Micro syllabus and Model Question
Machine Learning

Course No.: CMP 364 (3 credits)	Full Marks: 100
Course Title: Machine Learning	Pass Marks: 45
Nature of Course: Theory + Lab	Total Lecture: 45 hours
Level: Bachelor	Program: B.E. Computer

Course Description

This course is designed to provide the fundamental principles and methodologies of machine learning. Students will learn to develop algorithms that can automatically learn from data, improve with experience, and make predictions or decisions. The course covers supervised, unsupervised machine learning alongside in-depth concepts of neural networks, and model evaluation and validation with a focus on both theoretical understanding and practical implementation.

Course Objectives

The main objectives of this course are to introduce the basic concepts of machine learning, algorithms involved in machine learning, frameworks, apply machine learning in real world applications and to interpret and optimize the result

Detail Contents:

Unit 1: Introduction to Machine Learning (5 hours)		
1.1 Definition and Evolution of Machine Learning	Define ML as an AI subset where systems learn from data without explicit programming. Trace evolution: 1950s perceptrons, 1990s statistical methods (e.g., SVMs), 2010s deep learning boom. Highlight applications like spam detection, image recognition.	30 min
1.2 Types of Machine Learning	Introduce four types: supervised, unsupervised, reinforcement, active learning with examples. Explain the importance of matching type to problem. Activities: Brief lecture with comparison diagram. Understand the basics of reinforcement learning. Differentiate active learning from other types.	80 min
1.3 Machine Learning Workflow	Formulate ML problems clearly. Set objectives, identify data needs. Apply preprocessing techniques: handle missing values, normalize, encode categories etc. Select appropriate models based on problem type. Evaluate models effectively using accuracy, precision, recall,	120 min

	RMSE etc. Understand deployment fundamentals: basics of serving models (e.g., Flask APIs, Cloud).	
1.4 Challenges in Machine Learning	Recognize data quality problems: Noise, bias, imbalanced data, missing values. Understand computational constraints like large datasets, complex models, GPUs, optimization. Grasp need for model transparency. Discuss ethical implications: Privacy, fairness, societal impact. Guidelines: Bias mitigation, transparency	70 min
Unit 2: Supervised Learning (10 hours)		
2.1 Types of Supervised Learning: Regression and Classification	Define supervised learning: Predicting outputs from labeled data Differentiate regression and classification tasks. Provide examples such as Regression: Continuous outputs (e.g., house prices). Classification: Discrete outputs (e.g., spam vs. not spam).	30 min
2.1.1 Simple and Multiple Linear Regression	Simple linear regression: $y = mx + b$, predicting one variable (e.g., height vs. weight). Multiple regression: Multiple predictors (e.g., house size, location → price). Explain least squares method. Lecture with derivation (simplified). Coding: Fit simple/multiple regression on a dataset using scikit-learn.	60 min
2.1.2 Polynomial Regression	Extend linear regression for non-linear data using polynomial features (e.g., $y = ax^2 + bx + c$). Discuss overfitting risks. Lecture with visualizations of polynomial fits. Live coding: Compare linear vs. polynomial regression on a curved dataset, Model non-linear relationships with polynomial regression.	60 min
2.2.1 Regularization Techniques: Lasso and Ridge Regression	Understand Ridge regression's role in regularization. Introduce regularization to prevent overfitting. Ridge: Adds L2 penalty (β^2) to cost function, shrinking coefficients.- Lecture with Ridge formula. Coding: Apply Ridge on a high-dimensional dataset. Understand Lasso regression. Uses L1 penalty (β) to cost function	90 min
2.2.2 Bias Variance Tradeoff	Grasp the tradeoff's impact on model performance. Explain bias (underfitting) vs. variance (overfitting). Regularization balances the tradeoff. Visualize model complexity vs. error. "How does regularization affect bias/variance?"	30 min
2.2.3 Support Vector Regression	Understand SVR's principles and implementation. Predict continuous values using a margin of tolerance (epsilon-tube). Explain kernel tricks briefly. Coding : SVR on a small dataset	30 min

2.3 Classification		
Logistic Regression	Implement logistic regression for binary tasks. Predict probabilities for two classes using sigmoid function. Explain log-loss. Coding: Binary classification on a dataset Handle multi-class problems with logistic regression. Extend to multiple classes using softmax (e.g., digit recognition). Discuss one-vs-rest vs. one-vs-one	90 min
K-Nearest Neighbor	Discuss lazy learning methods. Apply KNN for classification tasks. Classify based on k-nearest data points. Discuss distance metrics (e.g., Euclidean) and choosing optimal value of k.	45 min
Support Vector Machine (SVM)	Understand SVM's core concept. Maximize margin between classes using hyperplane. Support vectors define the boundary. Visualization of support vectors Grasp kernel functions' role. Kernels map data to higher dimensions: Linear, Polynomial, RBF. Explain non-linear separability Implement SVM for varied datasets. Apply SVM to linearly separable (e.g., binary) and non-linear data	75 min
Decision Tree	Implement ID3 algorithm, Construct and optimize decision trees, Build trees by splitting on features (e.g., Gini, entropy). Pruning reduces overfitting. Understand and apply ensemble methods, Bagging: Bootstrap aggregating (e.g., multiple trees). Random Forests: Bagging + random feature selection. Discuss robustness.	90 min
Unit 3: Unsupervised Learning (10 hours)		
3.1 Basic Concept of Unsupervised Learning	Grasp the essence and purpose of unsupervised learning. Define unsupervised learning: Finding patterns in unlabeled data without predefined outputs. Contrast with supervised learning.	30 min
3.2.1 Clustering	Understand clustering's role in unsupervised learning. Implement and tune K-Means clustering. Partition data into K clusters by minimizing variance. Explain algorithm: Initialize centroids, assign points, update centroids. Discuss choosing K (elbow method) Understand and apply agglomerative clustering Merge closest clusters iteratively. Explain linkage criteria (e.g., single, complete). Visualize with dendrograms. Divisive method: Split clusters recursively. Compare with agglomerative. Discuss computational trade-offs	270 min

3.2.2 DBSCAN	Implement DBSCAN and handle irregular clusters. Cluster based on dense regions (core points, border points, noise). Parameters: Epsilon (distance), MinPts. Advantages: Handles outliers, non-spherical clusters	75 min
3.3 Dimensionality Reduction	Apply PCA to simplify and visualize data. Reduce dimensions by projecting data onto principal components (max variance directions). Explain eigenvalues, eigenvectors (conceptually) with numerical example. Applications: Visualization, noise reduction Understand and implement LDA for classification tasks, Supervised dimensionality reduction (contrast with PCA). Maximize class separability. Numerical example of LDA on supervised dataset	225 min
Unit 4: Artificial Neural Network (12 hours)		
4.1 Introduction to Neural Network		
4.1.1 Neural Network Architectures	Define feedforward neural networks (FNNs): Layers of neurons (input, hidden, output) with no cycles. Explain data flow. Grasp CNN's role in spatial data processing. Introduce convolutional neural networks (CNNs): Process grid-like data (e.g., images) using convolutions. Overview of layers (convolution, pooling) Recognize RNN's suitability for sequential tasks. Explain recurrent neural networks (RNNs): Handle sequential data with loops (e.g., time series). Discuss vanishing gradient problem briefly.	100 min
4.1.2 Perceptrons	Define perceptron: Basic unit with weighted inputs, bias, and activation (e.g., step function). Limitations: Linear separability. Understand perceptron mechanics and limits. Build MLPs for complex tasks. Multiple hidden layers for nonlinear problems. Explain activation functions (sigmoid, ReLU) Introduce backpropagation, Grasp backpropagation's role in training: Compute gradients to update weights. Chain rule for error propagation. Conceptual overview of gradient descent	100 min
4.2.1 Forward and Backward Propagation	Explain forward pass. Understand how predictions are generated. Input through layers to output, applying weights, biases, activations. Compute predictions. Detail backprop. Implement backprop and optimize models. Backward pass to compute gradients. Gradient descent: Update weights to minimize loss. Discuss learning rate.	90 min

4.2.2 Loss Functions	Define loss, Understand loss function's purpose, Measure model error. Guides optimization. Types depend on task (regression vs. classification). Apply MSE for regression problems, Average squared difference for regression (e.g., house price prediction). Sensitive to outliers. Implement cross-entropy for classification. Measure probability divergence for classification, Binary vs. categorical.	90 min
4.2.3 Regularization Techniques	Define overfitting (model too complex) vs. underfitting (too simple). Identify model fit issues. Visualize training vs. test error. Apply regularization to improve generalization. L1/L2: Add penalty to loss (shrink weights). Dropout: Randomly disable neurons. Batch Normalization: Normalize layer inputs for stability.	120 min
4.3.1 Advanced Neural Network Architecture	Understand CNN building blocks. CNNs for grid-like data (images). Components: Convolution, pooling, fully connected layers. Discuss filters. Construct CNN layers for image tasks. Extract features (edges). Pooling: Reduce dimensions (max, average). Fully connected: Classify. Explain parameter reduction. Apply CNNs to vision tasks. Object detection, facial recognition, medical imaging. Discuss pre-trained models (e.g., VGG).	120 min
4.3.2 Recurrent Neural Network	Understand RNN's sequential processing.RNNs for sequences: Process inputs with memory. Discuss unfolded RNNs, vanishing gradients Implement LSTMs for long sequences. Address vanishing gradients with gates (forget, input, output). Use case: Text generation. Apply GRUs as an alternative to LSTMs. Simplified LSTM with update/reset gates. Compare efficiency with LSTMs Use RNNs for time-series tasks. Stock prices, weather forecasting, speech recognition. Discuss sequence-to-sequence models	100 min
Unit 5: Model Evaluation and Validation (8 hours)		
5.1 Need of Model Evaluation in ML	Explain why evaluation is critical: Ensure models generalize to new data, avoid overfitting, compare models fairly. Discuss real-world consequences of poor evaluation (e.g., biased AI). Introduce evaluation pipeline: Metrics, validation, tuning. Understand the role of evaluation in building reliable ML models.	40 min
5.2 Model Evaluation Metrics	Define accuracy: Correct predictions/total predictions. Discuss limitations: Misleading for imbalanced datasets (e.g., 99% negative class) Calculate and interpret accuracy Precision: Correct positive predictions/total positive predictions. Recall: Correct positive predictions/total actual positives. Fβ score: Harmonic mean, balancing precision/recall (β adjusts emphasis)	150 min

	Compute precision, recall, F1 on imbalanced data. Apply and interpret precision, recall, and F1 for classification Use a confusion matrix to analyze model performance. Matrix of true vs. predicted labels (TP, TN, FP, FN). Derive metrics like precision, recall. Use case: Medical testing Plot TPR vs. FPR at various thresholds. AUC: Summarize ROC performance. PR-Curve: Precision vs. recall for imbalanced data. Use case: Spam detection.Evaluate models using ROC and PR curves.	
5.2.1 Regression Metrics	Apply MAE to regression tasks. Average absolute difference between predictions and actuals. Robust to outliers. Penalizes large errors heavily. Use case: Continuous predictions Understand MSE's sensitivity to errors. RMSE: Square root of MSE, same units as target. Interpretable for error scale	110 min
5.3 Model Valuation	Split data into training (e.g., 80%) and test sets (20%) to evaluate generalization. Discuss random vs. stratified splits. Implement train-test split for validation. Divide data into K folds, train on K-1, test on 1, repeat K times. Average performance. Discuss K choice (e.g., 5, 10). Use case: Small datasets. Use K-fold CV for robust validation.	120 min
5.4 Hyperparameter tuning	Exhaustive search over hyperparameter combinations (e.g., learning rate, C in SVM). Discuss computational cost.Grid search for SVM hyperparameters. Optimize models with grid search. Sample random hyperparameter combinations. More efficient for high-dimensional spaces. Compare with grid search. Apply random search for efficient tuning.	60 min

Sample Question for Machine Learning
Pokhara University

Course Name: Machine Learning

1. a) What is machine learning? Explain its applications that are used in your daily lives. Also explain the workflow of machine learning. [7]
   b) Compare regression and classification with examples. Explain the different regularization techniques in machine learning. [8]
2. a) For the data in the given table, find the class of the new data using KNN algorithm. [7]

Name	**Age**	**Income**	**No. of Credit**	**Class**
Karina	20	5000	3	Yes
Shriya	21	4000	2	No
Riya	22	3000	1	Yes
Saugat	23	2500	3	No
Dil	19	3500	2	Yes
Priya	24	1700	1	Yes
Junet	22	2200	2	No
Manjil	25	3100	3	?

   b) How does the SVM classifier work? Explain different types of kernel function in SVM. [8]
   Or,
   Explain decision tree for classification. What are the steps involved in the ID3 algorithm?
3. a) Explain steps used in K means clustering. Apply K(=2) Means algorithm over the data (185, 72), (170, 56), (168, 60), (179,68), (182,72), (188,77) up to two iterations and show the clusters. Initially choose the first two objects as initial centroids. [8]
   Or,
   What is hierarchical clustering? For the given data create a dendrogram using agglomerative method single linkage.

Sample No.	X	Y
P1	0.4	0.53

P2	0.22	0.38
P3	0.35	0.32
P4	0.26	0.19
P5	0.08	0.41
P6	0.45	0.30

b) What are the different techniques of Anomaly Detection? Explain density based method. [7]

4. a) What is unsupervised learning? Explain the steps involved in PCA. [7]

   b) What is backpropagation? Explain how gradient descent optimizes the cost function. [8]

5. a) Differentiate between overfitting and underfitting with examples. Explain Lasso-Ridge Regression. [7]

   b) Explain the convolutional neural network architecture. How LSTM can be used for time series prediction. [8]

6. a) What is a confusion matrix? How does f1 score differ from accuracy model evaluation. How does ROC help in evaluating the binary classification model? [7]

   b) What are hyperparameters? Compare Grid search and random search with examples. [8]

7. Write short notes on: (Any two) [10]
   a) K-fold cross validation
   b) RNN
   c) Reinforcement learning