

Introduction

- These notes are for my 2nd time learning CS446 after reviewing linear algebra, and probabilities and statistics, aiming for my research work in 'Sun Lab UIUC'.
- Learning ML mathematically and statistically, then the code!
- ML: improving computational mechanisms by fitting them to data.
- Application: Image Classification
Recommender system
Machine Translation
Chess

Math Background:

- ① Linear Algebra
- ② Basic probability and statistics
- ③ Multivariate Calculus
- ④ Basic Proof writing

Coding:

- ① python3 ② numpy
- ③ sk-learn ④ Pytorch

- Main Setting:
- ① supervised learning
 - ② unsupervised learning
 - ③ Time series modeling
 - ④ Reinforcement learning

Supervised Learning

Training data: $(x_1, y_1) \dots (x_n, y_n)$

Goal: Learn predictor from labeled examples, that accurately predicts future labels.

Eg:

- Linear Regression
- Least squares
- Logistic Regression
- Support Vector Machine
- Deep net
- Structured prediction

Difficulties of supervised learning

- ① Clean / Augment Data
- ② Choose model for f
- challenge [③] fit \hat{f} to data
- ④ How to prevent overfitting
(fits data well, not on future data)

"pytorch mini-algorithm"

1. Clean / Augment data
2. Pick Model (Algorithm)
3. Pick a Loss function
4. Gradient Descent to fit to data.
5. Tweak 1-4 until training error small
6. Tweak 1-5, possibly reducing model complexity, until testing error small.