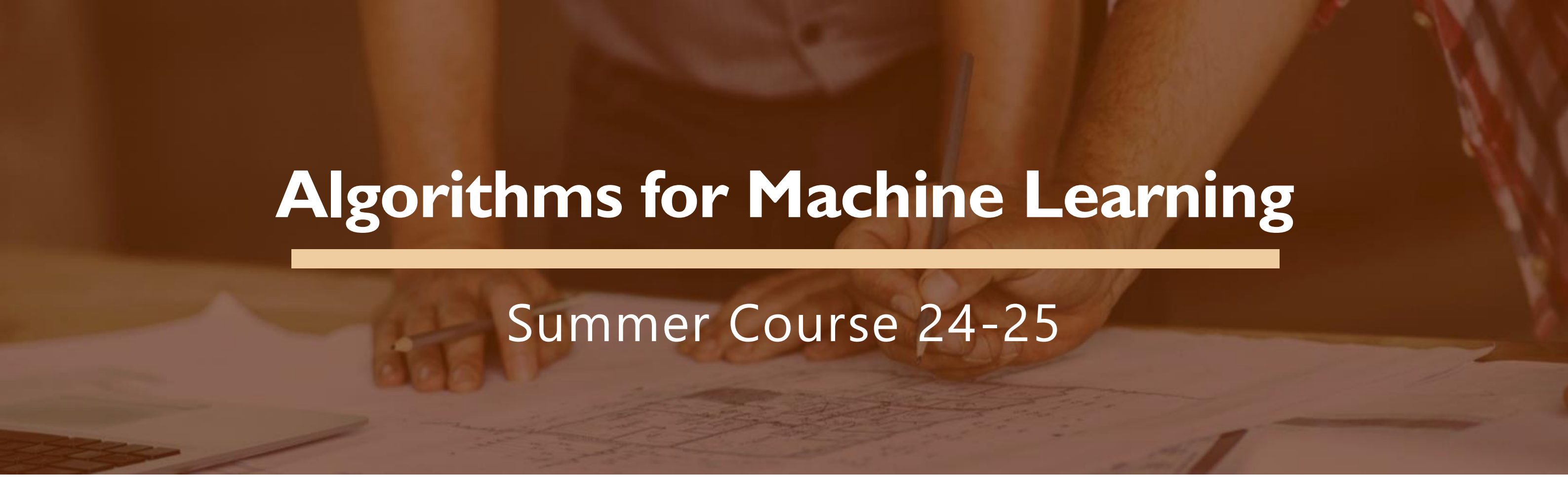

Algorithms for Machine Learning

Summer Course 24-25

INTRODUCTION

This course provides engineering students with a comprehensive introduction to algorithms used in machine learning (ML). It covers fundamental concepts, techniques, and applications, focusing on both theoretical and practical aspects. Students will gain hands-on experience through programming assignments and projects, using popular machine learning libraries.

Prerequisites:

- Basic programming knowledge (Python preferred)
- Understanding of basic statistics and linear algebra

Course Objectives:

- Understand the fundamental algorithms used in machine learning.
- Develop skills to implement and apply machine learning algorithms.
- Analyze the performance of different algorithms.
- Gain hands-on experience with machine learning tools and libraries.

Teaching Methodology

The course on Algorithms for Machine Learning will be conducted using a blended learning approach that combines theoretical lectures, hands-on programming labs, interactive discussions, and practical projects. This methodology ensures a comprehensive understanding of both the foundational concepts and practical implementations of machine learning algorithms.

1. **Lectures**
   - **Purpose:** Introduce and explain theoretical concepts.
   - **Approach:** Interactive presentations, real-world examples, Q&A sessions.
2. **Hands-on Labs**
   - **Purpose:** Provide practical experience in implementing ML algorithms.
   - **Approach:** Guided coding sessions, lab assignments, and code reviews.
3. **Interactive Discussions**
   - **Purpose:** Facilitate deeper understanding through peer-to-peer and instructor-student interactions.
   - **Approach:** Online discussion forums, in-class group discussions, and guest lectures from industry experts.

4. **Practical Projects**
   - **Purpose:** Apply learned concepts to real-world problems and develop project management skills.
   - **Approach:** Mid-term and final projects involving end-to-end machine learning pipelines, with presentations and peer feedback.
5. **Assessment and Evaluation**
   - **Purpose:** Evaluate students' understanding and application of course material.
   - **Approach:** Weekly quizzes, coding assignments, project evaluations, and participation assessment.
6. **Learning Resources**
   - **Purpose:** Provide additional materials to support learning.
   - **Approach:** Recommended textbooks, online resources, Kaggle competitions, and software tools like Jupyter Notebook, Scikit-learn, TensorFlow, etc.
7. **Feedback Mechanism**
   - **Purpose:** Continuously improve the course based on student feedback.
   - **Approach:** Regular surveys, feedback forms, and office hours for personalized guidance.

CONTENTS OF THE COURSE

Module 1: Introduction to Machine Learning (8 hours)

Lecture 1: Introduction to Machine Learning (2 hours)

Definition and types (supervised, unsupervised, reinforcement learning)
Applications of machine learning
Overview of the course structure

Lecture 2: Basics of Python for Machine Learning (2 hours)

Python programming basics
Introduction to libraries: NumPy, Pandas, Matplotlib

Lecture 3: Overview of Machine Learning Pipeline (2 hours)

Data collection and preprocessing
Model selection, training, evaluation, and deployment

Lecture 4: Data Preprocessing Techniques (2 hours)

Handling missing values, encoding categorical variables
Feature scaling and normalization

Module 2: Supervised Learning Algorithms (20 hours)

Lecture 5: Linear Regression (3 hours)

Simple and multiple linear regression
Gradient descent algorithm
Evaluation metrics: MSE, RMSE, R^2 score

Lecture 6: Logistic Regression (3 hours)

Binary classification
Sigmoid function
Cost function and optimization

Lecture 7: Decision Trees (3 hours)

Tree structure and properties
Information gain and Gini index
Overfitting and pruning techniques

Lecture 8: Support Vector Machines (3 hours)

Hyperplanes and support vectors
Kernel trick
Soft margin and hard margin

Lecture 9: k-Nearest Neighbors (3 hours)

Distance metrics
Weighted neighbors
Choosing the right k

Lecture 10: Naive Bayes Classifier (3 hours)

Bayes' theorem
Types of Naive Bayes classifiers
Assumptions and applications

Lecture 11: Ensemble Methods (2 hours)

Bagging and boosting
Random forests
Gradient boosting machines (GBM), XGBoost

Module 3: Unsupervised Learning Algorithms (14 hours)

Lecture 12: K-Means Clustering (3 hours)

Centroid-based clustering
Distance metrics
Elbow method and silhouette score

Lecture 13: Hierarchical Clustering (3 hours)

Agglomerative and divisive methods
Dendrograms
Linkage criteria

Lecture 14: Principal Component Analysis (PCA) (3 hours)

Dimensionality reduction
Eigenvalues and eigenvectors
Covariance matrix

Lecture 15: Association Rule Learning (3 hours)

Apriori algorithm
Support, confidence, and lift
Applications in market basket analysis

Lecture 16: Anomaly Detection (2 hours)

Techniques and algorithms
Applications in fraud detection

Module 4: Advanced Machine Learning Algorithms (14 hours)

Lecture 17: Neural Networks (4 hours)

Perceptron and multi-layer perceptron
Backpropagation algorithm
Activation functions

Lecture 18: Deep Learning Basics (4 hours)

Convolutional neural networks (CNNs)
Recurrent neural networks (RNNs)
Introduction to deep learning frameworks (TensorFlow, Keras, PyTorch)

Lecture 19: Reinforcement Learning (3 hours)

Markov decision processes
Q-learning

Applications in game playing and robotics

Lecture 20: Transfer Learning (3 hours)

Concept and applications
Pre-trained models
Fine-tuning techniques

Module 5: Model Evaluation and Optimization (10 hours)

Lecture 21: Model Evaluation Techniques (4 hours)

Cross-validation
ROC and AUC
Precision, recall, F1 score

Lecture 22: Hyperparameter Tuning (4 hours)

Grid search
Random search
Bayesian optimization

Lecture 23: Model Interpretability (2 hours)

SHAP values
LIME (Local Interpretable Model-agnostic Explanations)

Module 6: Practical Applications and Projects (10 hours)

Lecture 24: Case Study 1: Image Classification (3 hours)

Data preprocessing and augmentation
Implementing CNNs
Model evaluation

Lecture 25: Case Study 2: Natural Language Processing (NLP) (3 hours)

Text preprocessing
Implementing RNNs and LSTM
Sentiment analysis project

Lecture 26: Time Series Forecasting (2 hours)

ARIMA models

LSTM for time series
Evaluation metrics

Lecture 27: Recommender Systems (2 hours)

Collaborative filtering
Content-based filtering
Hybrid methods

Module 7: Special Topics in Machine Learning (4 hours)

Lecture 28: Explainable AI (XAI) (2 hours)

Importance of interpretability
Techniques and tools

Lecture 29: Ethical Issues in Machine Learning (2 hours)

Bias and fairness
Privacy concerns
AI ethics guidelinescourse requirements

Module 8: Review and Future Trends (4 hours)

Lecture 30: Course Review and Summary (2 hours)

Review of key concepts and techniques
Discussion of common pitfalls and best practices

Lecture 31: Future Trends in Machine Learning (2 hours)

Automated machine learning (AutoML)
Quantum machine learning
Emerging applications and research areas

Attendance is obligatory as well as any required reading or assignments. Throughout the course, in lecture or discussion sections we expect respect of one another, a positive environment in this class is up to each of the participants. To that point, if computers/tablets or other devices are used during this time, they should be used solely for course material and note-taking, so as not to distract others.

At least 2 years of study in an Engineering degree completed before the program start.

SCHEDULE AND CREDITS

- Contact hours: 80 h.
- Estimated time for homework and study: 40 h.
- ECTS Credits: 8 (4 credits in USA).

LEARNING OUTCOMES

Upon successful completion of this course, engineering students will be able to:

1. Understand Fundamental Concepts of Machine Learning

- **Define and Distinguish:** Clearly define machine learning and distinguish between its various types (supervised, unsupervised, and reinforcement learning).
- **Recognize Applications:** Identify and explain the applications of machine learning across different industries.

2. Implement and Apply Machine Learning Algorithms

- **Code Algorithms:** Write and implement machine learning algorithms in Python using libraries such as NumPy, Pandas, Scikit-learn, TensorFlow, and PyTorch.
- **Use Preprocessing Techniques:** Apply data preprocessing techniques including data cleaning, normalization, feature extraction, and dimensionality reduction.

3. Analyze and Evaluate Model Performance

- **Choose Evaluation Metrics:** Select appropriate evaluation metrics for different types of machine learning models.
- **Interpret Results:** Interpret and analyze the results of machine learning models to assess their performance and identify potential improvements.

4. Optimize Machine Learning Models

- **Tune Hyperparameters:** Perform hyperparameter tuning using techniques such as grid search, random search, and Bayesian optimization to improve model performance.
- **Prevent Overfitting:** Implement strategies like cross-validation, regularization, and pruning to prevent overfitting and enhance model generalization.

5. Develop Advanced Machine Learning Solutions

- **Implement Ensemble Methods:** Utilize ensemble methods (bagging, boosting, and stacking) to build robust and accurate models.
- **Construct Neural Networks:** Design, implement, and train neural networks including convolutional neural networks (CNNs) and recurrent neural networks (RNNs) for complex tasks such as image and text processing.

6. Apply Machine Learning to Real-world Problems

- **Work on Practical Projects:** Complete practical projects that involve end-to-end machine learning pipelines, including data collection, preprocessing, model training, and deployment.
- **Collaborate Effectively:** Work effectively in teams to tackle complex machine learning problems, demonstrating project management and teamwork skills.

7. Stay Informed About Emerging Trends

- **Explore Advanced Topics:** Gain insights into advanced topics like reinforcement learning, transfer learning, explainable AI (XAI), and automated machine learning (AutoML).
- **Ethical Awareness:** Understand and articulate the ethical implications of machine learning, including issues related to bias, fairness, and privacy.

8. Communicate Technical Information

- **Present Findings:** Present technical findings and project outcomes clearly and effectively, both in written reports and oral presentations.
- **Explain Concepts:** Explain complex machine learning concepts and techniques to both technical and non-technical audiences.

EVALUATION AND GRADING CRITERIA

The overall evaluation would be carried out as follows:

- Assignments and homework: 30% (Weekly coding assignments)
- Quizzes and individual evaluation tests: 20% (Weekly quizzes)
- Mid-term Project: 20% (Implementation of a machine learning algorithm)
- Final Project: 30% (Comprehensive project with report and presentation)

The specific topics evaluated are based on the sessions content and the weight varies depending on the dedication.