Sample Compressed SVM

(or *A sample compressed PAC-Bayes approach to kernel methods*)

Pascal Germain

Joint work with François Laviolette, Alexandre Lacasse,
Alexandre Lacoste, Mario Marchand and Sara Shanian

GRAAL
(Université Laval, Québec city)

February 18, 2010

Outline

In this lecture, we will :

- Review quickly the Sample-Compress theory
- See how we can describe a SVM as a **Majority Vote of Sample-Compressed classifiers** (the Sc-SVM)
- Use the **PAC-Bayes** theory to **upper-bound** the risk of our Sc-SVM
- Design a **learning algorithm** to minimise this PAC-Bayes bound
- Present some experimental **results**
- and Conclude...

The Classification problem

We consider a training set S of m examples

$$S \stackrel{\text{def}}{=} (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m)$$

where each $\mathbf{z}_i$ is a input-output pair:

$$\begin{array}{lcll} \mathbf{z}_i & \stackrel{\text{def}}{=} & (\mathbf{x}_i, y_i) & \\ \mathbf{x}_i & \in & \mathcal{X} \subseteq \mathbb{R}^n & \textit{(Real atttibutes)} \\ y_i & \in & \mathcal{Y} = \{-1, +1\} & \textit{(Binary classif.)} \end{array}$$

Each example $\mathbf{z}_i$ is drawn *IID* according to an unknown probability distribution D on $\mathcal{X} \times \mathcal{Y}$. Hence :

$$S \sim D^m$$

.

Elements of the Sample Compression theory

A **sc-classifier** $h_{\mathbf{i}}^{\mu}$ is a data-dependent classifier described by two variables:

- A **compression-set** $S_{\mathbf{i}}$ containing a subset of the training sequence S describing the classifier
  - $\mathbf{i} \stackrel{\mathrm{def}}{=} \langle i_1, i_2, \ldots, i_m \rangle \quad$ with $\quad 1 \leq i_1 < i_2 < \ldots < i_{|\mathbf{i}|} \leq m$
- A **message string** μ containing the additional information needed to construct the classifier.
  - μ is choosen among $\mathcal{M}_{\mathbf{i}}$, a predefined set of all messages that can be supplied with $S_{\mathbf{i}}$.

Given $S_{\mathbf{i}}$ and μ, a **reconstruction function** $\mathcal{R}$ outputs a classifier :

$$h_{\mathbf{i}}^{\mu} \stackrel{\mathrm{def}}{=} \mathcal{R}(S_{\mathbf{i}}, \mu)\,.$$

Risk of a sc-classifier

The **risk** (or generalization error) of a classifier h is defined as

$$R_D(h) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim D} I(h(\mathbf{x}) \neq y) = \Pr_{(\mathbf{x},y)\sim D} (h(\mathbf{x}) \neq y)$$

where $I(a) = 1$ if predicate a is true and 0 otherwise.

The **empirical risk** of a sc-classifier $h_{\mathbf{i}}^{\mu}$ on the training set S is defined by

$$R_S(h_{\mathbf{i}}^{\mu}) \stackrel{\text{def}}{=} \frac{1}{m} \sum_{j=1}^{m} R_{\langle(\mathbf{x}_j,y_j)\rangle}(h_{\mathbf{i}}^{\mu}),$$

where

$$R_{\langle(\mathbf{x}_j,y_j)\rangle}(h_{\mathbf{i}}^{\mu}) \stackrel{\text{def}}{=} \begin{cases} I(h_{\mathbf{i}}^{\mu}(\mathbf{x}_j) \neq y_j) & \text{if } j \notin \mathbf{i} \\ 0 & \text{otherwise.} \end{cases}$$

Thus, $\quad mR_s(h_{\mathbf{i}}^{\mu}) \sim \mathrm{Bin}\Big(m - \|\mathbf{i}\|, R_D(h_{\mathbf{i}}^{\mu})\Big)$.

Examples of compression sets and reconstruction functions

Support Vector Machine

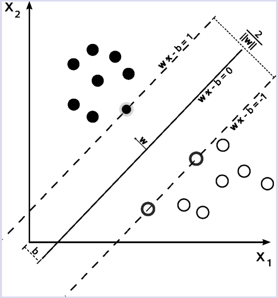


Image: Wikipedia

We can reconstruct a SVM by using the **support-vectors** as the compression-set.

In this example :

- Training set size: $|S| = 16$
- Compression-set size: $|\mathbf{i}| = 3$
- Message string: $\mu = \emptyset$

Perceptron

Once a perceptron is trained (until convergence) on S, we only need the $|\mathbf{i}|$ **examples implied in an update** to reconstruct the classifier.

- Again, $\mu = \emptyset$.

Outline

In this lecture, we will :

- Review quickly the Sample-Compress theory
- See how we can describe a SVM as a **Majority Vote of Sample-Compressed classifiers** (the Sc-SVM)
- Use the **PAC-Bayes** theory to **upper-bound** the risk of our Sc-SVM
- Design a **learning algorithm** to minimise this PAC-Bayes bound
- Present some experimental **results**
- and Conclude...

Redefining the SVM

- In an attempt to **upper-bound the risk** $R_D(h)$, usual sample-compression bounds **degrade with the size** of the compression-set $S_{\mathbf{i}}$ (as we expect).

- This gives a **bad interpretation of the generalization error** of the SVM, which can have a low risk even if there is a large number of support vectors.

- To overcome this issue, let's define the SVM as a **majority vote of sc-classifiers** of unitary compression-size.

Redefining the SVM

We denote $\mathcal{H}^S$ the set of all sc-classifiers. Each $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$ is such as :

- The **compression-set** contains only zero or one training example :

$$S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, S_{\langle 2\rangle}, \ldots, S_{\langle m\rangle}\}$$

- The **message string** is formed by a real number and a sign :

$$\mu \in \mathcal{M}_{\mathbf{i}} = [-1, 1]^{|\mathbf{i}|} \times \{+, -\}$$

We have $\mathcal{M}_{\langle i\rangle} = [-1, 1] \times \{+, -\}$ and $\mathcal{M}_{\langle\rangle} = \{\epsilon\} \times \{+, -\}$.

We consider **pairs of boolean complement classifiers** such as :

$$h_{\mathbf{i}}^{(\sigma,-)}(\mathbf{x}) = -h_{\mathbf{i}}^{(\sigma,+)}(\mathbf{x}) \quad \forall\, \mathbf{x} \in \mathcal{X},\, \sigma \in [-1, 1]\,.$$

We also have:

$$h_{\langle\rangle}^{(\epsilon,+)}(\mathbf{x}) = +1 \text{ and } h_{\langle\rangle}^{(\epsilon,-)}(\mathbf{x}) = -1 \quad \forall\, \mathbf{x} \in \mathcal{X}\,.$$

sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$
Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$
$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

Let Q be a **probability distribution** over $\mathcal{H}^S$. We denote

- $Q_{\mathcal{I}}$, the probability that a compression-set $S_{\mathbf{i}}$ is chosen by Q:

$$Q_{\mathcal{I}}(\mathbf{i}) \stackrel{\mathrm{def}}{=} \int_{\mu \in \mathcal{M}_{\mathbf{i}}} Q(h_{\mathbf{i}}^{\mu}) d\mu$$

- $Q_{S_{\mathbf{i}}}$, the probability of choosing message μ given $S_{\mathbf{i}}$:

$$Q_{S_{\mathbf{i}}}(\mu) \stackrel{\mathrm{def}}{=} Q(h_{\mathbf{i}}^{\mu} | S_{\mathbf{i}})$$

- Therefore, $Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$.

The **output** of the majority vote classifier (*bayes classifier*) is given by :

$$B_Q(\mathbf{x}) \stackrel{\mathrm{def}}{=} \mathrm{sgn}\left[\mathop{\mathbf{E}}_{h\sim Q} h(\mathbf{x})\right]$$

sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

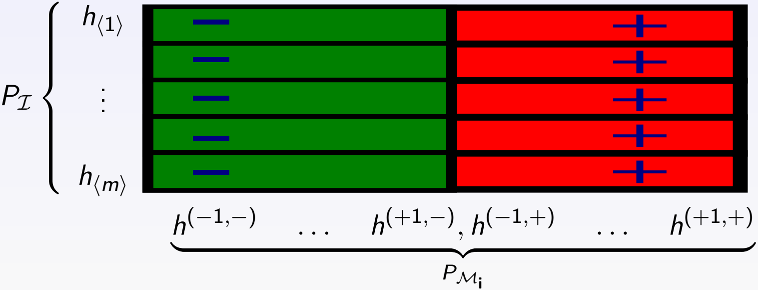
Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$
Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i})Q_{S_{\mathbf{i}}}(\mu)$
$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

Before seing the data, we define a **prior distribution** over the compression-sets and the message strings. This gives us indirectly a prior P over $\mathcal{H}^S$ such as :

- $P_{\mathcal{I}}$ is an uniform distribution over all possible compression-sets ;
- For each compression-set $S_{\mathbf{i}}$, $P_{S_{\mathbf{i}}}$ is uniform over all messages.



sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$
Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$
$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

We say that a posterior Q is **aligned on** a prior P when for all $\mathbf{i}$ and σ:

$$Q(h_{\mathbf{i}}^{(\sigma,+)}) + Q(h_{\mathbf{i}}^{(\sigma,-)}) = P(h_{\mathbf{i}}^{(\sigma,+)}) + P(h_{\mathbf{i}}^{(\sigma,-)})$$

Moreover, we say that a posterior Q is **strongly aligned** when for all $\mathbf{i}$, there is a $w_{\mathbf{i}}$ such that for all σ:

$$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$$

By restricting ourself to strongly aligned posterior, we obtain a posterior distribution totally defined by the $w_{\mathbf{i}}$'s :

$$\begin{aligned} Q(h_{\mathbf{i}}^{(\sigma,+)}) &= \frac{1}{2}\left(P(h_{\mathbf{i}}^{(\sigma,+)}) + P(h_{\mathbf{i}}^{(\sigma,-)}) + w_{\mathbf{i}}\right) \\ Q(h_{\mathbf{i}}^{(\sigma,-)}) &= \frac{1}{2}\left(P(h_{\mathbf{i}}^{(\sigma,+)}) + P(h_{\mathbf{i}}^{(\sigma,-)}) - w_{\mathbf{i}}\right) \end{aligned}$$

sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$

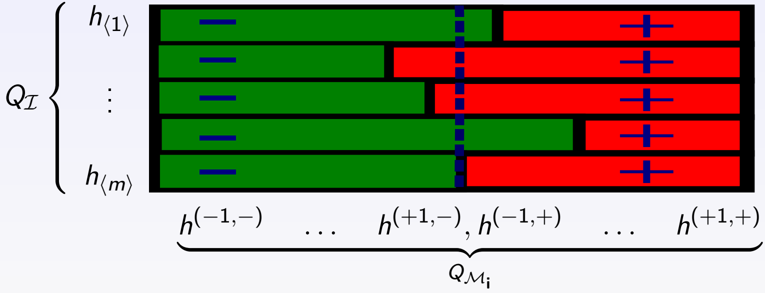
Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$

$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

$$
\begin{aligned}
Q(h_{\mathbf{i}}^{(\sigma,+)}) &= \frac{1}{2}\left(P(h_{\mathbf{i}}^{(\sigma,+)}) + P(h_{\mathbf{i}}^{(\sigma,-)}) + w_{\mathbf{i}}\right) \\
Q(h_{\mathbf{i}}^{(\sigma,-)}) &= \frac{1}{2}\left(P(h_{\mathbf{i}}^{(\sigma,+)}) + P(h_{\mathbf{i}}^{(\sigma,-)}) - w_{\mathbf{i}}\right)
\end{aligned}
$$



sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$

Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$

$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

There's almost **no loss of expressiveness** if we consider aligned posterior:

Proposition

Let P be a prior, S a training sequence, and Q a distribution on $\mathcal{H}^S$ for which there exists $A > 0$ such that for all $h_{\mathbf{i}}^{\mu}$:

$$Q(h_{\mathbf{i}}^{\mu}) + Q(-h_{\mathbf{i}}^{\mu}) \leq A\left(P(h_{\mathbf{i}}^{\mu}) + P(-h_{\mathbf{i}}^{\mu})\right).$$

Then there exists a distribution Q' aligned on P and Bayes-equivalent to Q

$$i.e.,\ B_{Q'}(\mathbf{x}) = B_Q(\mathbf{x})\ \forall \mathbf{x} \in \mathcal{X}.$$

Remember that $B_Q(\mathbf{x}) \stackrel{\text{def}}{=} \operatorname{sgn}\left[\mathop{\mathbf{E}}_{h\sim Q} h(\mathbf{x})\right]$.

From Q to Q', the margins will vary, but the outcome of the majority vote will stay the same!

sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$
Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$
$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

Consider any similarity function $k(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to [-1,1]$.
We say that **reconstruction function** $\mathcal{R}$ is associated to k when :

$$h_{\langle\rangle}^{(\epsilon,+)}(\mathbf{x}) \stackrel{\text{def}}{=} +1$$

$$h_{\langle i\rangle}^{(\sigma,+)}(\mathbf{x}) \stackrel{\text{def}}{=} \begin{cases} +1 & \text{if } \sigma < k(\mathbf{x}_i, \mathbf{x}) \\ -1 & \text{otherwise} \end{cases}$$

$$h_{\mathbf{i}}^{(\sigma,-)}(\mathbf{x}) \stackrel{\text{def}}{=} -h_{\mathbf{i}}^{(\sigma,+)}(\mathbf{x})\,.$$

For an uniform prior, this definition allows us to **recover the value** of k:

$$\frac{1}{2}\, k(\mathbf{x}_i, \mathbf{x}) = \int_{\sigma \in \mathcal{M}^1(S_{\mathbf{i}})} h_{\mathbf{i}}^{(\sigma,+)}(\mathbf{x}) \cdot P_{S_{\mathbf{i}}}(\sigma,+)\ d\mu\,,$$

as $\int_{-1}^{k(\mathbf{x}_i,\mathbf{x})} \frac{1/2}{(1-(-1))} d\mu - \int_{k(\mathbf{x}_i,\mathbf{x})}^{1} \frac{1/2}{(1-(-1))} d\mu = \frac{1}{2} k(\mathbf{x}_i, \mathbf{x})\,.$

sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$

Comp-set: $S_{\mathbf{i}} \in \{S_{\langle\rangle}, S_{\langle 1\rangle}, \ldots, S_{\langle m\rangle}\}$

Message: $\mu \in \mathcal{M}_{\mathbf{i}} = [-1,1]^{|\mathbf{i}|} \times \{+,-\}$

Distribution Q

$Q(h_{\mathbf{i}}^{\mu}) = Q_{\mathcal{I}}(\mathbf{i}) Q_{S_{\mathbf{i}}}(\mu)$

$Q(h_{\mathbf{i}}^{(\sigma,+)}) - Q(h_{\mathbf{i}}^{(\sigma,-)}) = w_{\mathbf{i}}$

We finally obtain that our strongly aligned posterior will be such that:

$$
\begin{aligned}
Q_{\mathcal{I}}(\langle\rangle) = Q_{\mathcal{I}}(\langle i\rangle) &= \frac{1}{m+1}\,, \\
w_{\langle i\rangle} \cdot k(\mathbf{x}_i, \mathbf{x}) &= \int_{\mu \in \mathcal{M}_{\langle i\rangle}} h_{\langle i\rangle}^{\mu}(\mathbf{x}) \cdot Q_{\langle i\rangle}(\mu)\ d\mu\,, \\
\text{and} \qquad w_{\langle\rangle} \cdot 1 &= \int_{\mu \in \mathcal{M}_{\langle\rangle}} h_{\langle\rangle}^{\mu}(\mathbf{x}) \cdot Q_{\langle\rangle}(\mu)\ d\mu\,.
\end{aligned}
$$

Thus, the output of this majority vote $B_Q(\mathbf{x}) = \operatorname{sgn}\left[\mathbf{E}_{\ h\sim Q} h(\mathbf{x})\right]$ will be the same as $f_{\mathrm{SVM}}(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{m} y_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b\right)$ when

$$
w_{\langle i\rangle} = \frac{y_i \alpha_i}{Z\,(m+1)} \quad \text{and} \quad w_{\langle\rangle} = \frac{b}{Z\,(m+1)}. \qquad \left(Z \stackrel{\text{def}}{=} \sum_{i=1}^{m} \alpha_i + |b|\right)
$$

Outline

In this lecture, we will :

- Review quickly the Sample-Compress theory
- See how we can describe a SVM as a **Majority Vote of Sample-Compressed classifiers** (the Sc-SVM)
- Use the **PAC-Bayes** theory to **upper-bound** the risk of our Sc-SVM
- Design a **learning algorithm** to minimise this PAC-Bayes bound
- Present some experimental **results**
- and Conclude...

PAC-Bayes bounds for Sc-SVM

Usuals PAC-Bayes theorems allow us to bound the risk of a majority vote classifier using two key ingredients:

- The **Kullback-Leibler divergence** $\mathrm{KL}(Q\|P)$ between prior distribution P and posterior distribution Q
- The **empirical risk of the Gibbs classifier** G_Q, related to the majority vote B_Q
  - Given any $\mathbf{x}$, G_Q draws h according to Q and classifies $\mathbf{x}$ according to h.
  - It follows that $R_D(B_Q) \leq 2R_D(G_Q)$.

In our setting, the Gibbs risk $R_D(G_Q)$ will be likely near $1/2$, even if the Bayes risk is close to 0.

- Each sc-classifier $h_{\mathbf{i}}^{\mu} \in \mathcal{H}^S$ might be really weak.

We want to bound a **more relevant risk**!

Inspired by [Germain et al. *PAC-Bayes bounds for general loss functions* (2006)].

Margin of the majority vote classifier

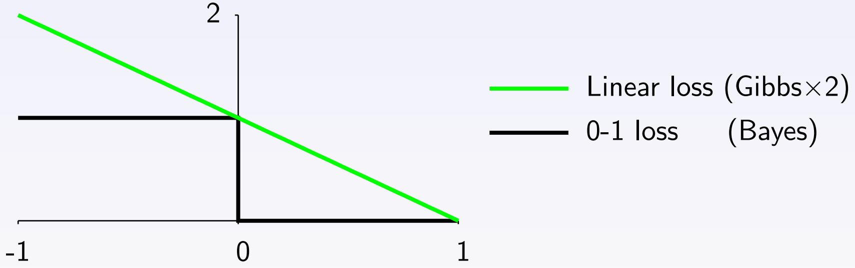
$$M_Q(\mathbf{x}, y) \stackrel{\text{def}}{=} \mathbf{E}_{h_{\mathbf{i}}^{\mu} \sim Q}\; y h_{\mathbf{i}}^{\mu}(\mathbf{x})$$

The margin is closely related to the Gibbs risk :

$$R_D(G_Q) = \tfrac{1}{2} - \tfrac{1}{2}\, \mathbf{E}_{(\mathbf{x},y)\sim D} M_Q(\mathbf{x}, y)\,.$$

For sc-classifiers, we define the **empirical margin** as:

$$\widehat{M}_Q(\mathbf{x}_j, y_j) \stackrel{\text{def}}{=} \mathbf{E}_{h_{\mathbf{i}}^{\mu} \sim Q}\left[y_j h_{\mathbf{i}}^{\mu}(\mathbf{x}_j) \cdot I(j \notin \mathbf{i}) + 1 \cdot I(j \in \mathbf{i}) \right].$$



Margin of the majority vote classifier

$$M_Q(\mathbf{x}, y) \stackrel{\text{def}}{=} \mathbf{E}_{h_{\mathbf{i}}^{\mu} \sim Q} \; y h_{\mathbf{i}}^{\mu}(\mathbf{x})$$

We consider **any non-negative loss** ζ that can be expended by a Taylor series around $M_Q(\mathbf{x}, y) = 0$ and **upper-bound the zero-one loss**:

$$\zeta(\alpha) = \sum_{k=0}^{\deg \zeta} a_k \, \alpha^k \quad \text{with} \quad a_k \geq 0$$

$$\zeta(\alpha) \geq I(\alpha \leq 0) \qquad \forall \alpha \in [-1, 1].$$

We obtain a risk value

$$\zeta_D^Q \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{(\mathbf{x},y) \sim D} \; \sum_{k=0}^{\deg \zeta} a_k \, (-M_Q(\mathbf{x}, y))^k$$

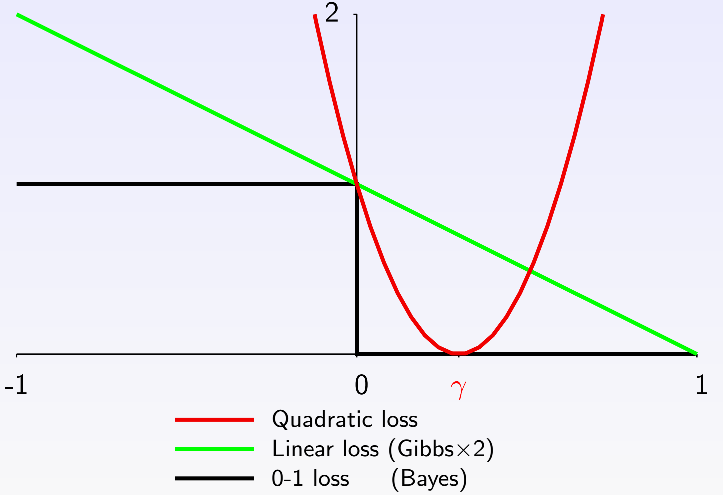
We express ζ_D^Q as the risk of a Gibbs classifier described by a transformed posterior $\overline{Q}$ on a augmented set of classiers $\overline{\mathcal{H}^S}$:

$$\zeta_D^Q = \zeta(1) \cdot \mathop{\mathbf{E}}_{(\mathbf{x},y) \sim D} \left[\tfrac{1}{2} - \tfrac{1}{2} M_{\overline{Q}}(\mathbf{x}, y) \right]$$

Margin of the majority vote classifier

$$M_Q(\mathbf{x}, y) \stackrel{\text{def}}{=} \mathbf{E}_{h_{\mathbf{i}}^{\mu} \sim Q}\; y\, h_{\mathbf{i}}^{\mu}(\mathbf{x})$$

We choose to use the **quadratic loss** function $\zeta_\gamma(\alpha) = \left(1 - \frac{1}{\gamma}\alpha\right)^2$.



First PAC-Bayes theorem

The following PAC-Bayes theorem (next slide) is an adapted version of a **Catoni's theorem** where the influence of the empirical risk (vs $\mathrm{KL}(Q\|P)$) is determined by an hyperparameter C.

- Generally less tight than the classic kl bound
- But useful to design a bound-minimization algorithm

In this adapted version, we consider:

- A general loss function ζ
- A set of (data-dependent) sc-classifiers of size $\leq l$

Theorem

For any D, any family $(\mathcal{H}^S)_{S\in\mathcal{D}^m}$ of sets of sc-classifiers of size at most l, any prior P any $\delta \in (0,1]$, any positive real number C_1 and any margin loss function ζ of degree $< m/l$, we have

$$\Pr_{S\sim D^m}\left(\begin{array}{l}\forall\, Q \text{ on } \mathcal{H}^S: \\ \zeta_D^Q \;\le\; C' \cdot \left(\zeta_S^Q + \frac{\zeta'(1)\cdot \mathrm{KL}(Q\|P) + \ln\frac{1}{\delta}}{\zeta(1)\cdot C_1 \cdot m}\right)\end{array}\right) \ge 1-\delta\,,$$

where $\mathrm{KL}(\cdot\|\cdot)$ is the Kullback-Leibler divergence, and

$$C' \;=\; \frac{C_1 \cdot \frac{m}{m - l\cdot \deg\zeta}}{1 - e^{-C_1\cdot\frac{m-l\cdot\deg\zeta}{m}}}\,.$$

Finding Q that minimizes this bound is equivalent to finding Q that minimizes :

$$f(Q) \stackrel{\text{def}}{=} C\cdot\zeta_S^Q + KL(Q\|P)$$

Second PAC-Bayes theorem

The next theorem is an adapted version of the **Langford and Seeger's theorem** where the influence of the empirical risk (vs $\mathrm{KL}(Q\|P)$) is given via the Kullback-Leibler divergence between two Bernoulli distributions of probability of success p and q :

$$\begin{aligned}\mathrm{kl}(q\|p) &\stackrel{\text{def}}{=} q\ln\frac{q}{p}+(1-q)\ln\frac{1-q}{1-p}\\ &= \mathrm{kl}(1-q\|1-p)\end{aligned}$$

We specialize the theorem for the case of aligned posterior:

$$Q(h)+Q(-h)=P(h)+P(-h)\quad \forall h\in\mathcal{H}$$

...And the term $\mathrm{KL}(Q\|P)$ disapears from the theorem!

Theorem

For any D, any family $(\mathcal{H}^S)_{S\in\mathcal{D}^m}$ of sets of sc-classifiers of size at most l, any prior P, any $\delta \in (0,1]$, any margin loss function ζ of degree $< m/l$, we have

$$\Pr_{S\sim D^m}\left(\begin{array}{l}\forall Q\in\mathcal{H}^S \ \textbf{aligned on } P: \\ \mathrm{kl}\left(\frac{1}{\zeta(1)}\cdot\zeta_S^Q \middle\| \frac{1}{\zeta(1)}\cdot\zeta_D^Q\right) \le \frac{\ln\frac{m+1}{\delta}}{m-l\cdot\deg\zeta}\end{array}\right)\ge 1-\delta$$

where $\mathrm{kl}(q\|p)$ is the KL-divergence between two Bernoulli distributions of respective succes probabilities q and p.

Finding Q that minimizes this bound is equivalent to finding Q that minimizes :

$$f(Q) \stackrel{\mathrm{def}}{=} \zeta_S^Q$$

We want to bound random variable $\mathbf{E}_{h\sim P} e^{m\cdot \mathrm{kl}(R_S(h)\|R(h))}$ in term of $R(G_Q)$.

General theorem

Term $\mathrm{KL}(Q\|P)$ arises when transforming expectation over P into expectation over Q:

$$
\begin{aligned}
& \ln \left[\mathbf{E}_{h\sim P} e^{m\cdot \mathrm{kl}(R_S(h)\|R(h))} \right] \\
= \ & \ln \left[\mathbf{E}_{h\sim Q} \frac{P(h)}{Q(h)} e^{m\cdot \mathrm{kl}(R_S(h),R(h))} \right] \\
\geq \ & \mathbf{E}_{h\sim Q} \ln \left[\frac{P(h)}{Q(h)} e^{m\cdot \mathrm{kl}(R_S(h),R(h))} \right] \\
= \ & m \mathbf{E}_{h\sim Q} \mathrm{kl}(R_S(h), R(h)) - \mathrm{KL}(Q\|P) \\
\geq \ & m \cdot \mathrm{kl}(\mathbf{E}_{h\sim Q} R_S(h), \mathbf{E}_{h\sim Q} R(h)) - \mathrm{KL}(Q\|P) \\
= \ & m \cdot \mathrm{kl}(R_S(G_Q), R(G_Q)) - \mathrm{KL}(Q\|P) \ .
\end{aligned}
$$

Aligned posterior theorem

Here, we do the same operation for "free" (proof on next slide):

$$
\begin{aligned}
& \ln \left[\mathbf{E}_{h\sim P} e^{m\cdot \mathrm{kl}(R_S(h)\|R(h))} \right] \\
= \ & \ln \left[\mathbf{E}_{h\sim Q} e^{m\cdot \mathrm{kl}(R_S(h)\|R(h))} \right] \\
\geq \ & \mathbf{E}_{h\sim Q} \ln \left[e^{m\cdot \mathrm{kl}(R_S(h),R(h))} \right] \\
= \ & m \mathbf{E}_{h\sim Q} \mathrm{kl}(R_S(h), R(h)) \\
\geq \ & m \cdot \mathrm{kl}(\mathbf{E}_{h\sim Q} R_S(h), \mathbf{E}_{h\sim Q} R(h)) \\
= \ & m \cdot \mathrm{kl}(R_S(G_Q), R(G_Q)) \ .
\end{aligned}
$$

The two "$\geq$" come from Jensen's inequality: $\mathbf{E} f(X) \geq f(\mathbf{E} X)$ for convex f.

First, note that as we have $h \in \mathcal{H}^S \Rightarrow -h \in \mathcal{H}^S$:

$$\mathop{\mathbf{E}}_{h \sim P} e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} = \int_{h \in \mathcal{H}} P(h) e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} = \int_{h \in \mathcal{H}} P(-h) e^{m \cdot \mathrm{kl}(R_S(-h) \| R(-h))} .$$

Then,

$$\begin{aligned}
& 2 \mathop{\mathbf{E}}_{h \sim P} e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} \\
= \; & \int_{h \in \mathcal{H}} P(h) e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} + \int_{h \in \mathcal{H}} P(-h) e^{m \cdot \mathrm{kl}(R_S(-h) \| R(-h))} \\
= \; & \int_{h \in \mathcal{H}} P(h) e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} + \int_{h \in \mathcal{H}} P(-h) e^{m \cdot \mathrm{kl}(1 - R_S(h) \| 1 - R(h))} \\
= \; & \int_{h \in \mathcal{H}} \left(P(h) + P(-h) \right) e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} \\
= \; & \int_{h \in \mathcal{H}} \left(Q(h) + Q(-h) \right) e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} \\
= \; & \int_{h \in \mathcal{H}} Q(h) e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} + \int_{h \in \mathcal{H}} Q(-h) e^{m \cdot \mathrm{kl}(R_S(-h) \| R(-h))} \\
= \; & 2 \mathop{\mathbf{E}}_{h \sim Q} e^{m \cdot \mathrm{kl}(R_S(h) \| R(h))} .
\end{aligned}$$

Outline

In this lecture, we will :

- Review quickly the Sample-Compress theory
- See how we can describe a SVM as a **Majority Vote of Sample-Compressed classifiers (the Sc-SVM)**
- Use the **PAC-Bayes** theory to **upper-bound** the risk of our Sc-SVM
- Design a **learning algorithm** to minimise this PAC-Bayes bound
- Present some experimental **results**
- and Conclude...

Let's design two learning algorithms

The task of the algorithms is to find a vector $\mathbf{w} = (w_0, w_1, \ldots, w_m)$,

$$\begin{aligned} w_0 &\stackrel{\text{def}}{=} w_{\langle\rangle} = Q(h_{\langle\rangle}^{(\sigma,+)}) - Q(h_{\langle\rangle}^{(\sigma,-)}) \\ w_i &\stackrel{\text{def}}{=} w_{\langle i\rangle} = Q(h_{\langle i\rangle}^{(\sigma,+)}) - Q(h_{\langle i\rangle}^{(\sigma,-)}) \\ |w_j| &\leq \frac{1}{m+1} \quad \forall j \in \{0, \ldots, m\} \end{aligned}$$

The empirical margin $\widehat{M}_Q$ will now be defined by

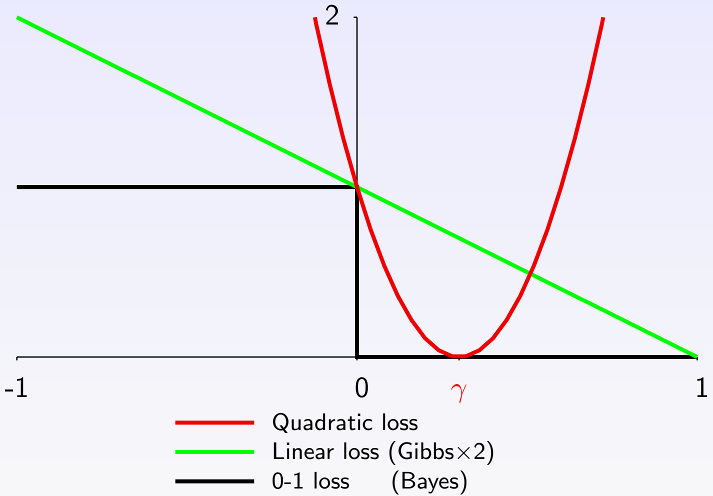
$$\widehat{M}_Q(\mathbf{x}_j, y_j) = \sum_{k=0}^{m} y_j \, w_k \, \widehat{G}(\mathbf{x}_k, \mathbf{x}_j) = y_j \, \mathbf{w} \, \widehat{\mathbf{G}}(\mathbf{x}_j)$$

where

$$\widehat{G}(\mathbf{x}_j, \mathbf{x}_l) \stackrel{\text{def}}{=} \begin{cases} k(\mathbf{x}_j, \mathbf{x}_l) & \forall\, j \in \{1, .., m\} \text{ and } j \neq l \\ 1 & \forall\, j \in \{1, .., m\} \text{ and } j = l \\ 1 & \text{for } j = 0 \end{cases}$$

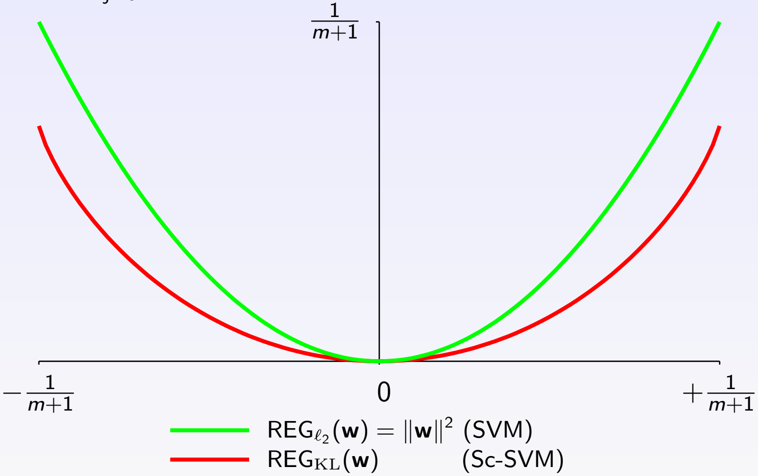
$$\widehat{\mathbf{G}}(\mathbf{x}_l) \stackrel{\text{def}}{=} (\widehat{G}(\mathbf{x}_0, \mathbf{x}_l), \ldots, \widehat{G}(\mathbf{x}_m, \mathbf{x}_l)).$$

Remember that we minimize the **quadratic loss** $\zeta_\gamma(\alpha) = \left(1 - \frac{1}{\gamma}\alpha\right)^2$, where α is the **margin** and γ is the **minimum** of the parabola.



The KL between an uniform prior and the posterior associated to **w** is

$$\mathrm{REG}_{\mathrm{KL}}(\mathbf{w}) = \frac{1}{2}\sum_{j=0}^{m}\Big[(c+w_i)\ln(c+w_i)+(c-w_i)\ln(c-w_i)\Big]-\ln c \quad \text{with } c=\tfrac{1}{m+1}$$



Algorithm with KL

Find $\mathbf{w}$ that minimizes $f(\mathbf{w}) \stackrel{\text{def}}{=} C \cdot \sum_{j=0}^{m} \zeta_\gamma \left(y_j \, \mathbf{w} \, \widehat{\mathbf{G}}(\mathbf{x}_j) \right) + \mathrm{REG}_{\mathrm{KL}}(\mathbf{w})$

Parameters to tune :

- C, the trade-off between the two terms to minimize
- γ, the minimum of the quadratic risk
- Kernel parameter(s), if any

Algorithm without KL

Find $\mathbf{w}$ that minimizes $f(\mathbf{w}) \stackrel{\text{def}}{=} \sum_{j=0}^{m} \zeta_\gamma \left(y_j \, \mathbf{w} \, \widehat{\mathbf{G}}(\mathbf{x}_j) \right)$

Parameters to tune :

- γ, the minimum of the quadratic risk
- Kernel parameter(s), if any

Optimization procedure

Both objective functions are **convex**. Starting from $\mathbf{w} = \mathbf{0}$, we optimize $f(\mathbf{w})$ **coordinate by coordinate**:

- Choose at random $i \in \{0, \ldots, m\}$
- Update $w_i \leftarrow w_i + \delta$ in order to minimize $f(\mathbf{w})$
- If $w_i > \frac{1}{m+1}$ then $w_i \leftarrow \frac{1}{m+1}$
- If $w_i < \frac{-1}{m+1}$ then $w_i \leftarrow \frac{-1}{m+1}$
- Repeat until convergence

Let $\mathbf{w}_\delta$ be the weight vector obtained after an update $w_i \leftarrow w_i + \delta$.

Then, the optimal δ is obtain when

$$\frac{\partial f(\mathbf{w}_\delta)}{\partial \delta} = 0$$

Algorithm with *KL*

Find $\mathbf{w}$ that minimizes $f(\mathbf{w}) \stackrel{\text{def}}{=} C \cdot \sum_{j=0}^{m} \zeta_\gamma \left(y_j \, \mathbf{w} \, \widehat{\mathbf{G}}(\mathbf{x}_j) \right) + \mathrm{REG}_{\mathrm{KL}}(\mathbf{w})$

$$\frac{df(\mathbf{w}_\delta)}{d\delta} = \frac{2C}{\gamma^2 m} \left(\delta \sum_{j=1}^{m} \widehat{G}^2(\mathbf{x}_i, \mathbf{x}_j) + \sum_{j=1}^{m} \widehat{G}(\mathbf{x}_i, \mathbf{x}_j) \, D_{\mathbf{w}}(j) \right) + \frac{1}{2} \ln \left[\frac{\frac{1}{m+1} + w_i + \delta}{\frac{1}{m+1} - w_i - \delta} \right]$$

where $D_{\mathbf{w}}(j) \stackrel{\text{def}}{=} \mathbf{w} \cdot \widehat{\mathbf{G}}(\mathbf{x}_j) - \gamma y_j$.

We find δ^* such as $\frac{df(\mathbf{w}_{\delta^*})}{d\delta} = 0$ using an iterative **root-finding method**.

Algorithm without *KL*

Find $\mathbf{w}$ that minimizes $f(\mathbf{w}) \stackrel{\text{def}}{=} \sum_{j=0}^{m} \zeta_\gamma \left(y_j \, \mathbf{w} \, \widehat{\mathbf{G}}(\mathbf{x}_j) \right)$

$$\frac{df(\mathbf{w}_\delta)}{d\delta} = \frac{2}{\gamma^2 m} \left(\delta \sum_{j=1}^{m} \widehat{G}^2(\mathbf{x}_i, \mathbf{x}_j) + \sum_{j=1}^{m} \widehat{G}(\mathbf{x}_i, \mathbf{x}_j) \, D_{\mathbf{w}}(j) \right)$$

We find δ^* such as $\frac{df(\mathbf{w}_{\delta^*})}{d\delta} = 0$ computing **directly** $\delta^* = \frac{-\delta \sum_{j=1}^{m} \widehat{G}^2(\mathbf{x}_i,\mathbf{x}_j)}{\sum_{j=1}^{m} \widehat{G}(\mathbf{x}_i,\mathbf{x}_j) \, D_{\mathbf{w}}(j)}$

Outline

In this lecture, we will :

- Review quickly the Sample-Compress theory
- See how we can describe a SVM as a **Majority Vote of Sample-Compressed classifiers** (the Sc-SVM)
- Use the **PAC-Bayes** theory to **upper-bound** the risk of our Sc-SVM
- Design a **learning algorithm** to minimise this PAC-Bayes bound
- Present some experimental **results**
- and Conclude...

Experimental results (RBF kernel, 10-folds CV)

Dataset	$\lvert T \rvert$	$\lvert S \rvert$	n	**Classic SVM**	**SC-SVM** (with KL)	**SC-SVM** (w/o KL)
Usvotes	200	235	16	0.065	**0.060**	**0.060**
Liver	175	170	6	**0.303**	0.371	**0.303**
Credit-A	300	353	15	0.187	0.170	**0.150**
Glass	107	107	9	0.159	**0.131**	0.178
Haberman	150	144	3	**0.273**	0.287	0.287
Heart	147	150	13	0.184	**0.163**	0.190
sonar	104	104	60	0.183	0.144	**0.135**
BreastCancer	340	343	9	0.038	**0.035**	**0.035**
Tic-tac-toe	479	479	9	0.023	**0.015**	**0.015**
Ionosphere	175	176	34	0.051	**0.029**	**0.029**
Wdbc	284	285	30	0.070	0.092	**0.067**
MNIST:0vs8	1916	500	784	0.005	**0.004**	**0.004**
MNIST:1vs7	1922	500	784	0.012	**0.008**	0.010
MNIST:1vs8	1936	500	784	0.013	**0.011**	**0.011**
MNIST:2vs3	1905	500	784	0.023	**0.016**	0.018
Letter:AB	1055	500	16	**0.001**	**0.001**	**0.001**
Letter:DO	1058	500	16	0.013	**0.009**	**0.009**
Letter:OQ	1036	500	16	**0.014**	0.017	0.017
Adult	10000	1809	14	0.160	**0.157**	**0.157**
Mushroom	4062	4062	22	**0.000**	**0.000**	**0.000**
Waveform	4000	4000	21	**0.068**	0.069	**0.068**
Ringnorm	3700	3700	20	0.015	0.016	**0.012**

Outline

In this lecture, we will :

- Review quickly the Sample-Compress theory
- See how we can describe a SVM as a **Majority Vote of Sample-Compressed classifiers** (the Sc-SVM)
- Use the **PAC-Bayes** theory to **upper-bound** the risk of our Sc-SVM
- Design a **learning algorithm** to minimise this PAC-Bayes bound
- Present some experimental **results**
- and Conclude...

Future works

Two future research ideas (among others) :

- Experimentations with **undefined similiraty measures** (non-PSD kernels)
- Consider a majority vote of sc-classifiers of **maximum size** > 1
  $\Rightarrow$ More general than the SVM

Image: http://www.mositronic.com/

The End!

Any Questions ?