Mathematical Tools in Machine Learning

Fadoua Balabdaoui

Seminar für Statistik, ETH

14 décembre 2019

Lecture 7 (Week 10)

VC dimension (continued)

Model Selection and Validation (Chapter 11)

Convex Learning Problems (Chapter 12)

Lecture 7

The Fundamental Theorem of PAC learning : Sauer's Lemma

- **Sauer's Lemma.** Let $\mathcal{H}$ be a hypothesis class with $\mathrm{VCdim}(\mathcal{H}) \leq d < \infty$. Then, for all $m \geq 1$

$$\tau_{\mathcal{H}}(m) \leq \sum_{i=0}^{d} \binom{m}{i}.$$

In particular if $m \geq d$ then $\tau_{\mathcal{H}}(m) \leq (em/d)^d$.

Proof (partial). For the first part, the proof is based on the fact that

$$|\mathcal{H}_C| \leq \left| \left\{ B \subseteq C : \mathcal{H} \text{ shatters } B \right\} \right|. \tag{1}$$

The inequality in (1) is shown by induction. To see that it gives the result, suppose $m \leq d$. Then,

$$\sum_{i=0}^{d} \binom{m}{i} \geq \sum_{i=0}^{m} \binom{m}{i} = 2^m \geq \tau_{\mathcal{H}}(m) \quad \text{(always true)}.$$

The Fundamental Theorem of PAC learning : Sauer's Lemma

Proof (partial) (continued). If $m > d$, then any $B \subseteq C$ such that $|B| \geq d + 1$ **cannot be shattered** by $\mathcal{H}$. Thus, if $B \subseteq C$ is shattered by $\mathcal{H}$ we **must have** $|B| \leq d$. Hence,

$$\{B \subseteq C : \mathcal{H} \text{ shatters } B\} \subseteq \cup_{i=0}^{d} \{B \subseteq C : |B| = i\}$$

with

$$\left| \cup_{i=0}^{d} \{B \subseteq C : |B| = i\} \right| = \sum_{i=0}^{d} \binom{m}{i}.$$

To show the 2nd assertion, note that for $m \geq d \Rightarrow d/m \in (0, 1]$. Hence

$$\left(\frac{d}{m}\right)^d \sum_{i=0}^{d} \binom{m}{i} \leq \sum_{i=0}^{d} \left(\frac{d}{m}\right)^i \binom{m}{i} \leq \sum_{i=0}^{m} \left(\frac{d}{m}\right)^i \binom{m}{i} = \left(1 + \frac{d}{m}\right)^m.$$

The proof follows from the fact hat $\exp\left(m \log(1 + \frac{d}{m})\right) \leq \exp(d)$. $\square$

The Fundamental Theorem of PAC learning : Uniform Convergence

- The following result links uniform approximation of the true error to the growth function of the class $\mathcal{H}$:

  Theorem. Let $\mathcal{H}$ be some hypothesis class with growth function $\tau_{\mathcal{H}}$. Also, suppose that $\ell \in [0,1]$. Then,

$$\mathbb{P}_{S\sim\mathcal{D}^m}\left(\sup_{h\in\mathcal{H}}|L_{\mathcal{D}}(h)-L_S(h)| \leq \frac{4+\sqrt{\log(\tau_{\mathcal{H}}(2m))}}{\delta\sqrt{2m}}\right) \geq 1-\delta, \forall\ \delta \in (0,1).$$

- Now, we can finish the proof of Theorem 6.7 : recall that the theorem states that the following assertions
  - **1.** $\mathcal{H}$ has the uniform convergence property.
  - **2.** Any ERM rule is a successful agnostic PAC learner for $\mathcal{H}$.
  - **3.** $\mathcal{H}$ is agnostic PAC learnable.
  - **4.** $\mathcal{H}$ is PAC learnable.

The Fundamental Theorem of PAC learning : Uniform Convergence

- **5.** Any ERM rule is a successful PAC learner for $\mathcal{H}$.
- **6.** $\mathcal{H}$ as a finite VC-dimension.

are all **equivalent** under the assumption that $\ell \in [0, 1]$. Furthermore, we have seen that we only need to show that $6 \Longrightarrow 1$.

Proof of $6 \Longrightarrow 1$ **:** It follows from Sauer's Lemma that for all $m \geq d/2$, $\tau_{\mathcal{H}}(2m) \leq (2em/d)^d$. Combining this with the preceding Theorem (on uniform convergence) gives

$$\mathbb{P}_{S\sim\mathcal{D}^m}\left(\sup_{h\in\mathcal{H}} |L_{\mathcal{D}}(h) - L_S(h)| \leq \frac{4 + \sqrt{d\log(2em/d)}}{\delta\sqrt{2m}}\right) \geq 1 - \delta.$$

Let us assume that m is large enough so that $\sqrt{d\log(2em/d)} \geq 4$.

The Fundamental Theorem of PAC learning : Uniform Convergence

Proof of $6 \implies 1$ **(continued).**

$$\mathbb{P}_{S\sim\mathcal{D}^m}\left(\sup_{h\in\mathcal{H}}|L_{\mathcal{D}}(h)-L_S(h)|\leq \frac{2\sqrt{d\log(2em/d)}}{\delta\sqrt{2m}}\right)\geq 1-\delta.$$

Now,

$$\frac{2\sqrt{d\log(2em/d)}}{\delta\sqrt{2m}}\leq\epsilon \iff m\geq\frac{2d}{(\epsilon\delta)^2}\log(m)+\frac{2d\log\left(\frac{2e}{d}\right)}{(\epsilon\delta)^2}.$$

Recall the inequality $\log(x)\leq\alpha x-\log(\alpha)-1,\quad\forall\ \alpha,x>0.$

$$\begin{aligned}\frac{2d}{(\epsilon\delta)^2}\log(m) &\leq \frac{2d}{(\epsilon\delta)^2}\left(\frac{(\epsilon\delta)^2}{4d}m-\log\left(\frac{(\epsilon\delta)^2}{4d}\right)-1\right)\\ &= \frac{m}{2}-\frac{2d}{(\epsilon\delta)^2}\log\left(\frac{(\epsilon\delta)^2}{4d}\right)-\frac{2d}{(\epsilon\delta)^2}\end{aligned}$$

The Fundamental Theorem of PAC learning : Uniform Convergence

Proof of $6 \Longrightarrow 1$ **(end).** Therefore, it is enough to take m such that

$$\begin{aligned} m/2 &\geq -\frac{2d}{(\epsilon\delta)^2}\log\left(\frac{(\epsilon\delta)^2}{4d}\right) - \frac{2d}{(\epsilon\delta)^2} + \frac{2d\log(2e/d)}{(\epsilon\delta)^2} \\ &= \frac{2d}{(\epsilon\delta)^2}\log\left(\frac{4d}{(\epsilon\delta)^2}\right) - \frac{2d}{(\epsilon\delta)^2} + \frac{2d\log(2e/d)}{(\epsilon\delta)^2} \end{aligned}$$

and hence, it is sufficient to take m such that

$$m \geq \begin{cases} \frac{4}{(\epsilon\delta)^2}\log\left(\frac{4}{(\epsilon\delta)^2}\right) + \frac{4(\log(2e)-1)}{(\epsilon\delta)^2}, & \text{if } d = 1 \\ \frac{4d}{(\epsilon\delta)^2}\log\left(\frac{4d}{(\epsilon\delta)^2}\right), & \text{if } d \geq 2. \end{cases}$$

We conclude that $\mathcal{H}$ has the **the uniform convergence property**. $\square$

The Fundamental Theorem of PAC learning : Uniform Convergence

- The previous proof implies that $m_{\mathcal{H}}^{UC}(\epsilon,\delta) \asymp 1/(\epsilon\delta)^2 \log(1/(\epsilon\delta))$. **But**, Theorem 6.8 (p. 48) gives a much better bound :

$$C_1 \frac{d+\log(1/\delta)}{\epsilon^2} \leq m_{\mathcal{H}}^{UC}(\epsilon,\delta) \leq C_2 \frac{d+\log(1/\delta)}{\epsilon^2}$$

for some $0 < C_1 < C_2$.

- Also, $\mathcal{H}$ is **agnostic PAC learnable** ($\min_{h\in\mathcal{H}} L_{\mathcal{D}}(h)$ **not** necessarily $=$ **0**) with sample complexity

$$C_1 \frac{d+\log(1/\delta)}{\epsilon^2} \leq m_{\mathcal{H}}(\epsilon,\delta) \leq C_2 \frac{d+\log(1/\delta)}{\epsilon^2}$$

for some $0 < C_1 < C_2$, and if $\mathcal{H}$ is **PAC learnable** ($\min_{h\in\mathcal{H}} L_{\mathcal{D}}(h) = \mathbf{0}$)

$$C_1 \frac{d+\log(1/\delta)}{\epsilon} \leq m_{\mathcal{H}}(\epsilon,\delta) \leq C_2 \frac{d+\log(1/\delta)}{\epsilon}$$

for some $0 < C_1 < C_2$.

Some conclusions

1. VC classes for classification are defined through the notion of shattering a subset of size m of the domain set $\mathcal{X}$.
2. The VC dimension of some class is finite if there exists some size d such that the class cannot shatter any set of size $> d$.
3. Examples include : class of thresholds, of intervals, rectangles, etc...
4. The Fundamental Theorem of Learning says that a class is PAC learnable iff it is VC.
5. VC classes be defined for other learning problems : the collection of all subgraphs $\{(x,t) \in \mathcal{X} \times \mathbb{R} : t < h(x), h \in \mathcal{H}\}$ cannot pick out all subsets of a set $C \subset \mathcal{X} \times \mathbb{R}$ as soon as $|C| > d$ for some integer $d \geq 1$ (*Weak Convergence and Empirical Processes* by van der Vaart and Wellner, Section 2.6).

Lecture 7

VC dimension (continued)

Model Selection and Validation (Chapter 11)

Convex Learning Problems (Chapter 12)

Validation methods : Some theory

- Recall from Theorem 6.8 (p. 48) that when $\mathrm{VCdim}(\mathcal{H}) = d < \infty$, $\mathcal{H}$ has the uniform convergence property with

$$m_{\mathcal{H}}^{UC}(\epsilon, \delta) \leq C_2 \frac{d + \log(1/\delta)}{\epsilon^2}$$

for some $C_2 > 0$.

- By definition of the uniform convergence property : we have for $m \geq C_2(d + \log(1/\delta))/\epsilon^2$

$$\mathbb{P}_{S \sim \mathcal{D}^m} \left(\sup_{h \in \mathcal{H}} |L_{\mathcal{D}}(h) - L_S(h)| \leq \epsilon \right) \geq 1 - \delta. \tag{2}$$

- Without loss of generality, assume that

$$\frac{C_2(d + \log(1/\delta))}{\epsilon^2} \geq 1.$$

Validation methods : Some theory

- If m is taken such that $m = \left\lfloor 2C_2(d + \log(1/\delta))/\epsilon^2 \right\rfloor$, then it is easy to check that

$$\frac{C_2(1 + \log(1/\delta))}{\epsilon^2} \leq m \leq \frac{2C_2(1 + \log(1/\delta))}{\epsilon^2},$$

where the left inequality follows from

$$\left\lfloor \frac{2C_2(d + \log(1/\delta))}{\epsilon^2} \right\rfloor \geq \frac{2C_2(d + \log(1/\delta))}{\epsilon^2} - 1 \geq \frac{C_2(d + \log(1/\delta))}{\epsilon^2}.$$

- With $C = 2C_2$, it follows from (2) that

$$\mathbb{P}_{S \sim \mathcal{D}^m} \left(\sup_{h \in \mathcal{H}} |L_{\mathcal{D}}(h) - L_S(h)| \leq \sqrt{C \frac{d + \log(1/\delta)}{m}} \right) \geq 1 - \delta.$$

Validation methods : Some theory

- Our main **goal** : get a **good estimation** for $L_{\mathcal{D}}(h_S)$, with h_S the prediction rule returned by a learning algorithm.

- Let S be a training set, and $V = \{(x_1^v, y_1^v), \dots, (x_{m_v}^v, y_{m_v}^v)\}$ be **another set of independent m_v examples**, which are **independent** of S and generated from the same distribution $\mathcal{D}$.

Theorem. Let $h = h_S$ be some predictor (based on S) and assume that the loss function $\ell \in [0,1]$. Then, for all $\delta \in (0,1)$, we have that

$$\mathbb{P}_{V \sim \mathcal{D}^{m_v}} \left(|L_{\mathcal{D}}(h_S) - L_V(h_S)| \leq \sqrt{\frac{\log(2/\delta)}{2m_v}} \,|S \right) \quad \geq \quad 1 - \delta, \text{a.s. } \mathcal{D}^m \text{ and}$$

$$\mathbb{P}_{(V,S) \sim \mathcal{D}^{m_v+m}} \left(|L_{\mathcal{D}}(h_S) - L_V(h_S)| \leq \sqrt{\frac{\log(2/\delta)}{2m_v}} \right) \quad \geq \quad 1 - \delta,$$

$L_V(h) = \frac{1}{m_v} \sum_{j=1}^{m_v} \ell(h, z_j^v)$, and $z_j^v = (x_j^v, y_j^v)$.

Validation methods : Some theory

- **Proof.** Recall that if $Y \perp Z$ are 2 random variables/vectors such that $(Y, Z) \sim \mathcal{D} = \mathcal{D}^Y \times \mathcal{D}^Z$, then for any measurable function $(y, z) \mapsto \psi(y, z)$ such that $\mathbb{E}[\psi(Y, Z)]$ exists, we have

$$\mathbb{E}_{\mathcal{D}}[\psi(Y, Z)|Z] = g(Z)$$

where $g(z) = \mathbb{E}_{\mathcal{D}^Y}[\psi(Y, z)]$.

- By the iterated law of expectations, we have $\mathbb{E}_{\mathcal{D}}[\psi(Y, Z)] = \mathbb{E}[g(Z)]$.

- Replace now Y and Z by S and V, and let

$$\psi(S, V) = \mathbb{1}_{\{|L_{\mathcal{D}}(h_S) - L_V(h_S)| \geq t\}}$$

for some threshold $t > 0$ to be determined.

Validation methods : Some theory

- **Proof (continued).** Then, a.s. $\mathcal{D}^m$

$$\mathbb{P}_{V\sim\mathcal{D}^{m_v}}\left(|L_{\mathcal{D}}(h_S) - L_V(h_S)| \geq t \ \ |S = s\right)$$
$$= \mathbb{P}_{V\sim\mathcal{D}^{m_v}}\left(|L_{\mathcal{D}}(h_s) - L_V(h_s)| \geq t\right)$$

and

$$\mathbb{P}_{(V,S)\sim\mathcal{D}^{m_v+m}}\left(|L_{\mathcal{D}}(h_S) - L_V(h_S)| \geq t\right)$$
$$= \int \mathbb{P}_{V\sim\mathcal{D}^{m_v}}\left(|L_{\mathcal{D}}(h_s) - L_V(h_s)| \geq t\right) d\mathcal{D}^m(s).$$

Since

$$L_{\mathcal{D}}(h_s) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(h_s,(x,y)), \quad \text{and } L_V(h_s) = \frac{1}{m_v}\sum_{j=1}^{m_v}\ell(h_s,(x_j^v, y_j^v))$$

Validation methods : The hold-out set

- **Proof (continued).** we can apply the Hoeffding's inequality (Chapter 4) to obtain

$$\mathbb{P}_{V\sim\mathcal{D}^{m_v}}\left(|L_{\mathcal{D}}(h_s)-L_V(h_s)|\geq\sqrt{\frac{\log(2/\delta)}{2m_v}}\right)$$

$$\leq 2\exp\left(-2m_v\sqrt{\frac{\log(2/\delta)}{2m_v}}^{\,2}\right)=2\exp\left(-2m_v\frac{\log(2/\delta)}{2m_v}\right)=\delta.\quad\square$$

- We get **a sharper bound/smaller error** if the size of V, m_v, is chosen such that

$$\frac{\log(2/\delta)}{2m_v}<C\frac{d+\log(1/\delta)}{m}\iff m_v>m\frac{\log(2/\delta)}{2C(d+\log(1/\delta))}.$$

- For example, it is enough to take $m_v\geq\frac{m}{2C}$, since we have $\log(2)+\log(1/\delta)<d+\log(1/\delta)$.

Validation methods : The hold-out set

- However, the price to pay is that we need the **additional sample** V.

- A cheaper method : split the original training set into 2 parts : (1) one part for **training (S)** and (2) the other part for **validation (V)**.

- The validation set, V, obtained this way is referred to as the **hold-out** set.

Validation methods : model selection

- Suppose we could use r different algorithms which output the following prediction rules $h_1, \ldots, h_r$. Note that for $i = 1, \ldots, r$ we have $h_i = h_{S,i}$ where S is the training set used to train the algorithms.

- Consider now $\mathcal{H} = \{h_1, \ldots, h_r\}$. To choose the best h_i, we look for i such that

$$L_V(h_i) = \min_{1 \le j \le r} L_V(h_j),$$

where V is a validation sample : a set which is **independent** of the original training S (used to obtain the prediction rules $h_1, \ldots, h_r$).

- In other words : to select the **best** *model*, we look for $\text{ERM}_{\mathcal{H}}$ using the validation sample V.

Validation methods : model selection

Theorem. Let $\mathcal{H} = \{h_1, \ldots, h_r\}$ be an arbitrary set of predictors (based on a training set S) and assume that the loss function $\ell \in [0,1]$. Assume that a validation set V of size m_v is sampled independently of any element in $\mathcal{H}$. Then,

$$\mathbb{P}_{V\sim\mathcal{D}^{m_v}}\left(\sup_{h\in\mathcal{H}} |L_{\mathcal{D}}(h) - L_V(h)| \leq \sqrt{\frac{\log(2r/\delta)}{2m_v}} \;\Big|\, S\right) \geq 1-\delta, \text{ (a.s. } \mathcal{D}^m)$$

and

$$\mathbb{P}_{(V,S)\sim\mathcal{D}^{m_v+m}}\left(\sup_{h\in\mathcal{H}} |L_{\mathcal{D}}(h) - L_V(h)| \leq \sqrt{\frac{\log(2r/\delta)}{2m_v}}\right) \geq 1-\delta.$$

- If $|\mathcal{H}| = r$ is **not too large**, then the bound on the maximal deviation between the true risk and the validation error of **any** h is sharp.

Validation methods : The model selection curve

- **Example** :

  $$x \sim \mathcal{U}[0,1], \quad y|x = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \epsilon, \quad \text{and} \quad \epsilon \sim \mathcal{U}[-1,1]$$

  with $\theta_0 = 5, \theta_1 = 2, \theta_2 = -1/2, \theta_3 = -3$.

- Let $\mathcal{P}_d$ be the set of **polynomials** of degree $d \geq 1$. Note that $\mathbb{E}[y|x] \in \mathcal{P}_3$. Let S be training set from $\mathcal{D}^m$.
- For $d \in \{1, \ldots, 9\}$ and $S \in \{S_1, \ldots, S_{20}\} : |S| = 50$ we compute
  - $h_d = h_{S,d} = \text{argmin}_{h \in \mathcal{P}_d} m^{-1} \sum_{i=1}^m (y_i - h(x_i))^2$,
  - $L_{\mathcal{D}}(h_{S,d}) = \mathbb{E}_{\mathcal{D}}[(y - h_{S,d}(x))^2]$ $(= \mathbb{E}_{(x,y)\sim\mathcal{D}}[(y - h_{S,d}(x))^2 \; |S]$ and $(x, y) \perp S)$.
- Also, we compute the averages $\sum_{i=1}^{20} L_{S_i}(h_{S_i,d})/20$ and $\sum_{i=1}^{20} L_{\mathcal{D}}(h_{S_i,d})/20$.

Validation methods : The model selection curve

- In this example, we have that

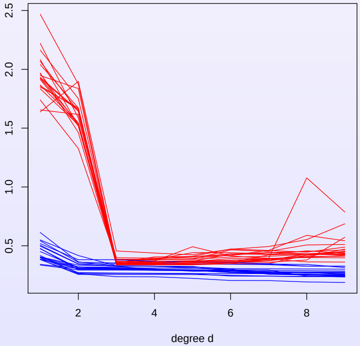
$$
\begin{aligned}
\mathbb{E}[(y-h_{S,d})^2] &= \mathbb{E}[(y-\mathbb{E}[y|x]+\mathbb{E}[y|x]-h_{S,d})^2] \\
&= \mathbb{E}[\epsilon^2]+2\mathbb{E}[(y-\mathbb{E}[y|x])(\mathbb{E}[y|x]-h_{S,d}(x))] \\
&\quad + \mathbb{E}[(\mathbb{E}[y|x]-h_{S,d})^2] \\
&= \mathbb{E}[\epsilon^2]+2\mathbb{E}\Big[\mathbb{E}[(y-\mathbb{E}[y|x])(\mathbb{E}[y|x]-h_{S,d}(x))|x]\Big] \\
&\quad + \mathbb{E}[(\mathbb{E}[y|x]-h_{S,d}(x))^2] \\
&= \frac{4}{12}+2\mathbb{E}\Big[(\mathbb{E}[y|x]-h_{S,d}(x))\underbrace{\mathbb{E}[(y-\mathbb{E}[y|x])|x]}_{=0}\Big] \\
&\quad + \mathbb{E}\Big[\Big(\theta_0 x+\theta_1 x+\theta_2 x^2+\theta_3 x^3-\sum_{i=0}^{d}\hat{\theta}_i(S)x^i\Big)^2\Big] \\
&= \frac{4}{12}+\mathbb{E}\Big[\Big(\theta_0 x+\theta_1 x+\theta_2 x^2+\theta_3 x^3-\sum_{i=0}^{d}\hat{\theta}_i(S)x^i\Big)^2\Big]
\end{aligned}
$$

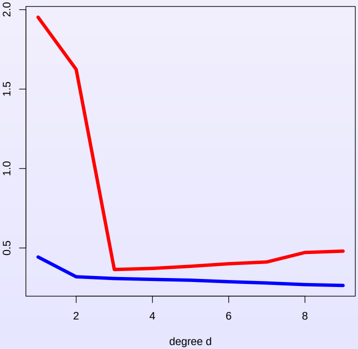
Validation methods : The model selection curve

- Recall that if $U \sim \mathcal{U}[0,1]$ then $\mathbb{E}[U^k] = 1/(k+1)$.

$$\mathbb{E}\Big[\Big(\sum_{i=0}^{3} \theta_i x^i - \sum_{i=0}^{d} \hat{\theta}_i(S) x^i\Big)^2\Big] = \sum_{0 \le i,j \le d} \frac{(\theta_i - \hat{\theta}_i(S))(\theta_j - \hat{\theta}_j(S))}{i+j+1}$$

with $\theta_i = 0$ if $i > 3$ (when $d > 3$).





Validation methods : The model selection curve

- Recall that $h_{S,d}$ is $\mathrm{ERM}_{\mathcal{P}_d}$ and so for a given training set S we have

$$L_S(h_{S,d}) = \min_{h \in \mathcal{P}_d} L_S(h).$$

- Since $\mathcal{P}_1 \subset \mathcal{P}_2 \ldots \subset \mathcal{P}_9$, $L_S(h_{S,1}) \geq L_S(h_{S,2}) \geq \ldots \geq L_S(h_{S,9})$, for any $S \sim \mathcal{D}^m$.
- However, the complexity of the model $\mathcal{P}_d$ **grows** : $L_S(h_{S,d})$ for large d **deviates** from its limit $L_{\mathcal{D}}(h_{S,d})$ because the sample size is **not big enough**. This is the reason $L_S(h_{S,d})$ **is not a good approximation** of $L_{\mathcal{D}}(h_{S,d})$ as d increases.

Validation methods : k-cross validation

- **Idea** : use the **same** training set for do both training and validation
- Let $A : S \mapsto \mathcal{H}$ be our learning algorithm. The k-cross validation method works the following way :
  - the training set S is partitioned into k different subsets (**folds**) of size $\approx m/k$,
  - for each fold S_i, we train A on the union of the remaining folds : $\cup_{j \neq i} S_j$, call the output $h^{(-i)}$
  - evaluate the error by computing $L_{S_i}(h^{(-i)})$,
  - compute the average error $1/k \sum_{i=1}^{k} L_{S_i}(h^{(-i)})$.
- If $k = m$, the procedure is also known under leave-one-out (LOO).

Validation methods : k-cross validation

- How to use the k-cross validation for model selection ? Suppose we have p **hypothesis classes** $\mathcal{H}_r, r = 1, \ldots, p$ and a **learning algorithm** A which can learn any of these class : for $r \in \{1, \ldots, p\}$ let

  $$A(\tilde{S}; r) \text{ be the rule in } \mathcal{H}_r \text{ returned by } A \text{ when receiving } \tilde{S}.$$

- For each $r \in \{1, \ldots, p\}$, do the following
  - for each $i = 1, \ldots, k$ compute $h_r^{(-i)}$ and evaluate $L_{S_i}(h_r^{(-i)})$,
  - compute $\mathrm{err}_r \equiv 1/k \sum_{i=1}^k L_{S_i}(h_r^{(-i)})$.

Then,

  - find $\hat{r}$ such that $\mathrm{err}_{\hat{r}} = \min_{1 \leq r \leq p} \mathrm{err}_r$,
  - output $A(S; \hat{r})$.

- There are settings under which the k-cross validation **works**, but a general theory is **hard** to establish (counterexample in Exercise 11.1).

Validation methods : what to do in practice and why

- Error decomposition (revisited) : Let $S \perp V$ be a training and validation sets. We have the decomposition

$$L_{\mathcal{D}}(h_S) = \underbrace{\Big(L_{\mathcal{D}}(h_S) - L_V(h_S)\Big)}_{\text{can be sharply bounded}} + \Big(L_V(h_S) - L_S(h_S)\Big) + \mathbf{L_S}(\mathbf{h_S}).$$

  - 1. $\mathbf{L_S}(\mathbf{h_S})$ small, but $\Big(L_V(h_S) - L_S(h_S)\Big)$ large : overfitting
  - 2. $\mathbf{L_S}(\mathbf{h_S})$ big : either underfitting / the hypothesis class is too small.

- Explanation of 2 : Let $h^* = \text{argmin}_{h \in \mathcal{H}} L_{\mathcal{D}}(h)$. We have the decomposition

$$\mathbf{L_S}(\mathbf{h_S}) = \underbrace{\Big(\mathbf{L_S}(\mathbf{h_S}) - L_S(h^*)\Big)}_{\leq 0} + \underbrace{\Big(\mathbf{L_S}(\mathbf{h}^*) - L_{\mathcal{D}}(h^*)\Big)}_{\text{can be sharply bounded}} + \underbrace{\mathbf{L}_{\mathcal{D}}(\mathbf{h}^*)}_{\text{the approximation error}}$$

Validation methods : what to do in practice and why

- Explanation of 2 (continued) : If $\mathbf{L_S(h_S)}$ is big, then so is $\mathbf{L_{\mathcal{D}}(h^*)}$. This means that the class $\mathcal{H}$ is too **small** (the bias is too **big**).
- If we are in situation 1 ($\mathbf{L_S(h_S)}$ small, maybe even $= 0$), we need to distinguish between
  - the fit is good
  - there is overfitting.
- The distinction can be done via plotting a **learning curve** :
  - compute the training errors occurring when we train the algorithm with $\eta S, 2\eta S, ...S$ (e.g. $\eta = 10\%$)
  - compute the validation errors with some validation set $V \perp S$ for $\eta S, 2\eta S, ...$

Validation methods : what to do in practice and why

- If the validation error **does not** drop with the increasing training sizes : indication that the **approximation error** is not 0 $\Longrightarrow$ we need to enlarge the hypothesis class.
- If the validation error shows decrease with the increasing training sizes but stays nevertheless large, then it is an indication that the size m is not enough $\Longrightarrow$ we need to get more examples.
- **Example.**

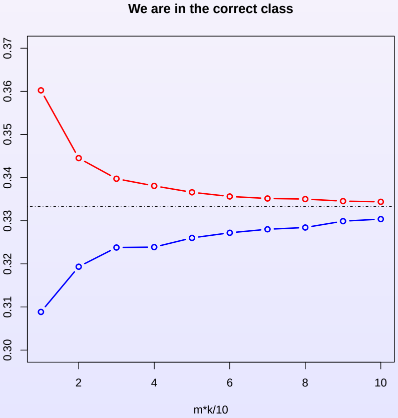
$$x \sim \mathcal{U}[0,1], \quad y|x = 5 + 2x - x^2/2 - 3x^3 + \epsilon, \quad \text{and} \quad \epsilon \sim \mathcal{U}[-1,1].$$

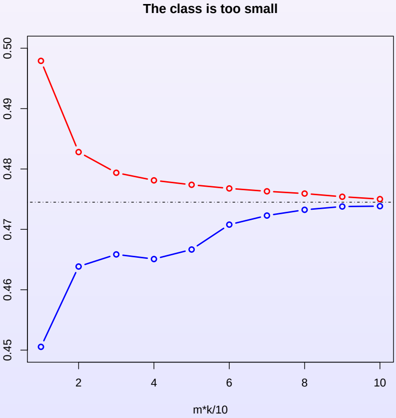
Consider prediction of y given x via learning one of the classes :

- $\mathcal{P}_1$: affine functions (wrong class – too small)
- $\mathcal{P}_3$: the class of polynomials of degree 3 (correct class).

Validation methods : what to do in practice and why

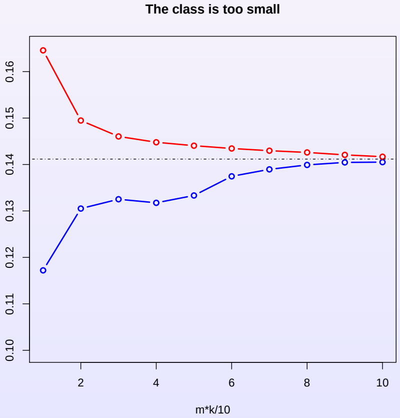
- Here, $m = m_v = 500$, and we learn using a ERM rule ($\ell = \ell_{\mathrm{sq}}$).





Validation methods : what to do in practice and why

Here is what the picture would look like for a classification problem :



Some conclusions

1. Validation is used to obtain a good estimation of $L_{\mathcal{D}}(h_S)$ (for complex classes)
2. When data collecting is expensive, the training set is split into 2 parts : one for training and the other one for validating the model
3. Validation is very useful for model selection : we choose the model which yields the smallest validation error
4. Cross-validation is a common method for model selection, but it still lacks a unifying theory
5. To decide whether the class is too small or the size of training set is not enough : we can use learning curves.

Lecture 7

VC dimension (continued)

Model Selection and Validation (Chapter 11)

Convex Learning Problems (Chapter 12)

Convexity, Lipschitzness and smoothness

- **Convexity** of the loss function, when it holds, makes learning **efficient**. Examples of convex learning problems include :
  - Linear regression with the quadratic loss $\ell_{sq}(h_w, (x, y)) = (h_w(x) - y)^2$ with $h_w(x) = \langle w, x \rangle$
  - Logistic regression with the loss $\ell(h_w, (x, y)) = \log(1 + \exp(-y\langle w, x \rangle))$

Classification with the ℓ_{0-1} is an example of a **non-convex** learning problem.

Definition (convex set). A set C in a a vector space is convex if for **any** two vectors $\mathbf{u}, \mathbf{v} \in C$, the line segment between u and v is contained in C : for any $\alpha \in [0, 1]$, $\alpha\mathbf{u} + (1 - \alpha)\mathbf{v} \in C$.

Convexity

Definition (convex function). Let C be a convex set. A function $f : C \mapsto \mathbb{R}$ is **convex** if $\forall\ \mathbf{u}, \mathbf{v} \in C$ and $\forall\ \alpha \in [0,1]$,
$f(\alpha \mathbf{u} + (1-\alpha)\mathbf{v}) \leq \alpha f(\mathbf{u}) + (1-\alpha) f(\mathbf{v})$.

- The following characterization can be shown :

epigraph$(f) = \{(\mathbf{x}, \beta) \in C \times \mathbb{R} : f(\mathbf{x}) \leq \beta\}$ is a convex set of $C \times \mathbb{R}$

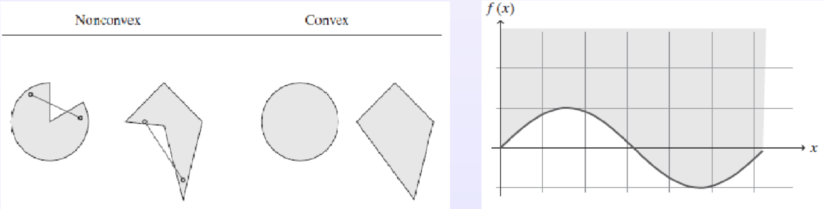


FIGURE – Left : examples for convex and non-convex 2-dimensional sets. Right : example of a non-convex function

Convexity

Property 1. An important consequence of convexity of some function f is that a local minimizer of f is necessarily a **global minimizer** of f.

Proof. Let $\mathbf{u}$ be a local minimum of f defined on C. Then, there exists $r > 0$ such that for all $\mathbf{v} \in B(\mathbf{u}, r)$, the Euclidean ball of radius r and centered at $\mathbf{u}$

$$f(\mathbf{u}) \leq f(\mathbf{v}).$$

Let $\mathbf{w} \in C$ (not necessarily in $B(\mathbf{u}, r)$). Then, we can find some small $\alpha > 0$ such that $\mathbf{u} + \alpha(\mathbf{w} - \mathbf{u}) \in B(\mathbf{u}, r)$. Therefore,

$$f(\mathbf{u}) \leq f\big(\mathbf{u} + \alpha(\mathbf{w} - \mathbf{u})\big) = f\big((1-\alpha)\mathbf{u} + \alpha\mathbf{w}\big).$$

If f is convex, the latter implies that $f(\mathbf{u}) \leq (1-\alpha)f(\mathbf{u}) + \alpha f(\mathbf{w})$, which is equivalent to

$f(\mathbf{u}) \leq f(\mathbf{w}) \iff \mathbf{u}$ is a global minimizer, since $\mathbf{w}$ was arbitrarily chosen. $\square$

Convexity

Property 2. Suppose that f is convex on a convex set $C \subset \mathbb{R}^d$ and is differentiable at $\mathbf{w} \in C$, that is

$$\nabla f(\mathbf{w}) = \left(\frac{\partial f(\mathbf{w})}{\partial w_1}, \ldots, \frac{\partial f(\mathbf{w})}{\partial w_d}\right)^T \quad \text{exists.}$$

Then, the function f stays **above** the tangent at $\mathbf{w}$, that is

$$\forall\ \mathbf{u} \quad f(\mathbf{u}) \geq f(\mathbf{w}) + \nabla f(\mathbf{w})^T(\mathbf{u} - \mathbf{w})$$

Lemma. Let $f : \mathbb{R} \to \mathbb{R}$ be a twice differential function. Then, the following assertions are equivalent :

1. f is convex.
2. f' is nondecreasing.
3. $f'' \geq 0$.

Convexity

- **Examples.** The functions $f(x) = x^2$ and $f(x) = \log(1 + \exp(x))$ are convex on $\mathbb{R}$ since their respective derivatives $f'(x) = 2x$ and $f'(x) = \exp(x)/(1 + \exp(x))$ are nondecreasing.

Result. Let $g : \mathbb{R} \to \mathbb{R}$ be convex. Then, the function $f(\mathbf{w}) = g(\langle \mathbf{w}, \mathbf{x} \rangle + y)$ for some fixed $\mathbf{x} \in \mathbb{R}^d$ and $y \in \mathbb{R}$ is **convex**.

Proof.

$$\begin{aligned} f(\alpha \mathbf{w}_1 + (1-\alpha)\mathbf{w}_2) &= g(\alpha\langle \mathbf{w}_1, \mathbf{x} \rangle + (1-\alpha)\langle \mathbf{w}_2, \mathbf{x} \rangle + y) \\ &= g\Big(\alpha(\langle \mathbf{w}_1, \mathbf{x} \rangle + y) + (1-\alpha)(\langle \mathbf{w}_2, \mathbf{x} \rangle + y)\Big) \\ &\leq \alpha g(\langle \mathbf{w}_1, \mathbf{x} \rangle + y) + (1-\alpha) g(\langle \mathbf{w}_2, \mathbf{x} \rangle + y) \\ &= \alpha f(\mathbf{w}_1) + (1-\alpha) f(\mathbf{w}_2). \qquad \square \end{aligned}$$

Convexity

- **Examples.** The previous result implies that
  - $f(\mathbf{w}) = (\langle \mathbf{w}, \mathbf{x} \rangle - y)^2$ is convex on $\mathbb{R}^d$ as the composition of $g(t) = t^2$ and the linear function $\mathbf{w} \mapsto \langle \mathbf{w}, \mathbf{x} \rangle - y$
  - $f(\mathbf{w}) = \log \left(1 + \exp(-y\langle \mathbf{w}, \mathbf{x} \rangle\right)$ is convex on $\mathbb{R}^d$ (with $y \in \{-1, 1\}$) as the composition of the convex function $g(t) = \log(1 + \exp(t))$ or $g(t) = \log(1 + \exp(-t))$ and the linear function $\mathbf{w} \mapsto \langle \mathbf{w}, \mathbf{x} \rangle$.

Result. For $i \in \{1, \ldots, r\}$, let $f_i : \mathbb{R}^d \mapsto \mathbb{R}$ be a convex function. Then, the functions

- $g(x) = \max_{1 \leq i \leq r} f_i(\mathbf{x})$,
- $g(x) = \sum_{i=1}^r w_i f_i(\mathbf{x})$, for $w_i \geq 0, i = 1, \ldots, r$

are also **convex**.

Still on Convexity... and Lipschitzness

Proof. We prove only the claim for the first function. We have that

$$\begin{aligned} g(\alpha \mathbf{x}_1 + (1-\alpha)\mathbf{x}_2) &= \max_{1 \le i \le r} f_i(\alpha \mathbf{x}_1 + (1-\alpha)\mathbf{x}_2) \\ &\le \max_{1 \le i \le r} [\alpha f_i(\mathbf{x}_1) + (1-\alpha) f_i(\mathbf{x}_2)] \\ &\le \alpha \max_{1 \le i \le r} f_i(\mathbf{x}_1) + (1-\alpha) \max_{1 \le i \le r} f_i(\mathbf{x}_2) \\ &= \alpha g(\mathbf{x}_1) + (1-\alpha) g(\mathbf{x}_2). \end{aligned}$$

Definition (Lipschitzness). Let $C \subset \mathbb{R}^d$. A function $f : \mathbb{R}^d \to \mathbb{R}^k$ is ρ-Lipschitz over C if $\forall\ \mathbf{w}_1, \mathbf{w}_2 \in C\ \|f(\mathbf{w}_2) - f(\mathbf{w}_1)\| \le \rho \|\mathbf{w}_2 - \mathbf{w}_1\|$.

- **Remark.** Lipschitz functions cannot change too fast. If f is a differentiable real function, then ρ-Lipschitzness of f implies that $\sup_t |f'(t)| \le \rho$ since $\lim_{x \to t} |(f(x) - f(t))/(x - t)| \le \rho$.

Lipschitzness

- **Remark (continued).**

$$\begin{aligned} |f(x) - f(y)| &= |f'(u^*)||x - y|, \text{ for some } u^* = \lambda^* x + (1 - \lambda^*)y \\ &\leq \rho|x - y|. \end{aligned}$$

Examples :

- $f(x) = |x|$ is 1-Lipschitz over $\mathbb{R}$ using the well-known inequality $\Big||x| - |y|\Big| \leq |x - y|$.
- $f(x) = \log(1 + \exp(x))$ is also 1-Lipschitz since for all $x \in \mathbb{R}$ $|f'(x)| = f'(x) = \exp(x)/(1 + \exp(x)) \leq 1$.
- $f(x) = x^2$ is not ρ-Lipschitz on $\mathbb{R}$ for any $\rho > 0$ since with $(x_1, x_2) = (0, 1 + \rho)$ we can check that $|f(x_2) - f(x_1)| > \rho|x_2 - x_1|$.

Liptschitzness

Examples (continued) :

- However, $f(x) = x^2$ is ρ-Lipschitz on $C_\rho = [-\rho/2, \rho/2]$ on which $|f'(x)| = 2|x| \leq \rho$.
- Consider $f(\mathbf{w}) = \langle \mathbf{v}, \mathbf{w} \rangle + b$ defined on $\mathbb{R}^d$ to $\mathbb{R}$ for some fixed $\mathbf{v} \in \mathbb{R}^d$. Then, $|f(\mathbf{w}_2) - f(\mathbf{w}_1)| = |\langle \mathbf{v}, \mathbf{w}_2 - \mathbf{w}_1 \rangle| \leq \|\mathbf{v}\| \, \|\mathbf{w}_2 - \mathbf{w}_1\|$ by the Cauchy-Schwartz inequality, so that f is $\|\mathbf{v}\|$-Lipschitz.

Result. Let $f(\mathbf{x}) = g_1(g_2(\mathbf{x}))$, where g_1 is ρ_1-Lipschitz and g_2 is ρ_2-Lipschitz. Then, f is $(\rho_1\rho_2)$-Lipschitz. In particular, if $g_2(\mathbf{x}) = \langle \mathbf{v}, \mathbf{x} \rangle + b$ for some $\mathbf{v} \in \mathbb{R}^d, b \in \mathbb{R}$, then f is $(\rho_1\|\mathbf{v}\|)$-Lipschitz.

Proof. Write $|f(\mathbf{w}_2) - f(\mathbf{w}_1)| = |g_1(g_2(\mathbf{w_2})) - g_1(g_2(\mathbf{w_1}))| \leq \rho_1|g_2(\mathbf{w_2}) - g_2(\mathbf{w_1})| \leq \rho_1\rho_2 \, \|\mathbf{w}_2 - \mathbf{w}_1\|$. $\square$