lions@epfl

EE-731: ADVANCED TOPICS IN DATA SCIENCES
LABORATORY FOR INFORMATION AND INFERENCE SYSTEMS
SPRING 2016

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

INSTRUCTOR: VOLKAN CEVHER

SCRIBER: SANKET DIWALE

OVERVIEW OF LEARNING THEORY

Index terms: *Probably Approximately Correct (PAC), Emperical Risk Minimization, Surrogate convex functions, Stability and Generalization*

The lecture formally addresses the question of feasibility of learning in a hypothesis space and links empirically observed prediction errors to the actual but unknown expected error when predicting quantities from an unknown distribution.

Section 1 presents the standard statistical learning model and casts some commonly encountered learning and estimation problems into the model's framework. Section 2 defines the notion of learnability in terms of Probably Approximately Correct (PAC) and Agnostic PAC learnability. Section 3 then presents the link between the process of Emperical Risk minimization and PAC learnability. Section 4 addresses the issue of selecting an appropriate hypothesis space and its effect on the expected risk. Many Empirical Risk minimization problems are in general NP hard, Section 5 presents the use of Convex Surrogate functions to solve such problems in a tractable way and yet achieve a bound on the risk function. Section 6 presents the theory of stability of learning algorithms and its to link to obtaining algorithm specific generalization error bounds.

1 The standard statistical learning model

In this section we describe a model that is used to describe and analyze the performance of learning algorithms. The model comprises of the following five components:

1. Training Data: The training data $\mathcal{D}_n := \{Z_i : 1 \leq i \leq n\}$ is a set of samples drawn independently and identically from an unknown distribution $\mathbb{P}$ defined on a set $\mathcal{Z}$.
2. Hypothesis Class: The hypothesis class given by $\mathcal{H}$, is a set of hypotheses h.
3. Loss Function: The loss function $f : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}$ is a function that describes the "loss" faced by choosing an hypothesis $h \in \mathcal{H}$ when a data sample $Z \in \mathcal{Z}$ is observed.
4. Risk: Given the loss function f, we would like to find an hypothesis $h \in \mathcal{H}$ that minimizes the "risk" or expected loss faced on drawing a new sample independently from the distribution $\mathbb{P}$. We thus define the risk function $F(h) := \mathbb{E}_{\mathbb{P}} f(h, Z)$, where $Z \sim \mathbb{P}$. Note the new is sample drawn independently from the same distribution as samples in $\mathcal{D}_n$. Note also that since the distribution $\mathbb{P}$ is unknown the function $F(h)$ is unknown and cannot be directly evaluated.
5. Goal: The goal of a learning algorithm is to find a "good hypothesis" $\hat{h}_n \in \mathcal{H}$ based on $\mathcal{D}_n$ such that $F(\hat{h}_n)$ is "small". We will look at how this is done in Section 3.

We now see how some of the most commonly encountered learning and estimation problems can be cast using the components of the above learning model.

Example 1. Binary Classification

1. The training data for a binary classification problem is a collection of input features X_i (say images) along with a binary (0-1) label Y_i.
2. The hypothesis class $\mathcal{H}$ is a set of classifiers $h : \mathcal{X} \to \{0, 1\}$. $\mathcal{H}$ can be a set of linear classifiers, a reproducing kernel Hilbert space, or all possible realizations of a deep network.
3. The loss function is a binary loss $f(h, Z) := \mathbb{1}_{\{Y_i \neq h(X_i)\}}$, where $Z \sim \mathbb{P}$ is independent of $\mathcal{D}_n$
4. Using the Binary loss function we define risk as $F(h) := \mathbb{E}_{\mathbb{P}} f(h, Z)$, which is the probability of false classification
5. The goal of binary classification is to find a "good" classifier based on the training data $\mathcal{D}_n$ such that the resulting probability of false classification is "small".

Remark. The goal of binary classification is to minimize the probability of false classification under the unknown distribution $\mathbb{P}$. However if the distribution $\mathbb{P}$ was known the the classifier that minimizes the probability of false classification under $\mathbb{P}$ would be the Bayes classifier,

$$h(x) = \mathbb{1}_{\{\mathbb{P}(Y=1|X=x) \geq 1/2\}},$$

In standard statistics this problem could be cast into a distribution estimation problem followed by Bayes classification, as is done in logistic regression.

Example 2. Logistic regression

1. The training data set considered is the same as that for binary classification.
2. For the hypothesis class, the class of linear classifiers $\mathcal{H} := \{\mathbb{1}_{\{\langle\cdot,\theta\rangle\leq 0\}} : \theta \in \Theta\}$ for some parameter space $\Theta \subseteq \mathbb{R}^p$ is considered. A statistical model relating the input X_i and output Y_i is assumed (canonical generalized linear model [1])
$$\mathsf{P}(Y_i = 1|X_i = x_i) = 1 - \mathsf{P}(Y_i = 0|X_i = x_i) = \frac{1}{1+\exp(-\langle x_i, \theta^\natural\rangle)},$$
for some $\theta^\natural \in \Theta$. (See [2] for a Bayesian interpretation.)
3. The same loss function and risk for a binary classification problem are considered.
4. The goal is to find the **maximum-likelihood** estimator
$$\hat{\theta}_n \in \arg\min_{\theta\in\Theta} L_n(\theta),$$
where L_n is the negative log-likelihood function. Output the classifier $\hat{h}_n(\cdot) = \mathbb{1}_{\{\langle\cdot,\hat{\theta}_n\rangle\leq 0\}}$.

Example 3. Linear regression

1. The training data, $Z_i = (X_i, Y_i) \in \mathcal{Z} = \mathbb{R}^p \times \mathbb{R}$, is a collection of input output data on a euclidean space.
2. A set of linear maps $\mathcal{H} = \{h_\theta(\cdot) = \langle\cdot,\theta\rangle : \theta \in \Theta\}$ parameterized using $\Theta \subseteq \mathbb{R}^p$ are considered as the hypothesis class
3. The loss function is the squared error $f(h_\theta, z) = (y - \langle x, \theta\rangle)^2$

Example 4. Density estimation

1. Training data: $Z_i \in \mathbb{R}$
2. Hypothesis class: A class of probability densities $\mathcal{P}$
3. Loss function: Negative log-likelihood $f(p, z) = -\log p(z)$

Example 5. K-means clustering/Vector quantization

1. The training data is a collection of points $Z_i \in \mathbb{R}^p$
2. A class of subsets of $\mathbb{R}^p$ of cardinality K is considered as the hypothesis class.
3. The loss function: $f(h, z) = \min_{c\in h} \|c - z\|_2^2$

2 Probably approximately correct (PAC) learnability

In this section we define the learnability of a function g in the hypothesis space $\mathcal{H}$ in the Probably approximately correct (PAC) sense.

Definition 1. *(PAC learnability [3])*
Assume that $Y_i = g(X_i)$ for some deterministic function *g, and that $g \in \mathcal{H}$. (Hence zero risk is possible.)*

*A hypothesis class $\mathcal{H}$ is PAC learnable, if there exists an algorithm $\mathcal{A}_\mathcal{H} : \mathcal{Z}^n \to \mathcal{H}$ and a function $n_\mathcal{H}(\varepsilon, \delta)$, such that for **every probability distribution** $\mathbb{P}$ and every $\varepsilon, \delta \in (0, 1)$, if $n \geq n_\mathcal{H}(\varepsilon, \delta)$, we have for the risk function $F(h)$*
$$F(\mathcal{A}(\mathcal{D}_n)) \leq \varepsilon, \quad \textit{(approximately correct)}$$
with probability at least $1 - \delta$ (probably).

Note that the above definition requires a $n_\mathcal{H}(\varepsilon, \delta)$ to exist for every ε, δ. But such an $n_\mathcal{H}(\varepsilon, \delta)$ will not exist as $\varepsilon \to 0$ for some $\delta < 1$ if g is not contained in $\mathcal{H}$ as none of the hypotheses in $\mathcal{H}$ can drive the risk to 0 with nonzero probability. The same is true if g is a mapping such that Y_i is a random variable. So to characterize learnability under such commonly occurring cases we define a weaker notion of learnability as follows.

Definition 2. *Definition (Agnostic PAC learnability [4])*
A hypothesis class $\mathcal{H}$ is agnostic PAC learnable, if there exist an algorithm $\mathcal{A}_\mathcal{H} : \mathcal{Z}^n \to \mathcal{H}$ and a function $n_\mathcal{H}(\varepsilon, \delta)$, such that for every probability distribution $\mathbb{P}$ and every $\varepsilon, \delta \in (0, 1)$, if $n \geq n_\mathcal{H}(\varepsilon, \delta)$, we have
$$F(\mathcal{A}(\mathcal{D}_n)) - \inf_{h\in\mathcal{H}} F(h) \leq \varepsilon,$$
with probability at least $1 - \delta$.

The agnostic PAC learnability simply requires the risk for the learned hypothesis to be within an ϵ distance from the infimum (minimum) risk possible within the given hypothesis class with probability $1-\delta$. The quantity $F(\mathcal{A}(\mathcal{D}_n)) - \inf_{h\in\mathcal{H}} F(h)$ is called the **excess risk**.

Observation 1. *PAC learnability implies $\mathbb{P}$ independent bound*
Note above that definitions of PAC learnability requires $n_{\mathcal{H}}(\varepsilon,\delta)$ to also be independent of the distribution $\mathbb{P}$ and thus the $n_{\mathcal{H}},\epsilon,\delta$ bounds on excess risk hold for any $\mathbb{P}$ even if the learning has been completed using data from one particular instance of $\mathbb{P}$ and thus PAC and agnostic PAC learnability is a strong property in this sense.

Observation 2. *Other types of learnability notions*
Distribution-dependent and localized formulations of PAC learnability unlike the ones above are presented in [5, 6, 7, 8].

A distribution dependent and localized PAC bound states that given an algorithm $\mathcal{A}_{\mathcal{H}} : \mathcal{Z}^n \to \mathcal{H}$, for every probability distribution $\mathbb{P}$ and every $\delta \in (0,1)$, we have

$$F(\mathcal{A}(\mathcal{D}_n)) - \inf_{h\in\mathcal{H}} F(h) \le \varepsilon_n(\mathbb{P}, h^\star; \mathcal{H}, \delta) \to 0,$$

with probability at least $1-\delta$, where $h^\star = \arg\min_{h\in\mathcal{H}} F(h)$ (assuming uniqueness).

3 Emperical Risk Minimization

Having defined the notion of PAC learnability in the previous section, we proceed to minimizing the unknown function $F(\mathcal{A}(\mathcal{D}_n))$ using an empirical approximation of the function. Since $\mathbb{P}$ is assumed unknown, we cannot directly solve the risk minimization problem

$$h^\star \in \arg\min_{h\in\mathcal{H}} F(h) := \mathbb{E}\, f(h, Z).$$

However, we can consider the empirical risk minimization problem as an approximation,

$$\hat{h}_n \in \arg\min_{h\in\mathcal{H}} \hat{F}_n(h) := \frac{1}{n}\sum_{i\le n} f(h, Z_i).$$

This is called the **ERM principle**, due to Vapnik and Chervonenkis.

By the strong law of large numbers (LLN), we know that $\hat{F}_n(h) \to F(h)$ almost surely for every $h \in \mathcal{H}$. So given convergence by the strong law of large numbers, what can we say about the $n_{\mathcal{H}}(\epsilon,\delta)$ for the PAC learnability.

Observation 3. *Convergence implied by the strong LLN*
For every $h \in \mathcal{H}$ and every probability distribution $\mathbb{P}$, there exists a function $n(\varepsilon,\delta; h, \mathbb{P})$, such that for every $\varepsilon,\delta \in (0,1)$, if $n \ge n(\varepsilon,\delta; h, \mathbb{P})$, we have

$$|\hat{F}_n(h) - F(h)| \le \varepsilon,$$

with probability at least $1-\delta$.

In other words the strong LLN only shows that for every hypothesis h, the emperical average of the $\hat{F}_n(h)$ converges almost surely to $F(h)$ for a large enough n that can change from one hypothesis to another. Thus we have still not shown PAC learnability, which requires the existence of an $n_{\mathcal{H}}(\epsilon,\delta)$ that is independent of the choice of h. To show this we can use the notion of convergence in a stronger sense.

Definition 3. *Uniform convergence*
A hypothesis class $\mathcal{H}$ has the uniform convergence property, if there exists a function $n_{\mathcal{H}}(\varepsilon,\delta)$, such that for every $\varepsilon,\delta \in (0,1)$ and any probability distribution $\mathbb{P}$, if $n \ge n_{\mathcal{H}}(\varepsilon,\delta)$, we have

$$\sup_{h\in\mathcal{H}} |\hat{F}_n(h) - F(h)| \le \varepsilon,$$

with probability at least $1-\delta$.

Proposition 3.1. *Uniform convergence implies learnability*
If the hypothesis class $\mathcal{H}$ has uniform convergence, i.e., for any $\varepsilon > 0$, $\exists\, n_{\mathcal{H}}(\varepsilon,\delta)$ s.t. $\forall\, n \ge n_{\mathcal{H}}(\varepsilon,\delta)$

$$\sup_{h\in\mathcal{H}} |\hat{F}_n(h) - F(h)| \le \varepsilon,$$

then for any $h^\star \in \arg\min_{h\in\mathcal{H}} F(h)$, we have

$$F(\hat{h}_n) - F(h^\star) \le 2\varepsilon.$$

Proof.

$$\begin{aligned} F(\hat{h}_n) - F(h^\star) &= F(\hat{h}_n) - \hat{F}_n(\hat{h}_n) + \hat{F}_n(\hat{h}_n) - \hat{F}_n(h^\star) + \hat{F}_n(h^\star) - F(h^\star) \\ &\le 2\sup_{h\in\mathcal{H}} |\hat{F}_n(h) - F(h)|. \end{aligned}$$

□

Remark. Note from the proof that Uniform convergence is only a sufficient condition for learnability.

The next theorem shows the relation of ERM and PAC learnability to uniform convergence for a binary classification problem.

Theorem 3.2 (See, e.g., [9]). *Assume that the hypothesis class $\mathcal{H}$ consists of only $\{0,1\}$-valued functions, and f is the $0-1$ loss. The following statements are equivalent.*

1. *The hypothesis class is agnostic PAC learnable.*
2. *The ERM is a good PAC learner.*
3. *The hypothesis class has the uniform convergence property.*

The above theorem states that for the binary classification problem with the 0-1 loss function, uniform convergence is both necessary and sufficient for PAC learnability. In general, uniform convergence may not be necessary for learnability and is only a sufficient condition.

Now having shown how the uniform convergence property of the hypothesis class affects learnability, we will see how to check for the uniform convergence property in a given class.

Proposition 3.3. *Uniform convergence property of a finite bounded hypothesis class*
Assume that the hypothesis class $\mathcal{H}$ consists of a finite number of functions taking values in $[0,1]$. Then $\mathcal{H}$ satisfies the uniform convergence property with

$$n_{\mathcal{H}}(\varepsilon,\delta) = \frac{\log(2|\mathcal{H}|/\delta)}{2\varepsilon^2}.$$

The proposition is a simple consequence of Hoeffding's inequality and the union bound.

Theorem 3.4 (Hoeffding's inequality (see, e.g., [11])).
Let $(\xi_i)_{1\le i\le m}$ be a sequence of independent $[0,1]$-valued random variables.
Let $S_n := (1/n)\sum_{1\le i\le n}(\xi_i - \mathbb{E}\,\xi_i)$.
Then for any $t>0$, $\mathbb{P}(|S_n|\ge t) \le 2\exp\left(-2nt^2\right)$.

From Hoeffding's inequality the proof for the proposition is as follows:

Proof. Define $\xi_i(h) = f(h,x_i)$, and define $S_n(h) := (1/n)\sum_{1\le i\le n}(\xi_i(h) - \mathbb{E}\,\xi_i(h))$ for every $h\in\mathcal{H}$. Notice that then

$$\sup_{h\in\mathcal{H}}|S_n(h)| = \sup_{h\in\mathcal{H}}|\hat{F}_n(h) - F(h)|.$$

By the union bound and Hoeffding's inequality, we have for any $t>0$,

$$\mathbb{P}\left(\sup_{h\in\mathcal{H}}|S_n(h)|\ge t\right) \le \sum_{h\in\mathcal{H}}\mathbb{P}\left(|S_n(h)|\ge t\right) \le |\mathcal{H}|\cdot 2\exp\left(-2nt^2\right).$$

Hence it suffices to choose

$$n_{\mathcal{H}}(\varepsilon,\delta) = \frac{\log(2|\mathcal{H}|/\delta)}{2\varepsilon^2}.$$

□

Observation 4. *Necessity of choosing a not-too-big hypothesis class*
We may write the proposition in another way:
For every probability distribution $\mathbb{P}$ and every $\delta\in(0,1)$, the ERM satisfies

$$\sup_{h\in\mathcal{H}}|\hat{F}_n(h) - F(h)| \le \varepsilon_n := \sqrt{\frac{\log(2|\mathcal{H}|/\delta)}{2n}},$$

with probability at least $1-\delta$.
If $|\mathcal{H}|$ is large, we need a large number of training data of the order $O(\log|\mathcal{H}|)$ to achieve a small excess risk ε_n.

*Otherwise, if ε_n is large, the values of $\hat{F}_n$ and F can be very different on certain hypotheses, and **overfitting** occurs. Thus to obtain reasonable fits with a small number of training samples, we need to keep $|\mathcal{H}|$ small as well. On the other hand if $|\mathcal{H}|$ is too small then $F(h)$ itself will be large and even though the excess risk is small the learned hypothesis will have a large risk.*

When $|\mathcal{H}|$ is not finite, the proof for learnability for the binary classification problem can be extended using the notion of the shattering coefficient. Consider the binary classification problem, in which $\mathcal{H}$ is a set of $\{0,1\}$-valued functions, and f is the $0-1$ loss.

Definition 4 (Shattering coefficient)**.** *The shattering coefficient of a hypothesis class $\mathcal{H}$ is defined as*

$$S_n(\mathcal{H}) := \sup_{x_1,\dots,x_n \in \mathcal{X}} |\{(h(x_i))_{1\le i\le n} : h \in \mathcal{H}\}|.$$

Definition 5 (Vapnik-Chervonenkis (VC) dimension)**.** *The VC dimension of a hypothesis class $\mathcal{H}$, denoted by* $\mathrm{VC}(\mathcal{H})$*, is defined as the largest integer k such that $S_k(\mathcal{H}) = 2^k$. If $S_k(\mathcal{H}) = 2^k$ for all k, then* $\mathrm{VC}(\mathcal{H}) := \infty$.

Theorem 3.5 ([12])**.** *Let $\mathcal{H}$ be a hypothesis class with VC dimension d. Then*

$$\sup_{h\in\mathcal{H}} |\hat{F}_n(h) - F(h)| \le 2\sqrt{\frac{2d\log(2en/d)}{n}} + \sqrt{\frac{\log(2/\delta)}{2n}},$$

with probability at least $1-\delta$.

4 Model Selection and Structural Risk Minimization

As we saw in the previous sections, an unknown function may only be agnostic PAC learnable in a given hypothesis space as the function may not belong to the hypothesis class selected. Further more as we increase the size of the hypothesis class the risk bound increases and for extremely large hypothesis classes the function may no longer remain learnable. In order to address the selection of an appropriate hypothesis space, we define the following terms. Let h_{opt} be a **global** minimizer of the risk $F(\cdot)$ which is not necessarily in $\mathcal{H}$. Let $h^\star$ be a minimizer of the risk $F(\cdot)$ on $\mathcal{H}$.
Then we can write

$$F(\hat{h}_n) - F(h_{\text{opt}}) = F(\hat{h}_n) - F(h^\star) + F(h^\star) - F(h_{\text{opt}}).$$

Definition 6 (Approximation error)**.** *The approximation error is defined as $\mathcal{E}_{app} = F(h^\star) - F(h_{opt})$.*

- The approximation error is fixed given a hypothesis class $\mathcal{H}$.
- The ERM can yield small risk only if $\mathcal{H}$ contains a "good enough" hypothesis.

Definition 7 (Estimation error)**.** *The estimation error is defined as $\mathcal{E}_{est} = F(\hat{h}_n) - F(h^\star)$.*

- The estimation error decreases with the training data size as the empirical risk minimizer approaches $h^\star$ (assuming a tractable solution to empirical risk minimization problem is available).

Observation 5. *Effect of shrinking the hypothesis class*
If we shrink the hypothesis class, while the estimation error $\mathcal{E}_{est}$ can be smaller, doing so can only increase the approximation error $\mathcal{E}_{app}$.

4.1 Model Selection

Model selection seeks a balance between approximation and estimation errors. Let $\mathcal{H}$ be a hypothesis class. Consider a countable family of sub-classes $\{\mathcal{H}_k : k \in \mathcal{K}\}$ such that $\bigcup_{k\in\mathcal{K}} \mathcal{H}_k = \mathcal{H}$. Denote by $\hat{h}_{n,k}$ an empirical risk minimizer chosen based on $\mathcal{D}_n$ in $\mathcal{H}_k$ for all $k \in \mathcal{K}$. The model selection problem asks to choose a $\hat{k}_n \in \mathcal{K}$ based on $\mathcal{D}_n$, such that

$$F(\hat{h}_{n,\hat{k}_n}) - F(h^\star) \le C \inf_{k\in\mathcal{K}} \left(\inf_{h\in\mathcal{H}_k} F(h) - F(h^\star) + \tilde{\pi}_n(k) \right),$$

with high probability for some constant $C > 0$ and $\tilde{\pi}_n(k) > 0$. Such an inequality on $F(\hat{h}_{n,\hat{k}_n}) - F(h^\star)$ is called an **oracle inequality**. If $C = 1$, the oracle inequality is called **sharp**.

4.2 Structural risk minimization (SRM)

The idea of structural risk minimization is to choose a subclass $k \in \mathcal{K}$ that **minimizes a risk estimate** (see, e.g., [13]). For this model selection process we choose,

$$\hat{k}_n \in \arg\min_{k\in\mathcal{K}} (\hat{F}_n(\hat{h}_{n,k}) + \pi_n(k)),$$

where $\pi_n(k)$ is some good estimate of $F(\hat{h}_{n,k}) - \hat{F}_n(\hat{h}_{n,k})$. We then output the hypothesis $\hat{h}_n = \hat{h}_{n,\hat{k}_n}$. The computational complexity of the selection process is completely ignored here.

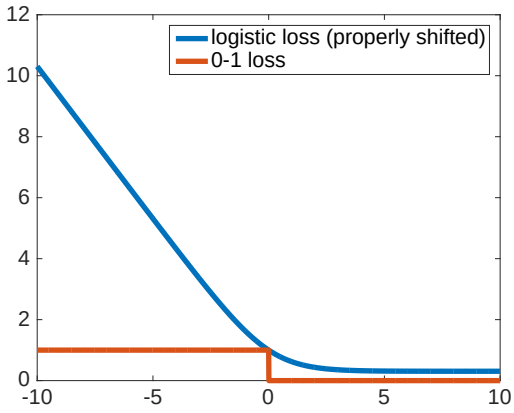

Figure 1: Logistic loss function

Theorem 4.1 ([14]).
Suppose there exists a double sequence $(R_{n,k})_{n\in\mathbb{N},k\in\mathcal{K}}$, such that for every $n \in \mathbb{N}$, $k \in \mathcal{K}$, and $\varepsilon > 0$,

$$\mathbb{P}\left(F(\hat{h}_{n,k}) > R_{n,k} + \varepsilon\right) \leq \alpha_n \exp(-2\beta_n\varepsilon^2),$$

for some constants $\alpha_n, \beta_n > 0$. Set $\pi_n(k) := R_{n,k} - \hat{F}_n(\hat{h}_{n,k}) + \sqrt{\beta_n^{-1}\log k}$. Then we have

$$F(\hat{h}_n) < \inf_k \left(\inf_{h\in\mathcal{H}_k} F(h) + \pi_n(k) + \sqrt{\frac{\log k}{n}} \right) + \varepsilon,$$

with probability at least $1 - 2\alpha_n \exp(-\beta_n\varepsilon^2/2) - 2\exp(-n\varepsilon^2/2)$.

In applying the above theorem the risk bound based on the VC dimension may be used [15, 13], but it can be loose since the bound is for the worst case. Hence it is important to find sharp **data dependent** risk estimates. See [16] for some recent advances.

5 Convex Surrogate Functions

Loss functions chosen for the emperical risk minimization often leads to NP hard optimization problems. In such cases convex surrogate functions are sought that provide a computationally tractable optimization problem for the risk minimization. The subsection below provides an example of a convex surrogate function used in binary classification.

5.1 Logistic regression as a learning algorithm

Given training data $(x_i, y_i) \in \mathbb{R}^p \times \{\pm 1\}$, $1 \leq i \leq n$ and a hypothesis class $\{\text{sign}(\langle x, \theta\rangle) : \theta \in \Theta\}$ (linear classifiers) for some $\Theta \subset \mathbb{R}^p$, we seek to solve the **empirical risk minimization** problem:

$$\hat{\theta}_n \in \arg\min_{\theta\in\Theta} \frac{1}{n} \sum_{1\leq i\leq n} \log\left[1 + \exp\left(-y_i\langle x_i, \theta\rangle\right)\right].$$

The output of the classifier is $\hat{h}_n(x) = \text{sign}(\langle x, \hat{\theta}_n\rangle)$.

Unlike the empirical risk minimization problem with the $0-1$ loss, the logistic regression approach yields a **convex optimization problem** that can be efficiently solved (when Θ is also convex). The intuition behind using the logistic loss function instead of the 0-1 loss function is shown in figure 1, where the logistic loss function behaves as a convex surrogate function that approximates the behavior of the 0-1 loss function. The Logistic loss function is given by, $\phi(t) = \log(1 + \exp(-t))$ while the $0-1$ loss function is given by $\phi(t) = \mathbb{1}_{\{t\leq 0\}}$. For logistic regression, t corresponds to $y\langle x, \theta\rangle$. The subsection below provides a general notion for the use of surrogate functions for binary classification.

5.2 Soft classification

Consider a **cost function** $g : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}$.

Definition 8 (Margin-based cost [17]).
A cost function g is called margin-based, if it can be written as $g(h, z) = \phi(yh(x))$ for some function ϕ.

For example, in logistic regression, $\phi(t) = \log(1 + \exp(-t))$, and $h \in \mathcal{H} = \{\langle \cdot, \theta \rangle : \theta \in \Theta\}$. The corresponding **empirical cost minimization** problem is given by

$$\hat{h}_n \in \arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{1 \le i \le n} \phi(y_i h(x_i)).$$

The corresponding **soft classifier** is given by

$$\tilde{h}_n(x) = \text{sign}(\hat{h}_n(x)).$$

Now if we define

$$H_\phi(\eta, \alpha) = \eta\phi(\alpha) + (1 - \eta)\phi(-\alpha), \quad \alpha_\phi^\star(\eta) = \arg\min_\alpha H_\phi(\eta, \alpha).$$

Let $F(h) = \mathbb{P}(\text{sign}(h(X)) \neq Y)$ denote the risk function, and $G(h) = \mathbb{E}\,\phi(Yh(X))$ be the expected cost function, then the following lemma provides a relation between the solution of the surrogate cost function minimization problem and the 0-1 loss risk minimization problem.

Lemma 5.1. *Zhang's lemma ([18])*
Assume that ϕ is convex, and $\alpha_\phi^\star(\eta) > 0$ when $\eta > 1/2$. If there exist $c > 0$ and $s \ge 1$ such that for all $\eta \in [0, 1]$,

$$|1/2 - \eta|^s \le c^s \left[H(\eta, 0) - H(\eta, \alpha^\star(\eta))\right]^{1/s},$$

Then for any hypothesis h,

$$F(h) - \min_h F(h) \le 2c \left[G(h) - \min_h G(h)\right]^{1/s}.$$

The lemma can be used to justify the use of soft classifiers as done in the previous subsection where the deviation of the surrogate cost from the minimized surrogate cost for any hypothesis bounds the deviation of the 0-1 loss function from its minimum. For the logistic regression it can be shown that, $c = 1/\sqrt{2}$ and $s = 2$ in the statement for Zhang's lemma can be used to justify the use of the logistic loss function. The lemma is also useful in several other problems as will be shown below.

5.3 Risk bound for ℓ_1-regularized logistic regression

Theorem 5.2. *Consider the ℓ_1-regularized logistic regression with $\Theta = \{\theta : \|\theta\|_1 \le \nu\}$ for some $\nu > 0$. Assume that $\|x\|_\infty \le 1$ for all $x \in \mathcal{X}$. Then there exists a constant $C > 0$ depending only on p, such that with probability at least $1 - \delta$,*

$$F(\hat{h}_n) - \inf_h F(h) \le 4\left(\nu\sqrt{\frac{C}{n}} + \sqrt{\frac{2\log(1/\delta)}{n}}\right)^{1/2} + \sqrt{2}\left[\left(\inf_{h \in \mathcal{H}} G(h)\right) - \left(\inf_h G(h)\right)\right]^{1/2}.$$

Proof. Similar to Theorem 4.4 in [19]. □

The right-hand side may be viewed as the sum of the estimation error and approximation error (w.r.t. the cost).

5.4 Other examples

Example 6. AdaBoost (See, e.g., [20])
Adaboost is equivalent to solving an empirical cost minimization problem with $\phi(t) = \exp(-t)$, for which $s = 2$ and $c = 1/\sqrt{2}$.

In practice however, AdaBoost may not be implemented by directly solving the empirical cost minimization problem.

Example 7. Support vector machine (See, e.g., [21]) The hinge cost function used by the support vector machine (SVM) corresponds to $\phi(t) = \max(0, 1 - t)$, for which $s = 1$ and $c = 1/2$.

6 Stability

An important property of learning algorithms is given by the notion of stability of its hypothesis choice under changing training sets. Below we look at a commonly used learning algorithm SVM (Support Vector Machine) and analyze its performance using the notion of stability. A linear SVM is given by

$$\hat{\theta}_n \in \arg\min_{\theta \in \Theta} \frac{1}{n} \sum_{1 \le i \le n} \phi(y_i \langle x_i, \theta \rangle) + \lambda \|\theta\|_2^2,$$

for some $\lambda > 0$, where $\phi(t) = \max(0, 1 - t)$ is the hinge loss. The output classifier is given by $\tilde{h}_n(\cdot) = \text{sign}(\langle \cdot, \hat{\theta}_n \rangle)$.

Since we have added a ℓ_2 regularizer on θ in the SVM formulation this is no longer a empirical cost minimization problem as considered by in the previous section. In order to analyse the risk bounds for such an algorithm, we instead resort to an algorithm dependent analysis as opposed to considering a full class of algorithms for which the bounds should hold. The notion of stability of a learning algorithm as defined below plays a key role in deriving such algorithm specific risk bounds.

Definition 9 (Classification stability [22])**.**
Consider the soft classification setting, where $\mathcal{H}$ is a class of soft classifiers (i.e., $\tilde{h}_n = \text{sign}(\hat{h}_n)$). For any $\mathcal{D}_n = \{z_1, \dots, z_n\}$, define $\mathcal{D}_n^{\setminus i}$ as $\mathcal{D}_n$ with the i-th element z_i removed. An algorithm $\mathcal{A}$ has classification stability with parameter $\beta > 0$, if for all $\mathcal{D}_n \subset \mathcal{Z}$ and for all $1 \le i \le n$,

$$\|\mathcal{A}(\mathcal{D}_n) - \mathcal{A}(\mathcal{D}_n^{\setminus i})\|_{L_\infty} \le \beta.$$

Then by the triangle inequality,

$$\|\mathcal{A}(\mathcal{D}_n) - \mathcal{A}(\mathcal{D}_n \cup \{z\})\|_{L_\infty} \le 2\beta, \quad \text{for all } z \in \mathcal{Z},$$

meaning the algorithm is robust to a small change of the training data.

Now, consider the $0-1$ loss $f(h,z) = \mathbb{1}_{\{\text{sign}(h(x)) \neq y\}}$. Then the risk $F(h) = \mathbb{E}\, f(h,Z)$ is the probability of classification error. Defining the margin-based loss

$$f^\gamma(h,z) = \begin{cases} 1 & \text{for } yh(x) \neq 0 \\ 1 - yh(x)/\gamma & \text{for } 0 \le yh(x) \le \gamma \\ 0 & \text{for } yh(x) \ge \gamma \end{cases},$$

and the corresponding margin-based empirical risk $\hat{F}_n^\gamma(h) = (1/n) \sum_{1 \le i \le n} f_\gamma(h, z_i)$.

Theorem 6.1 ([22])**.** *Stability implies generalization*
Let $\mathcal{A}$ be a soft classification algorithm that possesses classification stability with parameter $\beta_n > 0$. Then for any $\gamma > 0$, $n \in \mathbb{N}$, and any $\delta \in (0,1)$,

$$F(\mathcal{A}(\mathcal{D}_n)) \le \hat{F}_n^\gamma(\mathcal{A}(\mathcal{D}_n)) + 2\frac{\beta_n}{\gamma} + \left(1 + 4n\frac{\beta_n}{\gamma}\right)\sqrt{\frac{\log(1/\delta)}{2n}}.$$

Remark.
Notice that the uniform convergence property is not required.

Now to analyze the performance of the linear SVM classifier we look at the stability properties of SVM.

Theorem 6.2. *Risk bound for the linear SVM*
Assume that $\|x\|_2 \le \kappa$ for all $x \in \mathcal{X}$ for some $\kappa > 0$. Then $\mathcal{A}_{SVM}$ has classification stability with parameter $\beta_n = \kappa^2/(2\lambda n)$. Hence for any $n \in \mathbb{N}$ and $\delta \in (0,1)$,

$$F(\mathcal{A}_{SVM}(\mathcal{D}_n)) \le \hat{F}_n^1(\mathcal{A}_{SVM}(\mathcal{D}_n)) + \frac{\kappa^2}{\lambda n} + \left(1 + \frac{2\kappa^2}{\lambda}\right)\sqrt{\frac{\log(1/\delta)}{2n}},$$

with probability at least $1 - \delta$.

Proof. Similar to Example 2 in [22]. □

7 Summary

- Learning theory is concerned with developing learning algorithms that have **distribution-free** guarantees.
- The **ERM** principle provides a principled approach, if the **uniform convergence** property holds.
- **SRM** is an extension of the ERM principle that seeks a balance between the **estimation error** and the **approximation error**.
- In practice, we may replace the loss by an **convex surrogate** to yield an efficiently solvable **empirical cost minimization** problem.
- **Stability** provides another algorithm-wise analysis framework.

References

[1] P. McCullagh and J. A. Nelder, *Generalized Linear Models*, 2nd ed. London: Chapman and Hall, 1989.

[2] M. I. Jordan, "Why the logistic function? a tutorial discussion on probabilities and neural networks," MIT Computational Cognitive Science Report 9503, 1995.

[3] L. G. Valiant, "A theory of the learnable," *Commun. ACM*, vol. 27, no. 11, pp. 1134–1142, November 1984.

[4] D. Haussler, "Decision theoretic generalizations of the PAC model for neural net and other learning applications," *Inf. Comput.*, vol. 100, pp. 78–150, 1992.

[5] P. L. Bartlett, O. Bousquet, and S. Mendelson, "Local Rademacher complexities," *Ann. Stat.*, vol. 33, no. 4, pp. 1497–1537, 2005.

[6] P. L. Bartlett and S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results," *J. Mach. Learn. Res.*, vol. 3, 2002.

[7] V. Koltchinskii and D. Panchenko, "Rademacher processes and bounding the risk of function learning," 2004, arXiv:math/0405338v1 [math.PR].

[8] V. Koltchinskii, "Local Rademacher complexities and oracle inequalities in risk minimization," *Ann. Stat.*, vol. 34, no. 6, pp. 2593–2656, 2006.

[9] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning*. Cambridge, UK: Cambridge Univ. Press, 2014.

[10] K.-U. Höffgen and H.-U. Simon, "Robust trainability of single neurons," *J. Comput. Syst. Sci.*, vol. 50, pp. 114–125, 1995.

[11] P. Massart, *Concentration Inequalities and Model Selection*. Berlin: Springer-Verl., 2007.

[12] V. N. Vapnik and A. Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," *Theory Probab. Appl.*, vol. XVI, no. 2, pp. 264–280, 1971.

[13] V. N. Vapnik, *Statistical Learning Theory*. New York, NY: John Wiley & Sons, 1998.

[14] P. L. Bartlett, S. Boucheron, and G. Lugosi, "Model selection and error estimation," *Mach. Learn.*, vol. 48, pp. 85–113, 2002.

[15] G. Lugosi and K. Zeger, "Concept learning using complexity regularization," *IEEE Trans. Inf. Theory*, vol. 42, no. 1, pp. 48–54, 1996.

[16] V. Koltchinskii, *Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems*. Berlin: Springer-Verl., 2011.

[17] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, 1995.

[18] T. Zhang, "Statistical behavior and consistency of classification methods based on convex risk minimization," *Ann. Stat.*, vol. 32, no. 1, pp. 56–134, 2004.

[19] S. Boucheron, O. Bousquet, and G. Lugosi, "Theory of classification: A survey of some recent advances," *ESAIM: Probab. Stat.*, vol. 9, pp. 323–375, Nov. 2005.

[20] R. E. Schapire and Y. Freund, *Boosting*. Cambridge, MA: MIT Press, 2012.

[21] I. Steinwart and A. Christmann, *Support vector machines*. New York, NY: Springer, 2008.

[22] O. Bousquet and A. Elisseeff, "Stability and generalization," *J. Mach. Learn. Res.*, vol. 2, pp. 499–526, 2002.