## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# STATISTICS AND PROBABILITY FOR DATA SCIENCE

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – I**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1101 | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:** This Course will enable students to:

1. To understand fundamental concepts of probability and statistics.
2. To analyze and interpret data using statistical methods.
3. To apply statistical inference techniques for decision-making.
4. To develop predictive models using regression techniques.
5. To introduce probabilistic models used in data science applications.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: PROBABILITY THEORY** | **09 Hours** |
|---|---|

Sample Spaces- Events - Axioms – Counting - Conditional Probability and Bayes' Theorem – The Binomial Theorem – Random variable and distributions: Mean and Variance of a Random variable. R Commands for central tendency, variability, and structure of your numerical data. Binomial-Poisson-Exponential and Normal distributions. R commands for probability distribution.

| MODULE – II: SAMPLING DISTRIBUTIONS & DESCRIPTIVE STATISTICS | 09 Hours |
|---|---|
| Curve Fitting and Principles of Least Squares- Regression and correlation. Core R commands for regression and correlation analysis. The Central Limit Theorem, Applying CLT in R, distributions of the sample mean and the sample variance for a normal population, Sampling distributions (Chi-Square, t, F, z). Using R for Sampling Distributions. Test of Hypothesis- Testing for Attributes –Mean of Normal Population – One-tailed and two-tailed tests. Hypothesis testing in R. | |
| | |
| **MODULE – III: TABULAR DATA ANALYSIS** | **09 Hours** |
| F-test and Chi-Square test-Analysis of variance ANOVA – One-way and two-way classifications. F-tests, Chi-Square tests, and ANOVA in R. Tabular data- Power and the computation of sample size-Advanced data handling, Working with tabular data in R. | |
| | |
| **MODULE – IV: DENSITY ESTIMATION** | **09 Hours** |
| Multiple regression- Linear models- Logistic regression- Rates and Poisson regression Nonlinear curve fitting. Multiple regression and Logistic regression using R. Density Estimation- Recursive Partitioning- Smoothers and Generalised Additive Models. Applying density estimation, Recursive partitioning and Generalized additive model using R. | |
| | |
| **MODULE – V: SURVIVALS ANALYSIS** | **09 Hours** |
| Survivals Analysis- Analysing Longitudinal Data- Simultaneous Inference and Multiple Comparisons- Meta-Analysis- Survivals analysis using R. Principal Component Analysis (PCA)-Multidimensional Scaling Cluster Analysis (MSC). PCA and MSC using R. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Apply probability concepts to real-world problems | L2 & L3 |
| 2 | Perform exploratory data analysis using statistical techniques | L2 & L3 |
| 3 | Conduct hypothesis testing and interpret results | L3 |
| 4 | Build regression models for prediction | L4 |
| 5 | Apply basic probabilistic models in data science | L4 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 3 | 1 | 2 | 2 | - | - | - | - | - | 2 | 3 | 2 | - |
| CO2 | 3 | 3 | 2 | 3 | 3 | - | - | 1 | 1 | - | 2 | 3 | 3 | - |
| CO3 | 3 | 3 | 2 | 3 | 2 | - | 1 | 1 | 1 | - | 2 | 3 | 2 | 1 |
| CO4 | 3 | 3 | 3 | 3 | 3 | - | - | 1 | - | - | 3 | 3 | 3 | - |
| CO5 | 3 | 3 | 2 | 3 | 3 | - | - | 1 | - | - | 3 | 3 | 3 | - |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS**:

1. Statistical Inference – Casella, G., & Berger, R.
2. Applied Statistics and Probability for Engineers – Montgomery, D. C., & Runger, G.

**REFERENCE BOOKS:**

1. Richard Cotton, “Learning R”, O’Reilly, 2013.
2. Dalgaard, Peter, “Introductory statistics with R”, Springer Science & Business Media, 2008.
3. Brain S. Everitt, “A Handbook of Statistical Analysis Using R”, Second Edition, LLC, 2014.
4. Samir Madhavan, “Mastering Python for Data Science”, Packt, 2015.
5. Sheldon M. Ross,”Introduction to Probability and Statistics for Engineers and Scientists”, 4th edition, Academic Press; 2009.
6. Paul Teetor, “R Cookbook, O’Reilly, 2011.
7. Mark Lutz ,”Learning Python”, O’Reilly,5th Edition,2013

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# ADVANCED MACHINE LEARNING

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – I**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1102 | **Credits** | : | 04 |
| **Hours / Week** | : | 05 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–2–0 | | | |

**Course Learning Objectives:**

1. Develop a rigorous mathematical understanding of advanced supervised learning paradigms including kernel methods, Gaussian Processes, and Bayesian ensemble optimization.
2. Analyze and apply advanced unsupervised and semi-supervised learning algorithms including contrastive learning, variational autoencoders, and deep anomaly detection.
3. Model uncertainty in complex systems using probabilistic graphical models, Monte Carlo inference, and Bayesian deep learning techniques.
4. Design and evaluate state-of-the-art neural architectures including attention-based transformers for non-NLP domains, Graph Neural Networks, and meta-learning frameworks.
5. Assess the interpretability, fairness, calibration, and ethical implications of advanced machine learning systems deployed in high-stakes applications.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: ADVANCED SUPERVISED LEARNING AND KERNEL METHODS | 09 Hours |
|---|---|
| Bias-variance tradeoff; Ridge, Lasso, Elastic Net; kernel SVM and SVR; Gaussian Process regression (RBF, Matérn); Bayesian linear regression; XGBoost, LightGBM, CatBoost; Bayesian hyperparameter optimization (GP surrogate, EI, UCB). | |
| | |
| **MODULE – II: ADVANCED UNSUPERVISED AND SEMI-SUPERVISED LEARNING** | **09 Hours** |
| DBSCAN, HDBSCAN, GMM; t-SNE and UMAP; semi-supervised learning with FixMatch; contrastive learning with SimCLR; Variational Autoencoders (ELBO, reparameterization trick); Isolation Forest for anomaly detection. | |
| | |
| **MODULE – III: PROBABILISTIC GRAPHICAL MODELS AND BAYESIAN INFERENCE** | **09 Hours** |
| Bayesian networks and d-separation; Hidden Markov Models (Viterbi, Baum-Welch); variational inference; MCMC: Gibbs sampling, HMC, NUTS; EM algorithm for GMM; probabilistic programming with PyMC; Monte Carlo Dropout for uncertainty quantification. | |
| | |
| **MODULE – IV: ADVANCED NEURAL ARCHITECTURES AND MODERN TRAINING PARADIGMS** | **09 Hours** |
| AdamW, SAM, cosine annealing; multi-head attention and Flash Attention; Vision Transformer (ViT), Swin Transformer; GCN, GAT (PyTorch Geometric); MAML for few-shot learning; knowledge distillation; quantization (QAT) and LoRA; FedAvg; SHAP and GradCAM. | |
| | |
| **MODULE – V: SCALABLE AND GPU-ACCELERATED MACHINE LEARNING** | **09 Hours** |
| RAPIDS cuDF, cuML, cuGraph; GPU-accelerated feature engineering; PyTorch DDP distributed training; Automatic Mixed Precision (AMP, FP16/BF16); TensorRT INT8 inference optimization; MLflow experiment tracking. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Demonstrate the ability to design and implement advanced supervised learning models including kernel methods, Gaussian Processes, and Bayesian ensemble techniques for complex, high-dimensional real-world problems. | L3 |
| 2 | Apply probabilistic graphical models, variational inference, and Monte Carlo methods to model uncertainty and perform Bayesian reasoning in structured and sequential learning tasks. | L4 |
| 3 | Develop and evaluate state-of-the-art neural architectures including attention-based transformers, Graph Neural Networks, meta-learning frameworks, and model compression pipelines. | L5 |
| 4 | Build scalable GPU-accelerated machine learning pipelines, distributed training with PyTorch DDP, mixed-precision training, and production deployment using TensorRT and NIM. | L3 |
| 5 | Critically assess model interpretability, fairness, calibration, and ethical implications of advanced machine learning systems deployed in high-stakes real-world domains. | L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 3 | 3 | 1 |
| **CO2** | 3 | 3 | 2 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 3 | 3 | 1 |
| **CO3** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | 1 | 3 | 3 | 3 | 2 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | 1 | 3 | 3 | 3 | 1 |
| **CO5** | 2 | 3 | 2 | 3 | 2 | 3 | 3 | 1 | 1 | - | 3 | 2 | 2 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

## LAB EXPERIMENT

1. Bayesian Hyperparameter Optimization: implement GP-based HPO using Optuna on a classification dataset; compare convergence with grid and random search.
2. Gaussian Process Regression: fit RBF and Matérn-5/2 kernels using GPyTorch; visualize posterior mean and uncertainty bounds on a regression dataset.
3. UMAP + HDBSCAN Clustering: apply UMAP on MNIST; cluster with HDBSCAN; evaluate

Silhouette and Davies-Bouldin scores.

4. Semi-Supervised Learning with FixMatch: train on CIFAR-10 with 1% labeled data; compare accuracy against fully supervised baseline.
5. Bayesian Logistic Regression with PyMC: build and sample a Bayesian classification model using NUTS; perform posterior predictive checks.
6. Monte Carlo Dropout for Uncertainty: implement MC Dropout in PyTorch; compute epistemic uncertainty; plot reliability diagrams and compute ECE.
7. MAML Few-Shot Classification: implement Model-Agnostic Meta-Learning on miniImageNet; evaluate 5-way 1-shot accuracy.
8. Graph Neural Networks: implement GCN and GAT using PyTorch Geometric on Cora dataset for node classification; compare accuracy.
9. Knowledge Distillation + QAT: distill ResNet-50 teacher to MobileNetV2 student on CIFAR-100; apply INT8 QAT; report accuracy vs. speed tradeoff.
10. Variational Autoencoder: implement VAE from scratch in PyTorch on Fashion-MNIST; visualize latent space interpolations.
11. GPU-Accelerated ML with RAPIDS: migrate a scikit-learn pipeline (PCA, K-Means, Random Forest) to RAPIDS cuML; benchmark CPU vs. GPU on 1M rows.
12. Distributed Training + TensorRT: implement multi-GPU training with PyTorch DDP and AMP; export to TensorRT INT8; measure inference latency reduction.

**TEXT BOOKS**:

1. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
2. Murphy, K. P. (2022). Probabilistic Machine Learning: An Introduction. MIT Press.
3. Murphy, K. P. (2023). Probabilistic Machine Learning: Advanced Topics. MIT Press.
4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
5. Rasmussen, C. E., & Williams, C. K. I. (2006). Gaussian Processes for Machine Learning. MIT Press.

**REFERENCE BOOKS**

1. Hamilton, W. L. (2020). Graph Representation Learning. Morgan & Claypool / Synthesis Lectures on AI and ML.
2. Finn, C., Abbeel, P., & Levine, S. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. ICML 2017.
3. Chen, T. et al. (2020). A Simple Framework for Contrastive Learning of Visual Representations (SimCLR). ICML 2020.
4. Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions (SHAP). NeurIPS 2017.
5. McInnes, L., Healy, J., & Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection. arXiv:1802.03426.

# DATA ANALYTICS AND VISUALIZATION

**[As per Choice Based Credit System (CBCS) scheme]**

**SEMESTER – I**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1103 | **Credits** | : | 04 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–2–0 | | | |

**Course Learning Objectives:**

**This course will enable students to:**

1. Understand the fundamental differences and relationships among data science concepts such as statistics, data mining, analytics, and data science itself.
2. Familiarize with Python-based data analytics libraries like NumPy, SciPy, Pandas, and visualization tools such as Matplotlib, Plotly, and Bokeh.
3. Apply advanced visualization methodologies such as dendrograms, linked views, and projection-based techniques for exploratory data analysis.
4. Visualize clustering techniques and complex structures like mixture models, contingency tables, and kernel-based representations.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: INTRODUCTION TO DATA ANALYTICS** | **09 Hours** |
|---|---|
| Statistics vs Data Mining vs Data Analytics vs Data Science; Understanding data: Introduction, Types of Data, Types of Data Distribution: Normal distribution, Poisson Distribution, Binomial Distribution, Uniform distribution, Data Architecture and its components, Structured and Unstructured data. Steps of data analysis, Python Packages for Data Analysis: Numpy, Scipy, Matplotlib, Ploty/Bokeh. Data Frames, Usage of frames analytical roles, File handling and reading data for processing. | |
| | |
| **MODULE – II: VISUALIZATION IN HIGH-DIMENSIONAL DATA** | **09 Hours** |
| Introduction, Computational Statistics and Data Visualization; A Brief History of Data Visualization, Good Graphics, Static Graphics, Data Visualization Through Their Graph Representations. Graph-theoretic Graphics, High-dimensional Data Visualization, Multivariate Data Glyphs: Principles and Practice, Linked Views for Visual Exploration, Linked Data Views, Visualizing Trees and Forests. | |
| | |
| **MODULE – III: VISUALIZATION METHODOLOGIES** | **09 Hours** |
| Interactive Linked Micromap Plots for the Display of Geographically Referenced Statistical Data, Grand Tours, Projection Pursuit Guided Tours and Manual Controls, Multidimensional Scaling, Huge Multidimensional Data Visualization: Back to the Virtue of Principal Coordinates and Dendrograms in the New Computer Age, Multivariate Visualization by Density Estimation. | |
| | |
| **MODULE – IV: VISUALIZATION OF CLUSTER ANALYSIS** | **09 Hours** |
| Structured Sets of Graphs, Regression by Parts: Fitting Visually Interpretable Models with GUIDE, Structural Adaptive Smoothing by Propagation–Separation Methods, Smoothing Techniques for Visualization, Data Visualization via Kernel Machines, Visualizing Cluster Analysis and Finite Mixture Models, Visualizing Contingency Tables. | |
| | |
| **MODULE – V: APPLICATIONS** | **09 Hours** |
| Visualization for Genetic Network Reconstruction, Reconstruction, Visualization and Analysis of Medical Images, Exploratory Graphics of a Financial Dataset, Graphical Data Representation in Bankruptcy Analysis, and Visualizing Functional Data with an Application to eBay's Online Auctions, Visualization Tools for Insurance Risk Processes. | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Demonstrate a clear understanding of the various types of data, distributions, and the overall analytics process. | L2 |
| 2 | Utilize Python libraries to perform data analysis and generate fundamental visualizations. | L3 |
| 3 | Analyze and interpret high-dimensional data using advanced visualization techniques including graph-theoretic and glyph-based methods. | L4 |
| 4 | Evaluate and compare different visualization methodologies for cluster analysis, regression, smoothing, and kernel machines. | L5 |
| 5 | Investigate real-world applications of data visualization in various domains like medical imaging, finance, and insurance, etc. | L4 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | - | 2 | 2 | - | - | - | - | - | 2 | 3 | 2 | - |
| **CO2** | 3 | 3 | 2 | 3 | 3 | - | - | 1 | 1 | - | 2 | 3 | 3 | - |
| **CO3** | 3 | 3 | 2 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 3 | 3 | 1 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | - |
| **CO5** | 2 | 3 | 2 | 3 | 2 | 2 | 1 | 1 | 1 | - | 3 | 3 | 2 | 2 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

## LAB EXPERIMENT

1. **Retail Sales Analysis**: A retail company wants to understand its sales performance across different product categories, customer segments, and regions. Import the provided retail sales dataset into Power BI/Tableau, explore the tool interface, and create appropriate visualizations to identify top-performing products, regions, and customer groups. Summarize the key business insights derived from the analysis.
2. **Customer Transaction Data Cleaning**: A company has collected customer transaction data from multiple branches, but the dataset contains missing values, duplicate records, and inconsistent formats. Using Power BI/Tableau, clean and preprocess the dataset by handling data quality issues, transforming attributes where necessary, and preparing a reliable dataset suitable for further analysis and reporting.

3. **E-Commerce Customer Behavior Analysis**: An e-commerce organization seeks to understand customer purchasing patterns and spending behavior. Perform exploratory data analysis on the provided dataset using suitable statistical and visualization techniques to identify trends, correlations, outliers, and customer segments, and present the key findings through meaningful visualizations.
4. **Retail Performance Dashboard Development**: A national retail chain requires an interactive dashboard to monitor business performance. Using the given sales dataset, design a dashboard that displays key performance indicators such as revenue, profit, profit margin, and regional sales performance, incorporating interactive filters and visualizations to support managerial decision-making.
5. **Electricity Consumption Trend Analysis**: An electricity distribution company wants to analyze historical energy consumption patterns and predict future demand. Using the provided time-series dataset, create visualizations to identify trends, seasonal variations, and consumption patterns, and generate forecasts to support resource planning and operational decision-making.
6. **Food Delivery Geographic Analysis**: A food delivery service aims to evaluate its regional performance and identify areas with high and low demand. Using geographic and order-related datasets, create map-based visualizations to analyze delivery patterns, customer concentration, and service coverage, and provide recommendations for improving operational efficiency.
7. **Hospital Performance Analytics**: A hospital administration intends to monitor patient admissions, treatment outcomes, and resource utilization. Using the provided healthcare dataset, develop advanced visualizations employing calculated fields, measures, and parameters to evaluate departmental performance and identify opportunities for improving healthcare service delivery.
8. **Student Performance Dashboard**: A university seeks to analyze student academic performance across departments, programs, and semesters. Develop an interactive dashboard with drill-down and drill-through capabilities that enables stakeholders to explore performance metrics at various levels and identify factors influencing academic outcomes.
9. **Smart City Traffic Monitoring**: A municipal authority plans to monitor urban traffic conditions using live sensor data. Create a real-time dashboard in Power BI that displays traffic volume, congestion levels, and vehicle movement patterns, enabling city administrators to monitor and respond to traffic conditions effectively.
10. **COVID-19 Public Health Analytics**: A public health agency requires a comprehensive analysis of COVID-19 data to support policy decisions. Using publicly available datasets, perform data preprocessing, exploratory analysis, trend analysis, and geographic visualization, and develop a comprehensive dashboard that presents key indicators, patterns, and actionable insights related to disease spread and vaccination progress.

**TEXT BOOKS**

1. Bharti Motwani, Data Analytics using Python (1 ed.), Wiley, 2020. ISBN 8126502959
2. JosDirksen, "Expert Data Visualization", Packt Publishing, 2017.

**REFERENCES**

1. Chun-houh Chen, Wolfgang Härdle, Antony Unwin, "Handbook of Data Visualization", Springer- Verlag Berlin Heidelberg, Germany, 2008.
2. Andy Kirk, "Data Visualization: A Handbook for Data Driven Design", SAGE publishing, 2016.
3. Julie Steele and Noah Iliinsky, "Designing Data Visualizations: Representing Informational Relationships", O'REILLY, 2011.

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# ADVANCED DATA STRUCTURES AND ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – I**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1104 | **Credits** | : 04 |
| **Hours /Week** | : 05 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–2–0 | | |

**Course Learning Objectives:**

1. Introduce and apply advanced algorithms and programming techniques for building complex software systems.
2. Design and evaluate direct access methods, hash tables, and balanced tree structures.
3. Gain proficiency in algorithmic paradigms such as divide-and-conquer, backtracking, and dynamic programming.
4. Develop understanding of B-trees, disjoint set structures, and fundamental graph algorithms.
5. Acquire skills to solve problems efficiently while analyzing time and space complexity.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| | |
|---|---|
| **MODULE – I: REVIEW OF ALGORITHMS & DATA STRUCTURES** | **09 Hours** |
| Review of order notations and growth of functions, Analysis of Non-recursive & Recursive Algorithms, Average case analysis of algorithms, Basic data structures such as Stacks, Queues, Linked lists, and Applications. | |
| **MODULE – II: ADVANCED TREE OPERATIONS** | **09 Hours** |
| Direct access tables and hash tables, Hash functions and related analysis, Binary search trees and Operations, AVL Trees and balancing operations, R B Trees, properties, operations. | |
| **MODULE – III: GRAPH ALGORITHMS** | **09 Hours** |
| B-Trees: Definition, Properties, Operations, Data structures for disjoint sets, Graph algorithms, MST single source all pair shortest paths, BFS, DFS, Topological sorting, Strongly connected components. | |
| **MODULE – IV: SEARCHING & SORTING** | **09 Hours** |
| Quick sort randomized version, Searching in linear time, More graph algorithms – Maximal independent sets, Coloring vertex cover, Introduction to perfect graphs. | |
| **MODULE – V: ALGORITHMIC PARADIGMS** | **09 Hours** |
| Algorithmic paradigms: Greedy Strategy, Dynamic programming, Backtracking, Branch-and-Bound, Randomized algorithms. | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | To introduce and practice advanced algorithms and programming techniques necessary for developing sophisticated computer application programs. | L2 |
| 2 | To design and analyze direct access, hash tables, and balanced trees. | L3, L4 |
| 3 | To become familiar with various programming constructs, such as divide-and-conquer, backtracking, and dynamic programming. | L4 |
| 4 | To understand B-trees, disjoint sets, and core graph algorithms concepts. | L5 |
| 5 | To learn new techniques for solving specific problems more efficiently and for analyzing space and time requirements. | L4 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 3 | 2 | 2 | 3 | - | - | 1 | - | - | 3 | 2 | 3 | - |
| CO2 | 3 | 3 | 3 | 3 | 3 | - | - | 1 | - | - | 3 | 2 | 3 | - |
| CO3 | 3 | 3 | 2 | 3 | 3 | - | - | 1 | - | - | 3 | 2 | 3 | - |
| CO4 | 3 | 3 | 2 | 3 | 3 | - | - | 1 | - | - | 3 | 2 | 3 | - |
| CO5 | 3 | 3 | 3 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | - |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

## LAB EXPERIMENT

1. Implement and analyze basic data structures (stack, queue, and linked list) with real-time applications.
2. Solve recurrence relations and empirically verify time complexity using sample algorithms.
3. Implement hash tables using different collision resolution techniques and analyze their performance.
4. Implement Binary Search Trees, AVL Trees, and Red-Black Trees with insertion and deletion operations.
5. Implement B-Tree operations including insertion, deletion, and search.
6. Implement graph traversal algorithms (BFS and DFS) and detect strongly connected components.
7. Implement randomized quicksort and compare its performance with deterministic quicksort.
8. Implement linear-time selection algorithm to find the kth smallest element.
9. Implement Minimum Spanning Tree algorithms (Prim's and Kruskal's) using disjoint sets.
10. Implement shortest path algorithms (Dijkstra's and Floyd–Warshall) and analyze their efficiency.

**TEXT BOOKS**

1. H. S. Wilf, Algorithms and complexity, Prentice hall.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, Introduction to Algorithms, Prentice hall.

**REFERENCES**

1. Data Structures and Algorithms Made Easy – Narasimha Karumanchi
2. The Algorithm Design Manual by Steven S. Skiena, 3rd Edition, Springer, 2020, ISBN: 978-3-030-54255-9

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# IOT SENSORS AND APPLICATIONS

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – I**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1105 | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. To provide in-depth knowledge of physical principles applied in sensing and measurement.
2. To introduce the evolution of internet technology and the fundamental architecture of IoT.
3. To train students in building IoT systems using sensors, single-board computers, and open-source platforms.
4. To impart competence in the design and execution of mechanical and optical measurements.
5. To enable students to apply IoT data for business solutions across various domains such as healthcare and agriculture.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: SENSOR FUNDAMENTALS AND IOT ARCHITECTURE | 09 Hours |
|---|---|
| Introduction to Sensors and Transducers; Classification of Sensors (Active, Passive, Analog, Digital); Sensor Characteristics and Performance Parameters; Static and Dynamic Characteristics of Sensors; Sensor Error Analysis and Calibration; Accuracy, Precision, Sensitivity, and Resolution; Review of Computer Communication Concepts; Network Topologies and Packet Communication; OSI Reference Model and TCP/IP Stack; IPv4 and IPv6 Addressing; IoT Architecture Reference Layer; Perception, Network, Middleware, and Application Layers in IoT; Data Acquisition Systems; Introduction to Embedded Systems in IoT. | |
| | |
| **MODULE – II: PHYSICAL SENSING MECHANISMS** | **09 Hours** |
| Optical Sensors and Principles; Light Emitting Diodes (LED); Semiconductor Lasers; Photodiodes and Phototransistors; Fiber Optic Sensors and Applications; Infrared and Ultrasonic Sensors; Mechanical Sensors for Strain Measurement; Force and Torque Sensors; Pressure and Piezoelectric Sensors; Capacitive and Resistive Sensors; Signal Conditioning Circuits; Wheatstone Bridge Circuits; Amplifiers and Filters for Sensors; Noise Reduction Techniques in Sensor Circuits. | |
| | |
| **MODULE – III: IOT COMPONENTS AND CONNECTIVITY** | **09 Hours** |
| Characteristics of IoT Sensor Nodes; IoT Edge Computing Concepts; Cloud and Peripheral Computing; Single-Board Computers (Raspberry Pi, Arduino, ESP32); Microcontrollers and Interfacing Techniques; Open-Source IoT Hardware Platforms; Wireless Sensor Networks (WSN); Zigbee Communication Protocol; Wi-Fi Standards and IoT Connectivity; Bluetooth and BLE Communication; LoRa and LoRaWAN Technologies; 6LoWPAN Protocol; NFC and RFID Basics; Power Management in IoT Devices. | |
| | |
| **MODULE – IV: PROTOCOLS AND CLOUD INTEGRATION** | **09 Hours** |
| IoT Communication Protocols; MQTT Architecture and Publish-Subscribe Model; MQTT Brokers and QoS Levels; User Datagram Protocol (UDP); Hypertext Transfer Protocol (HTTP); Constrained Application Protocol (CoAP); XMPP and Gateway Protocols; RESTful APIs for IoT; Cloud Computing Fundamentals; Cloud Service Models (IaaS, PaaS, SaaS); Cloud Dashboards using ThingSpeak and Blynk; Data Logging and Visualization; Introduction to Big Data Analytics in IoT; Security Challenges in Cloud-Based IoT Systems. | |
| | |

| MODULE – V: IOT APPLICATIONS AND CASE STUDIES | 09 Hours |
|---|---|
| Smart City Architecture and Applications; Smart Energy Meters and Grid Monitoring; Home Automation Systems; Smart Lighting and Surveillance Systems; Healthcare IoT and Wearable Devices; ECG and EEG Monitoring Systems; Remote Patient Monitoring; Precision Agriculture and Smart Irrigation; Drone-Based Agricultural Monitoring; Industrial IoT Applications; Environmental Monitoring Systems; Intelligent Transportation Systems; Smart Waste Management; Case Studies in Real-Time IoT Deployments. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| At the end of the course the student will be able to: | | |
| 1 | Identify IoT networking components and build schematics for IoT solutions. | L2 |
| 2 | Choose appropriate sensors by comparing standards for measuring parameters like pressure, flow, and acceleration. | L3, L4 |
| 3 | Design and develop IoT-based sensor systems using microcontrollers and single-board computers. | L3, L4 |
| 4 | Evaluate performance characteristics of different sensors and their signal conditioning circuits. | L5 |
| 5 | Devise multidisciplinary models for IoT applications in smart cities, healthcare, and industrial environments. | L4, L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 2 | 2 | 3 | 1 | - | 1 | - | - | 2 | 2 | 3 | - |
| CO2 | 3 | 3 | 2 | 3 | 3 | 1 | - | - | - | - | 2 | 2 | 3 | - |
| CO3 | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | 1 | 3 | 3 | 3 | - |
| CO4 | 3 | 3 | 2 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 2 | 3 | - |
| CO5 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 2 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS**

1. Jacob Fraden, "Hand Book of Modern Sensors: Physics, Designs and Applications," Springer.
2. Alessandro Bassi et al., "Enabling Things to Talk – Designing IoT Solutions with the IoT Architecture Reference Model," Springer Open.

**REFERENCES**

1. Vijay Madisetti and Arshdeep Bahga, "Internet of Things: A Hands-on-Approach".
2. Jon S. Wilson, "Sensor Technology Hand Book," Elsevier.

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# RESEARCH METHODOLOGY AND IPR

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – I**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1106 | **Credits** | : 02 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 30 Hours |
| **L–T–P–J** | : 1–0–0–2 | | |

**Course Learning Objectives:**

1. To understand the fundamentals and types of research.
2. To develop skills in literature review and research planning.
3. To apply statistical techniques for data analysis.
4. To understand Intellectual Property Rights and their significance.
5. To gain knowledge of patents and research publication processes.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: RESEARCH DESIGN, ETHICS & PROBLEM FORMULATION** | **06 Hours** |
|---|---|

Research types (basic, applied, interdisciplinary), research process, problem identification and formulation, research gap analysis, hypothesis development, research ethics, plagiarism, research integrity, reproducibility in research.

| MODULE – II: ADVANCED LITERATURE REVIEW & RESEARCH PLANNING | 06 Hours |
|---|---|
| Systematic Literature Review (SLR), meta-analysis basics, research databases (Scopus, Web of Science), citation analysis, referencing tools (Mendeley/Zotero), research proposal writing, problem validation techniques. | |
| | |
| **MODULE – III: DATA ANALYTICS & STATISTICAL METHODS FOR RESEARCH** | **06 Hours** |
| Data collection methods, sampling techniques, data preprocessing, exploratory data analysis, regression models, ANOVA, multivariate analysis, hypothesis testing, tools (Python/SPSS/R) for statistical analysis. | |
| | |
| **MODULE – IV: INTELLECTUAL PROPERTY RIGHTS & TECHNOLOGY MANAGEMENT** | **06 Hours** |
| Introduction to IPR, types (copyright, trademark, patents, trade secrets), IPR in India, global IPR frameworks (WIPO, WTO), technology transfer, licensing, IPR in academia and startups, case studies in engineering and AI innovations. | |
| | |
| **MODULE – V: PATENTS, RESEARCH PUBLICATION & SCHOLARLY COMMUNICATION** | **06 Hours** |
| Patent concepts, patent search (Google Patents), patent drafting basics, filing procedure (India & PCT), research paper writing, journal indexing (SCI, Scopus, quartiles), impact factor, h-index, plagiarism tools, predatory journals, LaTeX/Overleaf basics. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| **1** | Understand research concepts, types, and problem formulation. | L2 |
| **2** | Perform literature review and develop research proposals. | L2, L3 |
| **3** | Apply statistical methods for data analysis. | L3 |
| **4** | Explain concepts of Intellectual Property Rights. | L2 |
| **5** | Understand patent filing and research publication process. | L2, L3 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 3 | 1 | 2 | 1 | - | 1 | - | 1 | - | 3 | 2 | 1 | 1 |
| **CO2** | 3 | 3 | 2 | 3 | 2 | - | 1 | 1 | 2 | 1 | 3 | 2 | 2 | 1 |
| **CO3** | 3 | 3 | 2 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 2 | - |
| **CO4** | 2 | 2 | 1 | 1 | 1 | 2 | 3 | - | 1 | - | 2 | 1 | 1 | 3 |
| **CO5** | 2 | 2 | 1 | 2 | 1 | 2 | 3 | 1 | 2 | 1 | 3 | 1 | 1 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS**

1. C.R. Kothari – Research Methodology: Methods and Techniques, New Age International.
2. Cooper, D.R. & Schindler, P.S. – Business Research Methods, McGraw-Hill.
3. Deborah E. Bouchoux – Intellectual Property: The Law of Trademarks, Copyrights, Patents, and Trade Secrets, Cengage Learning..

**REFERENCES**

1. Ranjit Kumar – Research Methodology: A Step-by-Step Guide for Beginners, Sage Publications.
2. Booth, W.C., Colomb, G.G., & Williams, J.M. – The Craft of Research, University of Chicago Press.
3. Montgomery, D.C. & Runger, G.C. – Applied Statistics and Probability for Engineers, Wiley.
4. Andy Field – Discovering Statistics Using SPSS, Sage Publications.
5. P. Narayanan – Intellectual Property Law in India, Eastern Law House.
6. WIPO – Understanding Intellectual Property, World Intellectual Property Organization.
7. Day, R.A. & Gastel, B. – How to Write and Publish a Scientific Paper, Cambridge University Press.
8. Springer – Guide to Publishing Research.
9. IEEE Author Center – Author Guidelines (Online Resource).

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# ADVANCED DEEP LEARNING

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – II**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1201 | **Credits** | : | 04 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–2–0 | | | |

**Course Learning Objectives:**

1. Develop a strong understanding of advanced deep learning architectures and optimization strategies for complex problem solving.
2. Build and deploy convolutional and sequential models tailored to real-world applications.
3. Investigate attention-based approaches and transformer architectures to improve model performance.
4. Design and evaluate generative models such as VAEs and GANs for learning data representations.
5. Utilize deep learning techniques in emerging domains including reinforcement learning, scalable deployment, and real-world systems.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: INTRODUCTION | 09 Hours |
|---|---|
| Introduction to Deep Learning: Overview of neural networks and deep learning, Historical perspective and key milestones in deep learning. Perceptron Learning, Multilayer Perceptrons (MLPs) and feedforward neural networks, Backpropagation, Initialization, Training & Validation, Parameter Estimation - MLE, MAP, Bayesian Estimation, Activation functions, loss functions, and optimization techniques, Regularization methods and hyperparameter tuning. Commonly used frameworks (e.g., TensorFlow, PyTorch). | |
| | |
| **MODULE – II: CNN, IMAGE CLASSIFICATION AND SEGMENTATION** | **09 Hours** |
| Convolutional Neural Networks (CNNs): Architecture of CNNs and convolutional layers, Pooling layers, batch normalization, and dropout, Applications of CNNs in Image classification, transfer learning, object detection, encoder-decoder models, Image segmentation: semantic, instance, and panoptic segmentation using U-Net and its variants, Regularization methods, Data augmentation techniques. | |
| | |
| **MODULE – III: RECURRENT NEURAL NETWORKS** | **09 Hours** |
| Introduction to RNNs: Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU), Advantages in handling sequential data, Challenges: Vanishing and exploding gradients Sequence modeling, text generation, and time series prediction, Applications: Sequence modeling, text generation, and time series prediction. | |
| | |
| **MODULE – IV: TRANSFORMERS** | **09 Hours** |
| Transformer models: Introduction of Attention Mechanism, Queries, Keys, and Values of Attention Network, Self-Attention and Positional Encodings, Attention-Based Sequence Encoder, Coupling the Sequence Encoder and Decoder, Cross Attention in the Sequence-to-Sequence Model, Multi-Head Attention, The Complete Transformer Network, BERT based models, Nemo, Self-supervision techniques, masked language modelling, autoregressive modelling. | |
| | |
| **MODULE – V: GENERATIVE NEURAL MODELS** | **09 Hours** |
| Fundamentals of GANs and adversarial training, Conditional GANs, StyleGAN, and CycleGAN, Applications of GANs in image generation, style transfer, and data augmentation; Advanced Topics in Deep Learning: Transfer learning and domain adaptation, Autoencoders, Variational Autoencoders (VAEs), and unsupervised learning, Reinforcement learning with deep neural networks, Transformers, LLMs. | |
| | |

## COURSE OUTCOMES:

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| At the end of the course the student will be able to: | | |
| 1 | To understand advanced deep learning architectures and optimization techniques for complex problem solving. | L2 |
| 2 | To design and implement convolutional and sequence-based models for real-world applications. | L3, L4 |
| 3 | To explore attention mechanisms and transformer models for improved learning performance. | L3 |
| 4 | To analyze and develop generative models such as VAEs and GANs for data representation. | L4 |
| 5 | To apply deep learning techniques in emerging areas, including reinforcement learning, model deployment, and scalable systems. | L3, L4 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 3 | 2 | 3 | 3 | - | - | - | - | - | 3 | 3 | 2 | - |
| CO2 | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 3 | 3 | - |
| CO3 | 3 | 3 | 2 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | 1 |
| CO4 | 3 | 3 | 3 | 3 | 3 | - | 1 | 1 | 1 | - | 3 | 3 | 3 | 1 |
| CO5 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 2 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

## LAB EXPERIMENT

1. MNIST Digit Classification: Train a simple feedforward neural network on the MNIST dataset using forward and backpropagation.
2. Image Classification with CNN: Develop and evaluate a CNN model on datasets like CIFAR-10 or Fashion-MNIST.
3. Transfer Learning with Pre-trained CNN: Fine-tune models like VGG16, ResNet, or MobileNet on a custom image dataset.
4. Object Detection using YOLO: Implement YOLOv5 or YOLOv8 for real-time object detection on images or video frames.
5. Semantic Segmentation with U-Net: Use the U-Net architecture for medical or satellite image segmentation.

6. Instance Segmentation with Mask R-CNN: Apply Mask R-CNN for pixel-wise object segmentation on COCO or similar datasets.
7. Image Classification using ViT and Swin Transformer: Train Vision Transformer (ViT) and Swin Transformer models, and compare their performance with CNNs on image classification tasks.
8. Diffusion Model for Image Generation: Use a pretrained or simplified diffusion model (e.g., Stable Diffusion) to generate high-quality synthetic images.
9. Autoencoder and Variational Autoencoder (VAE): Build autoencoders for image denoising and extend to VAEs for generative modeling.
10. Generative Adversarial Networks (GANs): Implement a basic GAN or DCGAN to generate synthetic image data.
11. Time Series Forecasting with LSTM and SARIMA: Compare LSTM-based forecasting with classical SARIMA for time series prediction tasks.
12. Sentiment Analysis using LSTM and BERT: Perform sentiment classification using LSTM and fine-tuned BERT on datasets like IMDB or Twitter.

**TEXT BOOKS:**

1. Christopher M. Bishop & Hugh Bishop “Deep Learning: Foundations and Concepts”, Springer, 2023.
2. Ian Goodfellow and Yoshua Bengio and Aaron Courville “Deep Learning”, MIT Press, 2016.

**REFERENCES:**

1. Stephen Marsland “Machine Learning: An Algorithmic Perspective, Second Edition”, Chapman and Hall/CRC.
2. Stanley H. Chan “Introduction to Probability For Data Science”, Michigan Publishing.

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# IMAGE PROCESSING AND COMPUTER VISION

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – II**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1202 | **Credits** | : 03 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–0–0 | | |

**Course Learning Objectives:**

1. Study the basic image processing concepts and digital image formation fundamentals.
2. Understand image enhancement, restoration, and transformation techniques in both spatial and frequency domains.
3. Learn deep learning models and architectures specifically designed for computer vision tasks.
4. Visualize and interpret CNN architectures to understand the underlying mechanisms of feature extraction.
5. Enable students to apply computer vision algorithms to solve real-world problems in object detection, segmentation, and image generation.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: DIGITAL IMAGE FUNDAMENTALS & TRANSFORMS** | **09 Hours** |
|---|---|

Computer vision and its applications; Fundamentals of image formation; Digital image low-level processing; Image Transformation (Orthogonal, Euclidean, Affine); Fourier Transform and its properties; Fast Fourier Transform; Gray level transformation; Histogram processing.

| MODULE – II: FEATURE EXTRACTION AND LOW-LEVEL VISION | 09 Hours |
|---|---|
| Spatial Filters; Colour Image Processing; Image Segmentation; Texture Descriptors; Colour Features; Edges/Boundaries; Interest or Corner Point Detectors; Histogram of Oriented Gradients (HoG); SIFT and SURF; Scale Space and Scale Selection. | |
| | |
| **MODULE – III: PATTERN ANALYSIS AND CLASSICAL MACHINE LEARNING** | **09 Hours** |
| Basics of Probability and Statistics; Clustering (K-Means, K-Medoids, Mixture of Gaussians); Classification (Discriminant Functions, Supervised/Un-supervised); Classifiers (Bayes, Artificial Neural Networks); Dimensionality Reduction (Principal Component Analysis). | |
| | |
| **MODULE – IV: DEEP LEARNING ARCHITECTURES FOR VISION** | **09 Hours** |
| Multi-layer Perceptrons; Backpropagation; Evolution of CNN Architectures (AlexNet, VGG, Inception Nets, ResNets, DenseNets); Visualization and Understanding CNNs (Kernels, Backprop-to-image, Deconvolution methods); CAM and Grad-CAM++. | |
| | |
| **MODULE – V: ADVANCED RECOGNITION AND GENERATIVE MODELS** | **09 Hours** |
| CNNs for Recognition and Verification (Siamese Networks, Triplet Loss); Object Detection (R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD); Semantic Segmentation (FCN, SegNet, U-Net, Mask-RCNN); Deep Generative Models (GANs, VAEs, Pixel RNNs); Applications (Image Editing, Super-resolution, CycleGANs). | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Understand and explain the fundamental concepts of a digital image processing system. | L2 |
| 2 | Implement and evaluate various techniques and algorithms used in computer vision, such as SIFT and HoG. | L3, L5 |
| 3 | Critically analyze the building and integration of computer vision algorithms for complex systems. | L3 |
| 4 | Design, implement, and train deep learning models (CNNs, GANs) to address real-world vision problems effectively. | L3, L4 |
| 5 | Demonstrate awareness of current key research issues and emerging trends in the computer vision domain. | L3, L4 |

**CO-PO MAPPING**

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | - | 2 | 2 | - | - | - | - | - | 2 | 2 | 2 | - |
| **CO2** | 3 | 3 | 2 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | - |
| **CO3** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | 1 | 3 | 3 | 3 | 1 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | 1 | 3 | 3 | 3 | 1 |
| **CO5** | 2 | 3 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | - | 3 | 2 | 2 | 2 |

**3 – High Correlation 2 – Moderate Correlation 1 – Low Correlation**

**TEXT BOOKS**

1. Digital Image Processing by R. Gonzalez and R. E. Wood.
2. Deep Learning by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.
3. Computer Vision Algorithms and Applications by Richard Szeliski.

**REFERENCES**

1. Computer Vision: A Modern Approach by David Forsyth and Jean Ponce.
2. Neural Networks and Deep Learning by Michael Nielsen.
3. Digital Image Processing by W.K. Pratt.

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# NLP AND GENERATIVE AI

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – II**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1203 | **Credits** | : 04 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–2–0 | | |

**Course Learning Objectives:**

1. Understand fundamental concepts of Applied NLP, including preprocessing, feature extraction, and text representation
2. Apply machine learning techniques to build NLP-based applications such as classification and similarity systems
3. Understand the transition from traditional NLP to Generative AI and Large Language Models (LLMs)
4. Use transformer-based models and prompt engineering techniques for real-world applications
5. Critically evaluate AI-generated outputs with respect to bias, hallucination, and ethical considerations.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: FOUNDATIONS OF NATURAL LANGUAGE PROCESSING | 09 Hours |
|---|---|
| Introduction: What is Natural Language Processing? Origins of NLP, Language and Knowledge, The Challenges of NLP, Language and Grammar, Processing Indian Languages, NLP Applications.<br>Language Modeling: Statistical Language Model - N-gram model (unigram, bigram), Paninion Framework, Karaka theory. | |
| | |
| **MODULE – II: WORD-LEVEL ANALYSIS** | **09 Hours** |
| Word Level Analysis: Regular Expressions, Finite-State Automata, Morphological Parsing, Spelling Error Detection and Correction, Words and Word Classes, Part-of Speech Tagging. Syntactic Analysis: Context-Free Grammar, Constituency, Top-down and Bottom-up Parsing, CYK Parsing. | |
| | |
| **MODULE – III: NAÏVE BAYES AND MACHINE TRANSLATION** | **09 Hours** |
| Naive Bayes, Text Classification and Sentiment: Naive Bayes Classifiers, Training the Naive Bayes Classifier, Worked Example, Optimizing for Sentiment Analysis, Naive Bayes for Other Text Classification Tasks, Naive Bayes as a Language Model.<br>Machine Translation: Language Divergences and Typology, Machine Translation using Encoder Decoder, Details of the Encoder-Decoder Model, Translating in Low-Resource Situations, MT Evaluation. | |
| | |
| **MODULE – IV: INFORMATION RETRIEVAL** | **09 Hours** |
| Information Retrieval: Design Features of Information Retrieval Systems, Information Retrieval Models - Classical, Non-classical, Alternative Models of Information Retrieval - Custer model, Fuzzy model, LSTM model, Major Issues in Information Retrieval. Lexical Resources: WordNet, FrameNet, Stemmers, Parts-of-Speech Tagger, Research Corpora. | |
| | |
| **MODULE – V: INTRODUCTION TO GAN** | **09 Hours** |
| Introduction to Generative Adversarial Networks Introduction to GAN: Introduction to Generative Adversarial Networks (GANs), Working principles and training Autoencoders: Introduction to autoencoder, High-level functionality, distinction between autoencoder and GAN, Content generation using autoencoder, Variational autoencoders | |
| | |

## COURSE OUTCOMES:

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Apply the fundamental concept of NLP, grammar-based language model and statistical-based language model. | L2 |
| 2 | Model morphological analysis using Finite State Transducers and parsing using context-free grammar and different parsing approaches. | L3 |
| 3 | Develop the Naïve Bayes classifier and sentiment analysis for Natural language problems and text classifications and Identify the Machine Translation applications of NLP using Encode and Decoder. | L3, L4 |
| 4 | Use the concepts of information retrieval, lexical semantics, lexical dictionaries such as WordNet, lexical computational semantics, distributional word similarity. | L3, L4 |
| 5 | Implement the concepts and principles of Generative Artificial Intelligence to engineering\| requirements. | L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | - | 2 | 2 | - | - | - | - | - | 2 | 3 | 2 | - |
| **CO2** | 3 | 3 | 2 | 3 | 3 | - | - | 1 | - | - | 2 | 3 | 3 | - |
| **CO3** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 3 | 3 | 1 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 2 |
| **CO5** | 2 | 3 | 1 | 2 | 2 | 3 | 3 | 1 | 1 | - | 3 | 2 | 2 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

## LAB EXPERIMENT

1. Develop unigram and bigram language models for next-word prediction using a given text corpus.
2. Analyze linguistic challenges in processing Indian languages using the Paninian framework and Karaka theory.
3. Design a regular expression–based system for extracting structured information from unstructured text.
4. Implement POS tagging and CYK parsing for syntactic analysis of natural language sentences.
5. Build a sentiment analysis system using the Naive Bayes Classifier for text classification.

6. Implement a sequence-to-sequence encoder–decoder model for basic machine translation tasks.
7. Develop an information retrieval system using TF-IDF and cosine similarity techniques.
8. Explore semantic relationships in text using WordNet and FrameNet.
9. Implement Generative Adversarial Networks for synthetic image generation.
10. Develop autoencoder and variational autoencoder models for feature extraction and content generation.

**TEXT BOOKS**

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press
2. Daniel Jurafsky, James H. Martin, "Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", Pearson Education, 2023.
3. Jakub Langr, Vladimir Bok. GANs in Action, Manning, 2019

**REFERENCES**

1. Larry L. Peterson and Bruce S. Davie: Computer Networks - A Systems Approach, 4th Edition, Elsevier, 2019.
2. Nader F. Mir: Computer and Communication Networks, 2nd Edition, Pearson Education, 2015.
3. William Stallings, Data and Computer Communication 10th Edition, Pearson Education, Inc., 2014.

## SCHOOL OF ENGINEERING

(A State Private University under the Karnataka Act No. 20 of 2013) Approved by UGC & AICTE, New Delhi, Accredited by NAAC with A+ grade

## Department of CSE (Artificial Intelligence and Data Science)

# BIG DATA ENGINEERING AND ANALYTICS

[As per Choice Based Credit System (CBCS) scheme]

**SEMESTER – II**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1204 | **Credits** | : 04 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–2–0 | | |

**Course Learning Objectives:**

1. Explore the fundamental concepts of Big Data platforms and the challenges of conventional systems.
2. Impart the architectural concepts of Hadoop and the MapReduce programming paradigm.
3. Master the use of distributed processing tools like PIG and HIVE within the Hadoop ecosystem.
4. Implement high-performance data analysis using advanced frameworks like Apache Spark and Storm.
5. Apply predictive modeling and feature engineering techniques to extract actionable insights from massive datasets.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: BIG DATA FUNDAMENTALS AND DISTRIBUTED STORAGE** | **09 Hours** |
|---|---|
| Introduction to Big Data Platform; Challenges of Conventional Systems; Nature of Data; Big Data Characteristics (V's); Big Data Architecture; Distributed File Systems; Design of HDFS; Sharding and Replication; ACID and BASE properties; Introduction to NoSQL and Columnar Databases (HBase, Cassandra). | |
| | |
| **MODULE – II: DISTRIBUTED PROCESSING FRAMEWORKS** | **09 Hours** |
| Hadoop Architecture and YARN; MapReduce Paradigm; Anatomy of a MapReduce Job Run; Failures and Job Scheduling; Shuffle and Sort; Applications using Pig: Pig Latin, Executive types, and Grunt shell; Hive Services: HiveQL, Querying data, and file formats. | |
| | |
| **MODULE – III: ADVANCED ENGINEERING FRAMEWORKS** | **09 Hours** |
| Overview of Apache Spark vs. Hadoop; Spark Stack: Resilient Distributed Datasets (RDD), Spark Context, and RDD Operations; Spark SQL and DataFrames; GraphX overview and Graph algorithms; Introduction to Apache Storm and Apache Oozie for workflow management. | |
| | |
| **MODULE – IV: PREDICTIVE ANALYTICS AND MACHINE LEARNING** | **09 Hours** |
| Predictive Data Analytics Lifecycle; Data Exploring and Preprocessing: Identifying Quality Issues, Normalization, and Outlier detection; Feature Selection: PCA, Chi-Square, and ANOVA F-test; Supervised Learning: Decision Trees (ID3 Algorithm), Similarity-based learning (KNN), and Linear/Multiple Regression. | |
| | |
| **MODULE – V: ADVANCED ANALYTICS AND DOMAIN APPLICATIONS** | **09 Hours** |
| Distributed Machine Learning: Data and Model Parallelism, Parameter Servers; Federated Learning; Healthcare Analytics: EHR, Phenotyping, and Clinical Prediction Models; Social Network Analytics: Centrality, Community Detection, and Sentiment Analysis. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| At the end of the course the student will be able to: | | |
| 1 | Identify Big Data characteristics and design distributed storage solutions using HDFS. | L4 |
| 2 | Develop efficient MapReduce applications and data manipulation scripts using Pig and Hive. | L3 |
| 3 | Synthesize Spark-based architectures for real-time processing and graph algorithms. | L3, L4 |
| 4 | Perform data preprocessing and construct predictive models for large-scale data analysis. | L3, L4 |
| 5 | Analyze domain-specific data (Healthcare/Social Networks) to build complete business solutions. | L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 3 | 2 | 2 | 3 | - | - | 1 | - | - | 2 | 3 | 3 | - |
| CO2 | 3 | 3 | 3 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | - |
| CO3 | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | 1 | 3 | 3 | 3 | - |
| CO4 | 3 | 3 | 3 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | - |
| CO5 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 2 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

## LAB EXPERIMENT

1. Hadoop Environment: How do you set up and install Hadoop in Pseudo-distributed and Fully Distributed modes?
2. HDFS Operations: How can HDFS commands be used to perform file storage, replication, and distributed data management operations?
3. MapReduce Mining: Write a MapReduce program to mine weather data and compute average, maximum, and minimum temperatures.
4. Hadoop Ecosystem: How can Pig Latin scripts be used for TF-IDF value calculation and Hive for database/table management?

5. Hive Query Processing: How do you create databases, tables, partitions, and perform aggregate queries using HiveQL?
6. Spark Operations: Deploy an Apache Spark Cluster and manipulate RDDs for matrix algorithms and distributed computations.
7. Spark SQL Analytics: How can Spark SQL and DataFrames be used to process and analyze structured big data?
8. Predictive Modelling: Implement simple and multivariate linear regression to predict future values and analyze goodness of fit.
9. Clustering & Classification: Implement K-Means clustering and Decision Tree classification on healthcare or financial datasets.
10. Social Network & Healthcare Analytics: Perform sentiment analysis, community detection, and healthcare prediction using distributed analytics frameworks.

**TEXT BOOKS**

1. Tom White, "Hadoop: The Definitive Guide", O'Reilly Media.
2. Thomas Erl, "Big Data Fundamentals: Concepts, Drivers & Techniques", Pearson.
3. John D. Kelleher, "Fundamentals of Machine Learning for Predictive Data Analytics", MIT Press.

**REFERENCES**

4. Arshdeep Bahga, Vijay Madisetti, "Big Data Science & Analytics: A Hands-On Approach", VPT.
5. Chandan K. Reddy, "Healthcare Data Analytics", CRC Press.
6. Anand Rajaraman, "Mining of Massive Datasets", CUP.

**Signature**



**Chairperson CSE (AI & DS)**

# AGENTIC AI

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE – I**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1207 | **Credits** | : | 03 |
| **Hours/Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. Understand the theoretical foundations, evolution, and taxonomy of intelligent agent-based and agentic AI systems.
2. Analyze agent architectures including reactive, deliberative, hybrid, and BDI models, and design multi-agent coordination strategies.
3. Apply planning and reasoning paradigms including ReAct and chain-of-thought, and utilize agent design patterns to build structured agentic workflows.
4. Implement memory models, retrieval-augmented generation, and context-aware mechanisms for adaptive intelligent agent systems.
5. Evaluate agent performance, apply responsible AI principles including bias, fairness, and alignment, and analyze governance frameworks for agentic AI deployment.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: FOUNDATIONS OF AGENTIC AI AND INTELLIGENT SYSTEMS | 09 Hours |
|---|---|
| Topics: Introduction to Agentic AI; evolution from traditional AI to intelligent agent-based and autonomous systems; fundamentals of intelligent agents; characteristics of agents such as autonomy, rationality, reactivity, and proactiveness; agent–environment interaction; perception–action cycle; classification of environments including deterministic, stochastic, dynamic, and partially observable environments; introduction to goal-based and utility-based agents; human-in-the-loop systems; human-AI collaboration frameworks; and overview of real-world applications of agentic AI systems. | |

| MODULE – II: AGENT ARCHITECTURES AND MULTI-AGENT SYSTEM DESIGN | 09 Hours |
|---|---|
| Topics: Agent structure and core components including perception, reasoning, decision-making, action, and learning; reactive agents and simple reflex systems; deliberative agents and goal-based reasoning; hybrid agent architectures combining reactive and deliberative approaches; introduction to BDI (Belief–Desire–Intention) model; fundamentals of multi-agent systems; communication and coordination among agents; basic negotiation mechanisms; task decomposition and hierarchical goal-based system design; introduction to tool integration and API-based agent systems (conceptual level); and overview of security, privacy, safety, and ethical considerations in agent system design. | |

| MODULE – III: PLANNING, REASONING AND AGENT DESIGN PATTERNS | 10 Hours |
|---|---|
| Topics: Fundamentals of planning in intelligent agents; goal-based planning and introduction to utility-based decision-making; reasoning and acting using the ReAct paradigm; structured reasoning approaches including chain-of-thought reasoning (conceptual); introduction to tree-based and graph-based reasoning methods; tool-augmented agents and external function calling concepts; reflection and self-correction mechanisms in agents; workflow-based agent models including sequential, parallel, conditional, and graph-based workflows; introduction to state management in agent systems; multi-step reasoning pipelines and orchestration concepts; prompting strategies for controlling agent behavior; and standard design patterns and best practices for building agentic systems. | |

| MODULE – IV: KNOWLEDGE REPRESENTATION AND ADAPTIVE AGENT SYSTEMS | 09 Hours |
|---|---|
| Topics: Memory models in intelligent agents including short-term memory and long-term memory; concepts of episodic and semantic memory; context management and context window handling in agent systems; external knowledge integration using databases, APIs, and knowledge bases; introduction to retrieval-augmented generation (RAG) in agent systems; adaptive decision-making under changing environments; personalization and context-aware intelligent systems; and scalability challenges in agent systems including memory limitations, latency issues, and distributed coordination. | |

| MODULE – V: DEPLOYMENT, EVALUATION AND RESPONSIBLE AI IN AGENT SYSTEMS | 08 Hours |
|---|---|
| Topics: Deployment architectures for AI agents including cloud-based, edge-based, and hybrid systems; agent lifecycle management from development to deployment and updates; monitoring and evaluation of agent performance using qualitative and quantitative metrics; reliability and robustness considerations in real-world systems; observability concepts such as logging, tracing, debugging, and system diagnostics (theoretical overview); ethical AI considerations including bias, fairness, safety, and alignment; governance and regulatory aspects of AI systems; and future trends in agentic AI including autonomous ecosystems, large-scale multi-agent collaboration, and industry-scale applications. | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| At the end of the course the student will be able to: | | |
| CO1 | Explain the conceptual framework, taxonomy, and evolution of agentic AI systems. | L1, L2 |
| CO2 | Design and analyze agent architectures and apply coordination mechanisms, communication strategies, and task decomposition. | L3, L4 |
| CO3 | Apply planning and reasoning paradigms and utilize design patterns, and prompting strategies to build structured agentic systems. | L3, L5 |
| CO4 | Implement and evaluate memory models, context management strategies, RAG, and adaptive decision-making mechanisms. | L3, L4 |
| CO5 | Assess agent deployment architectures and lifecycle management strategies, apply responsible AI principles. | L4, L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO 1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 1 | 2 | 1 | 1 | | | 1 | 1 | 2 | 3 | 2 | 2 |
| **CO2** | 2 | 3 | 2 | 3 | 2 | 1 | | | 2 | 1 | 2 | 3 | 3 | 2 |
| **CO3** | 2 | 3 | 3 | 2 | 2 | 1 | | | 2 | 1 | 2 | 3 | 3 | 2 |
| **CO4** | 3 | 3 | 2 | 3 | 3 | 1 | | | 1 | 1 | 2 | 3 | 2 | 3 |
| **CO5** | 3 | 3 | 2 | 3 | 2 | 2 | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS:**

1. T1: Russell, S., & Norvig, P., Artificial Intelligence: A Modern Approach, Pearson, 4th Ed., 2020.
2. T2: Wooldridge, M., An Introduction to MultiAgent Systems, Wiley, 2nd Ed., 2009.
3. T3: Wei, J., et al., Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, NeurIPS, 2022.

**REFERENCE BOOKS:**

1. R1: Yao, S., et al., ReAct: Synergizing Reasoning and Acting in Language Models, ICLR, 2023.
2. R2: Lewis, P., et al., Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, NeurIPS, 2020.
3. R3: Wang, L., et al., A Survey on Large Language Model based Autonomous Agents, Frontiers of Computer Science, 2024.

**MOOCS/NPTEL/SWAYAM:**

1. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-39+V1

# PATTEREN RECOGNITION

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE – I**

| | | | | |
|---|---|---|---|---|
| **Course Code** | 26MAD1208 | **Credits** | : | 03 |
| **Hours/Week** | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. To introduce the theoretical foundations and architectures of pattern recognition systems.
2. To develop statistical and probabilistic approaches for pattern classification and decision making.
3. To enable learners to apply feature extraction and dimensionality reduction techniques for complex datasets.
4. To equip students with machine learning algorithms and evaluation techniques for pattern analysis.
5. To train students in advanced deep learning approaches and recent trends in pattern recognition applications.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: FOUNDATIONS OF PATTERN RECOGNITION** | **09 Hours** |
|---|---|

Introduction to Pattern Recognition, Pattern Recognition Systems and Architectures, Pattern Representation, Feature Space and Decision Boundaries, Types of Learning: Supervised, Unsupervised, Semi-Supervised and Reinforcement Learning, Bias–Variance Tradeoff, Applications in Biometrics, Speech and Medical Diagnosis, Challenges in Pattern Recognition: Noise, Variability, Overfitting and Underfitting.

| MODULE – II: STATISTICAL PATTERN RECOGNITION | 09 Hours |
|---|---|
| Probability Theory and Bayes Theorem, Probability Density Functions and Likelihood Estimation, Bayesian Decision Theory, Minimum Error Rate Classification, Maximum Likelihood Estimation (MLE), Maximum A Posteriori (MAP), Gaussian Mixture Models, Multivariate Analysis, Hidden Markov Models (HMM), Expectation Maximization (EM) Algorithm, Discriminant Functions and Covariance Estimation. | |
| | |
| **MODULE – III: FEATURE EXTRACTION AND DIMENSIONALITY REDUCTION** | **09 Hours** |
| Feature Selection and Feature Extraction Techniques, Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Independent Component Analysis (ICA), Kernel PCA, Feature Normalization and Transformation, Curse of Dimensionality, Manifold Learning, t-SNE and UMAP techniques. | |
| | |
| **MODULE – IV: MACHINE LEARNING TECHNIQUES FOR PATTERN RECOGNITION:** | **09 Hours** |
| K-Nearest Neighbour (KNN), Support Vector Machines (SVM), Decision Trees, Random Forests, Ensemble Learning and Gradient Boosting, Clustering Techniques: K-Means and Hierarchical Clustering, Hyperparameter Optimization, Cross Validation Methods, Bias–Variance Analysis, Performance Evaluation Metrics: Confusion Matrix, ROC Curve, Accuracy, Precision, Recall and F1-Score. | |
| | |
| **MODULE – V: ADVANCED PATTERN RECOGNITION AND DEEP LEARNING:** | **09 Hours** |
| Neural Network Fundamentals, Convolutional Neural Networks (CNN), CNN Architectures: LeNet, AlexNet and ResNet, Recurrent Neural Networks (RNN), LSTM and GRU Networks, Transfer Learning, Autoencoders, Generative Adversarial Networks (GANs), Explainable AI (XAI), Pattern Recognition in Time-Series Data, Attention Mechanism and Introduction to Transformers, Research Trends in Deep Learning. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| At the end of the course the student will be able to: | | |
| 1 | Explain the fundamental concepts, architectures, and learning paradigms used in pattern recognition systems. | L2 |
| 2 | Apply statistical and probabilistic approaches for pattern classification and decision making. | L3 |
| 3 | Analyze feature extraction and dimensionality reduction techniques for high-dimensional datasets. | L4 |
| 4 | Implement machine learning algorithms and evaluation techniques for classification and clustering problems. | L5 |
| 5 | Evaluate advanced deep learning models and recent pattern recognition techniques for real-world engineering applications | L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 1 | 1 | 2 | | | | 1 | | 1 | 3 | 2 | |
| CO2 | 3 | 3 | 2 | 2 | 3 | | | 1 | 1 | | 1 | 3 | 3 | |
| CO3 | 2 | 3 | 3 | 3 | 3 | | | 1 | 1 | | 2 | 3 | 3 | 1 |
| CO4 | 3 | 3 | 3 | 3 | 3 | 1 | | 1 | 1 | 1 | 2 | 3 | 3 | 1 |
| CO5 | 2 | 3 | 3 | 3 | 3 | 2 | 2 | 1 | 2 | 1 | 3 | 3 | 3 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS:**

1. Richard O. Duda, Peter E. Hart and David G. Stork, *Pattern Classification*, Wiley India, 2nd Edition, 2012.
2. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

**REFERENCE BOOKS:**

1. Sergios Theodoridis and Konstantinos Koutroumbas, *Introduction to Pattern Recognition*, Academic Press, 2nd Edition, 2003.
2. Ian Goodfellow, Yoshua Bengio and Aaron Courville, *Deep Learning*, MIT Press, 2016.
3. Tom M. Mitchell, *Machine Learning*, McGraw-Hill Education, 1997.
4. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow*, O'Reilly Media, 3rd Edition, 2022.

# QUANTUM MACHINE LEARNING

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE – I**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1209 | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. To introduce the foundations of quantum computing and quantum information processing for machine learning applications.
2. To develop an understanding of quantum machine learning concepts and quantum-enhanced learning models.
3. To enable learners to apply quantum supervised learning and optimization techniques for intelligent systems.
4. To equip students with advanced quantum machine learning algorithms and hybrid quantum-classical learning approaches.
5. To train students in the use of quantum machine learning tools, frameworks and emerging research trends.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: FOUNDATIONS OF QUANTUM COMPUTING FOR MACHINE LEARNING | 09 Hours |
|---|---|
| Introduction to Quantum Computing and Quantum Information Processing, Qubits, Quantum States and Measurement, Superposition, Entanglement and Quantum Interference, Quantum Gates and Quantum Circuits, Hilbert Space Representation and Basic Linear Algebra for Quantum Systems, Tensor Products and Bloch Sphere Representation, Encoding Classical Data into Quantum States and Quantum Data Representation. | |
| | |
| **MODULE – II: QUANTUM COMPUTING CONCEPTS FOR MACHINE LEARNING** | **09 Hours** |
| Review of Classical Machine Learning Concepts, Limitations of Classical Machine Learning for High-Dimensional Problems, Advantages of Quantum Machine Learning, Quantum Feature Spaces, Quantum Kernel Methods, Quantum Data Encoding Techniques, Variational Quantum Algorithms, Quantum Optimization Concepts, Introduction to Quantum Complexity and Computational Advantage. | |
| | |
| **MODULE – III: QUANTUM SUPERVISED LEARNING MODELS** | **09 Hours** |
| Quantum Support Vector Machines (QSVM), Quantum Kernel Estimation, Variational Quantum Classifiers (VQC), Quantum Regression Models, Quantum Circuit Training Methods, Cost Functions and Optimization in Quantum Learning, Gradient-Based Optimization Techniques, Introduction to Barren Plateau Problem, Performance Evaluation of Quantum Learning Models. | |
| | |
| **MODULE – IV: ADVANCED QUANTUM MACHINE LEARNING ALGORITHMS** | **09 Hours** |
| Quantum Neural Networks (QNN), Parameterized Quantum Circuits (PQC), Quantum Principal Component Analysis (QPCA), Quantum k-Nearest Neighbours (QkNN), Hybrid Quantum-Classical Learning Models, Quantum Generative Models, Introduction to Quantum Reinforcement Learning, Quantum Optimization Algorithms, Deep Learning Acceleration using Quantum Computing. | |
| | |
| **MODULE – V: QUANTUM ML TOOLS, APPLICATIONS AND FUTURE SCOPE** | **09 Hours** |
| Applications of Quantum Machine Learning in Healthcare, Finance, Drug Discovery and Natural Language Processing, Noise and Decoherence in Quantum Systems, Quantum Error Mitigation Concepts, Quantum Software Frameworks: Qiskit, PennyLane and TensorFlow Quantum, Cloud-based Quantum Computing Platforms, Explainable Quantum AI Concepts, Challenges and Future Scope of Quantum Machine Learning, Current Research Trends in Quantum AI Systems. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| | **At the end of the course the student will be able to:** | |
| 1 | Explain the foundations of quantum computing and quantum information processing used in machine learning systems | L2 |
| 2 | Apply quantum machine learning concepts and quantum-enhanced optimization techniques for intelligent systems. | L3 |
| 3 | Analyze quantum supervised learning models and quantum kernel methods for classification and prediction problems. | L4 |
| 4 | Implement advanced quantum machine learning algorithms and hybrid quantum-classical learning models. | L5 |
| 5 | Evaluate quantum machine learning tools, applications and emerging research trends for real-world engineering systems | L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 1 | 1 | 2 | | | | 1 | | 1 | 3 | 2 | |
| CO2 | 3 | 3 | 2 | 2 | 3 | | | 1 | 1 | | 1 | 3 | 3 | |
| CO3 | 2 | 3 | 3 | 3 | 3 | | | 1 | 1 | | 2 | 3 | 3 | 1 |
| CO4 | 3 | 3 | 3 | 3 | 3 | 1 | | 1 | 1 | 1 | 2 | 3 | 3 | 1 |
| CO5 | 2 | 3 | 3 | 3 | 3 | 2 | 2 | 1 | 2 | 1 | 3 | 3 | 3 | 3 |

**3 – High Correlation 2 – Moderate Correlation 1 – Low Correlation**

**TEXT BOOKS:**

1. Michael A. Nielsen and Isaac L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, 10th Anniversary Edition, 2010.
2. Maria Schuld and Francesco Petruccione, *Machine Learning with Quantum Computers*, Springer International Publishing, 2nd Edition, 2021.

**REFERENCE BOOKS:**

1. Peter Wittek, *Quantum Machine Learning: What Quantum Computing Means to Data Mining*, Academic Press, 2014.
2. Maria Schuld and Francesco Petruccione, *Supervised Learning with Quantum Computers*, Springer, 2018.
3. Santanu Pattanayak, *Hands-On Quantum Machine Learning with Python*, Packt Publishing, 2021.

# HEALTHCARE ANALYTICS AND MEDICAL IMAGING TECHNIQUES

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE-I**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1210 | **Credits** | : 03 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–0–0 | | |

**Course Learning Objectives:**

1. Acquire knowledge of the core concepts of healthcare data analytics and Electronic Health Records (EHR).
2. Understand the integration of IoT and pervasive health analysis in clinical environments.
3. Discuss the fundamentals of digital images and spatial/frequency domain filtering for enhancement.
4. Identify advanced reconstruction and segmentation techniques for CT and MRI modalities.
5. Apply AI and machine learning tools to synthesize diagnostic insights from medical images and health data.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: HEALTHCARE DATA SOURCES AND BASIC ANALYTICS** | **09 Hours** |
|---|---|

Introduction to Healthcare Data Analytics; Electronic Health Records (EHR)—Components, Coding Systems, and Benefits; Barriers to Adopting EHR; Phenotyping Algorithms; Challenges in Data Acquisition and Pre-processing; Social Media Analytics for Healthcare.

| MODULE – II: ADVANCED HEALTHCARE SYSTEMS AND IOMT | 09 Hours |
|---|---|
| Clinical Trials and Survival Models; Predictive Models for integrating Clinical and Genomic Data; Data Analytics for Pervasive Health; Internet of Medical Things (IoMT)—Architecture, Devices, and Benefits; Tele-Health and Patient Health Monitoring Systems (PHMS); Cybersecurity vulnerabilities in health informatics. | |
| | |
| **MODULE – III: MEDICAL IMAGING FUNDAMENTALS AND ENHANCEMENT** | **09 Hours** |
| Digitizing Images and Medical Image Formats; Image Quality and Information Content—Histogram and Entropy; Digital Subtraction Angiography; Image Averaging; Contrast Enhancement; Spatial and Frequency domain filtering; Applications to X-ray and Ultrasound images | |
| | |
| **MODULE – IV: COMPUTED TOMOGRAPHY AND IMAGE RESTORATION** | **09 Hours** |
| Principles of Tomography; First to Fifth generation scanners; Spiral CT; 3D Imaging; Methods of Reconstruction—Back projection and Iterative methods; Modeling Image Degradation—Inverse filtering and Wiener filtering; Image Segmentation—Pixel, Edge, and Region-based. | |
| | |
| **MODULE – V: MRI AND ADVANCED IMAGING MODALITIES** | **09 Hours** |
| Fundamentals of Magnetic Resonance (NMR); Interaction of Nuclei with static magnetic fields; Relaxation processes (T1 and T2); MRI System Components—Magnets, RF Coils, and Shim Coils; Functional MRI; Nuclear Medicine (PET and SPECT); Thermography physics and computer-aided diagnosis | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Understand the components of EHR and phenotyping algorithms in healthcare data | L2 |
| 2 | Analyze the impact of IoMT architecture and predictive models for pervasive health | L4 |
| 3 | Examine image enhancement and restoration techniques for various medical imaging formats | L4 |
| 4 | Evaluate the principles and reconstruction methods of Computed Tomography (CT) | L5 |
| 5 | Synthesize MRI relaxation parameters and clinical applications for intelligent diagnosis | L6 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PSO1 | PSO2 | PSO3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 1 | 1 | 2 | 1 | 1 | – | 1 | – | 1 | 3 | 2 | 2 |
| CO2 | 3 | 3 | 2 | 2 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 2 |
| CO3 | 3 | 3 | 3 | 3 | 3 | 1 | – | 1 | 1 | 1 | 2 | 3 | 3 | 1 |
| CO4 | 3 | 3 | 2 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 2 |
| CO5 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 1 | 2 | 1 | 2 | 3 | 3 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS**:

1. Chandan K. Reddy and Charu C. Aggarwal, “Healthcare Data Analytics”, Taylor & Francis, 2015.
2. Paul Suetens, "Fundamentals of Medical Imaging", 3rd Edition, Cambridge University Press, 2017.
3. Rafael C. Gonzales, Richard E. Woods, “Digital Image Processing”, 4th edition, Pearson, 2018.

**REFERENCE BOOKS:**

1. Robert E. Hoyt, "Health Informatics: Practical Guide for Healthcare and Information Technology", 7th Edition, 2018.
2. Edward H. Shortliffe, James J. Cimino, "Biomedical Informatics", Springer, 2014

**MOOCS/NPTEL/SWAYAM:**

1. https://onlinecourses.nptel.ac.in/e-learning/preview/noc22_bt34

# MLOPS & MODEL DEPLOYMENT

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE-I**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1211 | **Credits** | : 03 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–0–0 | | |

**Course Learning Objectives:** This Course will enable students to:

1. Understand the principles of MLOps, ML lifecycle management, and the design of CI/CD/CT pipelines for production machine learning systems.
2. Design and implement reliable data and feature engineering pipelines with validation, versioning, and drift detection capabilities.
3. Apply model versioning, registry management, A/B testing, and automated retraining strategies to manage the complete model lifecycle.
4. Deploy ML models using containerization, Kubernetes, inference serving frameworks, and optimization techniques for scalable production inference.
5. Monitor, govern, and audit production ML systems for drift, fairness, explainability, and compliance with regulatory frameworks such as GDPR and the EU AI Act.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: MLOPS PRINCIPLES** | **09 Hours** |
|---|---|

MLOps principles; ML lifecycle and maturity model (Level 0, 1, 2); CI/CD/CT pipeline design; experiment tracking with MLflow; model registry (versioning, staging); DVC for data versioning; ML system design patterns (training-serving skew, pipeline debt).

| MODULE-2: OPERATIONAL PIPELINES | 09 Hours |
|---|---|
| Apache Airflow DAGs for pipeline orchestration; data validation with Great Expectations; Feast feature store; data drift detection (KS test, PSI); dataset versioning and data lineage; data quality monitoring. | |
| | |
| **MODULE – III: HYPERPARAMETER TUNING** | **09 Hours** |
| Hyperparameter tuning with Ray Tune and Optuna; MLflow Model Registry; Weights & Biases for experiment tracking; A/B testing and canary releases; champion-challenger model selection; automated retraining triggers. | |
| | |
| **MODULE – IV: ESSENTIAL APIs** | **09 Hours** |
| REST API with FastAPI; Docker containerization; Kubernetes for ML workloads; Triton Inference Server; ONNX and TensorRT optimization; serverless and edge ML deployment. | |
| | |
| **MODULE – V: MONITORING** | **09 Hours** |
| Model monitoring (metrics, logs, traces); data and concept drift detection; Prometheus and Grafana dashboards; SHAP explainability in production; fairness auditing; ML governance (model cards, audit trails); GDPR and EU AI Act compliance; NVIDIA NIM Microservices; LLMOps overview. | |
| | |

**COURSE OUTCOMES:**

| Course Outcome | | |
|---|---|---|
| **Course Outcome** | **Description** | **Bloom's Taxonomy Level** |
| **At the end of the course the student will be able to:** | | |
| 1 | Understand MLOps principles, ML lifecycle management, CI/CD/CT pipeline design. | L2 |
| 2 | Design and implement scalable data and feature engineering pipelines. | L3 |
| 3 | Apply model versioning, registry management, A/B testing, shadow deployment, and automated retraining strategies. | L3 |
| 4 | Deploy ML models using containerization, Kubernetes orchestration, inference serving frameworks. | L3 |
| 5 | Evaluate ML systems for data drift, concept drift, explainability, fairness, regulatory compliance. | L5 |

**CO-PO MAPPING**

| Table: Mapping Levels of COs to POs / PSOs | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **COs** | **Program Outcomes (POs)** | | | | | | | | | | | **PSOs** | | |
| **COs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **1** | **2** | **3** |
| CO1 | 3 | 2 | 2 | | 2 | 1 | | | | | | 3 | 2 | 2 |
| CO2 | 3 | 3 | 2 | 2 | 2 | | | 1 | | | | 3 | 3 | 2 |
| CO3 | 3 | 3 | 3 | 2 | 2 | | 1 | 2 | | | | 3 | 3 | 3 |
| CO4 | 3 | 3 | 3 | 3 | 1 | 1 | | 2 | | | 2 | 3 | 3 | 2 |
| CO5 | 2 | 3 | 2 | 2 | 3 | 3 | 2 | 1 | | | | 3 | 2 | 3 |

**3: Substantial (High)** **2: Moderate (Medium)** **1: Poor (Low)**

**Note:** This course requires Machine Learning, Python Programming, Basic Linux and Command Line, and Docker (basic) as prerequisites. Students are encouraged to complete the NVIDIA DLI modules referenced in Modules 4 and 5 and to practise using the full toolchain listed below.

**TEXTBOOKS**

1. Huyen, C. (2022). Designing Machine Learning Systems. O'Reilly Media.
2. Alla, S., & Adari, S. K. (2021). Beginning MLOps with MLflow. Apress.
3. Kleppmann, M. (2017). Designing Data-Intensive Applications. O'Reilly Media. [Module 2]
4. Sato, D., Wider, A., & Windheuser, C. (2019). Continuous Delivery for Machine Learning. martinfowler.com (online).

**REFERENCES**

1. Google Cloud (2021). MLOps: Continuous delivery and automation pipelines in machine learning. cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning
2. NVIDIA DLI: Deploying and Optimizing AI Inference at Scale (8 Hours).
3. NVIDIA DLI: Introduction to NVIDIA NIM Microservices (2 Hours | Free).
4. NVIDIA DLI: Sizing LLM Inference Systems (3 Hours).
5. MLflow Documentation: https://mlflow.org/docs/latest/index.html
6. NVIDIA Triton Inference Server: https://github.com/triton-inference-server/server
7. Evidently AI Documentation (drift monitoring): https://docs.evidentlyai.com/

# REINFORCEMENT LEARNING ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE – II**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1212 | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. Understand the fundamental theory of Reinforcement Learning, its problem formulation, and relationship to broader machine learning paradigms.
2. Develop the ability to model sequential decision-making problems as Markov Decision Processes and apply tabular solution methods including Dynamic Programming, Monte Carlo, and Temporal Difference learning.
3. Apply function approximation techniques and deep neural network-based methods to enable scalable reinforcement learning solutions in continuous state-action spaces.
4. Design, implement, and evaluate policy gradient and actor-critic algorithms for direct policy optimization in complex environments.
5. Analyze advanced RL paradigms including model-based RL, imitation learning, meta-learning, and multi-agent reinforcement learning, and apply them to real-world application domains.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: FOUNDATIONS OF REINFORCEMENT LEARNING AND MARKOV DECISION PROCESSES | 09 Hours |
|---|---|
| **Topics:** Introduction to Reinforcement Learning: historical perspective, definition, and elements of RL; comparison with supervised and unsupervised learning; terminology — agent, environment, state, action, reward, policy, and value function; Bellman Equations. Multi-armed Bandits: exploration vs. exploitation trade-off; ε-Greedy strategy; Upper Confidence Bound (UCB) action selection; Gradient Bandit Algorithm; contextual bandits. Markov Decision Processes (MDPs): Markov property; agent-environment interface; goals and rewards; returns and episodes; policies and value functions; Bellman optimality equations; optimality and approximation; infinite horizon discounted MDP; average cost criteria. Task Formulation: action space, state space, and environment definition; defining RL environments. | |
| | |
| **MODULE – II: TABULAR SOLUTION METHODS** | **09 Hours** |
| **Topics:** Dynamic Programming: principle of optimality; policy evaluation; policy improvement; policy iteration; value iteration; generalized policy iteration; asynchronous dynamic programming; efficiency of dynamic programming. Monte Carlo Methods: first-visit and every-visit Monte Carlo prediction; Monte Carlo control; on-policy and off-policy Monte Carlo learning; importance sampling; Monte Carlo with exploring starts. Temporal Difference Learning: TD(0) prediction; SARSA (on-policy TD control); Q-Learning (off-policy TD control); expected SARSA; double Q-learning; eligibility traces; TD(λ); n-step TD methods; bootstrapping. Markov Reward Process (MRP): overview and relationship to MDPs. | |
| | |
| **MODULE – III: FUNCTION APPROXIMATION METHODS** | **9 Hours** |
| **Topics:** Introduction to function approximation: motivation and challenges in large and continuous state spaces; on-policy prediction with function approximation. Linear Value Function Approximation: feature vectors; stochastic gradient descent; semi-gradient TD(0); linear function approximation for control. Non-linear Value Function Approximation: neural network function approximators; gradient MC and semi-gradient TD algorithms; after-states; least-squares TD. Deep Q-Networks (DQN): experience replay; target networks; DQN algorithm and variants; Double DQN; Dueling DQN; Prioritized Experience Replay. Controllability and stability in continuous state-action spaces; Linear Quadratic Regulator (LQR); policy iteration and value iteration for continuous control; integral reinforcement learning. | |
| | |
| **MODULE – IV: POLICY GRADIENT AND ACTOR-CRITIC METHODS** | **09 Hours** |
| **Topics:** Policy Approximation and advantages: policy parameterization; policy gradient theorem; REINFORCE algorithm (Monte Carlo policy gradient); bias and variance in RL; reducing variance using baselines; advantage function. Actor-Critic Methods: one-step and multi-step actor-critic; A3C (Asynchronous Advantage Actor-Critic); A2C; advantage estimation. Advanced Policy Gradient Algorithms: Proximal Policy Optimization (PPO); Trust Region Policy Optimization (TRPO); Deep Deterministic Policy Gradient (DDPG); Soft Actor-Critic (SAC); Twin Delayed DDPG (TD3). Introduction to Deep Reinforcement Learning: deep neural network-based methods for value-based and policy-based RL; integration of deep learning with RL frameworks. | |
| | |

| MODULE – V: ADVANCED RL PARADIGMS AND APPLICATIONS | 09 Hours |
|---|---|
| **Topics:** Model-Based Reinforcement Learning: model-based vs. model-free RL; Dyna-Q and Dyna-Q+; planning and learning integration; world models. Imitation Learning: behavioral cloning; dataset aggregation (DAgger); inverse reinforcement learning (IRL); reward shaping; generative adversarial imitation learning (GAIL). Meta-Learning in RL: introduction to meta-learning; model-agnostic meta-learning (MAML); applications of meta-learning in RL for fast adaptation. Multi-Agent Reinforcement Learning: multi-agent RL frameworks; cooperative and competitive multi-agent settings; partially observable environments; communication and coordination in multi-agent systems; emergent behaviors. Applications and Case Studies: games (Atari, Go, Chess); robotics; autonomous driving; healthcare; finance; and survey of current research directions and open problems in RL. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| CO1 | Explain the fundamental concepts of Reinforcement Learning, formulate decision-making problems using MDP frameworks, and analyze Markov chains and the agent-environment interaction model. | L1, L2 |
| CO2 | Apply tabular solution methods including Dynamic Programming, Monte Carlo, and Temporal Difference algorithms to solve finite state-space reinforcement learning problems. | L2, L3 |
| CO3 | Implement function approximation methods using linear and non-linear models including Deep Q-Networks to scale reinforcement learning solutions to large and continuous state spaces. | L3, L4 |
| CO4 | Design and evaluate policy gradient methods including REINFORCE, Proximal Policy Optimization, and actor-critic architectures for policy-based reinforcement learning. | L4, L5 |
| CO5 | Analyze and compare advanced RL paradigms such as model-based RL, imitation learning, meta-learning, and multi-agent RL, and evaluate their applicability to real-world domains including robotics, games, and healthcare. | L4, L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO10 | PO11 | PSO 1 | PSO 2 | PSO 3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 1 | 2 | 1 | 1 | | | 1 | 1 | 2 | 3 | 2 | 2 |
| **CO2** | 2 | 3 | 2 | 3 | 2 | 1 | | | 2 | 1 | 2 | 3 | 3 | 2 |
| **CO3** | 3 | 3 | 2 | 3 | 3 | 2 | | | 2 | 1 | 2 | 3 | 3 | 3 |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 2 | 1 | | 2 | 2 | 3 | 3 | 3 | 3 |
| **CO5** | 3 | 3 | 2 | 3 | 3 | 2 | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS:**

1. T1: Sutton, R. S., & Barto, A. G., Reinforcement Learning: An Introduction, MIT Press, 2nd Edition, 2018.
2. T2: Bilgin, E., Mastering Reinforcement Learning with Python, Packt Publishing, 2020.
3. T3: Lapan, M., Deep Reinforcement Learning Hands-On, Packt Publishing, 3rd Edition, 2024.

**REFERENCE BOOKS:**

1. Szepesvari, C., Algorithms for Reinforcement Learning, Morgan and Claypool Publishers, 2010.
2. Wiering, M., & Otterlo, M. V., Reinforcement Learning: State-of-the-Art, Springer, 2012.
3. Zai, A., & Brown, B., Deep Reinforcement Learning in Action, Manning Publications, 2020.
4. Winder, P., Reinforcement Learning: Industrial Applications of Intelligent Agents, O'Reilly Media, 2020.
5. Powell, W. B., Reinforcement Learning and Stochastic Optimization, Wiley, 2022.
6. Research Articles from IEEE Transactions on Neural Networks and Learning Systems, Journal of Machine Learning Research (JMLR), and NeurIPS/ICML/ICLR proceedings.

# ADVANCED DATA MINING & PREDICTIVE MODELING

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE-II**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1213 | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. Examine algorithms for mining frequent item sets and sequential patterns from large-scale datasets.
2. Identify and interpret patterns present in time-series data using analytical techniques.
3. Design and implement algorithms for discovering temporal patterns in evolving data.
4. Utilize graph mining techniques to perform effective web mining and network analysis.
5. Evaluate trajectory mining algorithms for analyzing spatial and movement-based data patterns.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: RULE AND SEQUENTIAL PATTERN MINING** | **09 Hours** |
|---|---|

Introduction to Association Rule Mining, Sequential Pattern Mining concepts, primitives, scalable methods; Transactional Patterns and other temporal based frequent patterns.

| **MODULE – II: MINING TIME SERIES DATA** | **09 Hours** |
|---|---|

Mining Time series Data, Periodicity Analysis for time related sequence data, Trend analysis, Similarity search in Time-series analysis.

| MODULE – III: GRAPH MINING | 09 Hours |
|---|---|
| Graph Mining, Mining frequent subgraphs, finding clusters in large graphs; Web Mining, Mining the web page layout structure, mining web link structure. | |
| | |
| **MODULE – IV: WEB MINING** | **09 Hours** |
| Automatic classification of web documents and web usage mining; Distributed Data Mining, Trajectory Pattern Mining: Moving together patterns, Sequential Pattern mining from trajectories. | |
| | |
| **MODULE – V: DISTRBUTED DATA MINING** | **09 Hours** |
| Distribute data mining framework, Distributed data source, Distributed data mining techniques, Distributed classifier learning, distributed clustering, distributed association rule mining. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Analyze Algorithms for frequent item sets and sequential patterns. | L4 |
| 2 | Determine patterns from time series data. | L3 |
| 3 | Develop algorithms for Temporal Patterns. | L5 |
| 4 | Apply Graph mining algorithms to Web Mining. | L3 |
| 5 | Analyze algorithms for trajectory patterns. | L4 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO10 | PO11 | PSO 1 | PSO 2 | PSO 3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 3 | 2 | 3 | 3 | - | - | 1 | - | - | 3 | 3 | 3 | - |
| **CO2** | 3 | 3 | 2 | 3 | 3 | - | - | 1 | - | - | 3 | 3 | 3 | - |
| **CO3** | 3 | 3 | 3 | 3 | 3 | - | - | 1 | 1 | - | 3 | 3 | 3 | - |
| **CO4** | 3 | 3 | 3 | 3 | 3 | 1 | - | 1 | 1 | - | 3 | 3 | 3 | 1 |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 1 | - | | 1 | - | 3 | 3 | 3 | 1 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS**:

1. Jiawei Han and M Kamber, Data Mining Concepts and techniques, Morgan Kaufmann Publishers In, 2022; Fourth Edition.
2. Chris Chatfield, The Analysis of Time Series: An Introduction, Chapman & Hall/CRC, 2003, Sixth Edition.
3. Bing Liu, Web Data Mining, Springer, 2011, Second Edition.

**REFERENCE BOOKS:**

1. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne and Vipin Kumar, Introduction to Data Mining, Pearson, 2018, Second Edition. 4. G. Dong and J Pei, Sequence Data Mining, Springer, 2007, First Edition.

# EDGE COMPUTING

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE– II**

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | : | 26MAD1214 | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours | **Total Hours** | : | 45 Hours |
| **L–T–P–J** | : | 3–0–0–0 | | | |

**Course Learning Objectives:**

1. To understand advanced concepts of cloud, fog, and edge computing paradigms.
2. To analyze edge architectures, orchestration frameworks, and resource management techniques.
3. To explore AI-enabled edge intelligence and real-time analytics systems.
4. To develop optimized edge AI applications using modern deployment frameworks and containers.
5. To design scalable, secure, and low-latency edge solutions for research and industrial applications.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: FOUNDATIONS OF CLOUD AND EDGE COMPUTING** | **09 Hours** |
|---|---|

Cloud computing fundamentals, distributed systems, virtualization and containerization, cloud service models, limitations of centralized cloud systems: case study, introduction to edge and fog computing, edge architecture, cloud-edge continuum, edge deployment models, latency-sensitive computing, resource-constrained environments, edge-enabled intelligent systems.

| MODULE – II: EDGE ARCHITECTURES AND RESOURCE MANAGEMENT | 09 Hours |
|---|---|
| Edge computing architectures, device edge and cloud edge, edge orchestration, edge deployment strategies, edge resource allocation, workload balancing, virtualization at edge, container orchestration, distributed resource management, scalability challenges, edge security fundamentals, energy-efficient edge systems, edge networking concepts. | |
| **MODULE – III: EDGE COMPUTING IN 5G AND INTELLIGENT NETWORKS** | **09 Hours** |
| 5G architecture fundamentals, edge integration in 5G networks, multi-access edge computing (MEC), mobile edge computing, software-defined networking, network slicing, intelligent edge communication, real-time edge networking, ultra-low latency systems, edge-enabled IoT ecosystems, smart communication infrastructures. | |
| **MODULE – IV: EDGE INTELLIGENCE AND AI AT EDGE** | **09 Hours** |
| Edge intelligence concepts, AI-enabled edge systems, real-time inference at edge, lightweight deep learning models, model compression techniques, pruning and quantization, federated learning at edge, edge analytics pipelines, on-device AI execution, real-time video analytics, speech AI systems, autonomous edge intelligence, performance optimization for edge AI. | |
| **MODULE – V: EDGE DEPLOYMENT, APPLICATIONS, AND RESEARCH TRENDS** | **09 Hours** |
| Jetson Nano and embedded edge platforms, Docker and containerization, Kubernetes for edge systems, Riva and NeMo frameworks, deployment of AI pipelines at edge, smart healthcare systems, industrial IoT applications, autonomous systems, cybersecurity in edge environments, ethical issues in edge AI, sustainable edge computing, current research trends and future directions in edge intelligence. | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Explain advanced edge computing architectures, deployment models, and distributed intelligence concepts. | L2 |
| 2 | Analyze resource optimization, orchestration, and real-time processing techniques in edge systems. | L3 |
| 3 | Design AI-enabled edge computing solutions for latency-sensitive applications. | L4 |
| 4 | Implement and deploy intelligent applications on edge devices using modern frameworks and tools. | L4 |
| 5 | Develop scalable and research-oriented edge AI systems integrating networking, analytics, and optimization techniques. | L6 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO10 | PO11 | PSO 1 | PSO 2 | PSO 3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | - | - | - | 1 | | | | | | | - | - | |
| **CO2** | 3 | 2 | - | 3 | 2 | | | | | | | - | - | |
| **CO3** | 3 | - | 3 | 2 | 3 | | | | | | | 3 | 3 | |
| **CO4** | 3 | - | 3 | - | 3 | | | | | | | - | 3 | |
| **CO5** | 3 | - | 3 | - | 3 | | | | | | | 3 | 3 | |

**3 – High Correlation 2 – Moderate Correlation 1 – Low Correlation**

**TEXT BOOKS**:

1. Weisong Shi, Schahram Dustdar, The Promise of Edge Computing, IEEE Computer Society, 2016.
2. Rajkumar Buyya, James Broberg, Andrzej Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011.
3. Perry Lea, Internet of Things for Architects, Packt Publishing, 2018.

**REFERENCE BOOKS:**

1. Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Morgan Kaufmann, 2012.
2. Satyanarayanan Mahadev, Edge Computing: Vision and Challenges, IEEE Internet Computing, 2017.

**MOOCS/NPTEL/SWAYAM:**

1. https://onlinecourses.nptel.ac.in/e-learning/preview/noc25_cs75

# VIDEO ANALYSIS AND SURVEILLANCE

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE – II**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1215 | **Credits** | : 03 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–0–0 | | |

**Course Learning Objectives:**

1. Understand the fundamentals of video representation, camera models, and motion analysis including optical flow and block matching.
2. Apply background modeling and video segmentation techniques to detect and extract moving objects in real-world surveillance scenarios.
3. Design and implement classical and deep learning-based object detection and multi-object tracking systems.
4. Analyse advanced video analytics using spatio-temporal deep learning models for action recognition, anomaly detection, and crowd analysis.
5. Evaluate intelligent surveillance system architectures addressing real-time processing, multi-camera networks, ethics, and emerging trends.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| MODULE – I: FOUNDATIONS OF VIDEO ANALYSIS AND MOTION | 09 Hours |
|---|---|
| Topics: Introduction to Video Analysis and Surveillance Systems; Applications in security, healthcare, smart cities; Video fundamentals: spatial vs. temporal information; Frame rate, resolution, video formats (MPEG, H.264, H.265); Color spaces (RGB, HSV, YCbCr); Image processing basics: sampling, quantization, noise models, filtering, edge detection; Camera models: pinhole camera model, intrinsic and extrinsic parameters, camera calibration; Geometric transformations and homography; Motion fundamentals: brightness constancy assumption; Optical Flow (Lucas–Kanade, Horn–Schunck); Block matching techniques. | |

| MODULE – II: MOTION DETECTION AND VIDEO SEGMENTATION | 08 Hours |
|---|---|
| Topics: Temporal differencing methods: frame differencing (2-frame, 3-frame), adaptive thresholding; Background modeling: static vs. dynamic backgrounds, running average model, Gaussian Mixture Models (GMM), kernel density estimation; Foreground extraction: blob detection, connected components; Morphological operations: erosion, dilation, opening, closing; Handling real-world challenges: illumination variation, shadow detection and removal, dynamic backgrounds, camera jitter; Performance evaluation metrics: precision, recall, F1-score, Intersection over Union (IoU). | |

| MODULE – III: OBJECT DETECTION AND MULTI-OBJECT TRACKING | 09 Hours |
|---|---|
| Topics: Classical object detection: Haar-like features, cascade classifiers, HOG + SVM; Deep learning-based detection: CNN fundamentals, R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD; Anchor boxes and non-max suppression; Object tracking: single-object tracking (Mean Shift, CAMShift); State estimation techniques: Kalman filter, Particle filter; Multi-object tracking (MOT): tracking-by-detection paradigm; Data association techniques: Hungarian algorithm, IoU-based matching, appearance models; Challenges: occlusion, scale variation, identity switching. | |

| MODULE – IV: ADVANCED VIDEO ANALYTICS AND DEEP LEARNING | 10 Hours |
|---|---|
| Topics: Spatio-temporal feature learning: temporal differencing vs. feature learning, optical flow-based features, trajectory-based descriptors; Action and activity recognition: human action recognition pipelines, feature descriptors (HOG3D, STIP), applications in surveillance; CNN + RNN models: CNN for spatial feature extraction, RNN architectures (LSTM, GRU), sequence modeling, CNN + LSTM pipelines; Deep learning architectures for video: 3D CNN (C3D, I3D), two-stream networks (RGB + optical flow), transformer-based models (ViViT, TimeSformer), architecture comparison; Anomaly detection: supervised vs. unsupervised methods, autoencoders, GAN-based methods; Crowd and behavior analysis: crowd density estimation, motion pattern analysis, abnormal behavior detection. | |

| MODULE – V: INTELLIGENT SURVEILLANCE SYSTEMS AND EMERGING TRENDS | 09 Hours |
|---|---|
| Topics: Surveillance system design: components (cameras, sensors, storage), centralized vs. distributed systems, edge vs. cloud-based analytics; Real-time video processing: latency, throughput, model optimization (quantization, pruning), hardware acceleration (GPU/TPU); Multi-camera systems: camera network calibration, cross-camera tracking, person re-identification, 3D scene understanding; Applications: smart city surveillance, traffic monitoring, industrial and healthcare surveillance; Ethics and privacy: privacy-preserving techniques, bias and fairness, legal frameworks; Emerging trends: self-supervised learning, federated learning, adversarial robustness, vision-language models for video understanding. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| CO1 | Explain the foundational concepts of video analysis using different techniques | L1, L2 |
| CO2 | Apply background modeling, foreground extraction, and video segmentation methods. | L3, L4 |
| CO3 | Design and implement classical and deep learning-based object detection pipelines and multi-object tracking systems. | L3, L5 |
| CO4 | Analyse spatio-temporal DL architectures for action recognition, anomaly detection, and crowd behavior analysis. | L3, L4 |
| CO5 | Evaluate intelligent surveillance system and understand emerging techniques like federated learning and vision-language models. | L4, L5 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO 10 | PO 11 | PSO 1 | PSO 2 | PSO 3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 1 | 2 | 1 | 1 | | | 1 | 1 | 2 | 3 | 2 | 2 |
| **CO2** | 2 | 3 | 2 | 3 | 2 | 1 | | | 2 | 1 | 2 | 3 | 3 | 2 |
| **CO3** | 2 | 3 | 3 | 2 | 2 | 1 | | | 2 | 1 | 2 | 3 | 3 | 2 |
| **CO4** | 3 | 3 | 2 | 3 | 3 | 1 | | | 1 | 1 | 2 | 3 | 2 | 3 |
| **CO5** | 3 | 3 | 2 | 3 | 2 | 2 | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 |

**3 – High Correlation  2 – Moderate Correlation  1 – Low Correlation**

**TEXT BOOKS**:

4. Forsyth, D. A., & Ponce, J., Computer Vision: A Modern Approach, 2nd Ed., Pearson, 2012.
5. Szeliski, R., Computer Vision: Algorithms and Applications, 2nd Ed., Springer, 2022.
6. Moeslund, T. B., Introduction to Video and Image Processing, Springer, 2012.

**REFERENCE BOOKS:**

4. Redmon, J., & Farhadi, A., YOLOv3: An Incremental Improvement, arXiv, 2018.
5. Tran, D., Bourdev, L., Fergus, R., et al., Learning Spatiotemporal Features with 3D Convolutional Networks (C3D), ICCV, 2015.
6. Arnab, A., Dehghani, M., Heigold, G., et al., ViViT: A Video Vision Transformer, ICCV, 2021.

# DATA ENGINEERING & PIPELINE

[As per Choice Based Credit System (CBCS) scheme]

**PROFESSIONAL ELECTIVE – I**

| | | | |
|---|---|---|---|
| **Course Code** | : 26MAD1216 | **Credits** | : 03 |
| **Hours / Week** | : 03 Hours | **Total Hours** | : 45 Hours |
| **L–T–P–J** | : 3–0–0–0 | | |

**Course Learning Objectives:**

1. Understand the architecture and components of modern data engineering systems.
2. Design and implement efficient ETL/ELT workflows and scalable data pipelines.
3. Apply distributed data storage and processing techniques for large-scale datasets.
4. Develop real-time and batch data processing applications using industry-standard tools.
5. Explore GPU-accelerated and AI-driven data engineering concepts aligned with NVIDIA DLI practices.

**Teaching-Learning Process (General Instructions)**

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

| **MODULE – I: INTRODUCTION TO DATA ENGINEERING AND MODERN DATA ECOSYSTEMS** | **09 Hours** |
|---|---|

Introduction to Data Engineering, Role of Data Engineers, Data Engineering Lifecycle, Structured, Semi-structured and Unstructured Data, Data Warehouses, Data Lakes and Lakehouse Architecture, Modern Data Stack, Big Data Ecosystems, Hadoop Ecosystem Overview, Introduction to Apache Spark, Data-driven Decision Systems.

**Case Study:** Architecture of a large-scale e-commerce analytics platform for customer behavior analysis.

| MODULE – II: DATA STORAGE, MODELLING AND MANAGEMENT | 09 Hours |
|---|---|
| Relational and NoSQL Databases, Data Modeling Techniques, Star and Snowflake Schemas, Distributed Storage Systems, HDFS Concepts, Object Storage, Data Serialization Formats (JSON, Avro, Parquet), Data Partitioning and Indexing, Metadata Management, Data Governance Basics. | |
| | |
| **MODULE – III: ETL/ELT AND DATA PIPELINE ENGINEERING** | **09 Hours** |
| ETL and ELT Architectures, Data Ingestion Methods, Batch Data Processing, Data Cleaning and Transformation, Workflow Scheduling, Pipeline Automation, Apache Airflow Fundamentals, Data Validation Techniques, Logging and Error Handling, Data Integration Strategies.<br>**Case Study:** Development of a healthcare ETL pipeline for integrating patient monitoring data from multiple hospital systems. | |
| | |
| **MODULE – IV: DISTRIBUTED AND REAL-TIME DATA PROCESSING** | **09 Hours** |
| Distributed Computing Fundamentals, Apache Spark Architecture, Spark DataFrames and RDDs, PySpark Programming, Real-time Stream Processing, Apache Kafka Fundamentals, Stream Analytics, Windowing Operations, Fault Tolerance and Scalability in Streaming Systems. | |
| | |
| **MODULE – V: GPU-ACCELERATED AND CLOUD DATA ENGINEERING** | **09 Hours** |
| Introduction to GPU Computing for Data Engineering, NVIDIA RAPIDS Ecosystem, GPU-accelerated DataFrames using cuDF, Accelerated ETL Pipelines, Introduction to CUDA for Data Processing, Distributed GPU Data Processing using Dask-cuDF, Cloud Data Engineering on AWS/Azure/GCP, Containerization using Docker, CI/CD for Data Pipelines, Pipeline Monitoring and Optimization. | |
| | |

## COURSE OUTCOMES

| Course Outcome | Description | Bloom's Taxonomy Level |
|---|---|---|
| **At the end of the course the student will be able to:** | | |
| 1 | Explain the principles and architecture of modern data engineering ecosystems. | L2 |
| 2 | Build and automate scalable ETL/ELT pipelines for enterprise applications | L4 |
| 3 | Utilize distributed and streaming frameworks for large-scale data processing | L3 |
| 4 | Develop cloud-native and GPU-accelerated data engineering workflows. | L5 |
| 5 | Analyse real-world case studies and optimize data pipelines for performance, reliability, and security. | L3 |

## CO-PO MAPPING

| Course Outcomes (COs) | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO 10 | PO 11 | PSO 1 | PSO 2 | PSO 3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | - | - | | - | | | | | | | 1 | 2 | |
| **CO2** | 3 | - | 3 | | 2 | | | | | | | 2 | 3 | |
| **CO3** | 3 | - | 3 | | 2 | | | | | | | 2 | 3 | |
| **CO4** | 3 | - | 3 | | 3 | | | | | | | 3 | 3 | |
| **CO5** | 3 | - | 3 | | 3 | | | | | | | 3 | 3 | |

**3 – High Correlation 2 – Moderate Correlation 1 – Low Correlation**

**TEXT BOOKS**:

1. Martin Kleppmann, "Designing Data-Intensive Applications", O'Reilly Media, 2017.
2. Joe Reis and Matt Housley, "Fundamentals of Data Engineering", O'Reilly Media, 2022.
3. Paul Crickard, "Data Engineering with Python", Packt Publishing, 2020.
4. Tyler Akidau, Slava Chernyak and Reuven Lax, "Streaming Systems", O'Reilly Media, 2018.

**REFERENCE BOOKS:**

1. Bill Chambers and Matei Zaharia, "Spark: The Definitive Guide", O'Reilly Media, 2018.
2. Tom White, "Hadoop: The Definitive Guide", O'Reilly Media, 2015.
3. Julitta Korol, "Apache Kafka Quick Start Guide", Packt Publishing, 2019.
4. Andreas Kretz, "Data Engineering Cookbook", Packt Publishing, 2021.

**MOOCS/NPTEL/SWAYAM:**

1. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+C-DS-04+V1

**Signature**



**Chairperson CSE (AI & DS)**