Machine Learning Fundamentals

Course Overview

Welcome to Machine Learning Fundamentals! This course provides a thorough grounding in a wide range of machine learning methods, for classification, regression, conditional probability estimation, clustering, and dimensionality reduction. We will gain an intuitive understanding of these methods, get a hands-on feel for them using experiments with Jupyter notebooks, and delve into their mathematical foundations.

Instructor

Sanjoy Dasgupta, Professor of Computer Science and Engineering, UC San Diego

Prerequisites

The most important prerequisites for this course are:

- The ability to program in Python and to use Jupyter notebooks. This can be obtained by taking the course DSE 200X, Python for Data Science.
- Familiarity with calculus, especially derivatives of single-variable and multivariate functions.

The course will make heavy use of basic probability and linear algebra. Although we will introduce these concepts as needed, it will be easiest if learners already have some familiarity with them.

Learning objectives

This course is an intensive introduction to the most widely-used machine learning methods. The first goal is to provide a basic intuitive understanding of these techniques: what they are good for, how they work, how they relate to one another, and their strengths and weaknesses. The second goal is to provide a hands-on feel for these methods through experiments with suitable data sets, using Jupyter notebooks. The third goal is to understand machine learning methods at a deeper level by delving into their mathematical underpinnings. This is crucial to being able to adapt and modify existing methods and to creatively combining them.

Topics

- Taxonomy of prediction problems
- Nearest neighbor methods and families of distance functions
- Generalization: what it means; overfitting; selecting parameters using cross-validation
- Generative modeling for classification, especially using the multivariate Gaussian
- Linear regression and its variants
- Logistic regression
- Optimization: deriving stochastic gradient descent algorithms and testing convexity
- Linear classification using the support vector machine
- Nonlinear modeling using basis expansion and kernel methods
- Decision trees, boosting, and random forests

- Methods for flat and hierarchical clustering
- Principal component analysis
- Autoencoders, distributed representations, and deep learning

Course Outline
This is a ten-week course.

- Week 1: Introduction: nearest neighbor, and a host of prediction problems
- Week 2: Probability basics and generative modeling
- Week 3: Linear algebra basics, the multivariate Gaussian, and more generative modeling
- Week 4: Linear regression and logistic regression
- Week 5: Optimization
- Week 6: Support vector machines
- Week 7: Beyond linear prediction: kernel methods, decision trees, boosting, random forests
- Week 8: Clustering
- Week 9: Informative projections
- Week 10: Deep learning

Python notebooks
For each topic, we will post Jupyter notebooks with programs and illustrative examples. These can be run and modified to get a feel for the methods involved.

Discussion forums
Discussion forums provide an opportunity for learners to discuss course materials with each other and with course staff.

Assignments and exams

- Weekly assignments (75% of grade): these have five components, of which four are graded.
    - Engagement (5%). This consists of simply checking the "mark as complete" button after viewing each video and associated materials.
    - Poll questions (0%).
    - Comprehensive quizzes (10%). These are simple multiple-choice questions based on the week's videos.
    - Problem sets (30%). These conceptual and mathematical problems are meant to cement understanding of the week's material.
    - Programming assignments (30%). These develop the ability to design and use machine learning algorithms.
- Final exam (25% of grade). This consists of conceptual and mathematical problems that cover the material from the entire course.

Time and grading policies for weekly assignments

All assignments will be due the last day of the course.

The worst two problem set scores will be dropped and the worst two programming assignment scores will be dropped. This means, for instance, that it is possible to obtain a full score while skipping any two of the programming assignments and any two of the problem sets.

Verified Learners
Learners can earn a verified certificate for the course by enrolling as part of the verified track, completing identity verification, and earning a passing grade. The deadline to change from unverified to the verified track is the end of the sixth week.

Grading
Grades will be assigned based on final scores, according to the following rubric:
85-100%: A
70-85%: B
50-70%: C
Less than 50%: F

Effort
The weekly effort for the course is intended to be roughly 10-12 hours.

Pace and deadlines
The course is instructor-paced. Each week the relevant material (videos and assignments) will be released, and will remain online until the end of the course. We encourage learners to keep current with the videos and assignments; however, as described above, there is a six-week window for submitting each assignment.

Honor code
Beyond learning this important material, we hope learners will take the course seriously and respect fellow students. Please read and abide by the EdX honor code pledge.

We value your feedback
This is a new online course. We are committed to making it as accessible and educational as possible, and would appreciate any feedback about how we might improve it.

Thank you!
Thank you very much for taking the course. We hope you enjoy it.