

Machine Learning Theory (CS 6783)

Lecture 2 : Learning Frameworks, Examples

1 Setting up learning problems

1. \mathcal{X} : instance space or input space

Examples:

- Computer Vision: Raw $M \times N$ image vectorized $\mathcal{X} = [0, 255]^{M \times N}$, SIFT features (typically $\mathcal{X} \subseteq \mathbb{R}^d$)
- Speech recognition: Mel Cepstral co-efficients $\mathcal{X} \subset \mathbb{R}^{12 \times \text{length}}$
- Natural Language Processing: Bag-of-words features ($\mathcal{X} \subset \mathbb{N}^{\text{document size}}$), n-grams

2. \mathcal{Y} : Outcome space, label space

Examples: Binary classification $\mathcal{Y} = \{\pm 1\}$, multiclass classification $\mathcal{Y} = \{1, \dots, K\}$, regression $\mathcal{Y} \subset \mathbb{R}$

3. $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$: loss function (measures prediction error)

Examples: Classification $\ell(y', y) = \mathbf{1}_{\{y' \neq y\}}$, Support vector machines $\ell(y', y) = \max\{0, 1 - y' \cdot y\}$, regression $\ell(y', y) = (y - y')^2$

4. $\mathcal{F} \subset \mathcal{Y}^{\mathcal{X}}$: Model/ Hypothesis class (set of functions from input space to outcome space)

Examples:

- Linear classifier: $\mathcal{F} = \{x \mapsto \text{sign}(f^\top x) : f \in \mathbb{R}^d\}$
- Linear SVM: $\mathcal{F} = \{x \mapsto f^\top x : f \in \mathbb{R}^d, \|f\|_2 \leq R\}$
- Neural Networks (deep learning): $\mathcal{F} = \{x \mapsto \sigma(W_{out}\sigma(W_K\sigma(\dots\sigma(W_2(W_1\sigma(W_{in}x))))))\}$ where σ is some non-linear transformation (Eg. ReLU)

Learner observes sample: $S = (x_1, y_1), \dots, (x_n, y_n)$

Learning Algorithm : (forecasting strategy, estimation procedure)

$$\hat{y} : \mathcal{X} \times \bigcup_{t=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^t \mapsto \mathcal{Y}$$

Given new input instance x the learning algorithm predicts $\hat{y}(x, S)$. When context is clear (ie. sample S is understood) we will fudge notation and simply use notation $\hat{y}(\cdot) = \hat{y}(\cdot, S)$. \hat{y} is the predictor returned by the learning algorithm.

Example: linear SVM Learning algorithm solves the optimization problem:

$$\mathbf{w}_{\text{SVM}} = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{t=1}^n \max\{0, 1 - y_t \mathbf{w}^\top x_t\} + \lambda \|\mathbf{w}\|$$

and the predictor is $\hat{\mathbf{y}}(x) = \hat{\mathbf{y}}(x, S) = \mathbf{w}_{\text{SVM}}^\top x$

1.1 PAC framework

$$\mathcal{Y} = \{\pm 1\}, \quad \ell(y', y) = \mathbf{1}_{\{y' \neq y\}}$$

Input instances generated as $x_1, \dots, x_n \sim D_X$ where D_X is some unknown distribution over input space. The labels are generated as

$$y_t = f^*(x_t)$$

where target function $f^* \in \mathcal{F}$. Learning algorithm only gets sample S and does not know f^* or D_X .

Goal: Find $\hat{\mathbf{y}}$ that minimizes

$$\mathbb{P}_{x \sim D_X} (\hat{\mathbf{y}}(x) \neq f^*(x))$$

1.2 Non-parametric Regression

$$\mathcal{Y} \subseteq \mathbb{R}, \quad \ell(y', y) = (y' - y)^2$$

Input instances generated as $x_1, \dots, x_n \sim D_X$ where D_X is some unknown distribution over input space. The labels are generated as

$$y_t = f^*(x_t) + \varepsilon_t \quad \text{where } \varepsilon_t \sim N(0, \sigma)$$

where target function $f^* \in \mathcal{F}$. Learning algorithm only gets sample S and does not know f^* or D_X .

Goal: Find $\hat{\mathbf{y}}$ that minimizes

$$\mathbb{E}_{x \sim D_X} [(\hat{\mathbf{y}}(x) - f^*(x))^2] =: \|\hat{\mathbf{y}} - f^*\|_{L_2(D_X)}$$

1.3 Statistical Learning (Agnostic PAC)

Generic \mathcal{X} , \mathcal{Y} , ℓ and \mathcal{F}

Samples generated as $(x_1, y_1), \dots, (x_n, y_n) \sim D$ where D is some unknown distribution over $\mathcal{X} \times \mathcal{Y}$.

Goal: Find $\hat{\mathbf{y}}$ that minimizes

$$\mathbb{E}_{(x,y) \sim D} [\ell(\hat{\mathbf{y}}(x), y)] - \inf_{f \in \mathcal{F}} \mathbb{E}_{(x,y) \sim D} [\ell(f(x), y)]$$

For any mapping $g : \mathcal{X} \mapsto \mathcal{Y}$ we shall use the notation $L_D(g) = \mathbb{E}_{(x,y) \sim D} [\ell(g(x), y)]$ and so our goal can be re-written as:

$$L_D(\hat{\mathbf{y}}) - \inf_{f \in \mathcal{F}} L_D(f)$$

Remarks:

1. $\hat{\mathbf{y}}$ is a random quantity as it depends on the sample
2. Hence formal statements we make will be in high probability over the sample or in expectation over draw of samples

2 Minimax Rate

How well does the best learning algorithm do in the worst case scenario?

Minimax Rate = “Best Possible Guarantee”

PAC framework:

$$\mathcal{V}_n^{PAC}(\mathcal{F}) := \inf_{\hat{y}} \sup_{D_X, f^* \in \mathcal{F}} \mathbb{E}_{S:|S|=n} [\mathbb{P}_{x \sim D_x} (\hat{y}(x) \neq f^*(x))]$$

A problem is “PAC learnable” if $\mathcal{V}_n^{PAC} \rightarrow 0$. That is, there exists a learning algorithm that converges to 0 expected error as sample size increases.

Non-parametric Regression:

$$\mathcal{V}_n^{NR}(\mathcal{F}) := \inf_{\hat{y}} \sup_{D_X, f^* \in \mathcal{F}} \mathbb{E}_{S:|S|=n} [\mathbb{E}_{x \sim D_X} [(\hat{y}(x) - f^*(x))^2]]$$

A statistical estimation problem is consistent if $\mathcal{V}_n^{NR} \rightarrow 0$.

Statistical learning:

$$\mathcal{V}_n^{stat}(\mathcal{F}) := \inf_{\hat{y}} \sup_D \mathbb{E}_{S:|S|=n} \left[L_D(\hat{y}) - \inf_{f \in \mathcal{F}} L_D(f) \right]$$

A problem is “statistically learnable” if $\mathcal{V}_n^{stat} \rightarrow 0$.

Statistical learning:

$$\mathcal{V}_n^{stat}(\mathcal{F}) := \inf_{\hat{y}} \sup_D \mathbb{E}_{S:|S|=n} \left[L_D(\hat{y}) - \inf_{f \in \mathcal{F}} L_D(f) \right]$$

A problem is “statistically learnable” if $\mathcal{V}_n^{stat} \rightarrow 0$.

A statement in expectation implies statement in high probability by Markov inequality but more generally one can also easily convert to exponentially high probability.

2.1 Comparing the Minimax Rates

Proposition 1. For any class $\mathcal{F} \subset \{\pm 1\}^{\mathcal{X}}$,

$$4\mathcal{V}_n^{PAC}(\mathcal{F}) \leq \mathcal{V}_n^{NR}(\mathcal{F}) \leq \mathcal{V}_n^{stat}(\mathcal{F})$$

and for any $\mathcal{F} \subset \mathbb{R}^{\mathcal{X}}$,

$$\mathcal{V}_n^{NR}(\mathcal{F}) \leq \mathcal{V}_n^{stat}(\mathcal{F})$$

That is, if a class is statistically learnable then it is learnable under either the PAC model or the statistical estimation setting

Proof. Let us start with the PAC learning objective. Note that,

$$\mathbf{1}_{\{\hat{y}(x) \neq f^*(x)\}} = \frac{1}{4}(\hat{y}(x) - f^*(x))^2$$

Now note that,

$$\begin{aligned} \mathbb{P}_{x \sim D_x}(\hat{y}(x) \neq f^*(x)) &= \mathbb{E}_{x \sim D_X} [\mathbf{1}_{\{\hat{y}(x) \neq f^*(x)\}}] \\ &= \frac{1}{4} \mathbb{E}_{x \sim D_X} [(\hat{y}(x) - f^*(x))^2] \end{aligned}$$

Thus we conclude that

$$4\mathcal{V}_n^{PAC}(\mathcal{F}) \leq \mathcal{V}_n^{NR}(\mathcal{F})$$

Now to conclude the proposition we prove that the minimax rate for non-parametric regression is upper bounded by minimax rate for the statistical learning problem (under squared loss).

To this end, in NR we assume that $y = f^*(x) + \varepsilon$ for zero-mean noise ε . Now note that, Now note that, for any \hat{y} ,

$$\begin{aligned} (\hat{y}(x) - f^*(x))^2 &= (\hat{y}(x) - y + \varepsilon)^2 \\ &= (\hat{y}(x) - y)^2 - 2\varepsilon(\hat{y}(x) - y) + \varepsilon^2 \\ &= (\hat{y}(x) - y)^2 - (f^*(x) - y)^2 + (f^*(x) - y)^2 - 2\varepsilon(\hat{y}(x) - y) + \varepsilon^2 \\ &= (\hat{y}(x) - y)^2 - (f^*(x) - y)^2 + 2\varepsilon^2 - 2\varepsilon(\hat{y}(x) - y) \\ &= (\hat{y}(x) - y)^2 - (f^*(x) - y)^2 + 2\varepsilon^2 - 2\varepsilon(\hat{y}(x) - f^*(x) - \varepsilon) \\ &= (\hat{y}(x) - y)^2 - (f^*(x) - y)^2 - 2\varepsilon(\hat{y}(x) - f^*(x)) \end{aligned}$$

Taking expectation w.r.t. y (or ε) we conclude that,

$$\begin{aligned} \mathbb{E}_{x \sim D_X} [(\hat{y}(x) - f^*(x))^2] &= \mathbb{E}_{(x,y) \sim D} [(\hat{y}(x) - y)^2] - \mathbb{E}_{(x,y) \sim D} [(f^*(x) - y)^2] - \mathbb{E}_{x \sim D_X} [\mathbb{E}_\varepsilon [2\varepsilon(\hat{y}(x) - f^*(x))]] \\ &= \mathbb{E}_{(x,y) \sim D} [(\hat{y}(x) - y)^2] - \mathbb{E}_{(x,y) \sim D} [(f^*(x) - y)^2] \\ &= L_D(\hat{y}) - \inf_{f \in \mathcal{F}} L_D(f) \end{aligned}$$

where in the above distribution D has marginal D_X over \mathcal{X} and the conditional distribution $D_{Y|X=x} = N(f^*(x), \sigma)$. Hence we conclude that

$$\mathcal{V}_n^{NR}(\mathcal{F}) \leq \mathcal{V}_n^{stat}(\mathcal{F})$$

when we consider statistical learning under square loss. □