Model selection by resampling penalization

Sylvain Arlot

Model selection by resampling penalization

Sylvain Arlot
Univ Paris-Sud, Laboratoire de Mathématiques d'Orsay,
Orsay Cedex, F-91405; CNRS, Orsay cedex, F-91405
sylvain.arlot@math.u-psud.fr
INRIA Futurs, Projet Select

January 19, 2007

Abstract

We present a new family of model selection algorithms based on the resampling heuristics. It can be used in several frameworks, do not require any knowledge about the unknown law of the data, and may be seen as a generalization of local Rademacher complexities and V-fold cross-validation. In the case example of least-square regression on histograms, we prove oracle inequalities, and that these algorithms are naturally adaptive to both the smoothness of the regression function and the variability of the noise level. Then, interpretating V-fold cross-validation in terms of penalization, we enlighten the question of choosing V. Finally, a simulation study illustrates the strength of resampling penalization algorithms against some classical ones, in particular with heteroscedastic data.

1 Introduction

Choosing between the outputs of many learning algorithms, from the prediction viewpoint, remains to estimate their generalization abilities. A classical method for this is penalization, that comes from model selection theory. Basically, it states that a good choice can be made by minimizing the sum of the empirical risk (how does the algorithm fits the data) and some complexity measure of the algorithm (called the penalty). The ideal penalty for prediction is of course the difference between the true and empirical risks of the output, but it is unknown in general. It is thus crucial to obtain tight estimates of such a quantity.

Many penalties or complexity measures have been proposed, both in the classification and regression frameworks. Consider for instance regression and least-square estimators on finite-dimensional vector spaces (the models). When the design is fixed and the noise-level constant equal to σ, Mallows' C_p penalty [16] (equal to $2n^{-1}\sigma^2 D$ for a D-dimensional space, and it can be modified according to the number of models [5, 17]) has some optimality properties [18, 15, 2]. However, such a penalty linear in the dimension may be terrible in an heteroscedastic framework (as shown by (2) and experiment HSd2 in Sect. 6).

In classification, the VC-dimension has the drawback of being independent of the underlying measure, so that it is adapted to the worst case. It has been improved with data-dependent complexity estimates, such as Rademacher complexities [13, 3] (generalized by Fromont with resampling ideas [11]), but they may be too large because they are still global complexity measures. The localization idea then led to local Rademacher complexities [4, 14] which are tight estimates of the ideal penalty, but involve unknown constants and may be very difficult to compute in practice. On the other hand, the V-fold cross-validation (VFCV) is very popular for such purposes, but it is still poorly understood from the non-asymptotic viewpoint.

In this article, we propose a new family of penalties, based on Efron's bootstrap heuristics [10] (and its generalization to weighted bootstrap, i.e. resampling). It is a localized version of Fromont's penalties, which does not involve any unknown constant, and is easy to compute (at the price of some loss in accuracy) in its V-fold cross-validation version. We define it in a much general framework, so that it has a wide range of application. As a first theoretical step, we prove the efficiency of these algorithms in the case example of least-square regression on histograms, under reasonable assumptions. Indeed, they satisfy oracle inequalities with constant almost one, asymptotic optimality and adaptivity to the regularity of the regression function. This comes from explicit computations that allow us to deeply understand why these penalties are working well. Then, we compare the "classical" VFCV with the V-fold penalties, enlightening how V should be chosen. Finally, we illustrate these results with a few simulation experiments. In particular, we show that resampling penalties are competitive with classical methods for "easy problems", and may be much better for some harder ones (e.g. with a variable noise-level).

2 A general model selection algorithm

We consider the following general setting : $\mathcal{X} \times \mathcal{Y}$ is a measurable space, P an unknown probability measure on it and $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \mathcal{Y}$ some data of common law P. Let S be the set of predictors (measurable functions $\mathcal{X} \mapsto \mathcal{Y}$) and $\gamma : S \times (\mathcal{X} \times \mathcal{Y}) \mapsto \mathbb{R}$ a contrast function. Given a family $(\widehat{s}_m)_{m \in \mathcal{M}_n}$ of data-dependent predictors, our goal is to find the one minimizing the prediction loss $P\gamma(t)$. We will extensively use this functional notation $Q\gamma(t) := \mathbb{E}_{(X,Y)\sim Q}[\gamma(t, (X, Y))]$, for any probability measure Q on $\mathcal{X} \times \mathcal{Y}$. Notice that the expectation here is only taken w.r.t. (X, Y), so that $Q\gamma(t)$ is random when $t = \widehat{s}_m$ is random. Assuming that there exists a minimizer $s \in S$ of the loss (the Bayes predictor), we will often consider the excess loss $l(s, t) = P\gamma(t) - P\gamma(s) \geq 0$ instead of the loss.

Assume that each predictor $\widehat{s}_m$ may be written as a function $\widehat{s}_m(P_n)$ of the empirical distribution of the data $P_n = n^{-1} \sum_{i=1}^n \delta_{(X_i,Y_i)}$. The ideal choice for $\widehat{m}$ is the one which minimizes over $\mathcal{M}_n$ the true prediction risk $P\gamma(\widehat{s}_m(P_n)) = P_n\gamma(\widehat{s}_m(P_n)) + \mathrm{pen}_{\mathrm{id}}(m)$ where the ideal penalty is equal to

$$\mathrm{pen}_{\mathrm{id}}(m) = (P - P_n)\gamma(\widehat{s}_m(P_n)) \ .$$

The *resampling heuristics* (introduced by Efron [10]) states that the expectation of any functional $F(P, P_n)$ is close to its resampling counterpart $\mathbb{E}_W F(P_n, P_n^W)$, where $P_n^W = n^{-1} \sum_{i=1}^n W_i \delta_{(X_i,Y_i)}$ is the empirical distribution P_n weighted by an independent random vector $W \in [0; +\infty)^n$, with $\sum_i \mathbb{E}[W_i] = n$. The expectation $\mathbb{E}_W[\cdot]$ means that we only integrate w.r.t. the weights W.

We suggest here to use this heuristics for estimating $\mathrm{pen}_{\mathrm{id}}(m)$, and plug it into the penalized criterion $P_n\gamma(\widehat{s}_m) + \mathrm{pen}(m)$. This defines $\widehat{m} \in \mathcal{M}_n$ as follows.

Algorithm 1 (Resampling penalization).

1. Choose a resampling scheme, i.e. the law of a weight vector W.
2. Choose a constant $C \geq C_W \approx \left(n^{-1} \sum_{i=1}^n \mathbb{E}\left(W_i - 1\right)^2\right)^{-1}$.
3. Compute the following resampling penalty for each $m \in \mathcal{M}_n$:

$$\mathrm{pen}(m) = C\mathbb{E}_W\left[P_n\gamma\left(\widehat{s}_m\left(P_n^W\right)\right) - P_n^W\gamma\left(\widehat{s}_m\left(P_n^W\right)\right)\right] \ .$$

4. Minimize the penalized empirical criterion to choose $\widehat{m}$ and thus $\widehat{s}_{\widehat{m}}$:

$$\widehat{m} \in \arg\min_{m \in \mathcal{M}_n} \left\{P_n\gamma(\widehat{s}_m(P_n)) + \mathrm{pen}(m)\right\} \ .$$

Remark 1.

1. There is a constant $C \neq 1$ in front of the penalty, although there isn't any in Efron's heuristics, because we did not normalize W. The asymptotical value of the right normalizing constant C_W may be derived from Theorem 3.6.13 in [21]. In the case example of histograms, we give a non-asymptotic expression for it (3). In general, we suggest to use some data-driven method to choose C (see algorithm 3), whereas the resampling penalty only estimates the shape of the ideal one.

2. We allowed C to be larger than C_W because overpenalizing may be fruitful in a non-asymptotic viewpoint, e.g. when there is few noisy data.

3. Because of this plug-in method, algorithm 1 seems to be reasonable only if $\mathcal{M}_n$ is not too large, i.e. if it has a polynomial complexity : $\mathrm{Card}(\mathcal{M}_n) \leq c_{\mathcal{M}} n^{\alpha_{\mathcal{M}}}$. Otherwise, we can for instance group the models of similar complexities and reduce $\mathcal{M}_n$ to a polynomial family.

3 The histogram regression case

As studying algorithm 1 in general is a rather difficult question, we focus in this article on the case example of least-square regression on histograms. Although we do not consider histograms as a final goal, this first theoretical step will be useful to derive heuristics making the general algorithm 1 work.

We first precise the framework and some notations. The data $(X_i, Y_i) \in \mathcal{X} \times \mathbb{R}$ are i.i.d. of common law P. Denoting s the regression function, we have

$$Y_i = s(X_i) + \sigma(X_i)\epsilon_i \tag{1}$$

where $\sigma : \mathcal{X} \mapsto \mathbb{R}$ is the heteroscedastic noise-level and ϵ_i are i.i.d. centered noise terms, possibly dependent from X_i, but with variance 1 conditionally to X_i. The feature space $\mathcal{X}$ is typically a compact set of $\mathbb{R}^d$. We use the least-square contrast $\gamma : (t, (x, y)) \mapsto (t(x) - y)^2$ to measure the quality of a predictor $t : \mathcal{X} \mapsto \mathcal{Y}$. As a consequence, the Bayes predictor is the regression function s, and the excess loss is $l(s, t) = \mathbb{E}_{(X,Y)\sim P} \left(t(X) - s(X)\right)^2$. To each model S_m, we associate the empirical risk minimizer $\widehat{s}_m = \widehat{s}_m(P_n) = \arg\min_{t \in S_m} \{P_n \gamma(t)\}$ (when it exists and is unique).

Each model in $(S_m)_{m \in \mathcal{M}_n}$ is the set of piecewise constant functions (histograms) on some partition $(I_\lambda)_{\lambda \in \Lambda_m}$ of $\mathcal{X}$. It is thus a vector space of dimension $D_m = \mathrm{Card}(\Lambda_m)$, spanned by the family $(\mathbb{1}_{I_\lambda})_{\lambda \in \Lambda_m}$. As this basis is orthogonal in $L^2(\mu)$ for any probability measure on $\mathcal{X}$, we can make explicit computations that will be useful to understand algorithm 1. The following

notations will be useful throughout this article.

$$p_\lambda := P(X \in I_\lambda) \qquad \widehat{p}_\lambda := P_n(X \in I_\lambda) \qquad \widehat{p}_\lambda^W = \widehat{p}_\lambda W_\lambda := P_n^W(X \in I_\lambda)$$

$$\begin{aligned}
s_m &:= \arg\min_{t \in S_m} P\gamma(t) = \sum_{\lambda \in \Lambda_m} \beta_\lambda \mathbb{1}_{I_\lambda} & \beta_\lambda &= \mathbb{E}_P[Y | X \in I_\lambda] \\
\widehat{s}_m &:= \arg\min_{t \in S_m} P_n\gamma(t) = \sum_{\lambda \in \Lambda_m} \widehat{\beta}_\lambda \mathbb{1}_{I_\lambda} & \widehat{\beta}_\lambda &= \frac{1}{n\widehat{p}_\lambda} \sum_{X_i \in I_\lambda} Y_i \\
\widehat{s}_m^W &:= \arg\min_{t \in S_m} P_n^W\gamma(t) = \sum_{\lambda \in \Lambda_m} \widehat{\beta}_\lambda^W \mathbb{1}_{I_\lambda} & \widehat{\beta}_\lambda^W &= \frac{1}{n\widehat{p}_\lambda^W} \sum_{X_i \in I_\lambda} W_i Y_i
\end{aligned}$$

Remark that $\widehat{s}_m$ is uniquely defined if and only if each I_λ contains at least one of the X_i, and the same problem arises for $\widehat{s}_m^W$. This is why we will slightly modify the general algorithm for histograms. Before this, we compute the ideal penalty (assuming that $\min_{\lambda \in \Lambda_m} \widehat{p}_\lambda > 0$; otherwise, the model m should clearly not be chosen) :

$$\mathrm{pen}_{\mathrm{id}}(m) = (P - P_n)\gamma(\widehat{s}_m) = \sum_{\lambda \in \Lambda_m} (p_\lambda + \widehat{p}_\lambda) \left(\widehat{\beta}_\lambda - \beta_\lambda\right)^2 + (P - P_n)\gamma(s_m) \ .$$

The last term in the sum being centered, it is estimated as zero by the resampling version of $\mathrm{pen}_{\mathrm{id}}$. The first term is a sum of D_m terms, each one depending only on the restrictions of P and P_n to I_λ. Thus, if we assume that $\widehat{p}_\lambda > 0$ and if we compute separately all those terms, conditionally to $\widehat{p}_\lambda^W > 0$, we can define the resampling version of $\mathrm{pen}_{\mathrm{id}}(m)$. This leads to the following algorithm.

Algorithm 2 (Resampling penalization for histograms). 0. Choose a threshold $A_n \geq 1$ and replace $\mathcal{M}_n$ by

$$\widehat{\mathcal{M}}_n = \left\{ m \in \mathcal{M}_n \text{ s.t. } \min_{\lambda \in \Lambda_m} \{n\widehat{p}_\lambda\} \geq A_n \right\} \ .$$

1. Choose a resampling scheme $\mathcal{L}(W)$.
2. Choose a constant $C \geq C_W(A_n)$ where C_W is defined by (3).

3'. Compute the following resampling penalty for each $m \in \widehat{\mathcal{M}}_n$:

$$\mathrm{pen}(m) = C \sum_{\lambda \in \Lambda_m} \mathbb{E}_W \left[\left(\widehat{p}_\lambda + \widehat{p}_\lambda^W\right) \left(\widehat{\beta}_\lambda^W - \widehat{\beta}_\lambda\right)^2 \middle| W_\lambda > 0 \right] \ .$$

4'. Minimize the penalized empirical criterion to choose $\widehat{m}$ and thus $\widehat{s}_{\widehat{m}}$:

$$\widehat{m} \in \arg\min_{m \in \widehat{\mathcal{M}}_n} \{P_n \gamma(\widehat{s}_m(P_n)) + \mathrm{pen}(m)\} \quad .$$

Remark 2. 1. The two modifications of the algorithm for histograms do not affect much the result if A_n is of the order $\ln(n)$. Indeed, models with very few data are not relevant in general, and if $\min_{\lambda \in \Lambda_m}\{n\widehat{p}_\lambda\} \geq A_n$ is not too small, the event $\{W_\lambda = 0\}$ has a very small probability.

2. We allow C to depend on A_n since the "optimal" constant C_W may depend on it, but this dependence is mild according to our computations.

When the resampling weights are exchangeable (see definition below), we are able to compute pen explicitly. It is enlightening to compare it with $\mathrm{pen}_{\mathrm{id}}$ in expectation, conditionally to $(\widehat{p}_\lambda)_{\lambda \in \Lambda_m}$ (we denote by $\mathbb{E}^m[\cdot]$ this conditional expectation) :

$$\mathbb{E}^m\left[\mathrm{pen}_{\mathrm{id}}(m)\right] = \frac{1}{n}\sum_{\lambda \in \Lambda_m}\left(1 + \frac{p_\lambda}{\widehat{p}_\lambda}\right)\left((\sigma_\lambda^r)^2 + \left(\sigma_\lambda^d\right)^2\right) \tag{2}$$

$$\mathbb{E}^m\left[\mathrm{pen}(m)\right] = \frac{C}{n}\sum_{\lambda \in \Lambda_m}\left(R_{1,W}(n,\widehat{p}_\lambda) + R_{2,W}(n,\widehat{p}_\lambda)\right)\left((\sigma_\lambda^r)^2 + \left(\sigma_\lambda^d\right)^2\right)$$

$$\text{with} \quad (\sigma_\lambda^r)^2 := \mathbb{E}[\sigma(x)^2 | X \in I_\lambda] \;\; ; \quad \left(\sigma_\lambda^d\right)^2 := \mathbb{E}[(s(X) - s_m(X))^2 | X \in I_\lambda]$$

$$\text{and for } k = 1, 2 \qquad R_{k,W}(n,\widehat{p}_\lambda) = \mathbb{E}\left[\frac{(W_i - W_\lambda)^2}{W_\lambda^{3-k}}\Big| W_\lambda > 0\right] \quad .$$

Hence, contrary to Mallows' penalty (with σ^2 known or estimated), resampling penalties really take into account the heteroscedasticity of the noise (σ_λ^r depends on λ) and the bias terms $\left(\sigma_\lambda^d\right)^2$. We then define

$$C_W(A_n) := \sup_{n\widehat{p}_\lambda \geq A_n}\left\{\frac{2}{R_{1,W}(n,\widehat{p}_\lambda) + R_{2,W}(n,\widehat{p}_\lambda)}\right\} \tag{3}$$

and $C'_W(A_n)$ is the infimum of the same quantity.

Examples of resampling weights

In this article, we consider resampling weights $W = (W_1, \ldots, W_n) \in [0; +\infty)^n$ such that $\mathbb{E}[W_i] = 1$ for all i and $\mathbb{E}[W_i^2] < \infty$. We mainly consider the following exchangeable weights (i.e. such that for any permutation τ, $(W_{\tau(1)}, \ldots, W_{\tau(n)}) \overset{(d)}{=} (W_1, \ldots, W_n)$).

1. *Efron* (q) : multinomial vector with parameters $(q; n^{-1}, \ldots, n^{-1})$. Then, $R_{2,W}(n, \widehat{p}_\lambda) = (n/q) \times (1 - (n\widehat{p}_\lambda)^{-1})$. A classical choice is $q = n$.

2. *Rademacher* : W_i i.i.d., 2 times Bernoulli$(1/2)$. Then, $R_{2,W}(n, \widehat{p}_\lambda) = 1$.

3. *Random hold-out* (q) (or cross-validation) : $W_i = \frac{n}{q} \mathbb{1}_{i \in I}$ with I uniform random subset (of cardinality q) of $\{1, \ldots, n\}$. $R_{2,W}(n, \widehat{p}_\lambda) = (n/q) - 1$. A classical choice is $q = n/2$.

4. *Leave-one-out* = Random hold-out $(n-1)$. Then, $R_{2,W}(n, \widehat{p}_\lambda) = (n-1)^{-1}$.

In each case, we can show that $R_{1,W} = R_{2,W}(1 + \delta^{(W)}_{n,\widehat{p}_\lambda})$ for some explicit small term $\delta^{(W)}_{n,\widehat{p}_\lambda}$ (numerically of the same order as $\mathbb{E}[p_\lambda/\widehat{p}_\lambda | \widehat{p}_\lambda > 0] - 1$ in expectation for the three first resamplings, and slightly smaller in the Leave-one-out case). Thus, $C_W \approx C'_W \approx R^{-1}_{2,W}$ (asymptotically in A_n).

For computational reasons, it is also convenient to introduce the following *V-fold cross-validation* resampling weights : given a partition $(B_j)_{1 \leq j \leq V}$ of $\{1, \ldots, n\}$ and $W^B \in \mathbb{R}^V$ leave-one-out weights, we define $W_i = W^B_j$ for each $i \in B_j$. The partition should be taken as regular as possible, and then we can compute $\mathbb{E}[\mathrm{pen}(m)]$ and show that $C_W \approx V - 1$.

The Rademacher weights lead to penalties close in spirit to local Rademacher complexities (the link between global Rademacher complexities and global resampling penalties with Rademacher weights can be found in [11]). The links with the classical leave-one-out and VFCV algorithms are given in Sect. 5.

4 Main results

In this section, we prove that algorithm 2 has some optimality properties under the following restrictions for some non-negative constants $\alpha_{\mathcal{M}}$, $c_{\mathcal{M}}$, c_A, c_{rich} :

(P1) Polynomial complexity of $\mathcal{M}_n$: $\mathrm{Card}(\mathcal{M}_n) \leq c_{\mathcal{M}} n^{\alpha_{\mathcal{M}}}$.

(P2) Richness of $\mathcal{M}_n$: $\forall x \in [1, nc^{-1}_{\mathrm{rich}}]$, $\exists m \in \mathcal{M}_n$ s.t. $D_m \in [x; c_{\mathrm{rich}} x]$.

(P3) The weights are exchangeable, among the examples given in Sect. 3.

(P4) The threshold is large enough : $C_A \ln(n) \geq A_n \geq (26 + 7\alpha_{\mathcal{M}}) \ln(n)$.

Assumption **(P1)** is almost necessary, since too large families of models need larger penalties than polynomial families [5, 2, 17]. Assumption **(P2)** is necessary but it is always satisfied in practice. Assumption **(P3)** is only here

to ensure that we have an explicit formula for the penalty, and sharp bounds on $R_{1,W}$ and $R_{2,W}$. The constant $(26 + 7\alpha_{\mathcal{M}})$ in **(P4)** is quite large due to technical reasons, but much smaller values (larger than 2) should suffice in practice.

Theorem 1. *Assume that the (X_i, Y_i)'s satisfy the following assumptions :*

(Ab) *Bounded data :* $\|Y_i\|_\infty \leq A < \infty$.

(An) *Noise-level bounded from below :* $\sigma(X_i) \geq \sigma_{\min} > 0$ *a.s.*

(Ap) *Polynomial decreasing of the bias :*

$$\exists \beta_1 \geq \beta_2 > 0,\, C_s, c_s > 0 \quad s.t. \quad c_s D_m^{-\beta_1} \leq l(s, s_m) \leq C_s D_m^{-\beta_2} \quad .$$

(Ar) *(pseudo)-Regular histograms :* $\forall m \in \mathcal{M}_n$, $\min_{\lambda \in \Lambda_m}\{p_\lambda\} \geq c_{\text{reg}} D_m^{-1}$.

*Let $\widehat{m}$ be the model chosen by algorithm 2 (under restrictions **(P1–4)**), with $\eta' C'_W(A_n) \geq C \geq \eta C_W(A_n)$ for some $\eta, \eta' > \frac{1}{2}$. It satisfies, with probability at least $1 - L_{(\mathbf{A}),(\mathbf{P})} n^{-2}$ ($L_{(\mathbf{A}),(\mathbf{P})}$ may depend on constants in **(A)** and **(P)**, but not on n),*

$$l(s, \widehat{s}_{\widehat{m}}) \leq K(\eta, \eta') \inf_{m \in \mathcal{M}_n} \{l(s, \widehat{s}_m)\} \quad . \tag{4}$$

At the price of enlarging $L_{(\mathbf{A}),(\mathbf{P})}$, the constant $K(\eta, \eta')$ can be taken close to $(1 + 2(\eta' - 1)_+)(1 - 2(1 - \eta)_+)^{-1}$, where $x_+ := \max(x, 0)$. In particular, $K(\eta, \eta')$ is almost 1 if η and η' are close to 1.

Moreover, we have the oracle inequality

$$\mathbb{E}\left[l(s, \widehat{s}_{\widehat{m}})\right] \leq K(\eta, \eta') \mathbb{E}\left[\inf_{m \in \mathcal{M}_n} \{l(s, \widehat{s}_m)\}\right] + \frac{A^2 L_{(\mathbf{A}),(\mathbf{P})}}{n^2} \quad . \tag{5}$$

sketch. By definition of $\widehat{m}$,

$$\forall m \in \widehat{\mathcal{M}}_n, \quad (\text{pen} - \text{pen}'_{\text{id}})(\widehat{m}) + l(s, \widehat{s}_{\widehat{m}}) \leq l(s, \widehat{s}_m) + (\text{pen} - \text{pen}'_{\text{id}})(m)$$

where we replaced pen_{id} by $\text{pen}'_{\text{id}} := \text{pen}_{\text{id}} - (P_n - P)\gamma(s)$. In order to obtain (4) with $\widehat{\mathcal{M}}_n$ instead of $\mathcal{M}_n$, we show concentration inequalities for $\text{pen}(m) - \text{pen}'_{\text{id}}(m)$ around zero, with remainders $\ll l(s, \widehat{s}_m)$ if D_m is large (larger than some power of $\ln(n)$). We use the following steps :

1. explicit computation of pen'_{id} and pen when W is exchangeable.

2. accurate bounds on $R_{1,W}$ and $R_{2,W}$, so that $(1-\delta(A_n))\mathbb{E}^m[2p_2(m)] \leq \mathbb{E}^m[\mathrm{pen}(m)] \leq (1+\delta(A_n))\mathbb{E}^m[2p_2(m)]$ with $p_2(m) = P_n\left(\gamma(s_m) - \gamma(\widehat{s}_m)\right)$ and $\lim_{A_n\to\infty}\delta = 0$. This needs sharp bounds on $\mathbb{E}[Z^{-1}|Z>0]$ with $\mathcal{L}(Z) = \mathcal{L}(W_\lambda|\widehat{p}_\lambda)$, for each resampling scheme introduced in Sect. 3.

3. moment inequalities for pen, p_2 and $p_1(m) = P\left(\gamma(\widehat{s}_m) - \gamma(s_m)\right)$, conditionally to $(\widehat{p}_\lambda)_{\lambda\in\Lambda_m}$, around their conditional expectations. This step uses results from [7], or can be derived from [12], since all those quantites are U-statistics of order 2 (this last fact is not true without the conditioning). This implies (unconditional) concentration inequalities.

4. concentration inequality for $(P_n - P)(\gamma(s_m) - \gamma(s))$ (Bernstein's inequality suffices in the bounded case).

5. since $\mathbb{E}^m\left[p_2(m)\right] = \mathbb{E}\left[p_2(m)\right]$, it only remains to prove that $\mathbb{E}^m\left[p_1(m)\right] \approx \mathbb{E}\left[p_1(m)\right]$ and $p_2 \approx p_1$ with high probability. We here use the Cramér-Chernoff method (it can be used since the $(\widehat{p}_\lambda)_{\lambda\in\Lambda_m}$ are negatively associated [9]), together with estimates of the exponential moments of the inverse of a binomial random variable. Controlling the remainder needs a lower bound on $\min_{\lambda\in\Lambda_m}\{np_\lambda\}$ that comes from **(P4)** (and Bernstein's inequality).

6. using the assumptions, all the remainders in our concentration inequalities are much smaller than $\mathbb{E}[l(s,\widehat{s}_{\widehat{m}})]$ when $D_m \geq D_0(n) = c_1(\ln(n))^{c_2}$ (with c_1, c_2 depending on the constants in the assumptions).

Let $m^\star$ be a minimizer of $l(s,\widehat{s}_m)$ over $\mathcal{M}_n$ (with an infinite loss when $\widehat{s}_m$ is not uniquely defined). It remains to prove that, with large probability, $D_{\widehat{m}} \geq D_0(n)$, $D_{m^\star} \geq D_0(n)$ and $m^\star \in \widehat{\mathcal{M}}_n$. These hold for n large enough thanks to **(Ap)** and **(Ar)** (we did not use **(Ar)** before).

We finally show that (4) implies (5) : let Ω_n be the event of probability $1 - L_{(\mathbf{A}),(\mathbf{P})}n^{-2}$ on which (4) occurs. On Ω_n^c, $l(s,\widehat{s}_{\widehat{m}})$ is bounded by A^2, so that

$$\begin{aligned}\mathbb{E}\left[l(s,\widehat{s}_{\widehat{m}})\right] &= \mathbb{E}\left[l(s,\widehat{s}_{\widehat{m}})\mathbb{1}_{\Omega_n}\right] + \mathbb{E}\left[l(s,\widehat{s}_{\widehat{m}})\mathbb{1}_{\Omega_n^c}\right] \\ &\leq K(\eta,\eta')\mathbb{E}\left[\inf_{m\in\mathcal{M}_n} l(s,\widehat{s}_m)\right] + L_{(\mathbf{A}),(\mathbf{P})}A^2n^{-2} \ . \qquad \square\end{aligned}$$

$\square$

Theorem 1 implies the a.s. asymptotic optimality of algorithm 3 in this framework. This means that if s and $\sigma(X)$ do not make the model selection problem too hard, the resampling penalization algorithm is working, without

any knowledge on the smoothness of s, the heteroscedasticity of σ or any property that the unknown law P may satisfy. In that sense, it is a *naturally adaptive algorithm.*

The lower bound in assumption **(Ap)** may seem strange, but it is intuitive that when the bias is decreasing very fast, the optimal model is of quite small dimension. Then, bounds relying on the fact that this dimension is large can not work. The same kind of assumption has already been used in the density estimation framework for the same reason [20].

Moreover, we can prove that non-constant hölderian functions satisfy **(Ap)** when X has a lower-bounded density w.r.t. the Lebesgue measure on $\mathcal{X} \subset \mathbb{R}$. The following result states that resampling penalization is adaptive to the hölderian smoothness of s in an heteroscedastic framework, since it attains the minimax rate of convergence $n^{-2\alpha/(2\alpha+1)}$ [19].

Theorem 2. *Let $\mathcal{X}$ be a compact interval of $\mathbb{R}$ and $\mathcal{Y} \subset \mathbb{R}$. Assume that (X_i, Y_i) satisfy **(Ab)**, **(An)** and the following assumptions :*

(Ad) *Density bounded from below : $\exists c_{\min}^X > 0$, $\forall I \subset \mathcal{X}$, $P(X \in I) \geq c_{\min}^X \operatorname{Leb}(I)$.*

(Ah) *Hölderian regression function : there exists $\alpha \in (0;1]$ and $R > 0$ s.t.*

$$s \in \mathcal{H}(\alpha, R) \quad \textit{i.e.} \quad \forall x_1, x_2 \in \mathcal{X},\ |s(x_1) - s(x_2)| \leq R\,|x_1 - x_2|^\alpha \ .$$

*Let $\mathcal{M}_n$ be the family of regular histograms of dimensions $1 \leq D \leq n$, $\widehat{m}$ the model chosen by algorithm 2, with **(P3-4)** satisfied ($\alpha_{\mathcal{M}} = 0$) and C like in Theorem 1. Then, denoting $\sigma_{\max} = \sup_{\mathcal{X}} |\sigma| \leq A$, there are some constants $L_{2,(\mathbf{A}),(\mathbf{P})}$ (that may depend on all the constants in the assumptions) and $L_1(\eta, \eta', \alpha)$ such that*

$$\mathbb{E}\left[l(s, \widehat{s}_{\widehat{m}})\right] \leq L_1 n^{-2\alpha/(2\alpha+1)} R^{2\alpha/(2\alpha+1)} \sigma_{\max}^{4\alpha/(2\alpha+1)} + L_{2,(\mathbf{A}),(\mathbf{P})} n^{-2} \ . \qquad (6)$$

Moreover, if σ is K_σ-Lipschitz, the constant $\sigma_{\max}^2$ may be replaced by $\int_{\mathcal{X}} \sigma(t)^2 dt$ (at the price of enlarging $L_{2,(\mathbf{A}),(\mathbf{P})}$).

sketch.

1. Since $\alpha \in (0;1]$, any non-constant function $s \in \mathcal{H}(\alpha, R)$ satisfies **(Ap)** with $\beta_2 = 2\alpha$ and $\beta_1 = 1 + \alpha^{-1}$ (the lower bound uses **(Ad)**).

2. Assumptions **(P1)**, **(P2)** and **(Ar)** are automatically satisfied by the regular family, so we can use (5). From the proof of Theorem 1, we obtain estimations of $\mathbb{E}\left[l(s, \widehat{s}_m)\right]$. Optimizing in D_m gives (6) for non-constant functions.

3. When s is constant, a direct proof shows that $D_{\widehat{m}}$ is at most of order $\ln(n)^{\xi_1}$ with large probability. This ensures that $\mathbb{E}\left[l(s,\widehat{s}_{\widehat{m}})\right]$ is at most of order $(\ln(n))^{\xi_2}n^{-1} \ll n^{-2\alpha/(2\alpha+1)}$ for every $\alpha > 0$. □

□

Other results like Theorem 1 may be proved under other assumptions : unbounded data (with moment inequalities for the noise, regularity assumptions on s and an upper bound on σ), $\sigma(x)$ that can vanish (with the unbounded assumptions, $\mathbb{E}[\sigma^2(X)] > 0$ and some regularity on σ), etc. We skip their detailed statements in order to focus on the last two sections, where we give a new look on V-fold cross-validation (seen from the penalization viewpoint) and illustrate theoretical results with a simulation study.

5 Links with V-fold cross-validation

The results of Sect. 4 assume that the weights are exchangeable. However, computing exactly the resampling penalties with such weights may be quite long : without a closed formula for pen, $\widehat{s}_m^W$ has to be computed for at least n (and up to 2^n) different weight vectors. Using the V-fold idea, we defined VFCV weights in Sect. 3, that allows to compute each penalty by considering only V different weight vectors. We call the resulting algorithm penVFCV.

It is quite enlightening to compare penVFCV to a more classical version of VFCV, where the final estimator is $\widehat{s}_{\widehat{m}}$ with

$$\widehat{m} \in \arg\min_{m\in\mathcal{M}_n}\left\{\mathrm{crit}_{\mathrm{VFCV}}(m)\right\} = \arg\min_{m\in\mathcal{M}_n}\left\{\frac{1}{V}\sum_{j=1}^{V} P_n^{(j)}\gamma(\widehat{s}_m^{(-j)})\right\} \quad . \tag{7}$$

The superscript (j) (resp. $(-j)$) above means that P_n and $\widehat{s}_m$ are computed with the data belonging to the block B_j (resp. to B_j^c). Assuming that the V blocks have the same size (and forgetting unicity issues of $\widehat{s}_m^{(-j)}$, that may be solved as before), we have (for any j)

$$\begin{aligned}\mathbb{E}\left[\mathrm{crit}_{\mathrm{VFCV}}(m)\right] &= P\gamma(s_m) + \mathbb{E}\left[P\gamma(\widehat{s}_m^{(-j)}) - P\gamma(s_m)\right] \\ &= P\gamma(s_m) + \frac{V}{(V-1)n}\sum_{\lambda\in\Lambda_m}\left(1+\delta_{n,p_\lambda}^{(V)}\right)\left((\sigma_\lambda^r)^2 + \left(\sigma_\lambda^d\right)^2\right)\end{aligned} \tag{8}$$

where $\delta_{n,p_\lambda}^{(V)}$ is typically small and non-negative (when np_λ is large enough).

On the other hand, we can compute exactly the expectation of the penVFCV criterion (with a constant $C = C_W = V - 1$) when the blocks have

the same size :

$$\mathbb{E}\left[\mathrm{crit}_{\mathrm{penVFCV}}(m)\right] = P\gamma(s_m) + \frac{1}{n}\sum_{\lambda\in\Lambda_m}\left(1+\delta_{n,p_\lambda}^{(penV)}\right)\left(\left(\sigma_\lambda^r\right)^2 + \left(\sigma_\lambda^d\right)^2\right) \qquad (9)$$

for some typically small non-negative $\delta_{n,p_\lambda}^{(penV)}$.

Comparing (8) and (9) with (2), one can see that up to small terms, both criterions are in expectation the sum of the bias and a variance term. The main difference between them lies in the constant in front of the variance : it is equal to $C/(V-1) = 1$ for penVFCV, whereas it is equal to $V/(V-1) > 1$ for VFCV.

The classical V-fold cross-validation is thus "overpenalizing" within a factor $V/(V-1)$ because it estimates the generalization ability of $\widehat{s}_m^{(-j)}$, which is built upon less data than $\widehat{s}_m$. This enlightens some clues for the choice of V : *computational issues* (the smaller V, the faster will be the algorithm), *stability* of the algorithm ($V = 2$ is known to be quite unstable, and leave-one-out much more stable), and *overpenalization* ($V/(V-1)$ should not be too far from 1). Our analysis do not quantify the stability issue, but it is sufficient to explain why the asymptotic optimality of leave-p-out needs $p \ll n$ for a prediction purpose [15] and $p \sim n$ for an identification purpose [22]. Indeed, the overpenalization factor is $n/(n-p) = (1-p/n)^{-1}$ should go to 1 for optimal prediction and to infinity for a.s. identification. Moreover, from the non-asymptotic viewpoint (n small and σ large, or s irregular), it is known that overpenalization (i.e. positively biased penalties) gives better results. This means that the better V may not always be the largest one for classical V-fold, independently from computational issues.

On the contrary, penVFCV is not overpenalizing, unless we explicitly choose $C > C_W$. We thus do not have to take into account the third factor for choosing V, so that it may be more accurate than VFCV within a smaller computation time. In the non-asymptotic viewpoint (or for an identification purpose), it is also easier to overpenalize when we need to, without destabilizing the algorithm by taking a small V.

A refined analysis of the "negligible" terms such as $\delta_{n,p_\lambda}^{(penV)}$, compared to the expectation of $p_\lambda/\widehat{p}_\lambda$, explains why the leave-one-out may be overfitting a little (see the simulations hereafter). We do not detail this phenomenon since it disappears when $V/(V-1)$ stays away from 1.

6 Simulations

To illustrate the results of Sect. 4 and the analysis of Sect. 5, we compare the performances of algorithm 2 (with several resampling schemes), Mallows'

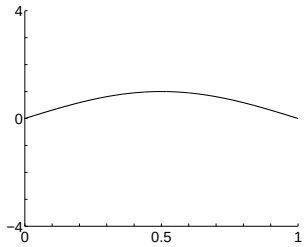

Figure 1: $s(x) = \sin(\pi x)$

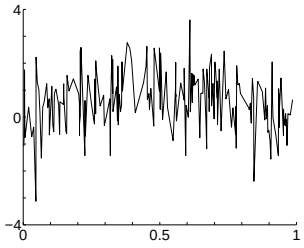

Figure 2: S1

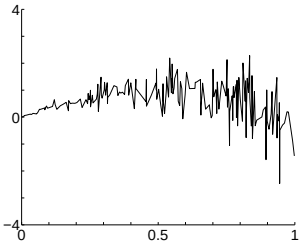

Figure 3: S2

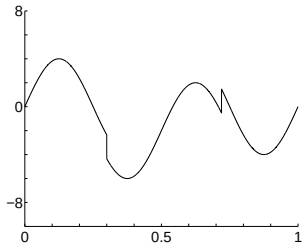

Figure 4: s = HeaviSine (see [8])

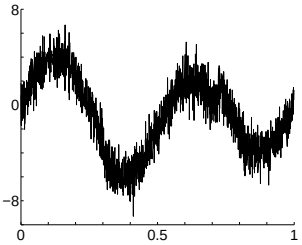

Figure 5: HSd1

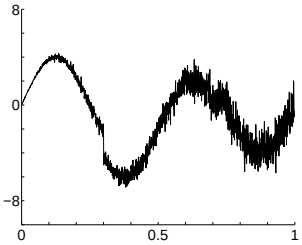

Figure 6: HSd2

C_p and VFCV on some simulated data.

We report here four experiments, called S1, S2, HSd1 and HSd2. Data are generated according to (1) with X_i i.i.d. uniform on $\mathcal{X} = [0; 1]$ and $\epsilon_i \sim \mathcal{N}(0, 1)$ independent from X_i. They differ from the regression function s (smooth for S, see Fig. 1 ; smooth with jumps for HS, see Fig. 4), the noise type (homoscedastic for S1 and HSd1, heteroscedastic for S2 and HSd2), the number n of data, and are repeated $N = 1000$ times. Instances of data sets are given in Fig. 2-3 and 5-6. Their last difference lies in the families of models $\mathcal{M}_n$:

S1 regular histograms with $1 \leq D \leq \frac{n}{\ln(n)}$ pieces.

S2 histograms regular on $\left[0; \frac{1}{2}\right]$ and on $\left[\frac{1}{2}; 1\right]$, with D_1 (resp. D_2) pieces, $1 \leq D_1, D_2 \leq \frac{n}{2\ln(n)}$. The model of constant functions is added to $\mathcal{M}_n$.

HSd1 dyadic regular histograms with 2^k pieces, $0 \leq k \leq \ln_2(n) - 1$.

HSd2 dyadic regular histograms with bin sizes 2^{-k_1} and 2^{-k_2}, $0 \leq k_1, k_2 \leq \ln_2(n) - 1$ (dyadic version of S2). The model of constant functions is added to $\mathcal{M}_n$.

We compare the following algorithms :

Mal Mallows' C_p penalty : $\text{pen}(m) = 2\widehat{\sigma}^2 D_m n^{-1}$ where $\widehat{\sigma}^2$ is the variance estimator used in [1], Sect. 6.

VFCV Classical V-fold cross-validation, defined by (7), with $V \in \{2, 5, 10, 20\}$.

penEfr Efron (n) penalty, $C = C_W = 1$.

penRad Rademacher penalty, $C = C_W = 1$.

penRHO Random hold-out ($n/2$) penalty, $C = C_W = 1$.

penLOO Leave-one-out penalty, $C = C_W = n - 1$.

penVFCV V-fold penalty, with $V \in \{2, 5, 10, 20\}$. $C = C_W = V - 1$.

For each of these except VFCV, we also consider the same penalties multiplied by 5/4 (denoted by a + symbol added after its shortened name). This intends to test for overpenalization.

In each experiment, for each simulated data set, we first remove the models with less than $A_n = 2$ data points in one piece of their associated partition. Then, we compute the least-square estimators $\widehat{s}_m$ for each $m \in \widehat{\mathcal{M}}_n$. Finally, we select $\widehat{m} \in \widehat{\mathcal{M}}_n$ using each algorithm and compute its true excess risk $l(s, \widehat{s}_{\widehat{m}})$ (and the excess risk of each model $m \in \mathcal{M}_n$). Since we simulate N data sets, we can then estimate the two following benchmarks :

$$C_{\text{or}} = \frac{\mathbb{E}\left[l(s, \widehat{s}_{\widehat{m}})\right]}{\mathbb{E}\left[\inf_{m \in \mathcal{M}_n} l(s, \widehat{s}_m)\right]} \qquad C_{\text{path}-\text{or}} = \mathbb{E}\left[\frac{l(s, \widehat{s}_{\widehat{m}})}{\inf_{m \in \mathcal{M}_n} l(s, \widehat{s}_m)}\right]$$

Basically, C_{or} is the constant that should appear in an oracle inequality like (4), and $C_{\text{path}-\text{or}}$ corresponds to a pathwise oracle inequality like (5). As C_{or} and $C_{\text{path}-\text{or}}$ approximatively give the same rankings between algorithms, we only report C_{or} in Tab. 1.

We always observe that penRad and penRHO are competitive with Mal (S1) and much better for more "difficult" problems (S2 is heteroscedastic ; jumps in HSd1 and HSd2 induce much bias). On the other hand, VFCV is a little worse than Mal for easy problems (S1) and better for more difficult ones, but never better than penRad or penRHO.

The best resampling schemes (not taking overpenalization into account) are penRad and penRHO, in view of S1 and S2 (dyadic models do not induce much differences between them in HSd1 and HSd2). Then, penLOO is slightly underpenalizing and penEfr strongly overfits. The comparison penRad $\approx$ penRHO $>$ penLOO $\gg$ penEfr can also be derived from Sect. 3.

In the four experiments, overpenalizing within a factor 5/4 leads to better results, mainly because n is quite small for the noisy (S1, S2) or irregular

(HSd1, HSd2) signals observed. This is no longer the case for some larger n or smaller σ.

We consider now V-fold algorithms. VFCV is slightly better than penVFCV, but worse than penVFCV+. The influence of V on C_{or} confirms the discussion of Sect. 5. For VFCV, the best V may be $V = 2$ (which overpenalizes, HSd1) or $V = 20$ (which is more stable, HSd2), or even both (S1,S2). On the contrary, penVFCV (and penVFCV+) is always improved when V increases, or at least it does not get worse. Then, the best one is penLOO (or penLOO+), i.e. $V = n$, the small terms $\delta_{n,p_\lambda}^{(penV)}$ being far less important than stability. This enlightens the interest of defining V-fold penalties, for which it is easier to solve the complexity-accuracy trade-off.

Remark 3. We only report here the result of 4 experiments, but several other ones (with n larger, σ smaller, $\sigma(x) = \mathbb{1}_{x\in\left[\frac{1}{2};1\right]}$ or other regression functions s such as Doppler, $\sqrt{\cdot}$ and a regular histogram) give the same kind of results. The constants C_{or} and $C_{\text{path}-\text{or}}$ are decreasing to 1 when n increases and σ decreases.

The overpenalization factor 5/4 is generally not optimal, and even not always better than 1 (in particular when n is large or σ small). We have for instance $C_{\text{or}}(\text{penLOO}) < C_{\text{or}}(\text{penRHO}) < C_{\text{or}}(\text{penRHO+})$ in S1 with $\sigma \equiv 0.1$ (with only small differences).

On the tuning parameters

The above simulations confirm that the best weights (for accuracy) are Random hold-out ($n/2$) and Rademacher, whereas V-fold or leave-one-out weights may be of interest for computational purposes. The second tuning parameter, A_n, may be taken equal to 2 (its "minimal" value because terms of the penalty with $n\widehat{p}_\lambda = 1$ would be zero) without serious consequences on C_{or} in practice.

On the contrary, the constant $C \geq C_W$ is quite important, and the best ratio C/C_W strongly depends on n, σ, s and $\mathcal{M}_n$. Moreover, there is no reason for C_W(histograms) to be the right non-asymptotic constant in the general algorithm 1. Our suggest is to choose C with the so-called "slope heuristics", proposed by Birgé and Massart [6] for penalties linear in dimension. Their claim is that the optimal penalty is twice the minimal penalty, i.e. the one under which the selected model is obviously too large. This leads to estimating the shape of pen_{id} by resampling, and the constant C with the slope heuristics, as follows.

Table 1: Accuracy indexes C_{or} for each algorithm in four experiments, $\pm$ a rough estimate of uncertainty of the value reported (i.e. the empirical standard deviation divided by $\sqrt{N}$). In each column, the more accurate algorithms (taking the uncertainty into account) are bolded.

Experiment	S1	S2	HSd1	HSd2
s	sin	sin	HeaviSine	HeaviSine
$\sigma(x)$	1	x	1	x
n (data)	200	200	2048	2048
$\mathcal{M}_n$	regular	2 bin sizes	dyadic, regular	dyadic, 2 bin sizes
Mal	1.928 ± 0.04	3.864 ± 0.02	1.606 ± 0.015	1.487 ± 0.011
Mal+	$\mathbf{1.800 \pm 0.03}$	4.047 ± 0.02	1.606 ± 0.015	1.487 ± 0.011
2−FCV	2.078 ± 0.04	2.542 ± 0.05	$\mathbf{1.002 \pm 0.003}$	1.184 ± 0.004
5−FCV	2.137 ± 0.04	2.582 ± 0.06	1.014 ± 0.003	1.115 ± 0.005
10−FCV	2.097 ± 0.05	2.603 ± 0.06	1.021 ± 0.003	1.109 ± 0.004
20−FCV	2.088 ± 0.04	2.578 ± 0.06	1.029 ± 0.004	1.105 ± 0.004
penEfr	2.597 ± 0.07	3.152 ± 0.07	1.067 ± 0.005	1.114 ± 0.005
penRad	1.973 ± 0.04	2.485 ± 0.06	1.018 ± 0.003	$\mathbf{1.102 \pm 0.004}$
penRHO	1.982 ± 0.04	2.502 ± 0.06	1.018 ± 0.003	1.103 ± 0.004
penLOO	2.080 ± 0.05	2.593 ± 0.06	1.034 ± 0.004	1.105 ± 0.004
pen2−FCV	2.578 ± 0.06	3.061 ± 0.07	1.038 ± 0.004	1.103 ± 0.005
pen5−FCV	2.219 ± 0.05	2.750 ± 0.06	1.037 ± 0.004	1.104 ± 0.004
pen10−FCV	2.121 ± 0.05	2.653 ± 0.06	1.034 ± 0.004	1.104 ± 0.004
pen20−FCV	2.085 ± 0.04	2.639 ± 0.06	1.034 ± 0.004	1.105 ± 0.004
penEfr+	2.016 ± 0.05	2.605 ± 0.06	1.011 ± 0.003	$\mathbf{1.097 \pm 0.004}$
penRad+	$\mathbf{1.799 \pm 0.03}$	$\mathbf{2.137 \pm 0.05}$	$\mathbf{1.002 \pm 0.003}$	$\mathbf{1.095 \pm 0.004}$
penRHO+	$\mathbf{1.798 \pm 0.03}$	$\mathbf{2.142 \pm 0.05}$	$\mathbf{1.002 \pm 0.003}$	$\mathbf{1.095 \pm 0.004}$
penLOO+	$\mathbf{1.844 \pm 0.03}$	$\mathbf{2.215 \pm 0.05}$	$\mathbf{1.004 \pm 0.003}$	$\mathbf{1.096 \pm 0.004}$
pen2−FCV+	2.175 ± 0.05	2.748 ± 0.06	1.011 ± 0.003	1.106 ± 0.004
pen5−FCV+	1.913 ± 0.03	2.378 ± 0.05	$\mathbf{1.006 \pm 0.003}$	$\mathbf{1.102 \pm 0.004}$
pen10−FCV+	1.872 ± 0.03	2.285 ± 0.05	$\mathbf{1.005 \pm 0.003}$	$\mathbf{1.098 \pm 0.004}$
pen20−FCV+	1.898 ± 0.04	2.254 ± 0.05	$\mathbf{1.004 \pm 0.003}$	$\mathbf{1.098 \pm 0.004}$

Algorithm 3 (Resampling penalization with slope heuristics). 1. Choose a resampling scheme, i.e. the law of a weight vector W.

2. Compute the following resampling penalty for each $m \in \mathcal{M}_n$:

$$\mathrm{pen}_0(m) = \mathbb{E}_W \left[P_n \gamma \left(\widehat{s}_m \left(P_n^W \right) \right) - P_n^W \gamma \left(\widehat{s}_m \left(P_n^W \right) \right) \right] \ .$$

3. Compute the selected model $\widehat{m}(C)$ as a function of $C > 0$

$$\widehat{m}(C) \in \arg\min_{m \in \mathcal{M}_n} \left\{ P_n \gamma(\widehat{s}_m(P_n)) + C \, \mathrm{pen}_0(m) \right\} \ .$$

4. Choose the minimal $C = \widehat{C}$ such that $D_{\widehat{m}(C)}$ is "reasonably small", and take $\widehat{m} = \widehat{m}(2\widehat{C})$.

Step 4 may need to artificially introduce huge models in $\mathcal{M}_n$, all the other ones being considered as "reasonably small". Finally, notice that $C \mapsto \widehat{m}(C)$ is piecewise constant with at most $\mathrm{Card}(\mathcal{M}_n)$ jumps, so that steps 3–4 have a complexity $\mathcal{O}(\mathrm{Card}(\mathcal{M}_n))$. As a consequence, the V-fold algorithm 3 is fastly computable.

Acknowledgements

I gratefully thank Pascal Massart for many fruitful discussions.

References

[1] Yannick Baraud. Model selection for regression on a fixed design. *Probab. Theory Related Fields*, 117(4):467–493, 2000.

[2] Yannick Baraud. Model selection for regression on a random design. *ESAIM Probab. Statist.*, 6:127–146 (electronic), 2002.

[3] P. Bartlett, S. Boucheron, and G. Lugosi. Model selection and error estimation. *Machine Learning*, 48:85–113, 2002.

[4] Peter L. Bartlett, Olivier Bousquet, and Shahar Mendelson. Local Rademacher complexities. *Ann. Statist.*, 33(4):1497–1537, 2005.

[5] Lucien Birgé and Pascal Massart. Gaussian model selection. *J. Eur. Math. Soc. (JEMS)*, 3(3):203–268, 2001.

[6] Lucien Birgé and Pascal Massart. Minimal penalties for gaussian model selection. *Probab. Theory Related Fields*, 134(3), 2006.

[7] Stéphane Boucheron, Olivier Bousquet, Gábor Lugosi, and Pascal Massart. Moment inequalities for functions of independent random variables. *Ann. Probab.*, 33(2):514–560, 2005.

[8] David L. Donoho and Iain M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Statist. Assoc.*, 90(432):1200–1224, 1995.

[9] Devdatt Dubhashi and Desh Ranjan. Balls and bins: a study in negative dependence. *Random Structures Algorithms*, 13(2):99–124, 1998.

[10] B. Efron. Bootstrap methods: another look at the jackknife. *Ann. Statist.*, 7(1):1–26, 1979.

[11] Magalie Fromont. Model selection by bootstrap penalization for classification. In *Learning theory*, volume 3120 of *Lecture Notes in Comput. Sci.*, pages 285–299. Springer, Berlin, 2004.

[12] Evarist Giné, Rafał Latała, and Joel Zinn. Exponential and moment inequalities for U-statistics. In *High dimensional probability, II (Seattle, WA, 1999)*, volume 47 of *Progr. Probab.*, pages 13–38. Birkhäuser Boston, Boston, MA, 2000.

[13] Vladimir Koltchinskii. Rademacher penalties and structural risk minimization. *IEEE Trans. Inform. Theory*, 47(5):1902–1914, 2001.

[14] Vladimir Koltchinskii. 2004 IMS Medallion Lecture : Local Rademacher Complexities and Oracle Inequalities in Risk Minimization. *Ann. Statist.*, 34(6), 2006.

[15] Ker-Chau Li. Asymptotic optimality for C_p, C_L, cross-validation and generalized cross-validation: discrete index set. *Ann. Statist.*, 15(3):958–975, 1987.

[16] C. L. Mallows. Some comments on C_p. *Technometrics*, 15:661–675, 1973.

[17] Marie Sauvé. Histogram selection in non gaussian regression. Technical Report 5911, INRIA, may 2006.

[18] Ritei Shibata. An optimal selection of regression variables. *Biometrika*, 68(1):45–54, 1981.

[19] Charles J. Stone. Optimal rates of convergence for nonparametric estimators. *Ann. Statist.*, 8(6):1348–1360, 1980.

[20] Charles J. Stone. An asymptotically optimal histogram selection rule. In *Proceedings of the Berkeley conference in honor of Jerzy Neyman and Jack Kiefer, Vol. II (Berkeley, Calif., 1983)*, Wadsworth Statist./Probab. Ser., pages 513–520, Belmont, CA, 1985. Wadsworth.

[21] Aad W. van der Vaart and Jon A. Wellner. *Weak convergence and empirical processes*. Springer Series in Statistics. Springer-Verlag, New York, 1996. With applications to statistics.

[22] Ping Zhang. Model selection via multifold cross validation. *Ann. Statist.*, 21(1):299–313, 1993.