

REGULARIZATION, BOOSTING AND MIRROR AVERAGING

ALEXANDRE TSYBAKOV,
UNIVERSITÉ PARIS VI

In their paper, Peter Bickel and Bo Li give an interesting unified view of regularization methods in statistics. The literature on this subject is immense, so they outline a general conceptual approach, and then focus on some selected problems where regularization is used, such as regression and classification, or more generally, prediction. In this context, they discuss in detail a number of recently emerging techniques, in particular, boosting, estimation of large covariance matrices, estimation in the models where the dimension is larger than the sample size.

It is difficult to overestimate the importance of regularization in statistics, especially in nonparametrics. Most of nonparametric estimation problems are ill-posed, and common estimators (kernel, histogram, spline, orthogonal series etc.) are nothing but regularized methods of solving them. The corresponding regularization parameters are just smoothing parameters of the estimators.

The main ideas of statistical regularization can be very transparently explained for prediction problems. Assume that X_1, \dots, X_n are i.i.d. observations taking values in a space \mathcal{X} , and assume that the unknown underlying function f^* that we want to estimate belongs to a space \mathcal{F} . Consider a loss function $Q : \mathcal{X} \times \mathcal{F} \rightarrow \mathbb{R}$ and the associated prediction risk

$$R(f) = \mathbf{E} Q(X, f)$$

where X has the same distribution as X_i . Assume that f^* is a minimizer of the risk $R(f)$ over \mathcal{F} . Then a classical, but not always reasonable, estimator of f^* is a minimizer over $f \in \mathcal{F}$ of the corresponding empirical risk

$$R_n(f) = \frac{1}{n} \sum_{i=1}^n Q(X_i, f).$$

Clearly, if \mathcal{F} is too large, this can lead to overfitting and the minimizers can be nonsense. On the other extreme, if \mathcal{F} is chosen to be too small, we cannot be sure that the true function f^* belongs to \mathcal{F} . So, continuing in the logic of empirical risk minimization, we need to know rather precisely a class \mathcal{F} (the smaller, the better) where f^* lies. This, of course, is not very realistic in practice, but minimizing R_n over a suitable restricted class \mathcal{F} yields us a first way of statistical regularization. For example, we can minimize R_n over the class of twice differentiable functions such that $\int (f'')^2 \leq L$ where L is a given constant. Closely related is the second way of statistical regularization where a “roughness” penalty $\text{pen}(f)$ is added to $R_n(f)$, for example, $\text{pen}(f) = \lambda \int (f'')^2$ where $\lambda > 0$ is a smoothing parameter, and the estimator of f^* is defined as a minimizer of $R_n(f) + \text{pen}(f)$.

These examples illustrate a construction of estimators for a given (fixed) smoothness of the underlying function. To get adaptation to unknown smoothness or other types of adaptation, we need one more stage of regularization, typically realized as penalization but this time over the complexity (smoothing) parameter appearing at the first stage. For instance, the famous Mallows – Akaike or cross-validation type schemes can be used.

Such a two-stage procedure works well in many cases. However, it has been recently realized that often it does not take advantage of the sparseness property. On the other hand, sparseness is believed to be an inalienable feature of many modern problems of signal processing and classification where “ p is larger than n ”, in the terminology of Peter Bickel and Bo Li. A remedy can be then suggested in the form of alternative regularization procedures, with one stage only, which turn out to have optimal properties both in “classical” and “sparse” cases. One of the main ideas is to use an ℓ_1 penalization of the empirical risk or, on a closely related note, minimization of the empirical risk under an ℓ_1 constraint. In its earliest and simplest form, this idea appears in soft thresholding of Donoho and Johnstone for the gaussian sequence model. For other models, e.g., in regression and classification, it is realized in more recent procedures, such as Lasso, various versions of boosting or mirror averaging.

Let us focus here on boosting and mirror averaging. Consider a dictionary \mathcal{H} of functions on \mathcal{X} . Assume without loss of generality that the dictionary is finite: $\mathcal{H} = \{h_1, \dots, h_M\}$, but M can be very large, for example, much larger than the sample size n . We believe that the underlying function f^* is well approximated either by the linear span $\mathcal{L}(\mathcal{H})$ of \mathcal{H} or by its convex hull $\mathcal{C}(\mathcal{H})$. The aim is then to find an estimator \tilde{f}_n such that its risk $R(\tilde{f}_n)$ would be close to the oracle risks $\inf_{f \in \mathcal{L}(\mathcal{H})} R(f)$ or $\inf_{f \in \mathcal{C}(\mathcal{H})} R(f)$. To get such an estimator \tilde{f}_n , we can implement ℓ_1 regularization, in particular, some versions of boosting. We can also use the method of mirror averaging.

Boosting. It will be convenient to distinguish between *linear boosting* where the output \tilde{f} of the procedure belongs to the linear span of \mathcal{H} (not restricted to its convex hull), and *convex boosting* where \tilde{f} belongs to the convex hull $\mathcal{C}(\mathcal{H})$. Convex boosting methods can be viewed as ℓ_1 penalized procedures since the set $\mathcal{C}(\mathcal{H})$ is determined by an ℓ_1 constraint. Peter Bickel and Bo Li describe a basic linear boosting algorithm for the problem of classification (cf. (3.8)). Clearly, it can be also written for a general prediction problem:

- initialize: pick $F_0 \in \mathcal{L}(\mathcal{H})$,
- for $k = 0, 1, \dots, k^*$, find

$$(\hat{\gamma}_k, \hat{h}_k) = \operatorname{argmin}_{\gamma \in \mathbb{R}, h \in \mathcal{H}} R_n(F_k + \gamma h)$$

and set $F_{k+1} = F_k + \hat{\gamma}_k \hat{h}_k$,

- output $\tilde{f}_n = F_{k^*+1}$.

Here the stopping time $k^* \leq M - 1$ is a regularization parameter of the algorithm. It can be selected by classical methods, as mentioned above, by adding a second stage of the procedure, i.e., a minimization of some criterion penalizing for large values of k . This is realized for the regression problem with squared loss by Bühlmann and

Yu (2005*), Bickel et al. (2006*), Barron et al. (2005), and for classification with convex loss by Zhang and Yu (2005*) [here and below the * sign indicates references to the bibliography of the paper of Peter Bickel and Bo Li]. Peter Bickel and Bo Li suggest in (3.9) another boosting method which is based on ℓ_1 penalization. They also provide its heuristic motivation. Some questions remain open here: how to choose k' in (3.9)? Does the method require a “second stage”, i.e., a model selection step for early stopping?

For the regression problem with squared loss and for some linear boosting procedures \tilde{f}_n , Barron et al (2005), under mild assumptions on the functions h_j from the dictionary, prove oracle inequalities of the form

$$(1) \quad \mathbf{E}\{R(\tilde{f}_n)\} \leq C \inf_{f \in \mathcal{C}(\mathcal{H})} R(f) + \Delta_n$$

where $\Delta_n > 0$ tends to 0, but not faster than $n^{-1/2}$, and $C > 1$ is a constant. This shows that, in fact, their linear boosting procedures \tilde{f}_n mimic the convex oracle.

Mannor et al. (2003), Lugosi and Vayatis (2004*) and Klemelä (2006) establish oracle inequalities similar to (1) for some convex boosting procedures. However, there is no evidence that boosting mimics well linear oracles. For a particular linear boosting scheme, an inequality similar to (1) but involving linear oracle risk $\inf_{f \in \mathcal{L}(\mathcal{H})} R(f)$ has been obtained by Zhang and Yu (2005*), however, with a remainder term Δ_n far from optimality. It would be, indeed, interesting to investigate whether boosting can achieve optimal rates of aggregation given in [9]. This can be, in principle, obtained as a consequence of *sparsity oracle inequalities*, i.e., inequalities of the form

$$(2) \quad \mathbf{E}\{R(\tilde{f}_n)\} \leq C \inf_{f \in \mathcal{L}(\mathcal{H})} \left\{ R(f) + \frac{M(f)}{n} \log M \right\}$$

where $C \geq 1$ and $M(f)$ is the number of non-zero coefficients in the \mathcal{H} -representation of f :

$$M(f) = \min \left\{ \sum_{j=1}^M \mathbb{I}_{\{\lambda_j \neq 0\}} : f = \sum_{j=1}^M \lambda_j h_j \right\}$$

An open question is whether there exist a boosting procedure \tilde{f}_n satisfying (2). Note that, in fact, (2) can be proved for other procedures: a first example is given in [2, 3] where (2) is established for a Lasso type \tilde{f}_n in the regression model with squared loss.

Mirror averaging. A competitor of boosting is the mirror averaging (MA) algorithm [4, 5]. It aims to achieve the same goal as the boosting procedures discussed above which is to mimic the convex or linear oracles associated to a given dictionary of functions \mathcal{H} (or to mimic the model selection oracle). The following two properties give evidence in favor of MA, as compared to boosting:

- unlike boosting, MA is an on-line method: it is applicable with streaming data. The computational cost of MA is of the same order or even smaller than that of boosting;
- in the theory, at least at its actual stage, better oracle inequalities are available for MA than for boosting.

To define the MA algorithm we introduce some notation. For any $\theta = (\theta^{(1)}, \dots, \theta^{(M)}) \in \Theta \subseteq \mathbb{R}^M$ set $f_\theta = \sum_{j=1}^M \theta^{(j)} h_j$ and assume that Θ is convex and that $\theta \mapsto Q(X, f_\theta)$ is convex for all $X \in \mathcal{X}$, with (sub)gradient $\nabla_\theta Q(X, f_\theta)$. Given a sequence of positive numbers β_i , the basic MA algorithm is defined as follows:

- $i = 0$: initialize values $\zeta_0 \in \mathbb{R}^M$, $\bar{\theta}_0 \in \Theta$, $\tilde{\theta}_0 \in \Theta$,
- for $i = 1, \dots, n$, iterate:

$$\begin{aligned} \zeta_i &= \zeta_{i-1} + \nabla_\theta Q(X_i, f_{\bar{\theta}_{i-1}}) && \text{(GRADIENT DESCENT)} \\ \bar{\theta}_i &= G(\zeta_i / \beta_i) && \text{(MIRRORING)} \\ \tilde{\theta}_i &= \tilde{\theta}_{i-1} - (\tilde{\theta}_{i-1} - \bar{\theta}_{i-1}) / i && \text{(AVERAGING)} \end{aligned}$$

- output $\tilde{\theta}_n$ and set $\tilde{f}_n = f_{\tilde{\theta}_n}$.

Here $G : \mathbb{R}^M \rightarrow \Theta$ is a specially chosen “mirroring” mapping. When Θ is the simplex, $\Theta = \Lambda^M = \left\{ \theta = (\theta^{(1)}, \dots, \theta^{(M)}) : \theta^{(j)} \geq 0, \sum_{j=1}^M \theta^{(j)} = 1 \right\}$, a possible choice of G is

$$(3) \quad G(z) = \left(\frac{\exp(-z^{(1)})}{\sum_{j=1}^M \exp(-z^{(j)})}, \dots, \frac{\exp(-z^{(M)})}{\sum_{j=1}^M \exp(-z^{(j)})} \right),$$

where $z = (z^{(1)}, \dots, z^{(M)})$. Remark that choosing Θ as a simplex Λ^M can be viewed as an ℓ_1 regularization, this point is in common with the convex boosting procedures. Note also that the recursive averaging step of the MA algorithm is equivalent to

$$\tilde{\theta}_n = \frac{1}{n} \sum_{i=1}^n \bar{\theta}_{i-1}.$$

Therefore, when $\Theta = \Lambda^M$, the vector of weights $\tilde{\theta}_n$ belongs to the simplex Λ^M , so that \tilde{f}_n is a convex mixture of the functions h_j with data-dependent weights.

Under the appropriate choice of β_i , the MA estimator \tilde{f}_n with $\Theta = \Lambda^M$ and with $G(\cdot)$ as defined in (3) satisfies the following oracle inequality [4]:

$$(4) \quad \mathbf{E}\{R(\tilde{f}_n)\} \leq \inf_{f \in \mathcal{C}(\mathcal{H})} R(f) + 2\sqrt{Q^*} \sqrt{\frac{\log M}{n}}$$

where

$$Q^* = \sup_{\theta \in \Lambda^M} \mathbf{E} \|\nabla_\theta Q(Z, f_\theta)\|_\infty^2.$$

Here and below $\|\cdot\|_p$ stands for the ℓ_p norm in \mathbb{R}^M . Inequality (4) shows that the MA algorithm mimics the convex oracle with optimal rate $\sqrt{\frac{\log M}{n}}$. It is sharper than the corresponding bound for boosting (1) because the risk of the oracle $\inf_{f \in \mathcal{C}(\mathcal{H})} R(f)$ in (4) appears with the minimal possible constant $C = 1$. Furthermore, (1) is only proved for the regression model with squared loss, while (4) is valid for any prediction model with convex loss.

In general, MA applies to any convex loss function whereas boosting is usually operational with the squared loss or with some special loss functions [an exception seems

to be the gradient boosting of Mason et al. (2000) but not much is known about its theoretical properties].

There are also some computational advantages of MA as compared to boosting. The computational cost of boosting with finite dictionary of cardinality M is of the order nM^2 : the cost of each iteration is of the order nM since we have to compare M different values of R_n , and this is multiplied by M since we need to run M iterations in order to select the stopping time k^* by comparing their outputs. For some versions of boosting the cost is of the order nMk^* where the random stopping time $k^* \leq M - 1$ cannot be evaluated in advance. The computational cost of MA is just $O(nM)$, i.e., n iterations with vectors of dimension M . If M is very large, for example, $M \gg n$, the difference between the two costs becomes substantial.

For a general convex set Θ , the mirror mapping G is defined as

$$(5) \quad G(z) = \arg \min_{\theta \in \Theta} \left\{ (z, \theta) + V(\theta) \right\}$$

where (\cdot, \cdot) denotes the scalar product in \mathbb{R}^M and $V : \Theta \rightarrow \mathbb{R}^M$ is a convex penalty which is strongly convex w.r.t. the ℓ_1 norm in \mathbb{R}^M . The last requirement makes it impossible to take V as the ℓ_1 norm of θ , but a sensible choice [4] is to use a penalty based on a norm that are quite close to the ℓ_1 norm, for example,

$$(6) \quad V(\theta) = \frac{1}{2} \|\theta\|_p^2, \quad p = 1 + \frac{1}{\log M}.$$

With this penalty and $\Theta = \mathbb{R}^M$, the mirror mapping G in (5) has the form

$$G(z) = - \left(\sum_{j=1}^M |z^{(j)}|^{\frac{p}{p-1}} \right)^{1-\frac{2}{p}} \left(|z^{(1)}|^{\frac{1}{p-1}} \text{sign } z^{(1)}, \dots, |z^{(M)}|^{\frac{1}{p-1}} \text{sign } z^{(M)} \right).$$

To compare, the function G with exponential weights defined in (3) is a solution of (5) with $\Theta = \Lambda^M$ and the entropic penalty $V(\theta) = \sum_{j=1}^M \theta^{(j)} \log \theta^{(j)}$. This penalty also satisfies the strong convexity property w.r.t. the ℓ_1 norm (see [4]). We see that MA with exponential weights operates with two types of penalization: the first of them is an ℓ_1 penalization due to a restriction of θ to the simplex $\Theta = \Lambda^M$, and the second one comes with the entropic penalty $V(\theta)$.

It would be interesting to understand whether the sparsity oracle inequalities of the type (2) can be proved for the MA algorithm. Some additional conditions on the loss function Q , such as strong convexity, might be needed to make it possible.

REFERENCES

- [1] Barron, A., Cohen, A., Dahmen, W. and DeVore, R. (2005). Approximation and learning by greedy algorithms. Manuscript.
- [2] Bunea F., Tsybakov, A. and Wegkamp M. (2005). Aggregation for gaussian regression. *Annals of Statistics*, tentatively accepted.
- [3] Bunea F., Tsybakov, A. and Wegkamp M. (2006). In *Proceedings of 19th Annual Conference on Learning Theory (COLT 2006). Lecture Notes in Artificial Intelligence*, v.4005, 379–391 (Lugosi, G. and Simon, H.U.,eds.), Springer-Verlag, Berlin-Heidelberg.

- [4] Juditsky, A., Nazin, A., Tsybakov, A. and Vayatis, N. (2005). Recursive aggregation of estimators by mirror descent algorithm with averaging. *Problems of Information Transmission*, **41**, n.4, 368-384.
- [5] Juditsky, A, Rigollet, Ph., Tsybakov, A. (2005). Learning by mirror averaging. Preprint, Laboratoire de Probabilités et Modèles Aléatoires, Universités Paris 6 – Paris 7. <https://hal.ccsd.cnrs.fr/ccsd-00014097>
- [6] Klemelä, J. (2006). Density estimation with stagewise optimization of the empirical risk. Manuscript.
- [7] Mannor, S., Meir, R. and Zhang, T. (2003) Greedy algorithms for classification – consistency, convergence rates, and adaptivity. *Journal of Machine Learning Research*, **4** 713-742.
- [8] Mason, L., Baxter, J., Bartlett, P.L. and Frean, M. (2000) Functional gradient techniques for combining hypotheses. In *Advances in Large Margin Classifiers* (A.J.Smola et al. eds). MIT Press, Cambridge MA, 221-246.
- [9] Tsybakov, A.B. (2003). Optimal rates of aggregation. In *Proceedings of 16th Annual Conference on Learning Theory (COLT) and 7th Annual Workshop on Kernel Machines. Lecture Notes in Artificial Intelligence*, v. 2777, p.303–313. Springer-Verlag, Heidelberg.

ALEXANDRE B. TSYBAKOV, LABORATOIRE DE PROBABILITÉS ET MODÈLES ALÉATOIRES, UNIVERSITÉ PARIS VI, FRANCE.

E-mail address: tsybakov@ccr.jussieu.fr