ECEN 740:
Machine Learning Engineering

Lecture 0: Introduction

Tie Liu

Texas A&M University

Machine learning engineering

The main goal of machine learning engineering is to design *computer algorithms* that allow machines to *learn* to perform on a given task

- Learning is based on the *statistics* of the training data, while the machines do not necessarily understand the *semantic meaning* of the data (even though some machines trained by contemporary machine learning algorithms may appear to)

- Machine learning is driven by the development of algorithms over increasingly more complex *mathematical models*. In particular, contemporary machine learning algorithms are mostly based on *deep neural networks*

- The success of deep learning remains, to a large extent, a *mystery* from the theoretical standpoint (despite a great deal of research effort in recent years)

General setup

- Let $(\mathsf{z}_i : i \in [m])$ be a collection of m *training examples*, drawn independently and identically from a *data-generating distribution*
- The algorithm designer may have some (limited) prior knowledge on the data-generating distribution; otherwise, it is *unknown* to the algorithm designer (and the machines)
- The goal of the algorithm is to let the machines learn about the data-generating distribution from the training data examples and use the learned knowledge to perform on a given task
- While learning is through the training examples, the performance of the learner is measured by its performance on a *fresh* data example z that is drawn from the *same* data-generating distribution but is *independent* of the training examples $(\mathsf{z}_i : i \in [m])$

Learn to predict

- In this setting, each data example $\mathsf{z} = (\mathsf{x}, \mathsf{y})$ consists of an *observation* variable x, which is usually a *high-dimensional* vector, and a *target* variable y, which can be either *categorical* or *numerical*

- The task is to *predict* the value of y from the value of x

- When the target variable y is *categorical*, the task is known as *classification*; on the other hand, if y is *numerical*, the task is known as *regression*

- *Learn-to-predict* is an essential machine learning task and a major focus of our study in this course

Generative vs. discriminative approach

- Roughly speaking, there are two different approaches for learn-to-predict: the *generative* approach and the *discriminative* approach
- The generative approach is based on a model for the *joint* distribution between the observation x and the target y; this approach is called "generative" because a data example can be generated from the joint distribution
- The discriminative approach has two sub-approaches: the *indirect discriminative* approach and the *direct discriminative* approach
- The indirect discriminative approach is based on a model for the *posterior* distribution of the target y given the observation x. While a data example cannot be generated from the posterior distribution, a prediction rule can be made directly from it

- The direct discriminative approach is based on a model directly for the *predictor*

- How to make predictions based on the joint or the posterior distribution is the subject of *Bayesian point estimation*, which we will discuss in our next lecture

- Both generative and discriminative approaches require a *model*; the difference only lies at which level modeling is performed. The *necessity* of a model will be discussed carefully under the context of different approaches

- Model selection needs to consider both *sample complexity* and *algorithmic complexity*, and the best way to manage sample complexity is to leverage our prior knowledge on the data-generating distribution

Other learning tasks

In addition to *learn-to-predict*, in this course we will also study:

- learn to *rank*
- learn to *represent*
- learn to *generate*

What's not to cover

As a *first* course in machine learning, this course covers some of the most basic and important machine learning algorithms (and the logic and principle behind them). However, the following topics will *not* be covered despite being equally if not more important:

- deep learning and large language models
- reinforcement learning
- online learning
- statistical learning theory (in any serious manner)

Textbook and references

Textbook:

- Shai Shalev-Shwartz and Shai Ben-David: *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014

References:

- Christian P. Robert: *The Bayesian Choice, Second Edition*. Springer 2007
- Robert E. Schapire and Yoav Freund: *Boosting: Foundations and Algorithms*. MIT Press, 2012
- Carl Edward Rasmussen and Christopher K. I. Williams: *Gaussian Processes for Machine Learning*. MIT Press, 2006
- Various *tutorial papers* as we move along

Course requirement

- Bi-weekly homework assignments (75%)+ one exam (25%)
- Homework assignment and submission via Google Classroom. Late submissions will *not* be accepted
- The exam will be held after completing the study of learn-to-predict (roughly at the 2/3 mark of the semester)
- The exam may be replaced by a term project, subject to instructor's approval
- Letter grades: A: 90–100; B: 80–89; C: 70–79; D: 60–69; F: 0–59 (*no* curving or dropping of any bad homework)