Artificial Intelligence Course Outline

Prerequisites for Machine Learning and Artificial Intelligence

1. **Mathematics**
   - Linear Algebra
     - Vectors, Matrices, and Operations
     - Eigenvalues and Eigenvectors
     - Matrix Decomposition (SVD, PCA)
   - Probability and Statistics
     - Basics of Probability Theory (Conditional Probability, Bayes' Theorem)
     - Probability Distributions (Normal, Binomial, Poisson)
     - Hypothesis Testing
     - Descriptive and Inferential Statistics
   - Calculus
     - Derivatives and Partial Derivatives
     - Gradient and Hessian
     - Chain Rule in Backpropagation
   - Optimization
     - Convex and Non-Convex Optimization
     - Gradient Descent Variants
2. **Programming Skills**
   - Python Basics (Loops, Functions, OOPs)
   - Libraries: NumPy, Pandas, Matplotlib, Seaborn
3. **Data Handling**
   - Loading, Cleaning, and Transforming Datasets
   - Working with Structured and Unstructured Data

Machine Learning

1. **Introduction to Machine Learning**
   - What is Machine Learning?
   - Applications of Machine Learning
   - Types of Machine Learning (Supervised, Unsupervised, Reinforcement)
   - Machine Learning Workflow
2. **Data Preprocessing**

- Data Cleaning (handling missing values, outliers)
- Feature Scaling (standardization, normalization)
- Feature Selection and Extraction
- Encoding Categorical Data

3. **Exploratory Data Analysis (EDA)**
   - Descriptive Statistics
   - Data Visualization (Matplotlib, Seaborn)
   - Identifying Patterns and Trends
4. **Introduction to Supervised Learning**
   - Key Concepts (Features, Labels, Training vs Testing)
   - Evaluation Metrics (Accuracy, Precision, Recall, F1 Score)
5. **Simple Linear Regression**
   - Concept and Applications
   - Least Squares Method
   - Evaluation Metrics (MSE, RMSE, R^2 Score)
6. **Multiple Linear Regression**
   - Multivariate Analysis
   - Assumptions and Diagnostics
7. **Gradient Descent**
   - Concept and Intuition
   - Batch, Stochastic, and Mini-Batch Gradient Descent

End-to-End Project

- **House Price Prediction**
  Use linear regression and regularization techniques to predict house prices based on historical data.

8. **Regularization**
   - Ridge Regression
   - Lasso Regression
   - Elastic Net
9. **Logistic Regression**
   - Sigmoid Function
   - Decision Boundary
   - Applications (e.g., binary classification)

End-to-End Project

- **Customer Churn Prediction**
  Build a model to predict customer churn using real-world datasets. Apply preprocessing, EDA, and a supervised learning algorithm.

10. **K-Fold Cross Validation**
    - Concept of Cross Validation
    - Bias-Variance Tradeoff
11. **Naive Bayes**
    - Bayes' Theorem
    - Gaussian, Multinomial, and Bernoulli Naive Bayes
12. **Support Vector Machine (SVM)**
    - Hyperplanes and Margins
    - Kernel Trick (Linear, Polynomial, RBF)
13. **Decision Tree**
    - Entropy and Information Gain
    - Gini Impurity
14. **Random Forest**
    - Bagging and Ensemble Techniques
    - Feature Importance

End-to-End Project

- **Spam Email Classification**
  Build a classifier using Naive Bayes, SVM, or Random Forest to distinguish spam emails from non-spam emails.

15. **K-Nearest Neighbors (KNN)**
    - Distance Metrics (Euclidean, Manhattan)
    - Choosing K
16. **Clustering**
    - K-Means Clustering
    - Hierarchical Clustering
    - DBSCAN

End-to-End Project

- **Movie Recommendation System**
  Build a system to recommend movies using clustering techniques.

Deep Learning

1. **Introduction to Deep Learning**
   - What is Deep Learning?
   - Differences Between Machine Learning and Deep Learning
   - Applications of Deep Learning
   - Key Challenges in Deep Learning (Overfitting, Data Requirements, Computational Needs)
2. **Neural Networks Basics**
   - Biological Neurons vs. Artificial Neurons
   - Perceptron Model
   - Activation Functions (Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax)
   - Forward Propagation and Backpropagation
   - XOR Problem and the Need for Non-Linearity
3. **Training Neural Networks**
   - Gradient Descent Optimization
   - Loss Functions (MSE, Cross-Entropy)
   - Overfitting and Underfitting
   - Regularization Techniques (Dropout, Weight Decay, Data Augmentation)
   - Weight Initialization Techniques (Xavier, He Initialization)
4. **Deep Neural Networks (DNN)**
   - Multi-Layer Perceptrons (MLP)
   - Batch Normalization
   - Dropout Regularization
   - Vanishing and Exploding Gradient Problems

End-to-End Project

- **Digit Recognition**
  Build a model to classify handwritten digits using a deep neural network and the MNIST dataset.

5. **Convolutional Neural Networks (CNN)**
   - Convolution Operations
   - Pooling Layers (Max Pooling, Average Pooling)
   - Architectures (LeNet, AlexNet, VGG, ResNet, EfficientNet)
   - Applications (Image Classification, Object Detection, Image Segmentation)
6. **Recurrent Neural Networks (RNN)**
   - Sequence Modeling
   - Vanishing Gradient Problem
   - Long Short-Term Memory (LSTM)

- Gated Recurrent Units (GRU)
- Bidirectional RNNs

End-to-End Project

- **Sentiment Analysis**
  Build a sentiment analysis model for text data using RNNs or LSTMs.

7. **Autoencoders**
   - Concept and Applications
   - Denoising Autoencoders
   - Variational Autoencoders (VAEs)
   - Use Cases (Anomaly Detection, Feature Compression)
8. **Generative Adversarial Networks (GANs)**
   - Generator and Discriminator
   - Training Dynamics (Adversarial Loss)
   - Conditional GANs
   - Applications (Image Generation, Style Transfer, Data Augmentation)

End-to-End Project

- **Face Detection**
  Build a system to detect faces in images or videos using a Convolutional Neural Network (CNN).

9. **Natural Language Processing (NLP) with Deep Learning**
   - Text Preprocessing (Tokenization, Stemming, Lemmatization)
   - Word Embeddings (Word2Vec, GloVe, FastText)
   - Sequence Models for NLP (LSTM, GRU)
   - Attention Mechanism
   - Transformers (BERT, GPT models)
   - Applications (Text Summarization, Translation, Sentiment Analysis)
10. **Advanced Topics in Deep Learning**
    - Transfer Learning (Pretrained Models)
    - Reinforcement Learning Basics
    - Hyperparameter Tuning (Grid Search, Random Search, Bayesian Optimization)
    - Model Deployment and Monitoring

End-to-End Project

- **Chatbot Development**
  Build a chatbot using a transformer model like BERT or GPT.

Deep Learning Frameworks

1. **Introduction to Frameworks**
   - Importance of Deep Learning Frameworks
   - Overview of TensorFlow, Keras, and PyTorch
2. **TensorFlow and Keras Basics**
   - Introduction to TensorFlow Ecosystem
   - Building and Training Models with Keras (Sequential and Functional APIs)
   - TensorFlow Basics (Tensors, Computational Graphs)
3. **PyTorch: Core Framework**
   - Introduction to PyTorch
     - Tensors and Operations
     - Dynamic Computation Graphs
   - Building Models in PyTorch
     - Defining Neural Networks (nn.Module)
     - Loss Functions and Optimizers
     - Training Loops (Manual vs `torch.optim`)
   - Data Handling
     - Dataset and DataLoader Classes
     - Data Augmentation Techniques with `torchvision`
   - Model Evaluation and Testing
     - Handling Overfitting with Regularization
     - Visualization with Matplotlib and Custom Metrics
   - Deployment and Export
     - Saving and Loading Models
4. **Advanced Topics in PyTorch**
   - Transfer Learning with Pretrained Models (`torchvision.models`)
   - Distributed Training and Mixed Precision (`torch.distributed` and `torch.cuda.amp`)
   - Custom Layers and Operations (`torch.autograd.Function`)