Machine Learning and Statistical Learning Theory

Instructor: Nate Bade
Office: 539 Lake Hall
Office Hours: M. 4:30 - 6:00, Th. 4:30 - 5:45.

Primary Textbook: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* by Trevor Hastie, Robert Tibshirani, Jerome Friedman
Additional Texts:
Understanding Machine Learning: From Theory to Algorithms by Shai Shalev-Shwartz, Shai Ben-David
Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems - by Aurélien Géron

Overview

Introduces both the mathematical theory of learning and the implementation of modern machine-learning algorithms appropriate for data science. Modeling everything from social organization to financial predictions, machine-learning algorithms allow us to discover information about complex systems, even when the underlying probability distributions are unknown. Algorithms discussed include regression, decision trees, clustering, and dimensionality reduction. Offers students an opportunity to learn the implications of the mathematical choices underpinning the use of each algorithm, how the results can be interpreted in actionable ways, and how to apply their knowledge through the analysis of a variety of data sets and models.

Practicum: Algorithms and Implementation:

The first portion of this course will focus on defining and applying the a variety of learning algorithms. We will start the class discussing the problem of learning a partially complete data labeling (Supervised Learning) and develop algorithms such as regression, support vector machines, and decision trees. We will then move to detecting patterns in unlabeled data using clustering and principle component analysis. By the end of the course, students should be familiar with the mathematical structure of these algorithms, and implications of any choices made in implementing them.

The machine learning is almost defined by its application. We will have frequent in class labs wherein we implement the algorithms we have discussed in the theory portion on real world data sets. In these labs, you will learn to use Python to obtain data, view it, clean it, analyze it, and finally use machine learning algorithms to try to solve a problem. The labs are open ended, and students are expected to use their own ideas and intuition to try to get an honest, best fit on the data sets given. The lab portion will cumulate in a final project, where you (or possibly your team) will attempt a novel machine learning project within an area of interest.

Theory: Frameworks and Communication:

After we have a handle on high level methods of machine learning, we will develop we will use the PAC (partially approximately correct) model to define data, sampling, training, and the metrics we will use to evaluate a machine learning algorithms success. The PAC framework will allow us to discuss each component of the machine learning process, and the practical effects of all the relevant

choices. It will also allow us to state and discuss the No Free Lunch Theorem (the bias/complexity trade off) about the limits of learnability. We will cover the VC dimension and the conditions under which we can guarantee an algorithm can learn a problem. Finally, we will analyze regularization, gradient decent and ensambling as extensions of the PAC model, and show how they can often lead to better results without violating the theorems above.

The final portion of the class is about the communication of machine learning results and ideas. As an applied mathematician, part of your job will to communicate your results to coworkers, bosses or investors that don't understand the math as well as you do. To this end, you will be expected to communicate clearly and effectively in both your labs and in your final project. The goal of your final project is to produce something you can put on a resume, a git-hub, or give as a presentation. I will be available to help you throughout this process with writing, editing and communication.

This course requires Python, Jupyter Notebook Server and github, all of which are free and open source.

Grade Breakdown

Written Homework (30%) - There will be three written assignments which will focus on theory.

Labs (30%) - There will be roughly 6 labs of which 4 must be completed. Labs will focus on the implementation of algorithms on real world data sets. Class time will be allotted for labs, but students may finish labs at home. In each lab, we will fit a real world data set using the algorithms of techniques introduced in that weeks theory lecture. Labs will be graded out of 5 pts, 3 pts for completion, 2 pts for communication. There will also be a standing bonus of 2 pts for going above and beyond and exploring an interesting aspect of the parameter space, or getting a really good fit.

Final Project (40%) - The final project will consist of a project report (roughly 5 pages) and presentation (roughly 15 minutes). Project groups should contain 3-4 people.

Masters students: This class features an XN project with an industry partner on detecting Alzheimers from 3d MRI scans of human brains. Masters students are encouraged to participate in this project.

PhD students: If you would like to propose your own project, it can take one of three forms:

- A computational analysis of a data set using sufficiently complicated or novel techniques from this course.
- A theoretical presentation of a topic not covered in this course with a case study.
- Thesis or Lab project.

I am more than happy to discuss possible projects in any of these categories with you.

The written component with be 75% of the grade, the presentation component 25%.

Rough project timeline:

- Group Selection: January 27
- Project Proposal Deadline: February 3

- Progress Report: February 24
- First Draft: March 16
- Final Draft: April 6
- Presentations: April 13 - 24.

Machine Learning Algorithm Roadmap

Supervised - Labeled training data

- **Regression** - Predicting continuous values
  - Linear Regression
  - Nonlinear Regression and Functional Fitting
  - Radial Basis Functions
  - Neural Networks
- **Classification** - Predicting discrete values
  - Logistic Regression
  - Support Vector Machines (SVM)
  - Decision Trees
  - Neural Networks

Unsupervised - Unlabeled data, pattern detection and descriptive modeling

- Principle Component Analysis
- Clustering
- Smoothing and Spline Fitting
- Neural Networks

Further topics in Machine Learning (Not covered)

Semi-Supervised - Image segmentation, sparse training data

- Neural Networks (Regional-CNN, U-NET)
- k-means Clustering
- Graph Partitioning

Implementation

- Model Ensembles
- Online Learning
- Large Data Sets
- Distributed Computations

Natural Language Processing

Week By Week Schedule:

Week 1: Introductory Concepts for Statistical Learning (ESL Chap. 1 and 2)
Definition of terms: Machine Learning vs Statistics, The Statistical Learning Framework and Empirical Risk Minimization. Linear Regression using derivatives and linear algebra.

Week 2: Working with High Dimensional Data (ESL Chap. 14.5, HML Chap. 1, 8)
Descriptive statistics. Using Principle Component Analysis to reduce dimension. Three categorical labeling modes':k-nearest neighbors, clustering, mixture modeling.
Lab 0: *Assembling the tools: Python, Jupyter, Scikit Learn, Numpy, first steps in data processing*

Week 3: Linear Regression (ESL Chap. 3, HML Chap. 4)
Linear regression and its relation to descriptive statistics. Regularize Loss Minimization, Stability and Stabilizers, Tikhonov Regularization. Training using Gradient Decent and Stochastic Gradient Decent.
Lab 1: *Regression on Ames Housing Data*

Assignment 1

Week 4: Linear Methods for Classification (ESL Chap. 3, HML Chap. 3)
Maximum Likelihood, Linear Discriminant Analysis, Logistic Regression.
Lab 2: *MNIST Handwritten Number Data Set - Precision/recall trade off with different classifiers*

Week 5: Polynomial Regression and Basis Expansion (ESL Chap. 5, HML Chap. 4)
Polynomial Regression, Radial Basis Functions, Kernels, smoothing.

Week 6: Neural Networks (HML Chap. 9-10)
Theory of Neural Networks. Perceptions and deep networks. Training networks using back propagation.
Lab 4: *Implementing Neural Networks with Tensorflow*

Assignment 2

Week 7: Convolutional Neural Networks (HML Chap 11, 13)
Convolutional Neural Networks and Image Processing
Lab 5: *Implementing CNNs with Keras*

Week 8: Recurrent Neural Networks (HML Chap 11, 14)
Recurrent Neural Networks and text processing.
Lab 6: *Implementing RNNs with Keras*

Week 9: Support Vector Machines and Decision Trees (ESL Chap. 12, 9)
Algorithms: SVMs for separable and non-separable datasets. Decision trees.

Week 10: Classification: Ensemble Learning, Bagging, Boosting Bootstraping and Hyper-parameters (ESL Chap. 8, 16)

Voting Classifiers and Boosting, Fitting a voting method. Bootstraping data, bagging and hyper parameter tuning.

Week 11: Statistical Learning Theory (UML Chap. 1-5)

The PAC Learning Framework. The Bias/complexity tradeoff and the No Free Lunch theorem.

Week 12: Learnability and VC dimension. (UML Chap. 6-8)

Definition of learnability. VC Dimension as a hard bound on learnability. Application VC dimention to Linear Regression and Neural Networks.

Week 13-14: Special Topics and Final Project Presentations
Assignment 3