

# **MAHAKAUSHAL**

## **University**



**Faculty of & Computer Science  
Syllabus & Prescribed Books**

**Subject-M.SC Artificial  
Intelligence and Machine  
Learning**

**M.Sc. AI&ML Semester  
Examination 2023-24  
I, II, III & IV Semester**

## M.Sc. ( Artificial Intelligence and Machine Learning ) Scheme of Examination

### I<sup>ST</sup> SEMESTER

- |                         |               |                               |      |
|-------------------------|---------------|-------------------------------|------|
| 1. Course Code          | : MAIM        | 5. Total Practical            | 2    |
| 2. Course Name          | : M.Sc. AI&ML | 6. Total Practical Marks      | 100  |
| 3. Total Theory Subject | : 5           | 7. Total Marks                | 350  |
| 4. Total Theory Marks   | : 250         | 8. Minimum Passing Percentage | : 36 |

Sub.Code	Subject Name	Theory									
		Paper					CCE		Practical		
		Total	1st	2nd	3rd	Max.	Min.	Max	Min.	Max.	Min.
MAIM101-T	Mathematics for AI and ML	42	0	0	42	15	8	3	0	0	50
MAIM102-T	Machine Learning Fundamentals	42	0	0	42	15	8	3	0	0	50
MAIM103-T	Artificial Intelligence	42	0	0	42	15	8	3	0	0	50
MAIM103-P	Artificial Intelligence LAB	0	0	0	0	0	0	0	50	18	50
MAIM104-T	Object oriented programming using C++	42	0	0	42	15	8	3	0	0	50
MAIM104-P	Object oriented programming using C++LAB	0	0	0	0	0	0	0	50	18	50
MAIM105-T(A)	Problem-solving and Search Techniques	42	0	0	42	15	8	3	0	0	50
MAIM105-T(B)	Probability and Statistics	42	0	0	42	15	8	3	0	0	50

## MAIM- 101 Mathematics for AI and ML

Unit No.	Topics
Unit 1	<b>Introduction to Mathematics for AI and ML in Tourism Management:-</b> Overview of artificial intelligence (AI) and machine learning (ML) applications in tourism management, Introduction to mathematical concepts used in AI and ML: linear algebra, calculus, probability theory, and statistics, Basic Python programming for AI and ML applications in tourism.
Unit 2	<b>Data Preprocessing and Feature Engineering:-</b> Data preprocessing techniques: data cleaning, normalization, and transformation, Feature engineering: selecting, extracting, and transforming features for AI and ML models, Hands-on exercises using Python libraries (e.g., NumPy, Pandas) for data preprocessing and feature engineering.
Unit 3	<b>Supervised Learning Algorithms:-</b> Overview of supervised learning algorithms: regression and classification, Linear regression and logistic regression for predicting tourism demand and customer preferences, Decision trees, random forests, and support vector machines for tourism forecasting and decision-making, Practical implementation of supervised learning algorithms using Python (e.g., scikit-learn).
Unit 4	<b>Unsupervised Learning Algorithms:-</b> Introduction to unsupervised learning algorithms: clustering and dimensionality reduction, K-means clustering and hierarchical clustering for market segmentation and customer profiling, Principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE) for visualizing high-dimensional tourism data, Hands-on exercises applying unsupervised learning algorithms to tourism datasets.
Unit 5	<b>Advanced Topics in AI and ML for Tourism Management:-</b> Advanced machine learning techniques for tourism management: Neural networks and deep learning for image recognition and natural language processing in tourism, Reinforcement learning for optimizing tourism operations and personalized recommendations, Case studies and real-world applications of AI and ML in tourism management, Ethical considerations and challenges of using AI and ML in the tourism industry.

### Reference books :

Reference Book:

Title: "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow"

Author: Aurélien Géron

Publisher: O'Reilly Media

## MAIM102: Machine Learning Fundamentals

Unit No.	Topics
Unit 1	<b>Introduction to Machine Learning and its Applications in Tourism:-</b> Overview of machine learning concepts and algorithms, Introduction to supervised, unsupervised, and reinforcement learning, Applications of machine learning in tourism management: demand forecasting, customer segmentation, recommendation systems, etc.
Unit 2	<b>Data Preprocessing and Feature Engineering for Tourism Data:-</b> Data preprocessing techniques: cleaning, normalization, and transformation, Feature engineering: selecting, extracting, and transforming features from tourism datasets, Hands-on exercises using Python libraries (e.g., Pandas, NumPy) for data preprocessing and feature engineering.
Unit 3	<b>Supervised Learning Algorithms for Tourism Management:-</b> Overview of supervised learning algorithms: regression and classification, Linear regression and logistic regression for predicting tourism demand and customer behavior, Decision trees, random forests, and support vector machines for tourism forecasting and decision-making, Practical implementation of supervised learning algorithms using Python (e.g., scikit-learn).
Unit 4	<b>Unsupervised Learning Algorithms for Tourism Management:</b> Introduction to unsupervised learning algorithms: clustering and dimensionality reduction, K-means clustering and hierarchical clustering for market segmentation and customer profiling, Principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE) for visualizing tourism data, Hands-on exercises applying unsupervised learning algorithms to tourism datasets.
Unit 5	<b>Advanced Topics in Machine Learning for Tourism Management:-</b> Advanced machine learning techniques for tourism: Neural networks and deep learning for image recognition, natural language processing, and time series forecasting in tourism, Reinforcement learning for optimizing tourism operations and personalized recommendations, Case studies and real-world applications of machine learning in tourism management, Ethical considerations and challenges of using machine learning in the tourism industry.

### Reference books :

Title: "Python Machine Learning"  
Author: Sebastian Raschka, Vahid Mirjalili  
Publisher: Packt Publishing

## MAIM103: Artificial Intelligence

Unit No.	Topics
Unit 1	<b>Introduction to Artificial Intelligence in Tourism:-</b> Overview of artificial intelligence (AI) and its applications in the tourism industry, Importance of AI in improving customer experiences, optimizing operations, and enhancing marketing strategies, Ethical considerations and challenges of implementing AI in tourism management.
Unit 2	<b>Data Collection and Preprocessing for Tourism Data:-</b> Methods for collecting and preprocessing tourism-related data, including customer reviews, booking information, and demographic data, Data cleaning, normalization, and feature extraction techniques, Introduction to Python libraries such as Pandas and NumPy for data manipulation and preprocessing.
Unit 3	<b>Machine Learning for Tourism Analytics:-</b> Introduction to machine learning concepts and algorithms relevant to tourism management, such as regression, classification, and clustering, Applications of machine learning in tourism analytics, including demand forecasting, customer segmentation, and recommendation systems, Hands-on exercises using Python libraries such as scikit-learn for implementing machine learning models.
Unit 4	<b>Natural Language Processing (NLP) for Tourism:-</b> Overview of natural language processing (NLP) techniques for analyzing text data in the tourism industry, Sentiment analysis of customer reviews, chatbots for customer service, and text summarization for content analysis, Practical implementation of NLP techniques using Python libraries such as NLTK and spaCy.
Unit 5	<b>Advanced Topics in AI for Tourism Management:-</b> Advanced AI techniques and applications in tourism management, such as deep learning, reinforcement learning, and predictive analytics, Case studies and real-world examples of AI implementation in tourism, including smart destination management systems and personalized travel recommendations, Future trends and opportunities for AI-driven innovation in the tourism industry.

### Reference books :

Title: "Artificial Intelligence: A Modern Approach"  
Author: Stuart Russell and Peter Norvig  
Publisher: Pearson

## **Artificial Intelligence Lab:-**

### 1. Introduction to Sentiment Analysis:

- Brief overview of sentiment analysis and its applications in analyzing text data.
- Explanation of sentiment analysis techniques, including lexicon-based and machine learning-based approaches.

### 2. Data Preparation:

- Provide a dataset containing tourism-related text data, such as customer reviews of hotels, restaurants, or attractions.
- Preprocess the text data by removing stopwords, punctuation, and special characters, and tokenizing the text into words or phrases.

### 3. Exploratory Data Analysis (EDA):

- Conduct EDA to understand the distribution of sentiment in the text data.
- Visualize key insights, such as the distribution of positive, negative, and neutral sentiments.

### 4. Lexicon-based Sentiment Analysis:

- Implement a lexicon-based sentiment analysis approach using pre-built sentiment lexicons, such as the AFINN lexicon or VADER sentiment analyzer.
- Calculate sentiment scores for each review based on the presence of positive and negative words.

### 5. Machine Learning-based Sentiment Analysis:

- Train a machine learning model for sentiment analysis using supervised learning techniques.
- Represent text data using features such as Bag-of-Words, TF-IDF, or word embeddings.
- Train a classification model (e.g., Logistic Regression, Naive Bayes, or Support Vector Machine) to predict sentiment labels (positive, negative, neutral) based on the text features.

### 6. Model Evaluation:

- Split the dataset into training and testing sets.
- Evaluate the performance of the sentiment analysis models using metrics such as accuracy, precision, recall, and F1-score.
- Compare the performance of lexicon-based and machine learning-based approaches.

### 7. Application to Real-world Data:

- Apply the trained sentiment analysis model to analyze sentiment in real-world tourism reviews.
- Extract insights about customer sentiment towards specific tourist destinations, hotels, or attractions.

### 8. Interactive Visualization:

- Create interactive visualizations to display the sentiment analysis results, such as word clouds, sentiment histograms, or sentiment trends over time.
- Use libraries like matplotlib, seaborn, or Plotly for visualization.

### 9. Discussion and Interpretation:

- Facilitate a discussion on the insights derived from the sentiment analysis of tourism reviews.

- Interpret the implications of positive and negative sentiment for tourism businesses and destinations.
- Discuss potential strategies for improving customer satisfaction and addressing negative feedback.

#### 10. Wrap-Up and Next Steps:

- Summarize key takeaways from the practical session and highlight the importance of sentiment analysis in understanding customer sentiment in the tourism industry.
- Provide resources for further exploration of NLP and sentiment analysis techniques in tourism management and customer experience analysis.

## MAIM104: Object Oriented Concepts & programming using C++

Unit No.	Topics
Unit 1	<b>Introduction to Object-Oriented Programming (OOP):-</b> Overview of object-oriented programming and its importance in software development, Basic concepts of OOP: classes, objects, encapsulation, inheritance, and polymorphism, Explanation of how OOP principles can be applied in tourism management software development.
Unit 2	<b>Classes and Objects in C++:-</b> Introduction to classes and objects in C++, Defining classes, member functions, and data members, Creating and using objects of classes in C++, Examples of designing classes for tourism-related entities such as hotels, tourist attractions, and travel packages.
Unit 3	<b>Inheritance and Polymorphism:-</b> Understanding inheritance and its role in code reuse and extensibility, Implementing inheritance in C++: single inheritance, multiple inheritance, and hierarchical inheritance, Polymorphism and its implementation through function overloading and overriding in C++, Examples of using inheritance and polymorphism in tourism management software design.
Unit 4	<b>Encapsulation and Abstraction:-</b> Explaining encapsulation and how it helps in data hiding and abstraction, Implementing encapsulation using access specifiers in C++, Utilizing abstraction to model complex systems and hide implementation details, Examples of encapsulation and abstraction in tourism management software development.
Unit 5	<b>Advanced Topics in C++ Programming:-</b> Exception handling in C++: try-catch blocks, throw keyword, and handling exceptions, Templates and generic programming in C++, File handling in C++: reading from and writing to files for data storage and retrieval in tourism applications, Case studies and practical projects applying OOP principles and C++ programming in tourism management scenarios.

### Reference Books:

Title: "C++ Primer"

Author: Stanley B. Lippman,

Josée Lajoie, Barbara E. Moo



## **Object Oriented Concepts & programming using C++ Lab:-**

### **1.Designing Class Structures:**

- Identify the key entities in the hotel management system, such as Room, Guest, and Reservation.
- Design class structures for each entity, including data members and member functions.

### **2.Implementing Room Class:**

- Create a Room class to represent individual hotel rooms.
- Define data members to store room number, type (e.g., single, double), availability status, and price.
- Implement member functions to set room availability, update room details, and display room information.

### **3.Implementing Guest Class:**

- Create a Guest class to represent hotel guests.
- Define data members to store guest information such as name, contact details, and reservation details.
- Implement member functions to add, update, and display guest information.

### **4.Implementing Reservation Class:**

- Create a Reservation class to manage hotel reservations.
- Define data members to store reservation details such as room number, guest information, check-in/check-out date, and total cost.
- Implement member functions to create, update, and cancel reservations, as well as calculate reservation costs.

### **5.Integration and Testing:**

- Integrate the Room, Guest, and Reservation classes into a cohesive hotel management system.
- Implement a user interface for interacting with the system, such as a command-line interface.
- Test the functionality of the system by creating sample rooms, guests, and reservations, and performing operations such as room booking and guest check-in/out.

### **6.Error Handling and Validation:**

- Implement error handling mechanisms to handle invalid inputs and edge cases.
- Validate user inputs to ensure data integrity and consistency within the system.

### **7.File Handling for Data Persistence:**

- Implement file handling functionality to save and load hotel data to/from external files.
- Use file I/O operations to store room, guest, and reservation information persistently.

### **8.Advanced Features (Optional):**

- Implement additional features such as room categorization, room service management, or billing functionality.
- Enhance the user interface with more interactive and user-friendly features.

### **9.User Documentation:**

- Create user documentation or a user manual that explains how to use the hotel management system.
- Provide instructions for common tasks such as making reservations, checking room availability, and managing guest information.

### **10.Presentation and Demonstration:**

- Prepare a presentation to showcase the hotel management system and its features.
- Demonstrate the functionality of the system, including room booking, guest management, and reservation handling.

## MAIM105(A): Problem-solving and Search Techniques

Unit No.	Topics
Unit 1	<b>Introduction to Problem-solving and Search Techniques:-</b> Overview of problem-solving methodologies in the context of tourism management, Introduction to search algorithms and their applications in finding optimal solutions, Discussion on problem representation, state space, and search space in tourism-related scenarios.
Unit 2	<b>Uninformed Search Algorithms:-</b> Introduction to uninformed search algorithms: Breadth-First Search (BFS), Depth-First Search (DFS), and Iterative Deepening Depth-First Search (IDDFS), Implementation of uninformed search algorithms for solving tourism management problems, such as route planning and itinerary generation, Comparison of performance and efficiency of different uninformed search algorithms.
Unit 3	<b>Informed Search Algorithms:-</b> Introduction to informed search algorithms: A* search algorithm, Greedy Best-First Search, and Hill Climbing, Heuristic functions and their role in guiding search algorithms towards the goal state Application of informed search algorithms in tourism-related scenarios, such as optimizing travel routes and resource allocation.
Unit 4	<b>Constraint Satisfaction Problems (CSP):-</b> Introduction to Constraint Satisfaction Problems (CSP) and their relevance to tourism management, Constraint propagation and backtracking search algorithms for solving CSPs, Modeling tourism-related problems as CSPs, such as scheduling tours or allocating resources within constraints.
Unit 5	<b>Metaheuristic and Evolutionary Algorithms:-</b> Introduction to metaheuristic algorithms: Genetic Algorithms (GA), Simulated Annealing, and Tabu Search, Application of metaheuristic algorithms in solving optimization problems in tourism management, such as tour scheduling and facility location, Discussion on the advantages, limitations, and parameter tuning of metaheuristic algorithms.

### Reference Books:

Title: "Artificial Intelligence: A Guide to Intelligent Systems"

Author: Michael Negnevitsky

Publisher: Addison Wesley

## MAIM105(B): Probability and Statistics

Unit No.	Topics
Unit 1	<b>Introduction to Probability and Statistics:-</b> Overview of probability theory and its applications in tourism management ,Basic concepts of probability: events, sample spaces, and probability distributions, Introduction to descriptive and inferential statistics.
Unit 2	<b>Data Collection and Sampling Techniques:-</b> Methods for collecting tourism-related data, including surveys, observations, and secondary sources, Sampling techniques and sample design in tourism research, Hands-on exercises on data collection and sampling using real-world tourism datasets.
Unit 3	<b>Exploratory Data Analysis (EDA):-</b> Exploratory data analysis techniques for understanding tourism data, Graphical methods: histograms, box plots, scatter plots, etc. Numerical summaries: mean, median, variance, standard deviation, etc.
Unit 4	<b>Probability Distributions and Statistical Inference:-</b> Common probability distributions used in tourism management, such as the normal distribution, binomial distribution, and Poisson distribution, Statistical inference: estimation and hypothesis testing, Application of probability distributions and statistical inference in tourism research and decision-making.
Unit 5	<b>Advanced Topics in Probability and Statistics for Tourism:-</b> Multivariate analysis techniques: correlation analysis, regression analysis, and factor analysis, Time series analysis and forecasting methods for tourism demand prediction, Bayesian statistics and its applications in tourism management.

### Reference Books:

Title: "Statistics for Business and Economics"

Author: Paul Newbold, William L. Carlson, Betty Thorne

Publisher: Pearson

## M.Sc. (Artificial Intelligence and Machine Learning) Scheme of Examination

### II<sup>ND</sup> SEMESTER

- |                              |                                    |
|------------------------------|------------------------------------|
| 1. Course Code : MAIM        | 5. Total Practical : 2             |
| 2. Course Name : M.Sc. AI&ML | 6. Total Practical Marks : 100     |
| 3. Total Theory Subject : 5  | 7. Total Marks : 350               |
| 4. Total Theory Marks : 250  | 8. Minimum Passing Percentage : 36 |

Sub.Code	Subject Name	Theory										
		Paper				CCE				Practical		Total
		1st	2nd	3rd	Max.	Min.	Max	Min.	Max.	Min.	Max.	
M AIM201-T	Deep Learning and Neural Networks	42	0	0	42	15	8	3	50	18	50	
M AIM201-P	Deep Learning and Neural Networks LAB	0	0	0	0	0	0	0	50	18	50	
M AIM202-T	Text Classification and Sentiment Analysis	42	0	0	42	15	8	3	0	0	50	
M AIM203-T	Communication Skill	42	0	0	42	15	8	3	0	0	50	
M AIM204-T	Big Data Analytics and Machine Learning	42	0	0	42	15	8	3	0	0	50	
M AIM204-P	Big Data Analytics and Machine Learning LAB	0	0	0	0	0	0	0	50	18	50	
M AIM205-T(A)	Convolutional Neural Networks (CNNs)	42	0	0	42	15	8	3	0	0	50	
M AIM205-T(B)	Recurrent Neural Networks (RNNs)	42	0	0	42	15	8	3	0	0	50	

## MAIM201: Deep Learning and Neural Networks

Unit No.	Topics
Unit 1	<b>Introduction to Deep Learning and Neural Networks:-</b> Overview of deep learning and its applications in tourism management, Basics of neural networks: perceptrons, activation functions, and feedforward networks, Introduction to deep learning frameworks such as TensorFlow and PyTorch.
Unit 2	<b>Building Blocks of Deep Learning:-</b> Convolutional Neural Networks (CNNs) for image recognition and analysis in tourism (e.g., image-based tourism recommendation systems), Recurrent Neural Networks (RNNs) for sequence modeling and time-series analysis in tourism (e.g., predicting tourist arrivals), Hands-on exercises with CNNs and RNNs using deep learning frameworks.
Unit 3	<b>Advanced Architectures in Deep Learning:-</b> Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs) for handling sequential data in tourism applications (e.g., sentiment analysis of tourist reviews), Attention mechanisms for improving the performance of neural networks in tourism-related tasks (e.g., personalized travel recommendations), Case studies and real-world examples of advanced neural network architectures in tourism management.
Unit 4	<b>Transfer Learning and Pre-trained Models:-</b> Transfer learning techniques for leveraging pre-trained models in tourism applications (e.g., fine-tuning pre-trained image classification models for tourism image analysis), Introduction to popular pre-trained models such as VGG, ResNet, and BERT, Hands-on exercises on transfer learning and fine-tuning pre-trained models for tourism-related tasks.
Unit 5	<b>Ethical Considerations and Future Directions:-</b> Ethical considerations in deploying deep learning models in tourism management (e.g., privacy concerns, bias in AI systems), Emerging trends and future directions in deep learning for tourism, such as reinforcement learning for personalized travel planning and generative models for content creation, Discussion on the societal impact of deep learning in the tourism industry and responsible AI practices.

### Reference Book:

Title: "Deep Learning"

Author: Ian Goodfellow, Yoshua Bengio, Aaron Courville

Publisher: MIT Press

## **Deep Learning and Neural Networks Lab:-**

### **1.Introduction to Sentiment Analysis and Deep Learning:**

- Overview of sentiment analysis and its importance in understanding customer sentiment in the tourism industry.
- Introduction to deep learning models for sentiment analysis, including Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs).

### **2.Data Preparation:**

- Provide a dataset containing tourism-related text data, such as customer reviews of hotels, restaurants, or attractions.
- Preprocess the text data by tokenizing, removing stopwords, and converting words to numerical representations using techniques like word embeddings.

### **3.Building a Recurrent Neural Network (RNN) Model:**

- Implement a simple RNN model for sentiment analysis using libraries like TensorFlow or PyTorch.
- Design the architecture of the RNN model with layers such as Embedding, LSTM, and Dense.
- Train the RNN model on the preprocessed text data for sentiment classification (positive, negative, neutral).

### **4.Building a Convolutional Neural Network (CNN) Model:**

- Implement a CNN model for sentiment analysis, which can capture local patterns and features in the text data.
- Design the architecture of the CNN model with convolutional layers, pooling layers, and dense layers.
- Train the CNN model on the preprocessed text data for sentiment classification.

### **5.Model Evaluation and Comparison:**

- Split the dataset into training and testing sets.
- Evaluate the performance of the RNN and CNN models using metrics such as accuracy, precision, recall, and F1-score.
- Compare the performance of the two models and analyze their strengths and weaknesses.

### **6.Hyperparameter Tuning:**

- Perform hyperparameter tuning to optimize the performance of the deep learning models.
- Experiment with different configurations of hyperparameters such as learning rate, batch size, and dropout rate.

### **7.Transfer Learning with Pre-trained Word Embeddings:**

- Utilize pre-trained word embeddings (e.g., Word2Vec, GloVe) to enhance the performance of the sentiment analysis models.
- Fine-tune the deep learning models with pre-trained word embeddings and observe the impact on model performance.

### **8.Interactive Demo:**

- Create an interactive demo where users can input text reviews, and the trained deep learning model predicts the sentiment of the input text.
- Implement the demo using web development frameworks like Flask or Django.

### **9.Error Analysis and Interpretation:**

- Analyze misclassified examples and errors made by the deep learning models.
- Identify common patterns or challenges in sentiment analysis of tourism reviews and discuss potential improvements.

### **10.Presentation and Discussion:**

- Prepare a presentation to showcase the implemented deep learning models and their performance in sentiment analysis of tourism reviews.
- Facilitate a discussion on the implications of sentiment analysis in tourism management and the potential applications of deep learning models in understanding customer sentiment and feedback.

### **Practical Requirements:**

- Python environment with libraries such as TensorFlow or PyTorch for deep learning implementation
- Preprocessed text data containing tourism reviews
- Jupyter Notebook or any Python IDE for writing and executing code
- Web development tools for building an interactive demo (optional)

## MAIM-202 - Text Classification and Sentiment Analysis

Unit No.	Topics
Unit 1	<b>Introduction to Text Classification and Sentiment Analysis:-</b> Overview of text classification and sentiment analysis in the context of tourism management, Importance of understanding customer sentiment for improving tourism services and experiences, Introduction to common techniques and algorithms used in text classification and sentiment analysis.
Unit 2	<b>Text Preprocessing for Tourism Data:-</b> Methods for preprocessing tourism-related text data, including tokenization, stopword removal, and stemming/lemmatization, Techniques for handling noisy and unstructured text data in tourism management, Hands-on exercises using Python libraries like NLTK and spaCy for text preprocessing.
Unit 3	<b>Feature Engineering and Representation:-</b> Techniques for feature engineering and representation in text classification and sentiment analysis, Bag-of-Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), and word embeddings for representing text data, Comparison of different feature representation methods and their suitability for tourism-related text analysis tasks.
Unit 4	<b>Traditional Machine Learning for Text Classification:-</b> Introduction to traditional machine learning algorithms for text classification, including Naive Bayes, Support Vector Machines (SVM), and decision trees, Training and evaluation of machine learning models for text classification tasks in tourism management, Techniques for model evaluation, performance optimization, and dealing with imbalanced datasets.
Unit 5	<b>Sentiment Analysis in Tourism Management:-</b> Advanced techniques for sentiment analysis in tourism, such as aspect-based sentiment analysis and opinion mining, Case studies and real-world applications of sentiment analysis in tourism, including analyzing customer reviews, social media sentiment analysis, and sentiment-based recommendation systems, Ethical considerations and challenges in sentiment analysis, such as privacy concerns and bias detection.

### Reference Book:

Title: "Text Mining and Analysis: Practical Methods, Examples, and Case Studies Using SAS"

Author: Goutam Chakraborty, Murali Pagolu, Satish Garla

Publisher: Chapman and Hall/CRC

## MAIM-203 Communication Skill

Unit No.	Topics
Unit 1	<b>Introduction:</b> Definition, nature, objects, elements and importance of communication, principles and practices, models of communication, types of communication,.
Unit 2	<b>Communication Skills and Soft Skills</b> Interviewing and group discussion, resume preparation, etiquette and manners, self-management, body and sign language, presentations skills, feedback & questioning technique: objectiveness in argument (Both one on one and in groups).
Unit 3	<b>Concept to Effective Communication</b> Dimensions and directions of communication, means of communication, 7C's for effective communication.
Unit 4	<b>Listening Skills</b> Importance of listening skills, good & bad listening, communication channels, types of communication medium- audio, video, digital, barriers of communication.
Unit 5	<b>Public Speaking and Reporting</b> Effective Public Speaking and its principles, interpretation and techniques of report writing, letter writing, negotiation skills.

Text Book :

### Suggested Reading:

Business Communication Royan and V. lesikar, John D. Pettit, JR. Richard D. Irwin, I

NC Business communication- K.K. Sinha

Business Etiquettes – David Robinson

Business communication Dr. Nageshwar Rao and Dr. R.P.D

as Effective business communication- Morphy Richards



## MAIM204: Big Data Analytics and Machine Learning

Unit No.	Topics
Unit 1	<b>Introduction to Big Data Analytics and Machine Learning:-</b> Overview of big data analytics and machine learning and their applications in tourism management, Introduction to key concepts such as data mining, predictive modeling, and clustering, Discussion on the importance of data-driven decision-making in the tourism industry.
Unit 2	<b>Data Collection and Preprocessing for Tourism Data:-</b> Methods for collecting and preprocessing large-scale tourism data from various sources such as social media, booking platforms, and IoT devices, Techniques for cleaning, filtering, and transforming raw data into structured datasets suitable for analysis, Hands-on exercises using Python libraries like Pandas and NumPy for data preprocessing.
Unit 3	<b>Exploratory Data Analysis (EDA) for Tourism Data:-</b> Techniques for exploratory data analysis (EDA) to gain insights into tourism datasets, Visualization methods such as histograms, scatter plots, and heatmaps for analyzing data distributions and relationships, Case studies and examples illustrating the use of EDA in understanding tourist behaviors, preferences, and trends.
Unit 4	<b>Machine Learning Models for Tourism Prediction and Recommendation:-</b> Overview of machine learning algorithms for predictive modeling and recommendation systems, Techniques such as regression, classification, and clustering for analyzing tourism data, Hands-on implementation of machine learning models using libraries like scikit-learn for tasks such as tourist demand prediction and personalized recommendation.
Unit 5	<b>Advanced Topics in Big Data Analytics and Machine Learning:-</b> Advanced techniques for big data analytics and machine learning, such as ensemble learning, deep learning, and reinforcement learning, Real-world applications of advanced analytics in tourism management, such as sentiment analysis, image recognition, and anomaly detection, Ethical considerations and challenges in deploying machine learning models in tourism contexts, including privacy concerns and bias mitigation strategies.

**Reference Book:**

Title: "Big Data Analytics: A Hands-On Approach"

Author: Arshdeep Bahga, Vijay Madisetti

Publisher: VPT

**Big Data Analytics and Machine Learning Lab:-****1.Data Acquisition and Preprocessing:**

- Obtain a real-world dataset containing historical tourism data, including variables such as tourist arrivals, weather conditions, local events, and economic indicators.
- Preprocess the dataset by handling missing values, encoding categorical variables, and scaling numerical features.

**2.Exploratory Data Analysis (EDA):**

- Conduct exploratory data analysis to understand the distribution and relationships of variables in the dataset.
- Visualize key features such as tourist arrivals over time, seasonal patterns, and correlations between variables using matplotlib or seaborn.

**3.Feature Engineering:**

- Engineer relevant features from the dataset to improve the predictive performance of the model.
- Create lag features to capture temporal dependencies and seasonal trends in tourist demand.
- Explore domain-specific features such as holidays, special events, or cultural festivals that may influence tourist arrivals.

**4.Model Selection and Training:**

- Select appropriate machine learning algorithms for the predictive modeling task, such as regression, time series forecasting, or ensemble methods.
- Split the preprocessed dataset into training and testing sets for model evaluation.
- Train machine learning models using techniques like linear regression, decision trees, or gradient boosting.

**5.Model Evaluation and Validation:**

- Evaluate the performance of trained models using appropriate metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), or R-squared.
- Compare the performance of different models and identify the best-performing model for predicting tourist demand.
- Validate the model using cross-validation techniques to assess its generalization ability.

**6.Hyperparameter Tuning:**

- Perform hyperparameter tuning for the selected model to optimize its performance.
- Explore techniques such as grid search or random search to find the optimal hyperparameters.

**7.Predictive Analysis and Visualization:**

- Apply the trained model to make predictions on future tourist demand based on the available features.
- Visualize the predicted tourist arrivals alongside the actual values to assess the model's accuracy and reliability.
- Analyze the results and interpret the insights gained from the predictive analysis.

**8.Documentation and Reporting:**

- Document the entire process, including data preprocessing steps, model selection criteria, and evaluation metrics.
- Prepare a report summarizing the lab program, including the problem statement, methodology, results, and conclusions.
- Present the findings in a clear and concise manner, highlighting the significance of predictive modeling in tourism management.

## MAIM205(A): Convolutional Neural Networks (CNNs)

Unit No.	Topics
Unit 1	<b>Introduction to Convolutional Neural Networks (CNNs):-</b> Overview of deep learning and its applications in tourism management, Introduction to CNNs and their role in image processing and analysis, Basic architecture of CNNs: convolutional layers, pooling layers, and fully connected layers.
Unit 2	<b>Image Preprocessing for CNNs:-</b> Methods for preprocessing images in tourism management applications, Techniques for image resizing, normalization, and augmentation, Hands-on exercises using Python libraries like TensorFlow or PyTorch for image preprocessing.
Unit 3	<b>CNN Architectures and Training:-</b> Overview of popular CNN architectures such as LeNet, AlexNet, VGG, and ResNet, Training CNN models using backpropagation and gradient descent optimization, Techniques for model evaluation, performance metrics, and overfitting prevention.
Unit 4	<b>Transfer Learning and Fine-tuning:-</b> Introduction to transfer learning and its applications in tourism management, Fine-tuning pre-trained CNN models for specific tourism-related tasks, Hands-on implementation of transfer learning using pre-trained CNN models and real-world tourism datasets.
Unit 5	<b>Advanced Applications of CNNs in Tourism Management:-</b> Real-world applications of CNNs in tourism management, such as image-based tourism recommendation systems, tourist behavior analysis, and safety surveillance, Case studies and examples illustrating the use of CNNs in solving tourism-related problems, Ethical considerations and challenges in deploying CNNs in tourism contexts, including privacy concerns and bias mitigation strategies.

### ReferenceBooks:

Title: "Deep Learning for Computer Vision"

Author: Rajalingappaa Shanmugamani

Publisher: Packt Publishing

## MAIM205(B): Recurrent Neural Networks (RNNs)

Unit No.	Topics
Unit 1	<b>Introduction to Recurrent Neural Networks (RNNs):</b> -Overview of neural networks and their applications in tourism management, Introduction to RNNs and their architecture for sequential data processing, Basic concepts of RNNs: recurrent connections, hidden states, and time-series data representation.
Unit 2	<b>Long Short-Term Memory (LSTM) Networks:</b> - Introduction to LSTM networks and their advantages over traditional RNNs, Understanding the architecture of LSTM cells: input gate, forget gate, output gate, Applications of LSTM networks in tourism management, such as time-series forecasting and sequence modeling.
Unit 3	<b>Gated Recurrent Units (GRUs):</b> - Introduction to GRU networks and their architecture, Comparison between LSTM and GRU networks in terms of complexity and performance, Hands-on implementation of LSTM and GRU networks using Python and TensorFlow or PyTorch.
Unit 4	<b>Applications of RNNs in Tourism Management:</b> - Real-world applications of RNNs in tourism management, such as demand forecasting, customer behavior analysis, and natural language processing, Case studies and examples illustrating the use of RNNs in solving tourism-related problems Ethical considerations and challenges in deploying RNNs in tourism contexts, including privacy concerns and bias mitigation strategies.
Unit 5	<b>Advanced Topics in RNNs:</b> - Advanced techniques for improving the performance of RNNs, such as attention mechanisms and sequence-to-sequence models, Transfer learning and fine-tuning of pre-trained RNN models for tourism-specific tasks, Future directions and emerging trends in RNN research and applications in tourism management.

### Reference Books:

Title: "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow"

Author: Aurélien Géron

Publisher: O'Reilly Medi.

## M.Sc. (Artificial Intelligence and Machine Learning) Scheme of Examination

### III<sup>RD</sup> SEMESTER

- |                         |               |                               |      |
|-------------------------|---------------|-------------------------------|------|
| 1. Course Code          | : MAIM        | 5. Total Practical            | 3    |
| 2. Course Name          | : M.Sc. AI&ML | 6. Total Practical Marks      | 150  |
| 3. Total Theory Subject | : 5           | 7. Total Marks                | 400  |
| 4. Total Theory Marks   | : 250         | 8. Minimum Passing Percentage | : 36 |

Sub.Code	Subject Name	Theory									
		Paper					CCE		Practical		Total
		1st	2nd	3rd	Max.	Min.	Max	Min.	Max.	Min.	Max.
MAIM301-T	Advanced Machine Learning Techniques	42	0	0	42	15	8	3	50	18	50
MAIM302-T	Machine Learning in Finance	42	0	0	42	15	8	3	50	18	50
MAIM302-T	Machine Learning in Finance LAB	0	0	0	0	0	0	0	50	18	50
MAIM303-T	Mobile Application Security	42	0	0	42	15	8	3	0	0	50
MAIM303-T	Mobile Application Security LAB	0	0	0	0	0	0	0	50	18	50
MAIM304-T	Advanced java Programming	42	0	0	42	15	8	3	0	0	50
MAIM304-T	Advanced java Programming LAB	0	0	0	0	0	0	0	50	18	50
MAIM305-T(A)	Bayesian Networks	42	0	0	42	15	8	3	0	0	50
MAIM305-T(B)	Markov Networks	42	0	0	42	15	8	3	0	0	50

## MAIM-301 Advanced Machine Learning Techniques

Unit No.	Topics
Unit 1	<b>Introduction to Advanced Machine Learning Techniques:-</b> Overview of advanced machine learning techniques and their applications, Discussion on the importance of advanced techniques in solving complex problems and improving model performance, Introduction to key concepts such as ensemble learning, deep learning, and reinforcement learning.
Unit 2	<b>Ensemble Learning:-</b> Introduction to ensemble learning and its principles, Exploration of ensemble methods such as bagging, boosting, and stacking, Hands-on implementation of ensemble learning algorithms using Python libraries like scikit-learn.
Unit 3	<b>Deep Learning:-</b> Overview of deep learning architectures and techniques, Introduction to neural networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and deep belief networks (DBNs), Hands-on exercises to build and train deep learning models for various tasks using frameworks like TensorFlow or PyTorch.
Unit 4	<b>Reinforcement Learning:-</b> Introduction to reinforcement learning and its applications, Exploration of reinforcement learning algorithms such as Q-learning, policy gradients, and deep Q-networks (DQN), Hands-on implementation of reinforcement learning algorithms for solving control and decision-making problems.
Unit 5	<b>Advanced Topics in Machine Learning:-</b> Advanced techniques for model interpretation and explainability, Introduction to model distillation, adversarial learning, and generative adversarial networks (GANs), Exploration of cutting-edge research topics in machine learning and their potential applications.

### Reference Book:

Title: "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow"

Author: Aurélien Géron

Publisher: O'Reilly Media

## MAIM302: Machine Learning in Finance

Unit No.	Topics
Unit 1	<b>Introduction to Machine Learning in Finance:-</b> Overview of machine learning applications in finance. Introduction to financial data types and challenges. Ethical considerations and regulatory compliance in financial machine learning.
Unit 2	<b>Predictive Modeling in Finance:-</b> Regression and classification techniques for financial forecasting. Time series analysis and forecasting methods. Feature engineering for financial data.
Unit 3	<b>Risk Management and Portfolio Optimization:-</b> Value at Risk (VaR) and Conditional Value at Risk (CVaR) modeling. Portfolio optimization using machine learning techniques. Credit risk assessment and fraud detection.
Unit 4	<b>Algorithmic Trading Strategies:-</b> Overview of algorithmic trading and high-frequency trading. Market microstructure and order book modeling. Reinforcement learning for trading strategy optimization.
Unit 5	<b>Deep Learning Applications in Finance:-</b> Introduction to deep learning architectures for finance. Deep learning models for time series forecasting and risk management. Natural Language Processing (NLP) for sentiment analysis and news-based trading strategies.

### Reference Book:

"Machine Learning for Asset Managers" by Marcos López de Prado. "Advances in Financial Machine Learning" by Marcos López de Prado. "Machine Learning in Finance: From Theory to Practice" by Matthew F. Dixon, Igor Halperin, and Paul Bilokon. "Algorithmic Trading: Winning Strategies and Their Rationale" by Ernest P. Chan. "Deep Learning in Finance: Applications of Deep Neural Networks to Trading, Investment, and Risk Management" by Guido Germano.

## **Machine Learning in Finance LAB-**

1. Predictive Modeling for Stock Price Forecasting:
2. Use historical stock price data to build predictive models for forecasting future prices.
3. Credit Risk Assessment:
4. Develop models to assess credit risk for individuals or businesses based on financial data.
5. Fraud Detection:
6. Implement machine learning algorithms to detect fraudulent transactions or activities in financial systems.
7. Algorithmic Trading Strategies:
8. Design and backtest algorithmic trading strategies using machine learning techniques to make trading decisions.
9. Portfolio Optimization:
10. Build models to optimize investment portfolios based on risk tolerance, return objectives, and market conditions.
11. Market Sentiment Analysis:
12. Analyze news articles, social media data, and other textual sources to gauge market sentiment and its impact on asset prices.
13. High-Frequency Trading (HFT):
14. Develop algorithms and infrastructure to execute high-frequency trades based on real-time market data.
15. Option Pricing and Hedging:
16. Use machine learning to price options and develop hedging strategies to mitigate risk in options trading.
17. Customer Segmentation and Lifetime Value Prediction:
18. Segment customers based on their financial behavior and predict their lifetime value to tailor marketing strategies.
19. Robo-Advisors:
20. Build automated investment advisory platforms that leverage machine learning to provide personalized investment recommendations to clients.



## MAIM303: Mobile Application Security

Unit No.	Topics
Unit 1	<b>Introduction to Mobile Application Security:-</b> Overview of mobile application security threats and vulnerabilities. Common attack vectors targeting mobile apps. Principles of secure mobile application development.
Unit 2	<b>Secure Coding Practices:-</b> Best practices for secure coding in mobile app development. Secure data storage and transmission. Input validation and sanitization techniques.
Unit 3	<b>Authentication and Authorization:-</b> Authentication mechanisms for mobile apps (e.g., passwords, biometrics, multi-factor authentication). Authorization frameworks and access control mechanisms. Implementing secure session management.
Unit 4	<b>Secure Network Communication:-</b> Securing network communications in mobile apps using HTTPS and TLS. Mitigating common network-based attacks (e.g., man-in-the-middle attacks, SSL stripping). Implementing certificate pinning for enhanced security.
Unit 5	<b>Mobile App Reverse Engineering and Penetration Testing:-</b> Techniques for reverse engineering mobile apps. Performing penetration testing to identify security vulnerabilities. Mitigating reverse engineering and tampering attempts.

### Reference Book:

"Mobile Application Security: Protecting Against Hacks, Threats, and Breaches" by Himanshu Dwivedi, Chris Clark, and David Thiel. "The Mobile Application Hacker's Handbook" by Dominic Chell, Ollie Whitehouse, Shaun Colley, and Tyrone Erasmus. "OAuth 2.0: Getting Started in Web-API Security" by Matthias Biehl. "SSL and TLS: Designing and Building Secure Systems" by Eric Rescorla. "The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy" by Patrick Egebreton.

## **Mobile Application Security LAB:-**

1. Secure Code Review:
2. Perform a code review of a mobile application to identify security vulnerabilities and recommend mitigations.
3. Data Encryption Implementation:
4. Implement data encryption techniques such as AES for securing sensitive data stored on a mobile device.
5. Secure Authentication Mechanisms:
6. Implement secure authentication mechanisms like OAuth 2.0 or JWT to protect user credentials and sessions.
7. Secure Data Storage:
8. Implement secure storage mechanisms (e.g., Keychain on iOS, Keystore on Android) to protect sensitive data stored locally on the device.
9. Transport Layer Security (TLS) Implementation:
10. Configure and enforce TLS/SSL protocols to secure network communications between the mobile app and server.
11. Secure API Integration:
12. Integrate APIs securely, implement proper authentication, authorization, and input validation to prevent common vulnerabilities such as injection attacks.
13. Mobile App Penetration Testing:
14. Conduct penetration testing on a mobile application using tools like OWASP ZAP or Burp Suite to identify security flaws and weaknesses.
15. Root/Jailbreak Detection and Mitigation:
16. Implement techniques to detect if a device is rooted (Android) or jailbroken (iOS) and apply appropriate security measures to mitigate risks.
17. Secure Offline Storage:
18. Implement secure offline storage mechanisms to protect data integrity and confidentiality when the device is not connected to the internet.
19. Secure Push Notification Implementation:
20. Implement push notifications securely, ensuring that sensitive information is not leaked and that notifications are delivered securely.

### **MAIM304-T Advanced java Programming**

Unit No.	Topics
Unit 1	<b>JAVABASICSREVIEW</b> Java streaming-Networking-Event handling - Multithreading – Byte code Interpretation - Customizing application - Data Structures - Collection classes.
Unit 2	<b>DISTRIBUTED COMPUTING</b> Custom sockets - Remote Method Invocation - Activation - Object serialization -Distributed garbage collection - RMI - IIOP - Interface definition language – CORBA-JINI overview.
Unit 3	<b>JAVA BEANS AND SWING</b> Bean concepts - Events in bean box -Bean customization - Persistence - Application - deployment using swing - Advanced swing techniques - JAR file handling.
Unit 4	<b>JAVAENTERPRISEAPPLICATIONS</b> -JNI-Servlets-Java Server Pages - JDBC - Session beans - Entity beans - Programming and deploying enterprise Java Beans - Java transactions. <b>RELATEDJAVATECHNIQUES.</b>
Unit 5	<b>Graphics</b> Java Media Frame work - 3D graphics - Internationalization - Case study - Deploying n-tierapplication, E- commerce applications.

## REFERENCES:-

1. Deitel&Deitel, "JavaHowtoprogram", PrenticeHall, 4th Edition, 2000.
2. GaryCornellandCayS.Horstmann, "CoreJavaVol1andVol2", Sun Microsystems Press, 1999.
3. StephenAsbury, ScottR. Weiner, Wiley, "DevelopingJavaEnterpriseApplications", 1998.

## ADVANCED JAVA PROGRAMMING LAB:-

### 1. Multithreading and Concurrency:

- Implement multithreaded programs using Java's Thread class and Runnable interface.
- Explore synchronization mechanisms such as synchronized blocks and methods to ensure thread safety.
- Practical exercises on solving concurrency issues like race conditions and deadlocks.

### 2. Networking and Socket Programming:

- Develop client-server applications using Java's Socket API for network communication.
- Implement protocols like TCP and UDP for reliable and connectionless communication.
- Practical exercises on creating chat applications, file transfer utilities, or network-based games.

### 3. Java Database Connectivity (JDBC):

- Integrate Java applications with relational databases using JDBC.
- Perform database operations like querying, updating, and executing stored procedures.
- Practical exercises on connecting to a database, executing SQL queries, and handling result sets.

### 4. Java EE Technologies:

- Explore Java Enterprise Edition (Java EE) technologies like Servlets, JavaServer Pages (JSP), and JavaServer Faces (JSF).
- Develop web applications using Servlets and JSP for server-side processing and dynamic content generation.
- Practical exercises on building web forms, handling user input, and displaying dynamic content.

### 5. Enterprise Application Development with Spring Framework:

- Introduction to the Spring Framework for building enterprise Java applications.
- Implementing dependency injection, aspect-oriented programming (AOP), and inversion of control (IoC) with Spring.

- Practical exercises on developing Spring-based web applications, RESTful services, and data access layers.

#### 6. Web Services Development with JAX-RS and JAX-WS:

- Implement RESTful and SOAP web services using Java API for RESTful Web Services (JAX-RS) and Java API for XML Web Services (JAX-WS).
- Practical exercises on creating and consuming web services for inter-application communication.

#### 7. JavaFX GUI Development:

- Build modern and rich graphical user interfaces (GUIs) using JavaFX.
- Create interactive UI components, layouts, and animations with JavaFX.
- Practical exercises on developing desktop applications with advanced UI features using JavaFX.

#### 8. Integration Testing and Test-Driven Development (TDD):

- Implement unit tests and integration tests for Java applications using frameworks like JUnit and Mockito.
- Practice test-driven development (TDD) by writing tests before implementing code.
- Practical exercises on writing test cases, mocking dependencies, and ensuring code quality through automated testing.

#### 9. Security and Authentication in Java Applications:

- Explore security features and best practices for Java applications, including authentication, authorization, and encryption.
- Implement authentication mechanisms like username/password, OAuth, or JWT (JSON Web Tokens).
- Practical exercises on securing web applications, protecting sensitive data, and preventing common security vulnerabilities.

#### 10. Asynchronous Programming with CompletableFuture and Reactive Streams:

- Learn asynchronous programming techniques using CompletableFuture and Reactive Streams in Java.
- Implement non-blocking I/O operations and reactive programming patterns for handling asynchronous events.
- Practical exercises on developing responsive and scalable applications using asynchronous programming paradigms.

## Class Assignments:

1. Write a program that produces the following output:  
Hello  
World!  
"It's been nice knowing you "  
Goodbye world!
2. State the order of evaluation of the operations in each of the following Java statements and implement them to show the value of x after each statement.  
 $x = 7 + 3 * 6 / 2 - 1;$   
 $x = 2 \% 2 + 2 * 2 - 2 / 2;$   
 $x = ( 3 * 9 * ( 3 + ( 9 * 3 / (3) ) ) );$
3. Write an application that declares 5 integers, determines and prints the largest and smallest in the group.
  4. Write an application that takes 3 parameters as sides of triangle and calculate area of triangle.
5. Write an application that declares two integers, determines whether the first is a multiple of the second and print the result. [Hint: Use the remainder operator.]
  6. Write a program to find all prime numbers between 100 to 1000.
  7. Write a program to check whether the given number is palindrome or not.
  8. Write an application that evaluates the factorial of the integers from 1 to 5.
9. Write a program that accepts an integer from user and check whether the number is Fibonacci number or not.
10. Read a positive integer value, and compute the following sequence: If the number is even, halve it; if it's odd, multiply by 3 and add 1. Repeat this process until the value is 1, printing out each value. Finally print out how many of these operations you performed.  
Typical output might be:  
Initial value is 9  
Next value is 28  
Next value is 14  
Next value is 7  
Next value is 22  
Next value is 11  
Next value is 34  
Next value is 17  
Next value is 52  
Next value is 26  
Next value is 13  
Next value is 40  
Next value is 20  
Next value is 10  
Next value is 5  
Next value is 16  
Next value is 8  
Next value is 4  
Next value is 2  
Final value 1, number of steps 19  
If the input value is less than 1, print a message containing the word Error and perform an `exit(0)`;
11. Write a program which will read an integer value for a base, then read a positive integer written to that base and print its value. Read the second integer a character at a time; skip over any leading nonvalid (i.e. not a digit between zero and ``base-1'') characters, then read valid characters until an invalid one is encountered.  
Input Output

=====

10 1234 1234

8 77 63 (the value of 77 in base 8, octal)

2 1111 15 (the value of 1111 in base 2, binary)

The base will be less than or equal to 10.

12. Write an application that uses *String* method *compareTo* to compare two strings defined by the user.
13. Write an application that uses *String* method *equals* and *equalsIgnoreCase* to tests any two string objects for equality.
14. Write an application that uses *String* method *indexOf* to determine the total number of occurrences of any given alphabet in a defined text.
15. Write an application that uses *String* method *concat* to concatenate two defined strings.
16. Write an application that finds the length of a given string.
17. Write an application that uses *String* method *charAt* to reverse the string.
18. Write an application that finds the substring from any given string using *substring* method and *startsWith* & *endsWith* methods.
19. Write an application that changes any given string with uppercase letters, displays it , changes it back to lowercase letters and displays it.
20. Create a class called *Employee* that includes three pieces of information as instance variables – a first name (type *String*), a last name (type *String*) and a monthly salary (double)
  
21. Write an application that uses *String* method *compareTo* to compare two strings defined by the user.
22. Write an application that uses *String* method *equals* and *equalsIgnoreCase* to tests any two string objects for equality.
23. Write an application that uses *String* method *indexOf* to determine the total number of occurrences of any given alphabet in a defined text.
24. Write an application that uses *String* method *concat* to concatenate two defined strings.
25. Write an application that finds the length of a given string.
26. Write an application that uses *String* method *charAt* to reverse the string.
27. Write an application that finds the substring from any given string using *substring* method and *startsWith* & *endsWith* methods.
28. Write an application that changes any given string with uppercase letters, displays it , changes it back to lowercase letters and displays it.
29. Create a class called *Employee* that includes three pieces of information as instance variables – a first name (type *String*), a last name (type *String*) and a monthly salary (double)  
  
write application that take a series of integers in pair(e.g. (a,b)) and find the following
  - a. Is first one is greater
  - b. Is both are even
  - c. They are relatively prime or not
30. Write an application that uses *String* method *compareTo* to compare two strings defined by the user.
31. Write an application that uses *String* method *equals* and *equalsIgnoreCase* to tests any two string objects for equality.
32. Write an application that uses *String* method *indexOf* to determine the total number of occurrences of any given alphabet in a defined text.

33. Write an application that uses *String* method *concat* to concatenate two defined strings.
34. Write an application that finds the length of a given string.
35. Write an application that uses *String* method *charAt* to reverse the string.
36. Write an application that finds the substring from any given string using *substring* method and *startsWith* & *endsWith* methods.
37. Write an application that changes any given string with uppercase letters, displays it , changes it back to lowercase letters and displays it.
38. Create a class called *Employee* that includes three pieces of information as instance variables – a first name (type *String*), a last name (type *String*) and a monthly salary (double)
39. Create a constructor in above class to initialize the three instance variables. Provide a get method for each instance variable.
40. Write a test application named *EmployeeTest* that demonstrates class *Employee*'s capabilities. Create two employee objects and display each object's yearly salary.
41. Give each employee a 10% raise and display each *Employee*'s yearly salary again.
42. Create a class *Account* with an instance variable *balance* (double). It should contain a constructor that initializes the *balance*, ensure that the initial balance is greater than 0.0



## MAIM-305(A) - Bayesian Networks

Unit No.	Topics
Unit 1	<b>Introduction to Probabilistic Graphical Models:-</b> Overview of probabilistic graphical models (PGMs) and their applications, Introduction to Bayesian Networks (BNs) as a type of PGM, Basic concepts of probability theory and graphical representations of BNs.
Unit 2	<b>Bayesian Network Structure Learning:-</b> Methods for learning the structure of Bayesian Networks from data, Constraint-based algorithms such as PC (Peter & Clark) and FCI (Fast Causal Inference), Score-based algorithms like BIC (Bayesian Information Criterion) and MDL (Minimum Description Length).
Unit 3	<b>Bayesian Network Parameter Learning:-</b> Techniques for estimating the parameters of Bayesian Networks from data, Maximum Likelihood Estimation (MLE) and Maximum A Posteriori (MAP) estimation, Bayesian estimation methods such as Bayesian Parameter Estimation (BPE) and Bayesian Structure Learning (BSL).
Unit 4	<b>Inference and Reasoning in Bayesian Networks:-</b> Methods for performing inference and reasoning in Bayesian Networks, Exact inference algorithms like Variable Elimination and Junction Tree algorithms, Approximate inference techniques including Markov Chain Monte Carlo (MCMC) and Variational Inference.
Unit 5	<b>Advanced Topics in Bayesian Networks:-</b> Exploration of advanced topics and extensions of Bayesian Networks, Dynamic Bayesian Networks (DBNs) for modeling temporal dependencies, Bayesian Networks for decision making under uncertainty and influence diagrams, Causal Inference using Bayesian Networks and counterfactual reasoning.

### REFERENCES:

Title: "Probabilistic Graphical Models: Principles and Techniques"  
Author: Daphne Koller and Nir Friedman  
Publisher: The MIT Press

## **Bayesian Networks lab:-**

### **1. Introduction to Healthcare Decision Support with Bayesian Networks:**

- Overview of decision support systems in healthcare and the role of Bayesian Networks.
- Discussion on the advantages of Bayesian Networks in handling uncertainty and modeling complex dependencies in medical data.

### **2. Data Acquisition and Preprocessing:**

- Obtain a dataset containing medical records, patient symptoms, test results, and diagnoses.
- Preprocess the medical data by handling missing values, encoding categorical variables, and normalizing numerical features.

### **3. Construction of Bayesian Network Models:**

- Identify relevant medical variables and dependencies to construct a Bayesian Network model.
- Utilize domain knowledge and expert input to define the structure of the Bayesian Network.
- Implement the Bayesian Network using libraries like PyMC3 or Bayesian Network Toolbox.

### **4. Parameter Learning and Model Validation:**

- Estimate the parameters of the Bayesian Network model from the medical dataset.
- Perform model validation and evaluation using techniques such as cross-validation and likelihood-based metrics.

### **5. Inference and Diagnosis:**

- Apply the trained Bayesian Network model to perform inference and diagnose medical conditions.
- Demonstrate the use of probabilistic reasoning to calculate the probability of different diagnoses given observed symptoms and test results.

### **6. Treatment Planning and Decision Making:**

- Extend the Bayesian Network model to incorporate treatment options and their effects on patient outcomes.
- Use decision theoretic principles to recommend optimal treatment plans based on the Bayesian Network's predictions.

### **7. Case Studies and Real-world Applications:**

- Present case studies and examples of Bayesian Networks being used in healthcare decision support systems.
- Discuss real-world applications of Bayesian Networks in medical diagnosis, treatment planning, and patient management.

### **8. Ethical Considerations and Future Directions:**

- Explore ethical considerations surrounding the use of Bayesian Networks in healthcare decision support.
- Discuss challenges, limitations, and future directions for incorporating Bayesian Networks into clinical practice.

## MAIM305(B)- Markov Networks

Unit No.	Topics
Unit 1	<b>Introduction to Probabilistic Graphical Models:-</b> Overview of probabilistic graphical models (PGMs), Types of PGMs: Bayesian networks, Markov networks, Basic concepts: nodes, edges, conditional independence.
Unit 2	<b>Markov Networks Fundamentals:-</b> Definition and representation of Markov networks, Factors and factorization in Markov networks, Markov blanket and local structures, Examples of Markov networks in real-world applications.
Unit 3	<b>Inference in Markov Networks:-</b> Exact inference methods: variable elimination, belief propagation, Approximate inference methods: Gibbs sampling, mean field approximation, Junction tree algorithm and its applications, Comparison of inference methods in terms of complexity and accuracy.
Unit 4	<b>Learning in Markov Networks:-</b> Parameter learning: maximum likelihood estimation, maximum a posteriori estimation, Structure learning: score-based methods, constraint-based methods, hybrid methods, Model selection and evaluation techniques, Incorporating prior knowledge and expert opinions.
Unit 5	<b>Advanced Topics:-</b> Learning with hidden variables, Dynamic Markov networks and temporal modeling, Distributed and parallel inference algorithms, Applications of Markov networks in computer vision, natural language processing, and bioinformatics.

### Reference Book:

"Probabilistic Graphical Models: Principles and Techniques" by Daphne Koller and Nir Friedman

## Markov Networks Lab:-

- 1.Implementing Gibbs Sampling: Students can implement Gibbs sampling algorithm to perform approximate inference in Markov networks. They can experiment with different convergence criteria and explore its effectiveness on various network structures.
- 2.Parameter Estimation with Expectation-Maximization (EM): Students can learn how to use the EM algorithm to estimate parameters of a Markov network when some variables are hidden. They can apply this technique to real-world datasets and analyze its convergence behavior.
- 3.Learning the Structure of Markov Networks: Students can explore different methods for learning the structure of Markov networks from data, such as score-based methods (e.g., Bayesian Information Criterion), constraint-based methods (e.g., PC algorithm), and hybrid methods. They can compare the performance of these methods on synthetic and real-world datasets.
- 4.Dynamic Bayesian Networks: Extend the study to dynamic Markov networks, where variables change over time. Students can learn about models like Hidden Markov Models (HMMs) and Dynamic Bayesian Networks (DBNs), and apply them to time-series data in various domains like finance, speech recognition, or healthcare.
- 5.Graphical Models for Image Segmentation: Explore the application of Markov networks for image segmentation tasks. Students can implement algorithms like Markov Random Fields (MRFs) for image labeling and compare them with other segmentation techniques like convolutional neural networks (CNNs).
- 6.Natural Language Processing with Markov Networks: Investigate the use of Markov networks for tasks such as part-of-speech tagging, named entity recognition, or syntactic parsing in natural language processing. Students can build models using annotated corpora and evaluate their performance against baseline methods.
- 7.Gene Regulatory Network Inference: Apply Markov networks to infer gene regulatory networks from gene expression data. Students can explore different methods for modeling gene interactions, incorporating prior knowledge, and assessing the reliability of inferred networks using biological validation.
- 8.Social Network Analysis: Use Markov networks to model social networks and analyze properties such as community structure, influence propagation, and link prediction. Students can study algorithms like belief propagation and message passing for inference on large-scale networks.
- 9.Anomaly Detection in Network Traffic: Explore the use of Markov networks for anomaly detection in network traffic. Students can build models to capture normal network behavior and detect deviations that may indicate security breaches or system failures.
- 10.Bayesian Network Classifier: Implement a classification model based on Bayesian networks and compare its performance with other classifiers like decision trees, support vector machines, or deep learning models. Students can experiment with different network structures and learning algorithms for parameter estimation.

## M.Sc. (Artificial Intelligence and Machine Learning ) Scheme of Examination

### IV<sup>TH</sup> SEMESTER

- |                              |                                    |
|------------------------------|------------------------------------|
| 1. Course Code : MAIM        | 5. Total Practical : 4             |
| 2. Course Name : M.Sc. AI&ML | 6. Total Practical Marks : 200     |
| 3. Total Theory Subject : 4  | 7. Total Marks : 400               |
| 4. Total Theory Marks : 200  | 8. Minimum Passing Percentage : 36 |

Sub.Code	Subject Name	Theory									
		Paper					CCE		Practical		Total
		1st	2nd	3rd	Max.	Min.	Max	Min.	Max.	Min.	Max.
MAIM401-T	Privacy and Security Concerns	42	0	0	42	15	8	3	50	18	50
MAIM401-P	Privacy and Security Concerns LAB	0	0	0	0	0	0	0	50	18	50
MAIM402-T	Data Manipulation & Preprocessing	42	0	0	42	15	8	3	0	0	50
MAIM401-P	Data Manipulation & Preprocessing LAB	0	0	0	0	0	0	0	50	18	50
MAIM401-T	Programming in python	42	0	0	42	15	8	3	0	0	50
MAIM401-P	Programming in Python LAB	0	0	0	0	0	0	0	50	18	50
MAIM401-T	Special Topics in AI and ML	42	0	0	42	15	8	3	0	0	50
MCYS405-P	Major Project	0	0	0	0	0	0	0	100	50	100

## MSCAIM-401 Privacy and Security Concerns

Unit No	TOPIC
Unit 1:	<b>Introduction to Privacy and Security:-</b> Overview of privacy and security concerns in technology, Legal and ethical frameworks: GDPR, HIPAA, etc. Threat landscape: common security threats and attack vectors, Principles of data protection and risk management.
Unit 2	<b>Data Privacy Regulations and Compliance:-</b> Deep dive into data privacy regulations: GDPR, CCPA, etc. Compliance requirements for handling personal data, Data protection impact assessments (DPIAs) and privacy by design principles, Case studies of data breaches and regulatory fines.
Unit 3	<b>Cryptography and Data Encryption:-</b> Fundamentals of cryptography: symmetric and asymmetric encryption, hashing, digital signatures, Encryption techniques for data protection in transit and at rest, Secure communication protocols: TLS/SSL, SSH, etc. Practical implementation of encryption in applications and systems.
Unit 4	<b>Secure Software Development Practices:-</b> Secure software development lifecycle (SDLC) methodologies, Common vulnerabilities and secure coding practices: OWASP Top 10, Code review and static analysis tools for identifying security flaws, Secure deployment and configuration management techniques.
Unit 5	<b>Emerging Technologies and Privacy Challenges:-</b> Privacy implications of emerging technologies: AI, IoT, blockchain, etc. Privacy-preserving techniques: homomorphic encryption, differential privacy, federated learning, Ethical considerations in data collection, processing, and sharing Future trends and directions in privacy and security research

### ReferencesBook:

"Security Engineering: A Guide to Building Dependable Distributed Systems" by Ross J. Anderson

## **Privacy and Security Concerns Lab:-**

### **1.Data Encryption Tool:**

•Develop a program that encrypts sensitive data using symmetric or asymmetric encryption algorithms such as AES or RSA. Include features for key generation, encryption, decryption, and secure storage of keys.

### **2.Privacy Policy Generator:**

•Create a tool that generates privacy policies for websites or applications based on user inputs regarding data collection, processing, and sharing practices. Ensure compliance with relevant regulations like GDPR or CCPA.

### **3.Vulnerability Scanner:**

•Build a program that scans web applications or networks for common security vulnerabilities such as SQL injection, cross-site scripting (XSS), or insecure configurations. Provide reports with recommendations for remediation.

### **4.Secure File Transfer Application:**

•Develop a secure file transfer application that encrypts files before transmission and ensures integrity during transit using secure communication protocols like HTTPS or SFTP. Implement authentication mechanisms and access controls.

### **5.Password Manager:**

•Create a password manager application that securely stores and manages user credentials using strong encryption techniques. Include features for generating strong passwords, autofill capabilities, and multi-factor authentication.

### **6.Privacy-preserving Data Analytics Tool:**

•Design a tool that performs data analytics on sensitive datasets while preserving privacy using techniques like differential privacy or secure multi-party computation. Ensure that individual privacy is protected while deriving useful insights from the data.

### **7.Security Incident Response Dashboard:**

•Build a dashboard application for security incident response teams to track and manage security incidents in real-time. Include features for incident classification, prioritization, assignment, and resolution tracking.

### **8.Anonymous Communication Platform:**

•Develop a platform that enables anonymous communication between users while preserving privacy and anonymity. Implement features such as end-to-end encryption, message authentication, and forward secrecy to ensure secure communication.

### **9.Blockchain-based Data Auditing Tool:**

•Create a tool that leverages blockchain technology to provide tamper-proof audit trails for data transactions. Enable users to track data provenance, access history, and modifications while ensuring data integrity and authenticity.

### **10.Privacy-preserving Machine Learning Framework:**

•Build a framework that allows organizations to train machine learning models on sensitive data without compromising individual privacy. Implement techniques such as federated learning, homomorphic encryption, or differential privacy to protect privacy during model training and inference.

## MAIM402-T Data Manipulation & Preprocessing

Unit No	TOPIC
Unit 1:	<b>Introduction to Data Manipulation:-</b> Overview of data manipulation and preprocessing techniques. Introduction to data cleaning, transformation, and feature engineering. Importance of data quality and its impact on downstream analysis.
Unit 2	<b>Data Cleaning Techniques:-</b> Handling missing values: imputation methods, deletion strategies. Outlier detection and treatment techniques. Data normalization and standardization.
Unit 3	<b>Feature Engineering:-</b> Feature selection techniques: filtering, wrapper methods, embedded methods. Encoding categorical variables: one-hot encoding, label encoding, target encoding. Creating new features: binning, scaling, interaction terms.
Unit 4	<b>Dimensionality Reduction:-</b> Principal Component Analysis (PCA) and its applications. Singular Value Decomposition (SVD) and its role in dimensionality reduction. t-Distributed Stochastic Neighbor Embedding (t-SNE) for visualizing high-dimensional data.
Unit 5	<b>Time Series Preprocessing:-</b> Resampling and interpolation methods for irregular time series data. Handling seasonality and trends in time series. Feature extraction from time series data.

### Additional Book Recommendations:

"Pandas Cookbook" by Theodore Petrou: Offers a collection of practical recipes for performing various data manipulation tasks using pandas.

"Hands-On Data Analysis with Pandas" by Stefanie Molin: Provides hands-on exercises and projects to enhance your skills in data manipulation and analysis with pandas.

"Data Wrangling with Python" by Jacqueline Kazil and Katharine Jarmul: Focuses on data cleaning, transformation, and preparation techniques using Python libraries like pandas, NumPy, and more.



## **Data Manipulation & Preprocessing LAB:-**

1. **Data Cleaning Techniques:** Cover techniques such as handling missing values, outliers, and inconsistencies in the data. This includes imputation methods, outlier detection, and data validation.
2. **Data Normalization and Standardization:** Discuss methods to scale numerical features to a standard range, such as Min-Max scaling or z-score standardization. Explain when to use each method based on the characteristics of the data.
3. **Encoding Categorical Variables:** Explore different methods for encoding categorical variables, such as one-hot encoding, label encoding, and target encoding. Discuss the advantages and limitations of each approach.
4. **Feature Engineering:** Introduce techniques to create new features from existing ones to improve model performance. This could include techniques like polynomial features, interaction features, or domain-specific feature engineering.
5. **Dimensionality Reduction:** Explain methods to reduce the number of features in a dataset while preserving important information. Techniques such as Principal Component Analysis (PCA), Singular Value Decomposition (SVD), and t-Distributed Stochastic Neighbor Embedding (t-SNE) can be covered.
6. **Handling Date and Time Data:** Discuss how to preprocess date and time variables, including techniques for extracting features such as day of the week, month, or time of day. Cover libraries like pandas in Python for efficient handling of date and time data.
7. **Text Preprocessing:** Explore techniques for preprocessing text data, including tokenization, stemming, lemmatization, and stop word removal. Discuss how to convert text data into a format suitable for machine learning models.
8. **Handling Imbalanced Data:** Address techniques for dealing with imbalanced datasets, where one class is significantly more prevalent than others. Methods such as resampling (oversampling and undersampling), synthetic data generation, and cost-sensitive learning can be discussed.
9. **Data Transformation Pipelines:** Explain how to build data transformation pipelines to automate the preprocessing steps and ensure consistency across training and testing datasets. Libraries like scikit-learn provide tools for constructing such pipelines.
10. **Data Augmentation:** Discuss techniques to increase the size of the training dataset by generating synthetic data. This is particularly useful in domains such as computer vision and natural language processing. Methods like rotation, translation, flipping (for images), or adding noise (for text) can be covered.

## MAIM-403 Programming in python

Unit No	TOPIC
Unit 1:	<b>Introduction to Python Programming:-</b> Overview of Python programming language: history, features, and applications, Setting up the Python development environment: installing Python, IDE setup, Basics of Python syntax: variables, data types, operators, and expressions, Control flow: conditional statements (if-else), loops (for, while), and flow control statements.
Unit 2	<b>Data Structures and Functions:-</b> Working with built-in data structures in Python: lists, tuples, dictionaries, sets, Writing and calling functions in Python: defining functions, parameters, return values, Function scope and namespace: global vs. local variables, scope resolution, Advanced function concepts: lambda functions, higher-order functions, decorators.
Unit 3	<b>Object-Oriented Programming (OOP) in Python:-</b> Introduction to object-oriented programming concepts in Python, Defining and using classes and objects in Python, Inheritance and polymorphism: extending classes, method overriding, Encapsulation and access control: public, private, protected members.
Unit 4	<b>File Handling and Modules:-</b> Reading from and writing to files in Python: file objects, file modes, Working with text and binary files: reading, writing, appending, Organizing code into modules and packages: importing modules, creating packages, Exploring standard library modules and third-party libraries.
Unit 5	<b>Advanced Topics in Python Programming:-</b> Python iterators and generators: lazy evaluation, generator expressions, Error handling and exceptions: try-except blocks, raising exceptions, Regular expressions: pattern matching, searching, and replacing text, Introduction to multithreading, multiprocessing, and asynchronous programming in Python.

### References:

"Python Crash Course" by Eric Matthes

## **Programming in python Lab:-**

### **1.Basic Syntax and Data Types:**

•Write Python programs to practice basic syntax, including variables, data types, operators, and control structures.

### **2.Working with Lists and Dictionaries:**

•Create Python programs to manipulate lists and dictionaries, including operations like indexing, slicing, adding, removing, and iterating over elements.

### **3.Functions and Modules Practice:**

•Develop Python functions to perform various tasks and organize them into modules. Test importing and using functions from different modules.

### **4.File Handling and Text Processing:**

•Write Python programs to read from and write to text files, perform text processing tasks such as searching, replacing, and parsing text data.

### **5.Object-Oriented Programming (OOP) Implementation:**

•Implement classes and objects in Python to demonstrate concepts such as inheritance, polymorphism, encapsulation, and method overriding.

### **6.Exception Handling Practice:**

•Write Python programs to handle various types of exceptions using try-except blocks, and practice raising custom exceptions.

### **7.Regular Expressions and Pattern Matching:**

•Develop Python programs to search, extract, and manipulate text patterns using regular expressions.

### **8.Database Connectivity with SQLite:**

•Create Python programs to connect to an SQLite database, perform CRUD operations, and handle transactions using the sqlite3 module.

### **9.Web Scraping and Data Extraction:**

•Build Python scripts to extract data from websites using libraries like BeautifulSoup or Scrapy, and store the data in structured formats.

### **10.Mini-Project Development:**

•Work on a small-scale Python project that integrates multiple concepts learned throughout the course, such as file handling, OOP principles, exception handling, and external library usage.

## MAIM-404 Special Topics in AI and ML

Unit No	TOPIC
Unit 1:	<b>Introduction to Special Topics in AI and ML:-</b> Overview of advanced topics in artificial intelligence (AI) and machine learning (ML), Importance and applications of special topics in various domains, Challenges and recent advancements in AI and ML research, Ethical considerations and societal impact of advanced AI and ML technologies.
Unit 2	<b>Deep Reinforcement Learning:-</b> Fundamentals of reinforcement learning (RL): Markov decision processes, policy, value functions, Deep Q-Learning: Q-networks, experience replay, target networks, Policy gradients methods: REINFORCE algorithm, actor-critic architectures, Applications of deep reinforcement learning in gaming, robotics, and autonomous systems.
Unit 3	<b>Generative Adversarial Networks (GANs):-</b> Introduction to GANs: architecture, training procedure, loss functions, Variants of GANs: conditional GANs, Wasserstein GANs, cycle-consistent GANs, Applications of GANs in image generation, style transfer, data augmentation, and anomaly detection, Ethical considerations and potential risks associated with GANs.
Unit 4	<b>Natural Language Processing (NLP):-</b> Overview of NLP tasks: text classification, named entity recognition, sentiment analysis, machine translation, Word embeddings and language models: Word2Vec, GloVe, BERT, Sequence-to-sequence models: attention mechanisms, encoder-decoder architectures, Applications of NLP in conversational agents, document summarization, and language generation.
Unit 5	<b>Bayesian Machine Learning:-</b> Introduction to Bayesian inference: Bayes' theorem, prior, likelihood, posterior, Bayesian modeling and probabilistic programming: Bayesian networks, Markov chain Monte Carlo (MCMC) methods, Variational inference: variational autoencoders (VAEs), stochastic variational inference, Applications of Bayesian machine learning in uncertainty estimation, Bayesian optimization, and probabilistic reasoning.

### References:

"Deep Reinforcement Learning Hands-On" by Maxim Lapan

MARKU