massachusetts institute of technology — artificial intelligence laboratory

Bagging Regularizes

Tomaso Poggio, Ryan Rifkin,
Sayan Mukherjee and Alex Rakhlin

AI Memo 2002-003 March 2002
CBCL Memo 214

Abstract

Intuitively, we expect that averaging — or bagging — different regressors with low correlation should smooth their behavior and be somewhat similar to regularization. In this note we make this intuition precise. Using an almost classical definition of stability, we prove that a certain form of averaging provides generalization bounds with a rate of convergence of the same order as Tikhonov regularization — similar to fashionable RKHS-based learning algorithms.

This report describes research done within the Center for Biological & Computational Learning which is part of the McGovern Institute, the Department of Brain & Cognitive Sciences and the Artificial Intelligence Laboratory at the Massachusetts Institute of Technology.
This research was sponsored by grants from: Office of Naval Research (DARPA) under contract No. N00014-00-1-0907, National Science Foundation (ITR) under contract No. IIS-0085836, National Science Foundation (KDI) under contract No. DMS-9872936, and National Science Foundation under contract No. IIS-9800032
Additional support was provided by: Central Research Institute of Electric Power Industry, Eastman Kodak Company, DaimlerChrysler AG, Compaq, Honda R&D Co., Ltd., Komatsu Ltd., NEC Fund, Siemens Corporate Research, Inc., and The Whitaker Foundation.

Introduction

Learning from examples can be regarded [8] as the problem of approximating a multivariate function from sparse data[1]. The function can be real valued as in regression or binary valued as in classification.
The accuracy of the approximated function is based upon its performance on future data, measured in terms of its *generalization error*. Given $\mathbf{x} \in \mathbb{R}^d$ and $y \in \mathbb{R}$ with underlying probability distribution $P(\mathbf{x}, y)$, the generalization error of a function f is

$$I_{exp}[f] \equiv \int V(y, f(\mathbf{x}))P(\mathbf{x}, y)\, d\mathbf{x}dy, \tag{1}$$

where V is the loss function (a typical example is the square loss, $V(y, f(\mathbf{x})) = (y - f(\mathbf{x})^2)$. Usually we do not know the distribution $P(\mathbf{x}, y)$. We have only the ℓ training pairs drawn from $P(\mathbf{x}, y)$ from which we can measure the *empirical error*

$$I_{emp}[f] \equiv \frac{1}{\ell}\sum_{i=1}^{\ell} V(y_i, f(\mathbf{x}_i)). \tag{2}$$

The problem of approximating a function from sparse data is ill-posed and a classical way to solve it is regularization theory [12]. Regularization theory originates from Tikhonov's classical approach for solving ill-posed problems. Existence, uniqueness and especially stability[2] can be restored via a regularizing operator. The basic idea at the heart of the method — as in any approach to ill-posed problems — is to restrict appropriately the space of solutions f to an appropriately small hypothesis space[3]. Within the universe of ill-posed problems, the problem of learning theory has a specific need — the derivation of generalization bounds.

1 Definitions

This section and the next one (stolen from [10]) provide key definitions and theorems. Given an input space $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ and an output space $y \in \mathcal{Y} \subseteq \mathbb{R}$, a training set

$$S = \{z_1 = (\mathbf{x}_1, y_1), ..., z_\ell = (\mathbf{x}_\ell, y_\ell)\},$$

of size ℓ in $\mathcal{Z} \in \mathcal{X} \times \mathcal{Y}$ is drawn i.i.d. from an unknown distribution D. We will refer to a set

$$S^{i,u} = \{z_1, ..., u, ..., z_\ell\},$$

where the point z_i in set S is replaced with an arbitrary new point u.
Given the training set S we estimate a function $f_S : \mathcal{X} \to \mathcal{Y}$. The error of this function with respect to an example $z = (\mathbf{x}, y)$ is defined as

$$V(f_S, z) = V(f_S(\mathbf{x}), y).$$

[1] There is a large literature on the subject: useful reviews for this paper are [3, 9, 4, 13] and references therein.
[2] Stability is defined as continuous dependence of the solution f on the data $(\mathbf{x}_i, y_i)$, e.g. the approximating function must vary little with small perturbations of training data.
[3] The Ivanov method restricts the solution f to compact sets defined by $\|f\|_K^2 \leq A$ for any positive, finite A.

Thus the empirical error of the function is

$$I_{emp}[f_S, S] = \frac{1}{\ell} \sum_{i=1}^{\ell} V(f_S, z_i)$$

where f_S is the function the algorithm selects given a set S and S is the set of points the loss function is evaluated at. The expected or generalization error is

$$I_{exp}[f_S] = \int_{\mathcal{Z}} V(f_S, z) P(z)\, dz = \mathbb{E}_z[V(f_S, z)],$$

where $\mathbb{E}_z[\cdot]$ is the expectation for z sampled from the distribution D.
We state that a loss function V is σ-admissible if

$$\forall S^1, S^2 \in \mathcal{Z}^\ell, (\mathbf{x}, y) \in \mathcal{Z} \quad |V(f_{S^1}(\mathbf{x}), y) - V(f_{S^2}(\mathbf{x}), y)| \leq \sigma |f_1(\mathbf{x}) - f_2(\mathbf{x})|.$$

This condition was introduced in [1].

2 Stability: old and new definitions

A learning algorithm is a mapping from a training set S to a function f_S.

Definition 2.1 (Bousquet and Elisseeff, 2001)[1] *An algorithm has stability β with respect to the loss function V if*

$$\forall S, S^{i,u} \in \mathcal{Z}^\ell, \forall z \in \mathcal{Z}, \quad |V(f_S, z) - V(f_{S^{i,u}}, z)| \leq \beta.$$

Note that β will in general depend on ℓ, so we could more precisely define stability as a function from the integers to the reals, but the usage will be clear from context. This definition of stability is known as *uniform* stability. It is a restrictive condition, as it needs to hold on all possible training sets, even training sets that can only occur with probability 0. This motivates the weaker notion of (β, δ)-stability.

Definition 2.2 (Kearns and Ron, 1999) [11] *An algorithm is β-stable at S with respect to a loss function V if*

$$\forall z \in \mathcal{Z}, \quad |V(f_S, z) - V(f_{S^{i,u}}, z)| \leq \beta.$$

Definition 2.3 (Kutin and Niyogi, 2001) [5] *An algorithm A is (β, δ)-stable with respect to a loss function V if*

$$\mathbb{P}_{S \sim \mathcal{Z}^\ell}(A \text{ is } \beta\text{-stable at } S) \geq 1 - \delta.$$

It is obvious that a β-stable algorithm is also (β, δ)-stable for all $\delta \geq 0$. The following theorems provide generalization bounds for β-stable and (β, δ)-stable algorithms.

Theorem 2.1 (Bousquet and Elisseeff, 2001) [1] *Let A be a β-stable learning algorithm satisfying $0 \leq V(f_S, z) \leq M$ for all training sets S and for all $z \in \mathcal{Z}$. For all $\varepsilon > 0$ and all $\ell \geq 1$,*

$$\mathbb{P}_S \left\{ |I_{emp}[f_S, S] - I_{exp}[f_S]| > \varepsilon + 2\beta_\ell M \right\} \leq \exp\left(-\frac{2\ell\varepsilon^2}{(4\ell\beta_\ell + M)^2} \right).$$

Theorem 2.2 (Kutin and Niyogi, 2001) [5] *Let A be a (β, δ)-stable learning algorithm satisfying $0 \leq V(f_S, z) \leq M$ for all training set S and for all $z \in \mathcal{Z}$. For all $\varepsilon, \delta > 0$ and all $\ell \geq 1$,*

$$\mathbb{P}_S \left\{ |I_{emp}[f_S, S] - I_{exp}[f_S]| > \varepsilon + \beta_\ell + M\delta \right\} \leq 2 \exp\left(-\frac{\ell\varepsilon^2}{8(2\ell\beta_\ell + M)^2} \right) + \frac{4\ell^2 M\delta}{2\ell\beta_\ell + M}.$$

In general we are mainly interested in the case where $\beta_\ell = O\left(\frac{1}{\ell}\right)$ and $\delta = O(e^{-\ell})$. Throughout the rest of the paper, when we state that an algorithm is *strongly* β or (β, δ)-stable we mean that $\beta_\ell = O\left(\frac{1}{\ell}\right)$ and $\delta = O(e^{-\ell})$.
Using this convention, we note that strongly β-stable and strongly (β, δ)-stable algorithms have asymptotically identical generalization bounds, with differing constants. In both cases, we have (via a simple restatement of the theorems) that for any $\tau \in (0, 1)$, with probability $1 - \tau$,

$$|I_{emp}[f_S, S] - I_{exp}[f_S]| \leq O\left(\frac{1}{\sqrt{\ell}}\right),$$

which we also refer to as *fast convergence*.
It is interesting that several key learning algorithms are strongly β-stable[4]. In a similar spirit, we introduce what is an even more restrictive definition and remark that it applies to all cases considered by Bousquet and Elisseeff.

Definition 2.4 *An algorithm has α-stability if*

$$\forall S, S^{i,u} \in \mathcal{Z}^\ell, \forall z \in \mathcal{Z}, \quad |f_S(\mathbf{x}) - f_{S^{i,u}}(\mathbf{x})| \leq \alpha. \tag{3}$$

This definition – which corresponds to the *classification stability* introduced by [1] just for classification – describes stability of the actual functions. It is closer to the classical definition of stability — as continuous dependence on the initial data. It is clear that *(strong) α-stability implies (strong) β-stability for σ-admissible loss functions*. The converse is not true: in general, stability wrt the loss function does not imply stability of the functions, even for σ-admissible loss functions (see [10])[5]. However, published proofs of β-stability of various algorithms [1, 5], first prove that the functions are close in L_∞, then use the σ-admissibility condition on the loss function to show β-stability. For instance, the proof of Theorem 22 of Bousquet and Elisseeff leads directly to

Theorem 2.3 *Let $\mathcal{H}$ be a reproducing kernel Hilbert space on a compact domain X with kernel K s.t. for all x $K(x, x) \leq C_k \leq \infty$. Let the loss function V be σ-admissible. The learning algorithm defined by*

$$\min_{f \in \mathcal{H}} \frac{1}{\ell} \sum_{i=1}^{\ell} V(f(\mathbf{x}_i), y_i) + \lambda ||f||_K^2 \tag{4}$$

is α-stable with

$$\alpha \leq \frac{C_K \sigma}{2\lambda\ell}.$$

[4]In retrospect this is to be expected since regularization induces continuous dependence on the initial data which is a property very similar to β-stability.

[5]For example, consider the square loss, and the case where $f_1(x) = y + K$ and $f_2(x) = y - K$. The loss of the two functions is identical, but their L_∞ norms differ by $2K$ (Rifkin, Calder pers. comm.).

3 Stability of bagging

The intuition is that averaging regressors or classifiers trained on subsamples of a training set should increase stability with correspondingly better generalization bounds. Note that throughout this section we assume that the regressors that will be bagged are only α-stable, a very weak assumption. They are not assumed to be strongly α-stable. Notice that in the following we are not making any claims about the empirical error over the whole training set! Consider N regressors $f_i(x)$, each trained on (in general different) subsets of the training set. Each of the subsets has size p; the training set has overall size ℓ. We call f_i' the regressor corresponding to f_i but obtained when one of the data points in the whole training set is perturbed. The *average bagged regressor* is defined as $\frac{1}{N}\sum_{j=1}^{N} f_j$. It is straightforward to check that if each f_i has α-stability α_p then the bagged regressor has also α-stability $\leq \alpha_p$.
We now consider a *special* sampling scheme for bagging: each of the N regressors is trained on a disjoint subset of the training set. In this case $N = \ell/p$ with p fixed. Only one of the N regressors will be affected by a perturbation of one of the ℓ training points. Thus only one of the terms in $\frac{1}{N}|\sum_{j=1}^{N}(f_j - f_j)'|$ will be different from zero. In this special case the α-stability of the bagged regressor is $\frac{\alpha_p}{N}$. Formalizing this reasoning results in the following theorem.

Theorem 3.1 *Consider the bagged regressor $\frac{1}{N}\sum_{j=1}^{N} f_j$, in which each of the N regressors is α-stable for a training set of size p. There exist sampling schemes such that the bagged regressor is strongly α-stable with α-stability $(\frac{\alpha_p p}{\ell})$. Its β-stability with respect to the σ-admissible loss function V is then $(\frac{\alpha_p p \sigma}{\ell})$.*

A similar result extends to a simple boosting scheme in which the bagged classifier $(\frac{1}{N}\sum b_i f_i)$ is a weighted average of the individual regressors, with weights possibly found by optimization on the training set. We assume that there is an upper bound on the individual weight b_i for all i, i.e. $b_i \leq D$, as it is the case if the b_i are normalized (i.e. $\sum b_i = 1$). This means that the bound on the weight of each regressor in the resulting boosted function decreases with increasing N. Then the β-stability of the *weighted regressor* is $(\frac{\alpha_p p \sigma D}{\ell})$.
Now consider a bagging scheme where the subsets chosen for training are not necessarily disjoint. Consider two variants:

1. If we enforce the constraint that each point belongs to exactly k subsets, then k functions will be affected by the perturbation of one point. We can train $\frac{k\ell}{p} = kN$ regressors, and, therefore, have the same bound on α-stability for the bagged regressor $\frac{\alpha_p k}{Nk} = \frac{\alpha_p p}{\ell}$. Note that the bound on α-stability does not change with k for this scheme. It would be interesting to see empirically how the training error depends on k.

2. If we do not impose the above restriction on the number of subsets a given point can belong to, we might ask a question: *If we pick $N = \frac{\ell}{p}$ subsets at random, how many functions will be affected by perturbation of one point?* We can do a probabilistic analysis of this scheme and use the property of (β, δ)-stability defined previously. Note that the probability of each point being selected for a subset is $\frac{p}{\ell}$, and there are $\frac{\ell}{p}$ subsets, so the expected number of subsets a given point belongs to is 1. Nonetheless, we were unable to derive tight exponential (in ℓ) bounds that would allow us to use (β, δ)-stability results from [6].

4 Remarks

1. If the individual regressors are strongly stable, then bagging does not improve the rate of convergence, which can be achieved by using only one regressor or by bagging a fixed number of regressors trained on nonoverlapping training sets of size increasing with ℓ.

2. In the case of regularization with quadratic loss the stability of the solution depends on the condition number $(||K + \lambda\ell I||)(||(K + \lambda\ell I)^{-1}||) \leq \frac{C_K}{\lambda}$. Thus it is finite for finite λ but can be very large for $\lambda = 0$ and cannot be bounded a priori. In this case, it seems difficult to show in general that bagging helps. However, in the one-dimensional radial basis functions case with $\lambda = 0$ we can use results (for instance by Buhmann et al [2, 7]) to show that bagging can give an improvement in the condition number of order $O(N^3)$, where N is the number of bagged RBF, each trained on an optimal subset of the data (intercalated so that the distance between training points for each regressor is maximal).

5 Discussion

The observation captured by theorem 3.1 implies that there exist bagging and training schemes providing strong stability to ensembles of non-strongly stable algorithms. Thus bagging has a regularization effect and provides rates of convergence for the generalization error that are of the same order as Tikhonov regularization. It would be interesting to extend the previous analysis to various "boosting" schemes.
Another, probably more interesting issue in many practical situations, is whether and how bagging can improve stability for a given, fixed size ℓ of the training set. At the same time, the empirical error should also be minimized. Intuitively, the empirical error can be reduced by increasing the size of the subsamples used to train the individual classifiers; this however tends to worsen stability.

Acknowledgments: We wish to thank Matt Calder for key suggestions.

References

[1] O. Bousquet and A. Elisseeff. Stability and generalization. *Journal Machine Learning Research*, 2001. submitted.

[2] M. D. Buhmann. A new class of radial basis functions with compact support. *Mathematics of Computation*, 70(233):307–318, 2000.

[3] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13:1–50, 2000.

[4] F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7:219–269, 1995.

[5] S. Kutin and P. Niyogi. The interaction of stability and weakness in adaboost. Technical report TR-2001-30, University of Chicago, 2001.

[6] S. Kutin and P. Niyogi. Almost-everywhere algorithmic stability and generalization error. Technical report TR-2002-03, University of Chicago, 2002.

[7] F. Derrien M. D. Buhmann and A. Le Méhauté. Spectral properties and knot removal for interpolation by pure radial sums. In L. L. Schumaker M. Dahlen, T. Lyche, editor, *Mathematical Methods for Curves and Surfaces*, pages 55–62. Vanderbilt University Press, 1995.

[8] T. Poggio and F. Girosi. A theory of networks for approximation and learning. C.B.I.P. Memo No. 31, Center for Biological Information Processing, Whitaker College, 1989.

[9] T. Poggio and F. Girosi. Networks for Approximation and Learning. In C. Lau, editor, *Foundations of Neural Networks*, pages 91–106. IEEE Press, Piscataway, NJ, 1992.

[10] R. Rifkin, S. Mukherjee, and T. Poggio. Stability, generalization, and uniform convergence. Ai memo, Massachusetts Institute of Technology, 2002. in press.

[11] D. Ron and M. Kearns. Algorithmic stability and sanity-check bounds for leave-one-out crossvaildation. *Neural Computation*, 11(6):1427–1453, 1999.

[12] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-posed Problems*. W. H. Winston, Washington, D.C., 1977.

[13] V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.