Machine Learning

Learning Model

Fabio Vandin

October 14^{th}, 2022

Finite Hypothesis Classes

Assume $\mathcal{H}$ is a finite class: $|\mathcal{H}| < \infty$

Let h_S be the output of $\text{ERM}_{\mathcal{H}}(S)$, i.e. $h_S \in \arg\min_{h \in \mathcal{H}} L_S(h)$

Assumptions

- **Realizability:** there exists $h^* \in \mathcal{H}$ such that $L_{\mathcal{D},f}(h^*) = 0$
- **i.i.d.:** examples in the training set are independently and identically distributed (i.i.d) according to $\mathcal{D}$, that is $S \sim \mathcal{D}^m$

Observation: realizability assumption implies that $L_S(h^*) = 0$

Can we *learn* (i.e., find using ERM) h^*?

(Simplified) PAC learning

Probably Approximately Correct (PAC) learning

Since the training data comes from $\mathcal{D}$:

- we can only be **approximately** correct
- we can only be **probably** correct

Parameters:

- *accuracy parameter* ε: we are satisfied with a *good* h_S: $L_{\mathcal{D},f}(h_S) \leq \varepsilon$ (ε small)
- *confidence parameter* δ: want h_S to be a *good* hypothesis with probability $\geq 1 - \delta$ (δ small)

Theorem

Let $\mathcal{H}$ *be a finite hypothesis class. Let* $\delta \in (0,1)$, $\varepsilon \in (0,1)$, *and* $m \in \mathbb{N}$ *such that*

$$m \geq \frac{\log(|\mathcal{H}|/\delta)}{\varepsilon}.$$

m = # of training samples = |S|

Then for any f *and any* $\mathcal{D}$ *for which the realizability assumption holds, with probability* $\geq 1-\delta$ *we have that for every ERM hypothesis* h_S *it holds that*

$$L_{\mathcal{D},f}(h_S) \leq \varepsilon.$$

data distribution (unknown)

true labeling function (unknown)

Note: $\log$ = natural logarithm

With finite hypothesis classes, I can "always" find a good hypothesis $\to L_{\mathcal{D},f}(h_S) \leq \varepsilon$ with prob. $\geq 1-\delta$ if I have enough data.

for any $\mathcal{H}$

for any $\mathcal{D}$

for any f

if $m \geq \frac{\log(|\mathcal{H}|/\delta)}{\varepsilon}$

Proof (see book as well, Corollary 2.3)

Let $S|_x = \{x_1, x_2, \ldots, x_m\}$ be the instances in the training set S.

We want to bound (i.e., an upper bound) to: $\mathcal{D}^m(\{S|_x : L_{\mathcal{D},f}(h_S) > \varepsilon\})$.

Let $\mathcal{H}_B = \{h \in \mathcal{H} : L_{\mathcal{D},f}(h) > \varepsilon\}$ (BAD HYPOTHESES)

and $M = \{S|_x : \exists h \in \mathcal{H}_B, L_S(h) = 0\}$ (MISLEADING SAMPLES)

Since we have the realizability assumption: $L_S(h_S) = 0$

$\Rightarrow L_{\mathcal{D},f}(h_S) > \varepsilon$ only if some $h \in \mathcal{H}_B$ has $L_S(h) = 0$.

That is, our training data must be in the set M (for this to happen): $\{S|_x : L_{\mathcal{D},f}(h_S) > \varepsilon\} \subseteq M$.

Note that: $M = \bigcup_{h \in \mathcal{H}_B} \{S|_x : L_S(h) = 0\}$ because of

Therefore $\mathcal{D}^m(\{S|_x : L_{\mathcal{D},f}(h_S) > \varepsilon\}) \leq \mathcal{D}^m(M) = \mathcal{D}^m\left(\bigcup_{h \in \mathcal{H}_B} \{S|_x : L_S(h) = 0\}\right)$

(union bound) $\leq \sum_{h \in \mathcal{H}_B} \mathcal{D}^m(\{S|_x : L_S(h) = 0\})$ $(*)$

Now let's fix $h \in \mathcal{H}_B$: $L_S(h)=0 \iff \forall i=1,\dots,m : h(x_i)=f(x_i)$

Therefore: $\mathcal{D}^m(\{S|_x : L_S(h)=0\}) = \mathcal{D}^m(\{S|_x : \forall i=1,\dots,m,\ h(x_i)=f(x_i)\})$

(because $x_1,\dots,x_m$ are i.i.d. from $\mathcal{D}$) $\;= \prod_{i=1}^{m} \mathcal{D}(\{x_i : h(x_i)=f(x_i)\})$ (**)

Consider some i, $1 \le i \le m$: $\mathcal{D}(\{x_i : h(x_i)=f(x_i)\}) = 1 - \underbrace{\mathcal{D}(\{x_i : h(x_i)\neq f(x_i)\})}_{L_{\mathcal{D},f}(h)=\Pr_{x\sim\mathcal{D}}[h(x)\neq f(x)]}$

$= 1 - L_{\mathcal{D},f}(h)$

(since $h \in \mathcal{H}_B$) $\le 1-\varepsilon$

$\le e^{-\varepsilon}$

(because ... Taylor's expansion)

$$e^x = \sum_{n=0}^{+\infty} \left(\frac{x^n}{n!}\right) = 1 + x + \frac{x^2}{2!} + \dots \Rightarrow e^{-x} \geq 1-x$$

Combining this with (**):

$\mathcal{D}^m(\{S|_x : L_S(h)=0\}) \le \prod_{i=1}^{m} e^{-\varepsilon} = e^{-m\varepsilon}$

Combining the above with (*): $\mathcal{D}^m(\{S|_x : L_{\mathcal{D},f}(h_S) > \varepsilon\}) \le \sum_{h\in\mathcal{H}_B} e^{-m\varepsilon} = |\mathcal{H}_B| e^{-m\varepsilon}$

$\le |\mathcal{H}| e^{-m\varepsilon}$

Now, given the choice of m:

$\le |\mathcal{H}| e^{-\cancel{\varepsilon} \log(|\mathcal{H}|/\delta) \cdot \frac{1}{\cancel{\varepsilon}}} = |\mathcal{H}| \cdot \frac{\delta}{|\mathcal{H}|} = \delta$

requires $|\mathcal{H}| < +\infty$ □

PAC Learning

Definition (PAC learnability)

A hypothesis class $\mathcal{H}$ is *PAC learnable* if there exist a function $m_{\mathcal{H}}: (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\delta, \varepsilon \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f: \mathcal{X} \to \{0,1\}$, if the realizability assumption holds with respect to $\mathcal{H}$,$\mathcal{D}$,f, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples generate by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability $\geq 1 - \delta$ (over the choice of examples): $L_{\mathcal{D},f}(h) \leq \varepsilon$.

PAC Learning

Definition (PAC learnability)

A hypothesis class $\mathcal{H}$ is *PAC learnable* if there exist a function $m_{\mathcal{H}}$: $(0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\delta, \varepsilon \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function f:$\mathcal{X} \to \{0,1\}$, if the realizability assumption holds with respect to $\mathcal{H}$,$\mathcal{D}$,f, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples generate by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability $\geq 1 - \delta$ (over the choice of examples): $L_{\mathcal{D},f}(h) \leq \varepsilon$.

$m_{\mathcal{H}}$: $(0,1)^2 \to \mathbb{N}$: *sample complexity* of learning $\mathcal{H}$.

PAC Learning

Definition (PAC learnability)

A hypothesis class $\mathcal{H}$ is *PAC learnable* if there exist a function $m_{\mathcal{H}}: (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\delta, \varepsilon \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f: \mathcal{X} \to \{0,1\}$, if the realizability assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples generate by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability $\geq 1 - \delta$ (over the choice of examples): $L_{\mathcal{D},f}(h) \leq \varepsilon$.

$m_{\mathcal{H}}: (0,1)^2 \to \mathbb{N}$: *sample complexity* of learning $\mathcal{H}$.

- $m_{\mathcal{H}}$ is the minimal integer that satisfies the requirements.

Corollary

Every finite hypothesis class is PAC learnable with sample complexity $m_{\mathcal{H}}(\varepsilon, \delta) \leq \left\lceil \frac{\log(|\mathcal{H}|/\delta)}{\varepsilon} \right\rceil$. (What is the algorithm? ERM)

A More General Learning Model: Remove Realizability Assumption (Agnostic PAC Learning)

Realizability Assumption: there exists $h^* \in \mathcal{H}$ such that $L_{\mathcal{D},f}(h^*) = 0$

Informally: the label is fully determined by the instance x

A More General Learning Model: Remove Realizability Assumption (Agnostic PAC Learning)

Realizability Assumption: there exists $h^* \in \mathcal{H}$ such that $L_{\mathcal{D},f}(h^*) = 0$

Informally: the label is fully determined by the instance x

$\Rightarrow$ Too strong in many applications!

Relaxation: $\mathcal{D}$ is a probability distribution over $\mathcal{X} \times \mathcal{Y}$
$\Rightarrow$ $\mathcal{D}$ is the *joint distribution* over domain points and labels.

For example, two components of $\mathcal{D}$:

- $\mathcal{D}_x$: (marginal) distribution over domain points
- $\mathcal{D}((x, y)|x)$: conditional distribution over labels for each domain point

[Now $y \neq$ from $f(x)$ for some $f(\cdot)$]

A More General Learning Model: Remove Realizability Assumption (Agnostic PAC Learning)

Realizability Assumption: there exists $h^* \in \mathcal{H}$ such that $L_{\mathcal{D},f}(h^*) = 0$

Informally: the label is fully determined by the instance x

$\Rightarrow$ Too strong in many applications!

Relaxation: $\mathcal{D}$ is a probability distribution over $\mathcal{X} \times \mathcal{Y}$
$\Rightarrow$ $\mathcal{D}$ is the *joint distribution* over domain points and labels.

For example, two components of $\mathcal{D}$:

- $\mathcal{D}_x$: (marginal) distribution over domain points
- $\mathcal{D}((x, y)|x)$: conditional distribution over labels for each domain point

Given x, label y is obtained according to a conditional probability $\mathbb{P}[y|x]$.

The Empirical and True Error

With $\mathcal{D}$ that is a probability distribution over $\mathcal{X} \times \mathcal{Y}$ the *true error* (or risk) is:

$$L_{\mathcal{D}}(h) \stackrel{def}{=} \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y]$$

The Empirical and True Error

With $\mathcal{D}$ that is a probability distribution over $\mathcal{X} \times \mathcal{Y}$ the *true error* (or risk) is:

$$L_{\mathcal{D}}(h) \stackrel{def}{=} \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y]$$

As before $\mathcal{D}$ is not known to the learner; the learner only knows the training data S

Empirical risk: as before, that is

$$L_{\mathcal{S}}(h) \stackrel{def}{=} \frac{|\{i, 0 \leq i \leq m : h(x_i) \neq y_i\}|}{m}$$

The Empirical and True Error

With $\mathcal{D}$ that is a probability distribution over $\mathcal{X} \times \mathcal{Y}$ the *true error* (or risk) is:

$$L_{\mathcal{D}}(h) \stackrel{def}{=} \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y]$$

As before $\mathcal{D}$ is not known to the learner; the learner only knows the training data S

Empirical risk: as before, that is

$$L_{\mathcal{S}}(h) \stackrel{def}{=} \frac{|\{i, 0 \leq i \leq m : h(x_i) \neq y_i\}|}{m}$$

$\forall i$: probability pick (x_i, y_i) is $\frac{1}{m}$

Note: $L_{\mathcal{S}}(h)$ = probability that for a pair (x_i, y_i) taken uniformly at random from S the event "$h(x_i) \neq y_i$" holds.

$\hookrightarrow$ essentially: $\mathbb{E}[L_S(h)] = L_{\mathcal{D}}(h)$

Learner's goal: find $h : \mathcal{X} \to \mathcal{Y}$ minimizing $L_{\mathcal{D}}(h)$

An Optimal Predictor

Learner's goal: find $h : \mathcal{X} \to \mathcal{Y}$ minimizing $L_{\mathcal{D}}(h)$

Is there a *best predictor*?

Given a probability distribution $\mathcal{D}$ over $\mathcal{X} \times \{0, 1\}$, the best predictor is the **Bayes Optimal Predictor**

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \text{if } \mathbb{P}[y = 1|x] \geq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

Proposition

For any classifier $g : \mathcal{X} \to \{0, 1\}$, it holds $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

An Optimal Predictor

Learner's goal: find $h : \mathcal{X} \to \mathcal{Y}$ minimizing $L_{\mathcal{D}}(h)$

Is there a *best predictor*?

Given a probability distribution $\mathcal{D}$ over $\mathcal{X} \times \{0,1\}$, the best predictor is the **Bayes Optimal Predictor**

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \text{if } \mathbb{P}[y = 1|x] \geq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

Proposition

For any classifier $g : \mathcal{X} \to \{0,1\}$, it holds $L_{\mathcal{D}}(f_{\mathcal{D}}) \leq L_{\mathcal{D}}(g)$.

PROOF: Exercize

Can we use such predictor? No, because we don't know $\Pr[y=1|x]$ (we don't $\mathcal{D}$)

Agnostic PAC Learnability

Consider only predictors from a hypothesis class $\mathcal{H}$.

We are going to be ok with not finding the best predictor, but not being too far off.

Definition

A hypothesis class $\mathcal{H}$ is *agnostic PAC learnable* if there exist a function $m_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\delta, \varepsilon \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the learning algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ the algorithm returns a hypothesis h such that, with probability $\geq 1 - \delta$ (over the choice of the m training examples):

$$L_{\mathcal{D}}(h) \leq \min_{h' \in \mathcal{H}} L_{\mathcal{D}}(h') + \varepsilon.$$

Note: this is a generalization of the previous learning model.

Previously: - $\mathcal{D}$ was a distribution on $\mathcal{X}$ only
- $\min_{h' \in \mathcal{H}} L_{\mathcal{D}}(h') = 0$

A More General Learning Model: Beyond Binary Classification

Binary classification: $\mathcal{Y} = \{0, 1\}$

Other learning problems:

- multiclass classification: classification with > 2 labels
- regression: $\mathcal{Y} = \mathbb{R}$

Multiclass classification: same as before!

Regression

Domain set: $\mathcal{X}$ is usually $\mathbb{R}^p$ for some p.

Target set: $\mathcal{Y}$ is $\mathbb{R}$

Training data: (as before) $S = ((x_1, y_1), \ldots, (x_m, y_m))$

Learner's output: (as before) $h : \mathcal{X} \to \mathcal{Y}$

Loss: the previous one does not make much sense...

if the "observed value"/"true value" is 11.72, and I predict 11.71999, the predicted value $\neq$ "observed value", but as an error it is less than the error of predicting 1.

(Generalized) Loss Functions

Definition

Given any hypotheses set $\mathcal{H}$ and some domain Z, a *loss function* is any function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$

$Z = \mathcal{X} \times \mathcal{Y}$

(Generalized) Loss Functions

Definition

Given any hypotheses set $\mathcal{H}$ and some domain Z, a *loss function* is any function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$

Generalization error

Risk function = expected loss of a hypothesis $h \in \mathcal{H}$ with respect to $\mathcal{D}$ over Z:

$$L_{\mathcal{D}}(h) \stackrel{def}{=} \mathbb{E}_{z \sim \mathcal{D}}[\ell(h, z)]$$

(x, y)

(Generalized) Loss Functions

Definition

Given any hypotheses set $\mathcal{H}$ and some domain Z, a *loss function* is any function $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$

Risk function = expected loss of a hypothesis $h \in \mathcal{H}$ with respect to $\mathcal{D}$ over Z:

$$L_{\mathcal{D}}(h) \stackrel{def}{=} \mathbb{E}_{z \sim \mathcal{D}}[\ell(h, z)]$$

Empirical risk = expected loss over a given sample $S = (z_1, \ldots, z_m) \in Z^m$:

(x_1, y_1) (x_m, y_m)

$$L_S(h) \stackrel{def}{=} \frac{1}{m} \sum_{i=1}^{m} \ell(h, z_i)$$

(x_i, y_i)

Comparing $h(x_i)$ with y_i, how much do I lose? $\ell(h, z_i)$

Some Common Loss Functions

0-1 loss: $Z = \mathcal{X} \times \mathcal{Y}$

$$\ell_{0-1}(h, (x, y)) \stackrel{def}{=} \begin{cases} 0 & \text{if } h(x) = y \\ 1 & \text{if } h(x) \neq y \end{cases}$$

Some Common Loss Functions

0-1 loss: $Z = \mathcal{X} \times \mathcal{Y}$

$$\ell_{0-1}(h,(x,y)) \stackrel{def}{=} \begin{cases} 0 & \text{if } h(x) = y \\ 1 & \text{if } h(x) \neq y \end{cases}$$

Commonly used in binary or multiclass classification.

Squared loss: $Z = \mathcal{X} \times \mathcal{Y}$

$$\ell_{sq}(h,(x,y)) \stackrel{def}{=} (h(x) - y)^2$$

Some Common Loss Functions

0-1 loss: $Z = \mathcal{X} \times \mathcal{Y}$

$$\ell_{0-1}(h, (x, y)) \stackrel{def}{=} \begin{cases} 0 & \text{if } h(x) = y \\ 1 & \text{if } h(x) \neq y \end{cases}$$

Commonly used in binary or multiclass classification.

Squared loss: $Z = \mathcal{X} \times \mathcal{Y}$

$$\ell_{sq}(h, (x, y)) \stackrel{def}{=} (h(x) - y)^2$$

Commonly used in **regression**.

$\ell_{abs}(h, (x,y)) = |h(x) - y|$

Some Common Loss Functions

0-1 loss: $Z = \mathcal{X} \times \mathcal{Y}$

$$\ell_{0-1}(h, (x, y)) \stackrel{def}{=} \begin{cases} 0 & \text{if } h(x) = y \\ 1 & \text{if } h(x) \neq y \end{cases}$$

Commonly used in binary or multiclass classification.

Squared loss: $Z = \mathcal{X} \times \mathcal{Y}$

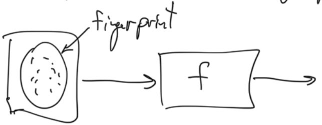
$$\ell_{sq}(h, (x, y)) \stackrel{def}{=} (h(x) - y)^2$$

Commonly used in **regression**.

Note: in general, the loss function may depend on the application!
But computational considerations play a role...

How to Choose the Loss Function?

Example classification of fingerprints



$$\begin{cases} +1 & \text{access} \\ -1 & \text{no access} \end{cases}$$

Two types of errors:
false accept and false reject

How do you choose the loss?

True value

Predicted value	+1	−1
+1	0 no error	1 false accept
−1	1 false reject	0 no error

What if you are a sig the system of CIA?

0	100
1	0

What about discounts at the supermarket?

0	1
10	0