unit IV Preparation of Dissertation and Research Papers, Tables and illustrations, Guidelines for writing the abstract, introduction, methodology, results and discussion, conclusion sections of a manuscript. References, Citation and listing system of documents. Intellectual property rights (IPR) patents copyrights Trademarks Industrial design geographical indication. Ethics of Research Scientific Misconduct Forms of Scientific Misconduct. Plagiarism, Unscientific practices in thesis work, Ethics in science.	**15 hours**
Pedagogy	Lectures/ tutorials/assignments/self-study	
References/ Readings	1. Bordens, K. S., & Abbott, B. B. (2002). Research design and methods: A process approach. McGraw-Hill. 2. Douglas C. M., & George C. R. (2007). Applied Statistics & probability for Engineers, 3rd edition, Wiley. 3. Kothari, C. R. (2004). Research methodology: Methods and techniques. New Age International. 4. Robert P. M., Peter S. M., & Mark A. L. (2012) Intellectual Property in New Technological Age. Aspen Law & Business; 6th edition 5. Shirore C. (2015). A Beginners Guide to Latex	
Course Outcomes	Upon completion of the course, students will be able to: 1. define research problems, articulate research objectives, and formulate relevant research questions. 2. Analyze research related information and statistical methods in research. 3. research design principles, distinguishing between quantitative and qualitative approaches, and appreciating the significance of building and validating theoretical models. 4. Understand the filing patent applications processes, Patent search, and various tools of IPR, Copyright, and Trademarks.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **601**
Title of the Course : **Generative Deep Learning Models**
Number of Credits : **4(2L-2T-0P)**
Contact hours : **60 hours**
Effective from AY : **2023-24**

Prerequisites for the course	Machine Learning and Deep Learning	
Course Objectives	1. Be able to build, train and apply fully connected deep neural networks and know how to implement efficient (vectorized) neural networks 2. Understand the key parameters in a neural network's architecture. 3. Be able to implement a neural network in TensorFlow. 4. to familiarize students with the current state-of-the-art in machine generation of speech, music, still images and video using Deep Learning architectures.	
Content:	**Unit I:** Introduction to Generative Deep Learning: Generative Modeling, Generative Versus Discriminative Modeling, Advances in Machine Learning, The Rise of Generative Modeling, The Generative Modeling Framework, Probabilistic Generative Models-Hello Wrodl!, Your First Probabilistic Generative Model, Naive Bayes The Challenges of Generative Modeling, Representation Learning, Setting Up Your Environment Deep Learning: Structured and Unstructured Data, Deep Neural Networks - Keras and TensorFlow, Your First Deep Neural Network, Loading the Data, Building the Model, Compiling the Model, Training the Model, Evaluating the Model, Improving the Model, Convolutional Layers, Batch Normalization, Dropout Layers. Variational Autoencoders: The Art Exhibition, Autoencoders - Your First Autoencoder, The Encoder, The Decoder, Joining the Encoder to the Decoder, Analysis of the Autoencoder The Variational Art Exhibition: Building a Variational Autoencoder- The Encoder, The Loss Function, Analysis of the Variational Autoencoder, Using VAEs to Generate Faces - Training the VAE, Analysis of the VAE, Generating New Faces, Latent Space Arithmetic, Morphing Between Faces. Generative Adversarial Networks Ganimals: Introduction to GANs Your First GAN - The Discriminator, The Generator, Training the GAN GAN Challenges - Oscillating Loss, Mode Collapse,	**15 hours**

	Uninformative Loss, Hyperparameters, Tackling the GAN Challenges Wasserstein GAN, Wasserstein Loss, The Lipschitz Constraint, Weight Clipping, Training the WGAN, Analysis of the WGAN WGAN-GP The Gradient Penalty Loss, Analysis of WGAN-GP	
	Unit II: Teaching Machines to Paint, Write, Compose, and Play, Paint-Apples and Organges - CycleGAN-Your First CycleGAN-The Generators (U-Net), The Discriminators, Compiling the CycleGAN, Training the CycleGAN, Analysis of the CycleGAN-Creating a CycleGAN to Paint Like Monet -The Generators (ResNet), Analysis of the CycleGAN-Neural Style Transfer -Content Loss, Style Loss, Total Variance Loss, Running the Neural St-Transfer, Analysis of the Neural Style Transfer Model Write-The Literary Society for Troublesome Miscreants, Long Short-Term Memory Networks, - LSTM Network Tokenization, Building the Dataset, The LSTM Architecture, The Embedding Layer, The LSTM Layer, The LSTM Cell -Generating New Text-RNN Extensions - Stacked Recurrent Networks, Gated Recurrent Units, Bidirectional Cells-Encoder–Decoder Models-A Question and Answer Generator-A Question-Answer Dataset-Model Architecture-Inference Compose Preliminaries - Musical Notation, Your First Music-Generating RNN - Attention, Building an Attention Mechanism in Keras, Analysis of the RNN with Attention, Attention in Encoder–Decoder Networks, Generating Polyphonic Music. The Musical Organ Your First MuseGAN The MuseGAN Generator Chords, Style, Melody, and Groove The Bar Generator-The Critic-Analysis of the MuseGAN	**15 hours**
Practicals to be discussed and implemented during tutorial slots	Suggested practical assignments. - Any 10 assignments can be given and the instructor can spend around 3 to 4 hours for each assignment. 1. Implement different parametric model (Generative model) for Wrodl data set. 2. Create your Anime character using GANS using the Anime data set. 3. Generate Fake Human faces using DCGANs for Data Augmentation. 4. Image Style Transfer using CycleGANs. 5. Semi-Supervised GAN (SGAN) using MNIST dataset 6. Medical Image Synthesis using GANs for Pulmonary Chest X-rays 7. Build Face Aging Application using Face Synthesis 8. Colourizing Black and White Images using GANs	**(10 *3=30 Hours)**

	9. Satellite Image to Google Map Image-to-Image Translation using DualGANs 10. Removing Unwanted Noises from Real Scene Images using GANs 11. Intent Classification using GAN-BERT 12. Generate New Human Poses using Deformable GANs 13. Create a Text-to-Image synthesizer using ST-GANs 14. Abstractive Text Summarizer using GANs 15. Liver Tumor Semantic Segmentation using SegAN 16. Anomaly Detection using GANs in MNIST Datasets	
Pedagogy	Lectures / tutorials / Assignments / ISA /seminar /Lab Assignments	
References/ Readings	1. Ahirwar, K. (2019). Generative adversarial networks projects: Build next-generation generative models using TensorFlow and Keras. Packt Publishing Ltd. 2. Atienza, R. (2018). Advanced Deep Learning with Keras: Apply deep learning techniques, autoencoders, GANs, variational autoencoders, deep reinforcement learning, policy gradients, and more. Packt Publishing Ltd. 3. Foster, D. (2022). Generative deep learning. O'Reilly Media, Inc. 4. Jakub, L. & Vladimir, B. (2019). GANs in Action: Deep learning with Generative Adversarial Networks. Manning Publications. 5. Kalin, J. (2018). Generative Adversarial Networks Cookbook: Over 100 Recipes to Build Generative Models Using Python, TensorFlow, and Keras. Packt Publishing Ltd. 6. Kuntal, G. (2017). Learning Generative Adversarial Networks: Next-generation deep learning simplified. Packt Publishing 7. Valle, R. (2019). Hands-On Generative Adversarial Networks with Keras: Your guide to implementing next-generation generative adversarial networks. Packt Publishing Ltd.	
Course Outcomes	The students will be able 1. to understand the principles of Adversarial Learning Models. 2. Know the architectural and training details of GANs. 3. Describe how Adversarial Attacks can cripple the neural networks.	

(Back to Index)

Name of the Programme : **MSc. Artificial Intelligence**
Course Code : **CSI-602**
Title of the Course : **MLOps**
Number of Credits : **4(2L-2T-0P)**
Contact hours : **60 hours**
Effective from AY : **2023-24**

Prerequisites for the course	Linear algebra, Probability theory, Machine learning, Python programming.	
Course Objectives	1. To delve into the practices and challenges in deployment of MLOP models and the role of DevOps practices in achieving this goal. 2. To deliver the previously developed ML model in production by using established DevOps practices such as testing, versioning, continuous delivery, and monitoring. 3. To automate the deployment of ML models into the core software system or as a service component. This means, to automate the complete ML-workflow steps without any manual intervention.	
Content: Theory	**Unit I:** Introduction to MLOps Rise of the Machine Learning Engineer and MLOps-What Is MLOps?-DevOps and MLOps-An MLOps Hierarchy of Needs-Implementing DevOps-Configuring-Continuous Integration with GitHub Actions-DataOps and Data Engineering-Platform Automation-MLOps MLOps Foundations-Bash and the Linux Command Line-Cloud Shell Development Environments-Bash Shell and Commands-List Files Run CommandsFiles and Navigation-Input/Output-Configuration-Writing a Script-Cloud Computing Foundations and Building Blocks-Getting Started with Cloud Computing-minimalistic python revision-Descriptive Statistics and Normal Distributions-Optimization-Machine Learning Key Concepts-Doing Data Science-Build an MLOps Pipeline from Zero MLOps for Containers and Edge Devices Containers-Container Runtime-Creating a Container Running a Container-Best Practices-Serving a Trained Model Over HTTP-Edge Devices-Coral Azure Percept-TFHub-Porting Over Non-TPU Models-Containers for Managed ML Systems-Containers in Monetizing MLOps-Build Once, Run Many MLOps Workflow Continuous Delivery for Machine Learning Models-Packaging for ML Models-Infrastructure as Code for Continuous Delivery of ML Models-Using Cloud Pipelines-Controlled Rollout of Models-Testing Techniques for Model Deployment AutoML and KaizenML-AutoML-MLOps Industrial Revolution-	**15 Hours**

	Kaizen Versus KaizenML-Feature Stores-Apple's Ecosystem-Apple's AutoML: Create ML-Apple's Core ML Tools orGoogle'sAutoML and Edge Computer Vision or Azure's AutoMLor AWS AutoML-Open Source AutoML Solutions-Ludwig-FLAML-Model Explainability Monitoring and Logging-Observability for Cloud MLOps-Introduction to Logging-Logging in Python-Modifying Log Levels-Logging Different Applications-Monitoring and Observability-Basics of Model Monitoring-Monitoring Drift with AWS Sage Maker-Monitoring Drift with Azure ML	
	Unit II MLOps for AWS-Introduction to AWS-Getting Started with AWS Services-MLOps on AWS-MLOps Cookbook on AWS-CLI Tools-Flask Microservice-AWS Lambda Recipes-AWS Lambda-SAM Local-AWS Lambda-SAM Containerized Deploy-Applying AWS Machine Learning to the Real World Machine Learning Interoperability-Why Interoperability Is Critical-ONNX: Open Neural Network Exchange-ONNX Model Zoo-Convert PyTorch into ONNX -Convert TensorFlow into ONNX-Deploy ONNX to Azure-Apple Core ML-Edge Integration. Building MLOps Command Line Tools and Microservices-Python Packaging-The Requirements File-Command Line Tools-Creating a Dataset Linter Modularizing a Command Line Tool-Microservices-Creating a Serverless Function-Authenticating to Cloud Functions-Building a Cloud-Based CLI-Machine Learning CLI Workflows Machine Learning Engineering and MLOps Case StudiesUnlikely Benefits of Ignorance in Building Machine Learning Models-MLOps Projects at Sqor Sports Social Network-Mechanical Turk Data Labeling-Influencer Rank-Athlete intelligence (AI product)-The perfect techniques versus the real world-critical challenges in MLops- Ethical and unintended consequences-lack of operational excellences- focus on prediction accuracy vs the big picture	**15 Hours**
Suggested Practical to be discussed and implemented during tutorials	1. Perfect Project Structure – Cookiecutter& readme.so 2. Speed Exploratory Data Analysis to Minutes – Pandas Profiling, SweetViz 3. Track Data Science Projects with CI, CD, CT, CM –Data Version Control (DVC) 4. Explainable AI / XAI – SHAP, LIME, SHAPASH 5. Deploy ML Projects in minutes – Docker, FastAPI 6. End to End Machine Learning – MLflow 7. Building Production Ready ML Pipelines - Model Registry,	**(3*10= 30 Hours)**

	Feature Store (Feast, ButterFlow) 8. Big Data using Python, instead of PySpark – DASK 9. Build a Chatbot and Deploy it (open-source) 10. FaaS Framework implementation – Apache OpenWhisk, OpenFaas	
Pedagogy	Lectures/ tutorials/lab assignments/self-study	
References/ Readings	1. Alla, S., & Adari, S. K. (2021). Beginning MLOps with MLFlow. Begin. MLOps with MLFlow. 2. Burkov, A. (2020). Machine Learning Engineering (Vol. 1). Montreal, QC, Canada: True Positive Incorporated. 3. David S., Dev K., Thomas H., Steven H., Dan R. & Michael O. (2020). ML Ops: Operationalizing Data Science. O'Reilly. 4. Gift, N., & Deza, A. (2021). Practical MLOps. O'Reilly Media, Inc. 5. Hapke, H., & Nelson, C. (2020). Building Machine Learning Pipelines. O'Reilly Media. 6. Treveil, M., Omont, N., Stenac, C., Lefevre, K., Phan, D., Zentici, J., & Heidmann, L. (2020). Introducing MLOps. O'Reilly Media.	
Course Outcomes	1. To understand the ML pipeline and the impact of concept of drift and data drift. 2. To figure out the gap between Proof of concept and deployment. 3. To understand the relevance of dev op practices for ML model deployment. 4. To be able to deploy ML models practically using right tools and framework like gitupactions,MLflow etc.	

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI 603**
Title of the Course : **Cloud Computing**
Number of Credits : **4(4L-0T-0P)**
Contact hours : **60 hours(60L)**
Effective from AY : **2023-24**

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

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

(Back to Index)

Name of the Programme : **M.Sc. Artificial Intelligence**
Course Code : **CSI 604**
Title of the Course : **Design thinking**
Number of Credits : **4(4L-0T-0P)**
Contact Hours : **60 hours**
Effective from AY : **2023-24**

Prerequisites for the course	None	
Course Objectives	The objective of this Course is to provide the new ways of creative thinking and Learn the innovation cycle of Design Thinking process for developing innovative products which are useful for a student in preparing for a career.	
Content	**Unit I** **An Insight to Learning** Understanding the Learning Process, Kolb's Learning Styles, Assessing and Interpreting **Remembering Memory** Understanding the Memory process, Problems in retention, Memory enhancement techniques	**15 hours**
	Unit II **Emotions: Experience & Expression** Understanding Emotions: Experience & Expression, Assessing Empathy, Application with Peers **Basics of Design Thinking** Definition of Design Thinking, Need for Design Thinking, Objective of Design Thinking, Concepts & Brainstorming, Stages of Design Thinking Process (explain with examples) – Empathize, Define, Ideate, Prototype, Test.	**15 hours**
	Unit III **Being Ingenious & Fixing Problem** Understanding Creative thinking process, Understanding Problem Solving, Testing Creative Problem Solving **Process of Product Design** Process of Engineering Product Design, Design Thinking Approach, Stages of Product Design, Examples of best product designs and functions, Assignment – Engineering Product Design **Unit III** **Prototyping & Testing** What is Prototype? Why Prototype? Rapid Prototype Development process, Testing, Sample Example, Test Group Marketing **Celebrating the Difference** Understanding Individual differences & Uniqueness, Group	**15 hours**

	Discussion and Activities to encourage the understanding, acceptance and appreciation of Individual differences	
	Unit IV **Design Thinking & Customer Centricity** Practical Examples of Customer Challenges, Use of Design Thinking to Enhance Customer Experience, Parameters of Product experience, Alignment of Customer Expectations with Product Design **Feedback, Re-Design & Re-Create** Feedback loop, Focus on User Experience, Address "ergonomic challenges, User focused design, rapid prototyping & testing, final product, Final Presentation – "Solving Practical Engineering, Problem through Innovative Product Design & Creative Solution".	**15 hours**
Pedagogy	Lectures/ Assignments/ Lab assignments/ mini project work	
References/ Readings	1. Knapp, J., Zeratsky, J., & Kowitz, B. (2016). Sprint: How to solve big problems and test new ideas in just five days. Simon and Schuster. 2. Martin, R. L. (2009). The design of business: Why design thinking is the next competitive advantage. Harvard Business Press. 3. Tim B. (2012). Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation. MUT Journal of Business Administration, 9(2), 190-194.	
Course Outcomes	Student will able to 1. Compare and classify the various learning styles and memory techniques and Apply them in their engineering education 2. Analyze emotional experience and Inspect emotional expressions to better understand users while designing innovative products 3. Develop new ways of creative thinking and Learn the innovation cycle of Design Thinking process for developing innovative products 4. Propose real-time innovative engineering product designs and Choose appropriate frameworks, strategies, techniques during prototype development	

(Back to Index)

GENERIC ELECTIVE COURSE

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

<table>
<tr><td>Prerequisites for the course</td><td colspan="2">Programme prerequisites</td></tr>
<tr><td>Course Objectives</td><td colspan="2">The course is aimed at learners to gain practical and essential skills to work effectively in the industry.</td></tr>
<tr><td rowspan="2">Content</td><td>Understanding the Industry and Companies
● Understanding the evolution of the industry and technology and methods used
● Understanding Innovation and how new Impactful ideas have evolved
● Types of companies and typical organization - Who does What
● Understanding companies - Domain, Offering, Customers, Strategy
● Company Culture & Professionalism
● Understanding companies financially</td><td>8 hours</td></tr>
<tr><td>Understanding Execution and day to day work in organizations
● Product Solutioning and Development - Understanding beyond the theory
● Product Management - Understanding beyond the theory
● Quality - Understanding beyond the theory
● Solutioning and Design - A key step between requirements and delivery
● Site Reliability, Devops, Support - Understanding beyond the theory
● Common Metrics and Measurements
● Key Tools in a Product Life Cycle
● Issues Management and Lifecycle - A key aspect of customer Satisfaction
● Software delivery models and Release cycles - how they work in the real world
● Usability by end user - UI/UX and other key concepts and its importance
● Understanding Data engineering and Data science
● Writing good product or service specifications which can be translated to building a good product
● Understanding data from collection to modeling to usage
● How to do effective product, competition or technical research and use it effectively</td><td>20 hours</td></tr>
</table>

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

	3. give reasoning. (To measure this outcome, Problem analysis and solving techniques would be taught and used, Question and answers and use cases would be utilized) 4. apply core concepts to new situations. (To measure this outcome, Group projects and Case studies based homework would be used)

(Back to Index)

SEMESTER IV
RESEARCH SPECIFIC ELECTIVE COURSES

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI-605
Title of the Course : Speech Processing
Number of Credits : 4 (3L-1T-0P)
Contact Hours : 60 hours
Effective from AY : 2023-24

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

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

(Back to Index)

Name of the Programme : **MSc Artificial Intelligence**
Course Code : **CSI-606**
Title of Course : **Advanced Machine Translation**
Number of Credits : **4(2L-2T-0P)**
Contact Hours : **60 hours**
Effective from AY :**2023-24**

Prerequisites for the course	Deep Learning, NLP, Python Programming	
Course Objectives	1. To help understand the knowledge and tools (python and deep learning) that are needed for effectively developing the key aspects of language understanding. 2. How transformers work and how to integrate them in your applications.	
Content	**Unit I** Transformers Introduction-The Encoder-Decoder Framework-Attention Mechanisms-Transfer Learning in -NLP-Hugging Face Transformers: Bridging the Gap-A Tour of Transformer Applications-Text Classification-Named Entity Recognition-Question AnsweringSummarization-Translation -Text Generation-The Hugging Face Ecosystem-The Hugging Face Hub-Hugging Face Tokenizers-Hugging Face Datasets-Hugging Face Accelerate-Main Challenges with Transformers Model architecture Architecture of Transformer – The Encoder stack -The decoder stack - The Encoder-Self-Attention-The Feed-Forward Layer-Adding Layer Normalization-Positional Embeddings-Adding a Classification Head-The Decoder-Meet the Transformers-The Transformer Tree of Life-The Encoder Branch-The Decoder Branch-The Encoder-Decoder Branch Text Generation-The Challenge with Generating Coherent Text-Greedy Search Decoding-Beam Search Decoding-Sampling Methods-Top-k and Nucleus SamplingWhich Decoding Method Is Best?	**15 Hours**
	Unit II Summarization -The CNN/DailyMail Dataset-Text Summarization Pipelines-Summarization Baseline-GPT-2-T5-BART-PEGASUS-Comparing Different Summaries-Measuring the Quality of Generated Text-BLEU-ROUGEEvaluating PEGASUS on the CNN/DailyMail Dataset-Training a Summarization Model-Evaluating PEGASUS on SAMSum-Fine-Tuning PEGASUS-Generating Dialogue Summaries Question Answering-Building a Review-Based QA System-The Dataset-Extracting Answers from Text-Using Haystack to Build a QA Pipeline-Improving Our QA Pipeline-Evaluating the Retriever-Evaluating the Reader-Domain Adaptation-Evaluating the Whole	**15 Hours**

	QA Pipeline-Going Beyond Extractive QA Training Transformer from scratch-Large Datasets and Where to Find Them-Challenges of Building a Large-Scale Corpus-Building a Custom Code Dataset-Working with Large Datasets-Adding Datasets to the Hugging Face Hub-Building a Tokenizer-The Tokenizer Model-Measuring Tokenizer Performance-A Tokenizer for Python-Training a Tokenizer-Saving a Custom Tokenizer on the Hub-Training a Model from Scratch-A Tale of Pretraining Objectives-Initializing the Model-Implementing the Dataloader-Defining the Training Loop-The Training R	
	Suggested lab work: 1. Machine Translation with transformers. 2. Applying Transformers to Legal and financial documents for AI Text summarization. 3. Detecting customer emotions to make predictions. 4. Analyzing Fake news with transformers. 5. Multilingual Named Entity Recognition	**(5 * 6 =30 Hours)**
Pedagogy	Lectures/ Tutorials/Lab assignments/ mini project work/ISA/Seminar.	
References/ Readings	1. Rothman, D. (2021). Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more. Packt Publishing Ltd. 2. Tunstall, L., Von Werra, L., & Wolf, T. (2022). Natural Language Processing with Transformers. O'Reilly Media, Inc.	
Course Outcomes	Student will able to 1. Build, debug, and optimize transformer models for core NLP tasks, such as text classification, named entity recognition, and question answering. 2. Learn how transformers can be used for cross-lingual transfer learning. 3. Apply transformers in real-world scenarios where labeled data is scarce. 4. Make transformer models efficient for deployment using techniques such as distillation, pruning, and quantization.	

(Back to Index)

Name of the Programme : **MSc Artificial Intelligence**
Course Code : **CSI-607**
Title of the Course : **Data Engineering**
Number of Credits : **4(2L-2T-0P)**
Contact hours : **60 hours**
Effective from AY : **2023-24**

Prerequisites for the course	Data Base Fundamentals, Programming skills, mathematics and statistics	
Course Objectives	The objective is to acquire proficiency in data preparation, data integration, data storage, and management, support for analytics, scalability, and real-time processing, encompassing the comprehensive skills needed for the effective handling and utilization of data in various contexts.	
Content	**Unit I:** Foundation and Building Blocks Data Engineering Described-What Is Data Engineering? -Data Engineering Defined-The Data Engineering Lifecycle-Evolution of the Data Engineer-Data Engineering and Data Science-Data Engineering Skills and Activities-Data Maturity and the Data Engineer The Data Engineering Lifecycle What Is the Data Engineering Lifecycle?-The Data Lifecycle Versus the Data Engineering Lifecycle-Generation: Source Systems-Storage-Ingestion-Transformation-Serving Data-Major Undercurrents Across the Data Engineering Lifecycle Designing Good Data Architecture What Is Data Architecture?-Enterprise Architecture Defined-Data Architecture Defined-“Good” Data Architecture-Principles of Good Data Architecture-Principles -Major Architecture Concepts-Domains and Services-Distributed Systems, Scalability, and Designing for Failure-Tight Versus Loose Coupling: Tiers, Monoliths, and Microservices-User Access: Single Versus Multitenant-Event-Driven Architecture-Brownfield Versus Greenfield Projects-Examples and Types of Data Architecture-Data Warehouse-Data Lake-Convergence, Next-Generation Data Lakes, and the Data Platform-Modern Data Stack-Lambda Architecture-Kappa Architecture-The Dataflow Model and Unified Batch and Streaming-Architecture for IoT-Data Mesh-Other Data Architecture Examples-Who’s Involved with Designing a Data Architecture? Data Generation in Source Systems Sources of Data: How Is Data Created?-Source Systems: Main Ideas-Files and Unstructured DataAPIs-Application Databases (OLTP Systems)-Online Analytical Processing System-Change Data Capture-Logs-Database Logs-CRUD-Insert-Only-Messages and Streams-Types of Time-Source System Practical Details-	**15 hours**

	Databases-APIs-Data Sharing-Third-Party Data Sources-Message Queues and Event-Streaming Platforms-Undercurrents and Their Impact on Source Systems Storage Raw Ingredients of Data Storage-Magnetic Disk Drive-Solid-State Drive-Random Access Memory-Networking and CPU-Serialization-Compression-Caching-Data Storage Systems-Single Machine Versus Distributed Storage-Eventual Versus Strong Consistency-File Storage-Block Storage-Object Storage-Cache and Memory-Based Storage Systems-The Hadoop Distributed File System-Streaming Storage-Indexes, Partitioning, and Clustering-Data Engineering Storage Abstractions-The Data Warehouse-The Data Lake-The Data Lakehouse-Data Platforms-Stream-to-Batch Storage -Architecture-Big Ideas and Trends in Storage-Data Catalog-Data Sharing-Schema-Separation of Compute from Storage-Data Storage Lifecycle and Data Retention-Single-Tenant Versus Multitenant Storage	
	Unit II Ingestion What Is Data Ingestion?-Key Engineering Considerations for the Ingestion Phase-Bounded Versus Unbounded Data-Frequency-Synchronous Versus Asynchronous Ingestion-Serialization and Deserialization-Throughput and Scalability-Reliability and Durability-Payload-Push Versus Pull Versus Poll Patterns-Batch Ingestion Considerations-Snapshot or Differential Extraction-File-Based Export and Ingestion-ETL Versus ELT-Inserts, Updates, and Batch Size-Data Migration-Message and Stream Ingestion Considerations-Schema Evolution-Late-Arriving Data-Ordering and Multiple Delivery-ReplayTime to Live-Message Size-Error Handling and Dead-Letter Queues-Consumer Pull and PushWays to Ingest Data-Direct Database Connection-Change Data Capture-APIs-Message Queues and Event-Streaming Platforms-Managed Data Connectors-Moving Data with Object Storage-EDIDatabases and File Export-Practical Issues with Common File Formats-Shell-SSH-SFTP and SCP-Webhooks-Web Interface-Web Scraping-Transfer Appliances for Data Migration-Data Sharing Queries, Modeling, and Transformation Queries-What Is a Query?-The Life of a Query-The Query Optimizer-Improving Query Performance-Queries on Streaming Data-Data Modeling-What Is a Data Model?-Conceptual, Logical, and Physical Data Models-Normalization-Techniques for Modeling Batch Analytical Data-Modeling Streaming Data-Transformations-Batch Transformations-Materialized Views, Federation, and Query Virtualization-Streaming Transformations and Processing Serving Data for Analytics, Machine Learning, and Reverse ETL-General Considerations for Serving Data-Trust-What's the Use	**15 hours**

	Case, and Who's the User?-Data Products-Self-Service or Not?-Data Definitions and Logic-Data Mesh-Analytics-Business Analytics-Operational Analytics-Embedded Analytics-Machine Learning-What a Data Engineer Should Know About ML-Ways to Serve Data for Analytics and ML-File Exchange-Databases-Streaming Systems-Query Federation-Data Sharing-Semantic and Metrics Layers-Serving Data in Notebooks-Reverse ETL Security and Privacy-People-The Power of Negative Thinking-Always Be Paranoid-Processes-Security Theater Versus Security Habit-Active Security-The Principle of Least Privilege-Shared Responsibility in the Cloud-Always Back Up Your Data-An Example Security Policy-Technology-Patch and Update Systems-Encryption-Logging, Monitoring, and Alerting-Network Access-Security for Low-Level Data Engineering The Future of Data Engineering The Data Engineering Lifecycle Isn't Going Away-The Decline of Complexity and the Rise of Easy-to-Use Data Tools-The Cloud-Scale Data OS and Improved Interoperability-"Enterprisey" Data Engineering	
	Tutorial sessions (ANY SIX): Preliminaries required to be understood - Python data processing.,csv, flat-file, parquet, json, etc.,SQL database table design.,Python + Postgres, data ingestion and retrieval.,PySpark,Data cleansing / dirty data. (How to work on the problems. You will need two things to work effectively on most of these problems.) Docker,docker-compose (All the tools and technologies you need will be packaged into the dockerfile for each exercise. For each exercise you will need to cd into that folder and run the docker build command, that command will be listed in the README for each exercise, follow those instructions.) Exercise 1 - Downloading files. The first exercise tests your ability to download a number of files 2 from an HTTP source and unzip them, storing them locally with Python. cd Exercises/Exercise-1 and see README in that location for instructions. Exercise 2 - Web Scraping + Downloading + Pandas The second exercise tests your ability to perform web scraping, build uris, download files, and use Pandas to do some simple cumulative actions. cd Exercises/Exercise-2 and see README in that location for instructions. Exercise 3 - Boto3 AWS + s3 + Python. The third exercise tests a few skills. This time we will be using a popular aws package called boto3 to try to perform a multi-step action to download some open source s3 data files. cd	**5x6=30 hours**

	Exercises/Exercise-3 and see README in that location for instructions. Exercise 4 - Convert JSON to CSV + Ragged Directories. The fourth exercise focuses more on file types json and csv, and working with them in Python. You will have to traverse a ragged directory structure, finding any json files and converting them to csv. Exercise 5 - Data Modeling for Postgres + Python. The fifth exercise is going to be a little different than the rest. In this problem you will be given a number of csv files. You must create a data model / schema to hold these data sets, including indexes, then create all the tables inside Postgres by connecting to the database with Python. Exercise 6 - Ingestion and Aggregation with PySpark. The sixth exercise Is going to step it up a little and move onto more popular tools. In this exercise we are going to load some files using PySpark and then be asked to do some basic aggregation Exercise 7 - Using Various PySpark Functions The seventh exercise Taking a page out of the previous exercise, this one is focus on using a few of the more common build in PySpark functions pyspark.sql.functions and applying their usage to real-life problems. Many times to solve simple problems we have to find and use multiple functions available from libraries. This will test your ability to do that. **Exercise 8: Project work (ANY ONE):** 1. Scrape Stock and Twitter Data Using Python, Kafka, and Spark 2. Scrape Real-Estate Properties With Python and Create a Dashboard With It 3. Focus on Analytics With Stack Overflow Data 4. Instead of Stocks, Predict Political and Financial Events WithPredictIt 5. Scraping Inflation Data and Developing a Model With Data From CommonCrawl	
Pedagogy	lectures/ tutorials/lab assignments/self-study/ flipped classroom	
References/ Readings	1. Burkov, A. (2020). Machine learning engineering (Vol. 1). Montreal, QC, Canada: True Positive Incorporated. 2. Chris, F., Antje B. (2021). Data Science on AWS: Implementing End-to-End, Continuous AI and Machine Learning Pipelines. O'Reilly Media, Inc, USA. 3. Densmore, J. (2021). Data pipelines pocket reference. O'Reilly Media. 4. Hamid, M.Q., Hammad, S. (2021). Snowflake Cookbook: Techniques for building modern cloud data warehousing solutions. Packt Publishing. 5. Macey, T. (2021). 97 Things Every Data Engineer Should Know. "	

	O'Reilly Media, Inc.". 6. Martin, K. (2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. 7. Paul, C. (2020). Data Engineering with Python: Work with massive datasets to design data models and automate data pipelines using Python. Packt Publishing Limited. 8. Ralph, K., Margy, R.(2013) The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. John Wiley & Sons 9. Walker, M. (2020). Python Data Cleaning Cookbook: Modern techniques and Python tools to detect and remove dirty data and extract key insights. Packt Publishing Ltd.
Course Outcomes	After completion of this course, students will be able to: 1. Design and implement data engineering solutions, applying analytic algorithms to sample datasets for practical insights and problem-solving. 2. Acquire proficiency in developing machine-learning models tailored for real-world datasets, understanding the intricacies of model development and evaluation. 3. Evaluate the effectiveness of analytic algorithms on diverse datasets, fostering a nuanced understanding of their performance in various contexts. 4. Demonstrate the ability to apply machine-learning models to real-world scenarios, showcasing a practical grasp of deploying and assessing models in practical applications.

(Back to Index)

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI-608
Title of the Course : Financial Machine Learning
Number of Credits : 4(2L-2T-0P)
Contact Hours : 60 hours
Effective from AY :2023-24

Prerequisites for the course	Machine learning and probability and statistics	
Course Objectives	The course aims to equip students to use machine learning for cost reduction through process automation, improve revenue generation with faster decision-making, enhance customer experiences by prioritizing critical issues automatically, and bolster security through expedited fraud detection in the financial domain.	
Content	**Unit I:** Financial Machine Learning as a distinct subject- DATA ANALYSIS-Financial Data Structure- Essential Types of Financial Data- Bars- Dealing with Multi-Product Series-Sampling Features LABELING -The Fixed-Time Horizon Method - Computing Dynamic Thresholds - The Triple-Barrier Method - Learning Side and Size - Meta-Labeling - How to Use Meta-Labeling - The Quantamental Way - Dropping Unnecessary Labels SAMPLE WEIGHTS - Overlapping Outcomes - Number of Concurrent Labels-Average Uniqueness of a Label-Bagging Classifiers and Uniqueness- Return Attribution-Time Decay- Class Weights Fractionally Differentiated Features: The Stationarity vs. Memory Dilemma- Literature Review - The Method - Implementation - Stationarity with Maximum Memory Preservation. Ensemble Methods - The Three Sources of Errors - Aggregation - Random Forest - Boosting - Bagging vs. Boosting in Finance - Bagging for Scalability	**15 hours**
	Unit II Cross-Validation in Finance - The Goal of Cross-Validation - Why K-Fold CV Fails in Finance - A Solution: Purged K-Fold CV - Bugs in Sklearn's Cross-Validation. Feature Importance - The Importance of Feature Importance - Feature Importance with Substitution Effects - Feature Importance without Substitution Effects - Parallelized vs. Stacked Feature Importance - Experiments with Synthetic Data Hyper-Parameter Tuning with Cross-Validation - Grid Search Cross-Validation - Randomized Search Cross-Validation - Scoring and Hyper-parameter Tuning HIGH-PERFORMANCE COMPUTING RECIPES	**15 hours**

	Multiprocessing and Vectorization - Vectorization Example Single-Thread vs. Multithreading vs. Multiprocessing,Atoms and Molecules, Multiprocessing Engines, Multiprocessing Example	
	Suggested tutorial assignments (ANY SIX): Assignment -1 :- Process Automation In finance and insurance, employees spend more than half their time collecting and processing data. By implementing machine learning tools, companies can automate a large part of routine and time-consuming processes, increase productivity, save costs, and free up employees so they can focus on higher value-added tasks. Assignment-2 :- Document Analysis Text analysis tools use machine learning to make sense of unstructured data. These tools are helping companies in the finance industry gain value from their data in a fast and cost-effective way while reducing human error. Applications range from automatically classifying data in emails, contracts, and reports, to extracting relevant information from legal documents, statements, and bills. Assignment-3:- Portfolio Management Robo-advisors are one of the most popular applications of machine learning in finance. A robo-advisor is an intelligent system that uses machine learning algorithms and statistics. Robo-advisors are often used to provide investment advice and portfolio management services to clients. By processing large amounts of data in a short space of time, robo-advisors can help customers stay ahead and make smart and well-informed investment decisions. Assignment-4: - Algorithmic Trading Algorithmic trading helps businesses make fast and highly accurate trading decisions. Machine learning algorithms are trained to identify trading opportunities, by recognizing patterns and behaviors in historical data. Assignment-5 :- Digital Assistants The use of machine learning bots is gaining momentum in the banking industry, helping companies create better experiences in customer service while saving money on call centers. Chatbots, for instance, are equipped with machine learning algorithms and trained to handle common and non-critical customer queries around the clock, scaling support, and improving customer satisfaction. Assignment -6 :- Risk Management There is a huge amount of risk involved in the finance sector: market risk, credit risk, operational risk, regulatory risk, and so on. In the last few years, financial companies have increasingly been adopting AI and machine learning to improve risk management,	**6x5= 30 hours**

	helping them to detect and quantify risks, and make the right decisions. Machine learning algorithms can constantly monitor and analyze large sets of data, in order to spot trends and patterns and deliver critical information in real-time. Assignment-7: - Fraud Detection & Money Laundering Prevention Machine learning is now a key player in the constant battle against fraudulent transactions and money laundering. This technology can detect anomalies in large sets of historical data, and monitor operations in real-time for suspicious behavior, alerting financial services to security threats and illegal activities in real time.	
Pedagogy	lectures/ tutorials/lab assignments/self-study/ flipped classroom	
References/ Readings	1. Burkov, A. (2019). The hundred-page machine learning book (Vol. 1, p. 32). Quebec City, QC, Canada: Andriy Burkov. 2. Cartea, Á., Jaimungal, S., &Penalva, J. (2015). Algorithmic and high-frequency trading. Cambridge University Press. 3. De Prado, M. L. (2018). Advances in financial machine learning. John Wiley & Sons. 4. Ruppert, D., & Matteson, D. S. (2011). Statistics and data analysis for financial engineering (Vol. 13). New York: Springer.	
Course Outcomes	Upon completion of the course, students will be able to: 1. Understand Financial Machine Learning, covering data analysis, financial data structures, and methods for handling multi-product series and sampling features. 2. develop expertise in labeling techniques, along with mastering the application of sample weights and Fractionally Differentiated Features to enhance data analysis in the financial domain. 3. Apply Cross-Validation in Finance domain 4. Understand ensemble Methods such as Random Forest, Boosting, and the application of bagging in the financial domain for improved scalability.	

(Back to Index)

Name of the Programme : M.Sc. Artificial Intelligence
Course Code : CSI 609
Title of Course : Recommender systems
Number of Credits : 4 (2L-2T-0P)
Contact Hours : 60 hours
Effective from AY : 2023-24

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

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

(Back to Index)