ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 475-480

Fast approximation of the bootstrap for model selection

G. Simon[1], A. Lendasse[2], V. Wertz[2], M. Verleysen[1, ‡]

Université Catholique de Louvain
[1] DICE - Place du Levant 3, B-1348 Louvain-la-Neuve, Belgium,
Phone : +32-10-47-25-40, Fax : +32-10-47-21-80
Email : {gsimon, verleysen}@dice.ucl.ac.be

[2] CESAME - Avenue G. Lemaître 4, B-1348 Louvain-la-Neuve, Belgium,
Email : {lendasse, wertz}@auto.ucl.ac.be

Abstract. The bootstrap resampling method may be efficiently used to estimate the generalization error of a family of nonlinear regression models, as artificial neural networks. The main difficulty associated with the bootstrap in real-world applications is the high computation load. In this paper we propose a simple procedure based on empirical evidence, to considerably reduce the computation time needed to estimate the generalization error of a family of models of increasing number of parameters.

1. Introduction

Model design has raised a considerable research effort since decades, on linear models, nonlinear ones, artificial neural networks, and many others. Model design includes the necessity to *compare* models (for example of different complexities) in order to select the "best" model among several ones. For this purpose, it is necessary to obtain a good approximation of the generalization error of each model (the generalization error being the average error that the model would make on an infinite-size and unknown test set independent from the learning one).
Nowadays there exist some well-known and widely used methods able to fulfil this task: among others the AIC or BIC criteria and the like [1], [2], [3] as well as cross-validation, leave-one-out [3, 6] and bootstrap [4]. All these methods have been proved to be roughly asymptotically equivalent (see for example [5] and [6]). A natural extension of the bootstrap, the .632 bootstrap [6], has also been proved to be unbiased.
Nevertheless, and while this is not an irrefutable question, it seems that the bootstrap is advantageous in many "real" modelling cases (i.e. when the number of samples is limited, the dimension of the space high, etc.) [6]. But the main problem when using the bootstrap is the computation of the results that could really be time consuming. Another limitation is that the conventional use of the bootstrap for model selection leads to a final model chosen from the restricted set of *a priori* selected models.

[‡] G. Simon is funded by the Belgian F.R.I.A. M. Verleysen is Senior Research Associate of the Belgian F.N.R.S. The work of A. Lendasse and V. Wertz are supported by the Interuniversity Attraction Poles (IAP), initiated by the Belgian Federal State, Ministry of Sciences, Technologies and Culture. The scientific responsibility rests with the authors.

In this paper we will show that, under reasonable and simple hypotheses usually fulfilled in real world applications, it is possible to provide a good estimate of the bootstrap results with a considerably reduced number of modelling stages, thus saving a considerable amount of computation time. Moreover the model selected with this bootstrap approach may be different from the ones used to compute the approximation

2. Bootstrap technique

The bootstrap [4] is based on the plug-in principle that permits to obtain an estimator of a statistic according to an empirical distribution. In our context we use the bootstrap to estimate the generalization error of several models in order to choose the "best" one.
The bootstrap estimator is computed over a finite number N of new samples $\boldsymbol{x}^*$ generated from the original sample $\boldsymbol{x}$ by drawing with replacement. The bootstrap estimate of the generalization error is given by

$$\hat{e}_{gen} = e_{app} + optimism\,, \qquad (1)$$

where e_{app} is the apparent error obtained when evaluating the model built (learned) on the original sample $\boldsymbol{x}$ on the same sample (learning error), and *optimism* is a correction term aiming to estimate the difference between a learning and a generalization error. The *optimism* is computed on the N bootstrap replications:

$$optimism = E\,[\, e_{\boldsymbol{x}^*}(\hat{F}_{\boldsymbol{x}}) - e_{\boldsymbol{x}^*}(\hat{F}_{\boldsymbol{x}^*})]\,, \qquad (2)$$

where $E[\ ]$ is the statistical expectation computed over all bootstrap replications and $e_{\boldsymbol{x}^*}(\hat{F}_{\boldsymbol{x}})$ is the error for a model developed (learned) on the $\boldsymbol{x}^*$ sample and evaluated on the $\hat{F}_{\boldsymbol{x}}$ empirical distribution. $\hat{F}_{\boldsymbol{x}}$ and $\hat{F}_{\boldsymbol{x}^*}$ are the empirical distribution functions in the real world and in the bootstrap world respectively.
Note that the .632 bootstrap [4] aims to reduce the slight bias introduced by the *optimism* correction of the basic bootstrap methodology. The acceleration method presented in this paper can be extended straightforwardly to the .632 bootstrap.

3. Methodology

3.1. Empirical argument

In numerous applications of the bootstrap for nonlinear model selection we have noticed two persistent facts.
First, it is well known that the apparent error e_{app} of a nonlinear regression model (like Multi-Layer Perceptrons (MLPs), Radial-Basis Function Networks (RBFNs), etc.) is usually roughly exponentially or quadratically decreasing with the number p of parameters in the model. Of course only parameters of the same nature have to be considered (weights in MLPs, centers in RBFNs, etc.). This comment must be kept in mind in the following.
With a good approximation, e_{app} can thus be expressed as one of the following expressions:

$$e_{app} \approx Ae^{-Bp} \qquad or \qquad e_{app} \approx \frac{1}{Ax^2 + Bx + C}\,. \qquad (3)$$

This empirical fact is usually confirmed on a reasonable range of possible values for p. When p is either nearby zero or very large, (3) is no longer valid, but this is not a problem for the following as we will use approximation (3) only in its validity range: p too small leads to a poor model with large apparent and generalization errors, while p too large leads to overfitting.
A second empirical fact is that the *optimism* increases roughly linearly with the number p of parameters, leading to:

$$optimism = Dp + E \ . \tag{4}$$

Here again a limited range of the p value must be considered.
Empirical evidence of approximations (3) and (4) will be illustrated in section 4.

3.2. Theoretical argument

Assuming a linear relation (4) for the *optimism* is certainly the most unexpected hypothesis of the method presented in this paper, although it is confirmed by experience. To strengthen this hypothesis, we can mention that a general formulation of structure selection criteria can also be written as

$$\hat{e}_{prediction} = e_{app} + correction\ term \tag{5}$$

where *correction term* is proportional to p in most cases ($2p\sigma/n$ for AIC and $\ln(n)p\sigma/n$ for BIC, where σ is the estimated quadratic error on the learning set). In all these situations (AIC, BIC and other experiments), we see that the *optimism* or correction to add to the apparent error is proportional to p; leaving the constants D and E resulting from the experimental procedure below is a way to avoid making hypotheses (usually based on asymptotical results) to fix these constants, and to allow adapting them to each specific problem or application.

3.3. Estimating the bootstrap results

Under the empirical argument developed in section 3.1 and according to the theoretical argument in section 3.2, rewriting the bootstrap estimate of the generalization error would then give, for some parameters A, B, C, D and E:

$$\begin{aligned} \hat{e}_{generalization} &= e_{app} + optimism \\ &= Ae^{-Bp} + Dp + E \quad . \\ \text{or} \quad &= \frac{1}{Ax^2 + Bx + C} + Dp + E \end{aligned} \tag{6}$$

The principle of the method is then to make a limited number of experiments to estimate A, B, C, D and E. A, B and C are evaluated by (3) with models (with different values of p) using the original sample $\boldsymbol{x}$ both for learning and test. The D and E values can be computed according to (4) with models built on bootstrap replicates $\boldsymbol{x}^*$ and evaluated on both the original sample $\boldsymbol{x}$ and the bootstrap samples $\boldsymbol{x}^*$.
In both cases experiments on three (two) different values of p are theoretically sufficient to fix parameters A, B and C, and D and E respectively. Nevertheless it is suggested to increase the number of experiments in order to decrease the influence of a single experiment. When A, B, C, D and E have been computed, the minimum of (6) gives the value of p that minimizes the generalization error.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 475-480

4. Experimental results

We illustrate the method described in the previous section on a standard benchmark in time-series prediction. The Santa Fe A time series [7] has been chosen mainly for the large number of data available for the training stage (1000) as well as for the test stage (9000). These two numbers correspond to the rules of the Santa Fe competition, as detailed in [7].
The model we used is chosen *a priori* to be:

$$\hat{y}(t+1) = f\big(y(t), y(t-1), y(t-2), y(t-3), y(t-5), y(t-6)\big). \qquad (7)$$

This regressor has been shown to be adequate for this series [7]. Note that we aim here to present an experimental validation of our method (approximation of the bootstrap results) and not to make a comparison of the most adequate model and regressor. We take a Radial-Basis Function Network (RBFN) for *f*(.) and the regressor given by (7) as *a priori* choices.
A RBFN is characterised by its number *n* of Gaussian kernels (or hidden units). A detailed description of the learning strategy we used to train the RBFN can be found in [8]. As the number *p* of parameters in a RBFN is proportional to its number *n* of units, the following results will be illustrated according to the value of *n* instead of *p*, without changing anything in the arguments detailed in the previous section. With respect to the comment in Section 3.1, the *n* Gaussian kernels are effectively parameters of the same nature.
We trained 7 RBFNs on the Santa Fe learning dataset, for *n* = 20, 40, 60, 80, 100, 120 and 140 respectively. With the apparent generalization error obtained for those values of *n* we can compute *A, B* and *C* (in the least mean square sense) and deduce an estimate of the apparent error. We have decided here to use the decreasing hyperbolic function in (3). Then bootstrap estimates of the *optimism* are evaluated for the same values of *n*. A linear interpolation gives *D* and *E*. The results of these two steps are shown in figure 2.a) and 2.b). Figure 3 shows the final estimate of the generalization error. The minimum of this function is attained for *n* = 103. Then we repeat the same experiment with four different *n* instead of seven, with values of *n* equals 20, 60, 100 and 140 respectively. The optimal *n* found in this case is a coherent result of 102. Finally, we made another experiment for RBFN networks with varying number *n* of units trained on the 1000 training data of the Santa Fe A series, and tested on the 9000 test data. Assuming that the time series is sufficiently stationary, using these 9000 test data gives a good estimate (in this specific application) of the effective generalization error. This last result is illustrated in Figure 4 where the minimum is *n* = 100, thus confirming our experimental estimation.
A second example is provided to illustrate the robustness of the proposed method. We have used this time the abalone dataset [9], with 1000 data for the training stage and the remaining 3177 for the test stage. We have applied the same methodology with different RBFN models with *n* equals 1, 17, 33 and 49. We choose the hyperbolic decreasing function in (3). Figure 5 shows a) the apparent error and b) the *optimism*. Figure 6 is the final bootstrap estimate. The selected model with the computed minimum prediction error has 32 Gaussians kernels, close to the best 27 kernels obtained from the 3177 data.
In comparison with the classical bootstrap methodology, the described method has two advantages. First, it is not necessary to test all potential models to find the “best” one,

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 475-480

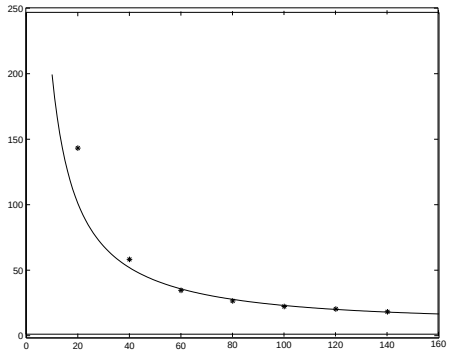

Figure 2. a) Apparent error for RBFN models with 20, 40, 60, 80, 100, 120 and 140 Gaussian kernels; A = -1 10^{-6}, B = 5.04 10^{-4} and C = 0

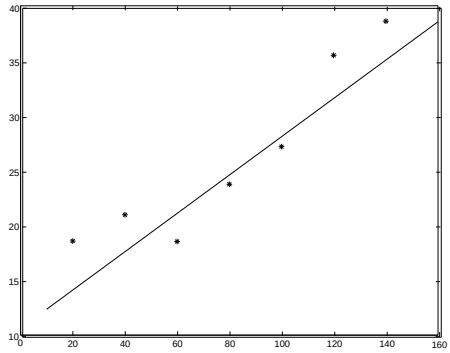

Figure 2. b) Optimism for RBFN models with 20, 40, 60, 80, 100, 120 and 140 Gaussian kernels; D = 0.17 and E = 12.18.

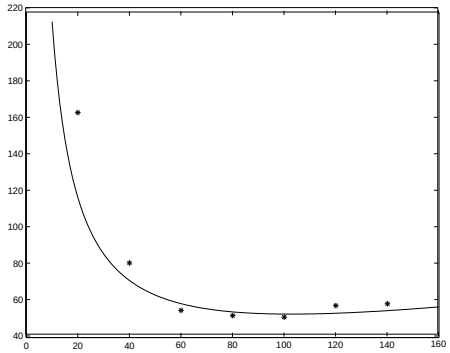

Figure 3. Interpolated graph for the bootstrap estimate of the generalization error.

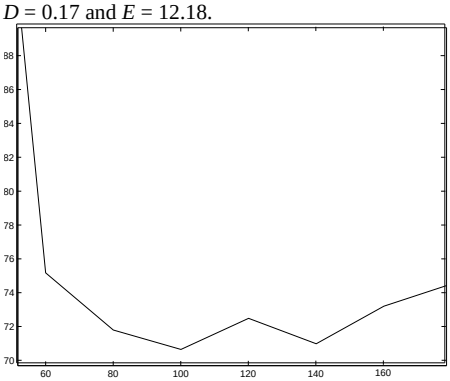

Figure 4. Estimate of the effective generalization error for the Santa Fe A time series.

thus reducing the computation time of an important factor since models have a computation cost proportional to n^2. Secondly, as the interpolation averages the possible variations (due to poor estimate) in the generalization error of each model, one can afford a much lower number of bootstrap replications to estimate the generalization error of each tested model. Reducing the number of bootstrap replication is a real breakthrough with respect to the elapsed time of a classical bootstrap approach.

5. Conclusion

In this paper we have proposed an effective procedure to reduce the computation time of a bootstrap approximation of the generalization error in a family of nonlinear regression models. The limited loss of accuracy is balanced by a considerable saving in computation load, the main shortcoming of using the bootstrap methodology for model selection. This saving is principally due to the number of replication that is here much lower than in a normal use of the bootstrap.

Although this procedure has only been tested in a neural network model selection context, this simple and time saving method could easily be extended to other contexts of nonlinear regression, classification, etc., where computation time and complexity play a role. It can also be applied to other resampling procedures, as the .632 bootstrap.

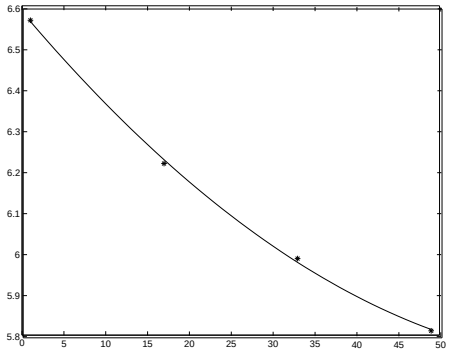

Figure 5. a) Apparent error for abalone case, n = 1, 17, 33 and 49; A =-3.43 10^{-6}, B = 5.81 10^{-4} and C = 1.51 10^{-1}.

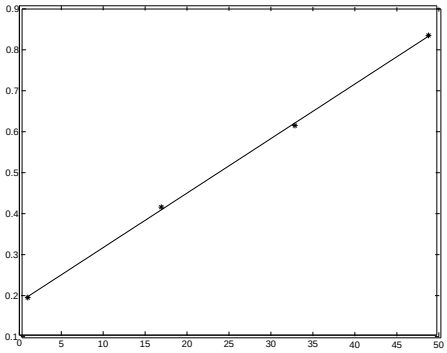

Figure 5. b) Optimism for the abalone case, with n = 1, 17, 33 and 49; D = 1.32 10^{-2} and E = 1.79 10^{-1}.

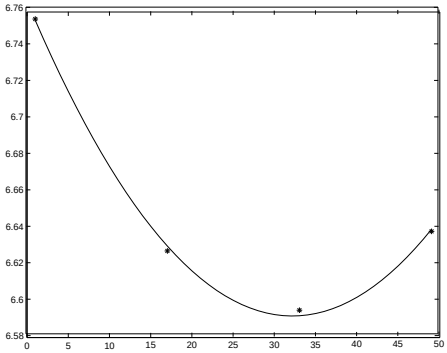

Figure 6. Bootstrap estimate of the generalization error for the abalone example.

References

[1] H. Akaike, "*Information theory and an extension of the maximum likelihood principle*", 2nd Int. Symp. on information Theory, 267-81, Budapest, 1973

[2] G. Schwarz, "*Estimating the dimension of a model*", Ann. Stat. 6, 461-464, 1978.

[3] L. Ljung, "*System Identification - Theory for the user*", 2nd ed, Prentice Hall, 1999.

[4] B. Efron, R. J. Tibshirani, "An introduction to the bootstrap", Chapman & Hall, 1993.

[5] M. Stone, *"An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion*", J. Royal. Statist. Soc., B39, 44-7, 1977.

[6] R. Kohavi, *"A study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection*", Proc. of the 14th Int. Joint Conf. on A.I., Vol. 2, Canada, 1995.

[7] A. S. Weigend and N.A. Gershenfeld, *"Times Series Prediction: Forcasting the future and Understanding the Past"*, Addison-Wesley Publishing Company, 1994.

[8] N. Benoudjit, C. Archambeau, A. Lendasse, J. Lee, M. Verleysen, *"Width optimization of the Gaussian kernels in Radial Basis Function Networks"*, Proc. of ESANN'2002, d-site, Brussels, 2002.

[9] W.J. Nash, T.L. Sellers, S.R. Talbot, A.J. Cawthorn and W.B. Ford, "*The Population Biology of Abalone (_Haliotis_ species) in Tasmania. I. Blacklip Abalone (_H. rubra_) from the North Coast and Islands of Bass Strait*", Sea Fisheries Division, Technical Report No. 48, 1994.