Lecture 13

Statistical Learning: First Steps

Sasha Rakhlin

Oct 22, 2018

Outline

Outline

Supervised Learning: data $\mathcal{S} = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ are i.i.d. from *unknown* distribution P.

> *Learning algorithm*: a mapping $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \longmapsto \widehat{f}_n$.

Goals:

- **Prediction:** small expected loss

$$\mathbf{L}(\widehat{f}_n) = \mathbb{E}_{X,Y}\ell(Y, \widehat{f}_n(X)).$$

  Here $(X, Y) \sim P$. Interpretation: good prediction on a random example from same population.

- **Estimation:** small $\|\widehat{f}_n - f^*\|$, or $\|\widehat{\theta} - \theta^*\|$, where f^* or θ^* are parameters of P (e.g. regression function $f^*(x) = \mathbb{E}[Y|X = x]$, or $f^*(x) = \langle \theta^*, x \rangle$, etc).

In this course, we mostly focus on prediction, but will also outline connections between prediction and estimation.

Why not estimate the underlying distribution P first?

This is in general a harder problem than prediction. Consider classification. We might be attempting to learn parts/properties of the distribution that are irrelevant, while all we care about is the "boundary" between the two classes.

Key difficulty: our goals are in terms of unknown quantities related to unknown P. Have to use empirical data instead. Purview of statistics.

For instance, we can calculate the *empirical loss* of $f : \mathcal{X} \to \mathcal{Y}$

$$\widehat{L}(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, f(X_i))$$

Quiz: what is random here?

1. $\widehat{\mathbf{L}}(\mathsf{f})$ for a given fixed f
2. $\widehat{\mathsf{f}}_n$
3. $\widehat{\mathbf{L}}(\widehat{\mathsf{f}}_n)$
4. $\mathbf{L}(\widehat{\mathsf{f}}_n)$
5. $\mathbf{L}(\mathsf{f})$ for a given fixed f

It is important that these are understood before we proceed further.

Theoretical analysis of performance is typically easier if $\widehat{f}_n$ has closed form (in terms of the training data).

E.g. ordinary least squares $\widehat{f}_n(x) = x^\top (X^\top X)^{-1} X^\top Y$.

Unfortunately, most ML and many Statistical procedures are not explicitly defined but arise as

- solutions to an optimization objective (e.g. logistic regression)
- as an iterative procedure without an immediately obvious objective function (e.g. AdaBoost, Random Forests, etc)

The Gold Standard

Within the framework we set up, the smallest expected loss is achieved by the *Bayes optimal* function

$$f^* = \arg\min_f \mathbf{L}(f)$$

where the minimization is over all (measurable) prediction rules $f : \mathcal{X} \to \mathcal{Y}$.

The value of the lowest expected loss is called the *Bayes error*:

$$\mathbf{L}(f^*) = \inf_f \mathbf{L}(f)$$

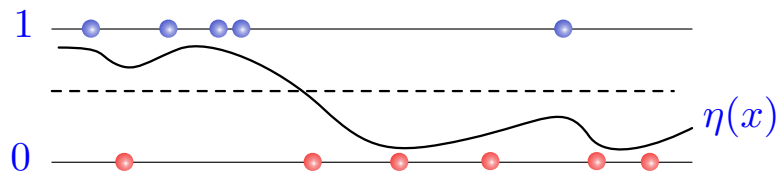
Of course, we cannot calculate any of these quantities since P is unknown.

Bayes Optimal Function

Bayes optimal function f^* takes on the following forms in these two particular cases:

- Binary classification ($\mathcal{Y} = \{0, 1\}$) with the indicator loss:

$$f^*(x) = \mathbf{I}\{\eta(x) \geq 1/2\}, \quad \text{where} \quad \eta(x) = \mathbb{E}[Y|X = x]$$



- Regression ($\mathcal{Y} = \mathbb{R}$) with squared loss:

$$f^*(x) = \eta(x), \quad \text{where} \quad \eta(x) = \mathbb{E}[Y|X = x]$$

The big question: is there a way to construct a learning algorithm with a guarantee that

$$\mathbf{L}(\widehat{f}_n) - \mathbf{L}(f^*)$$

is small for large enough sample size n?

Consistency

An algorithm that ensures

$$\lim_{n\to\infty} \mathbf{L}(\widehat{f}_n) = \mathbf{L}(f^*) \qquad \text{almost surely}$$

is called *consistent*. Consistency ensures that our algorithm is approaching the best possible prediction performance as the sample size increases.

The good news: consistency is possible to achieve.

- easy if $\mathcal{X}$ is a finite or countable set
- not too hard if $\mathcal{X}$ is infinite, and the underlying relationship between x and y is "continuous"

The bad news...

In general, we cannot prove anything "interesting" about $\mathbf{L}(\widehat{f}_n) - \mathbf{L}(f^*)$, unless we make further assumptions (incorporate *prior knowledge*).

What do we mean by "nothing interesting"? This is the subject of the so-called "No Free Lunch" Theorems. Unless we posit further assumptions,

- For any algorithm $\widehat{f}_n$, any n and any $\epsilon > 0$, there exists a distribution P such that $\mathbf{L}(f^*) = 0$ and

$$\mathbb{E}\mathbf{L}(\widehat{f}_n) \geq \frac{1}{2} - \epsilon$$

- For any algorithm $\widehat{f}_n$, and any sequence a_n that converges to 0, there exists a probability distribution P such that $\mathbf{L}(f^*) = 0$ and for all n

$$\mathbb{E}\mathbf{L}(\widehat{f}_n) \geq a_n$$

Reference: (Devroye, Györfi, Lugosi: *A Probabilistic Theory of Pattern Recognition*), (Bousquet, Boucheron, Lugosi, 2004)

is this really "bad news"?

Not really. We always have some domain knowledge.

Two ways of incorporating prior knowledge:

- Direct way: assumptions on distribution P (e.g. margin)
- Indirect way: redefine the goal to perform as well as a reference set $\mathcal{F}$ of predictors:

$$\mathbf{L}(\widehat{\mathsf{f}}_n) - \inf_{\mathsf{f} \in \mathcal{F}} \mathbf{L}(\mathsf{f})$$

  $\mathcal{F}$ encapsulates our *inductive bias*.

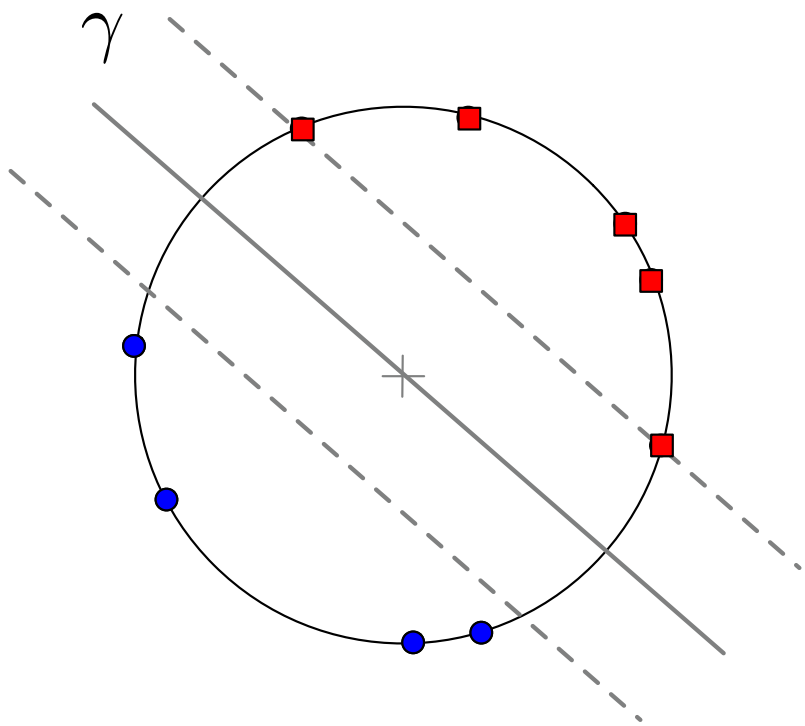
We often make both of these assumptions.

Outline

We start our study of Statistical Learning with the classical Perceptron algorithm.

Reason: simplicity. We will give a three-line proof of Perceptron, followed by two interesting consequences with one-line proofs each. These consequences are, perhaps, the easiest nontrivial statistical guarantees I can think of.

Perceptron



Perceptron

$(x_1, y_1), \ldots, (x_T, y_T) \quad \in \mathcal{X} \times \{\pm 1\}$ (T may or may not be same as n)

Maintain a hypothesis $w_t \in \mathbb{R}^d$ (initialize $w_1 = \mathbf{0}$).

On round t,

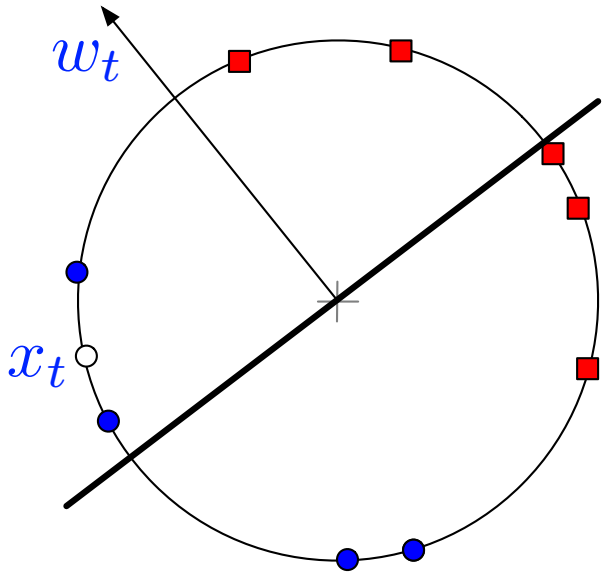
- Consider (x_t, y_t)
- Form prediction $\widehat{y}_t = \text{sign}(\langle w_t, x_t \rangle)$
- If $\widehat{y}_t \neq y_t$, update

$$w_{t+1} = w_t + y_t x_t$$

else

$$w_{t+1} = w_t$$

Perceptron



For simplicity, suppose all data are in a unit ball, $\|x_t\| \le 1$.

Definition of margin of $(x_1, y_1), \ldots, (x_T, y_T)$:

$$\gamma \;=\; \max_{\|w\|=1} \; \min_{i \in [T]} \; y_i \langle w, x_i \rangle$$

or $\gamma = 0$ if no margin.

Theorem (Novikoff '62): Perceptron makes at most $1/\gamma^2$ mistakes (and corrections) on **any** sequence of examples with margin γ.

Proof: Let m be the number of mistakes after T iterations. If a mistake is made on round t,

$$\|w_{t+1}\|^2 = \|w_t + y_t x_t\|^2 \leq \|w_t\|^2 + 2y_t \langle w_t, x_t \rangle + 1 \leq \|w_t\|^2 + 1.$$

Hence,

$$\|w_T\|^2 \leq m.$$

For optimal hyperplane w^*

$$\gamma \leq \langle w^*, y_t x_t \rangle = \langle w^*, w_{t+1} - w_t \rangle .$$

Hence (adding and canceling),

$$m\gamma \leq \langle w^*, w_T \rangle \leq \|w_T\| \leq \sqrt{m}.$$

More formally, for any T and $(x_1, y_1), \ldots, (x_T, y_T)$,

$$\sum_{t=1}^{T} \mathbf{I}\{y_t \langle w_t, x_t \rangle \leq 0\} \leq \frac{D^2}{\gamma^2}$$

where $\gamma = \gamma(x_{1:T}, y_{1:T})$ is margin and $D = D(x_{1:T}, y_{1:T}) = \max_t \|x_t\|$.