Chapter 1：The learning problem

1. define the learning problem
    a) Problem setup: examples: a viewer rates movie: reverse-engineer
2. Components of Learning (summarized from the credit approval example):
    a) the unknow target function,
    b) the training examples,
    c) the learning algorithm (based on hypothesis set. Eg. Neural network, algorithm: back propagation)
    d) the final hypothesis
3. Exercises 1.1 – formalize the learning problems

    learning components

    [The Solutions in "Learning From Data: A Short Course"(Last Updated: 2021-02-22) - XiangyunZHANG - 博客园 (cnblogs.com)]
4. A simple learning model -> Perceptron Learning Algorithm
    a) The meaning of the weights: reflect importance. Exercise 1.2: an example about weights
    b) Formula

    $$h(\mathbf{x}) = \text{sign}(\mathbf{w}^{\mathrm{T}}\mathbf{x}).$$

    c) Iterative method: update the weights

    $$\mathbf{w}(t+1) = \mathbf{w}(t) + y(t)\mathbf{x}(t).$$

    d) Exercise：prove the update rule
    e) Effectiveness of PLA: PLA 的有效性: learning algorithm 从 infinite 假设集中 用 finite (simple) steps 去搜索到 solution.

    总能找到合适的 weight solution, 不管

        1.每次 iteration 选哪个错误分类的点做 update

        2.initial weight 是什么样的
5. Exercise 1.4：PLA code and diagram
6. Learning and Design
    a) Difference between learning and design: learning from DATA; Design from specification (coin categorization) if less specification, need data to learn (examples in Exercise 1.5: differences between learning and design)
    b) Types of Learning:

        A: supervised learning: given input & correct output

        (inc. active learning: data set through queries, need some strategies to ask questions

        online learning: data set updates; limitations on computing and storage);

        B: reinforcement learning: (input, some output, grade for this output) 还有其他可能的 output，但不知道好坏程度；reinforce better actions;

        C: unsupervised learning: (input,?)

        Learn something from the inputs by themselves: eg. cluster

        (Examples from Exercise 1.6)
7. Is Learning Feasible?

a) Exercise 1.7: samples in data set tell nothing outside of data

b) Exercise 1.8: binomial distribution: using samples to infer outside data set in a probabilistic way

c) Bin experiment connects to learning: Hoeffding inequality

- Bond depends on N (sample size)
- In sample error (v) is random since it depends on the sample, out of sample error (μ) is constant. μ affects v, but v is used to infer μ
- 还是不知道真正μ，不保证μ就在这个范围内，但是 knowing that we are within ±E of μ most of the time is a significant improvement over not knowing anything at all.
- Hoeffding inequality: ("in sample mean = out of sample mean" is P.A.C: probably approximately correct)
  prerequisite:
  1. Sample randomly selected with replacement
  2. Hypothesis h is fixed
- Multiple hypothesis: coin analogy
  small test: 1-0.999^1000 = 0.63
  Exercise 1.10 coin flipping code explanation. verify 0.63;

[Learning-From-Data-A-Short-Course/Solutions to Chapter 1 The Learning Problem.ipynb at master · niuers/Learning-From-Data-A-Short-Course (github.com)]

Union bound:

$$\begin{aligned}\mathbb{P}[\,|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon\,] \;\le\; \mathbb{P}[\quad & |E_{\text{in}}(h_1) - E_{\text{out}}(h_1)| > \epsilon \\ & \textbf{or } |E_{\text{in}}(h_2) - E_{\text{out}}(h_2)| > \epsilon \\ & \cdots \\ & \textbf{or } |E_{\text{in}}(h_M) - E_{\text{out}}(h_M)| > \epsilon\,] \\ \le\; & \sum_{m=1}^{M} \mathbb{P}\left[|E_{\text{in}}(h_m) \quad E_{\text{out}}(h_m)| > \epsilon\right].\end{aligned}$$

$$\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon] \le 2Me^{-2\epsilon^2 N}.$$ M should be finite

d) Feasibility of learning:

Probabilistic view: Data set tells us something likely about f outside of Data set (assumption: examples in data set are generated independently) ->learning is feasible

The feasibility of learning is thus split into two questions:

1. Can we make sure that $E_{\text{out}}(g)$ is close enough to $E_{\text{in}}(g)$?
2. Can we make $E_{\text{in}}(g)$ small enough?

e) M: The complexity of Hypothesis set

M small， good bound， Ein ≈ Eout

M large， g fits the data well, Ein ≈ 0

Trade off in complexity of hypothesis set： major theme in learning theory

The complexity of f: no effect on bound

but complex functions hard to fit, it is difficult to make Ein ≈ 0

if ↑M, pick g which makes Ein ≈ 0, but bound↑， Ein not close to Eout

Recap:

1. The credit approval case, what is the in-sample error?
2. the out of sample error?
3. Explain the Hoeffding inequality
4. apply a bound to Eout wrt Ein, allowing the learning algorithm to choose any hypothesis in a set
5. How can we make learning feasible?
6. what is the contribution of Hoeffding inequality to the feasibility of learning?
7. How is the complexity of the hypothesis set affecting the feasibility of learning?
8. How can a complex target function affect the feasibility of learning?

8. Error and Noise
    a) Error measures: eg. Binary square error; different error measure -> different fimal hypothesis
    b) Fingerprint example: supermarket vs. CIA: so which error measure to use depends on how the system is going to be used
    c) Noise targets:
        - target distribution: P(y|X)
        - 同样的 x， 可能得到不同的 y, given x, there's a probability distribution over y， y 是什么结果可能有个概率
          或者理解成：y given x to be the deterministic f (x) , and consider y - f (x) as pure noise
        - data point: P(x, y) = P(x)P(y I x)
        - Exercise 1.13： 理解 P(y|X)， noise 的影响
        - 总结：noisy target harder to learn (hard to fit noise); inequality valid, but Ein worse