Learning GP-trees from Noisy Data

José L. Montaña[1⋆], César L. Alonso[2], and Cruz E. Borges[1⋆⋆]

[1] Departamento de Matemáticas, Estadística y Computación,
Universidad de Cantabria, 39005 Santander, Spain
{montanjl, borgesce}@unican.es
[2] Centro de Inteligencia Artificial, Universidad de Oviedo
Campus de Viesques, 33271 Gijón, Spain
calonso@aic.uniovi.es

Abstract. We discuss the problem of model selection in Genetic Programming using the framework provided by Statistical Learning Theory, i.e. Vapnik-Chervonenkis theory (VC). We present empirical comparisons between classical statistical methods (AIC, BIC) for model selection and the Structural Risk Minimization method (based on VC-theory) for symbolic regression problems. Empirical comparisons of different methods for model selection suggest practical advantages of using VC-based model selection when using genetic training.

keywords: Model selection, genetic programming, symbolic regression

1 Introduction

In the last years GP has been applied to a range of complex learning problems, including that of symbolic regression in a variety of fields like quantum computing, electronic design, sorting, searching, game playing, etc. For dealing with these problems GP evolves a population composed by symbolic expressions built from a set of functionals $F = \{f_1, \ldots, f_k\}$ and a set of terminals $T = \{x_1, \ldots, c_1, \ldots\}$ (including the variables and the constants). Once the functionals and the terminals have been selected, the regression task can be thought as a supervised learning problem where the hypothesis class $\mathcal{H}$ is the tree structured search space described from the set of leaves T and the set of nodes F. Analogously, the GP algorithm evolving symbolic expressions representing the concepts of class $\mathcal{H}$ can be regarded as a supervised learning algorithm that selects the best model inside the class $\mathcal{H}$.

Regarding this consideration of GP as a supervised learning task we propose the use of tools of Statistical Learning Theory (see [10]) with the purpose of model selection in Genetic Programming. This point of view is not new and has been suggested in [6]. This paper presents empirical comparisons between classical statistical methods, Akaike Information Criterium (AIC), Bayesian Information Criterium (BIC) and the Structural Risk Minimization (SRM) method

⋆ The First two authors are supported by spanish grant TIN2007-67466-C02-02
⋆⋆ Supported by MTM2007-62799 and FPU program

for symbolic regression problems. We consider symbolic regression formulation under general setting for predictive learning (see [8], [7], [10], [3]). The goal is to estimate unknown real-valued function in the relationship:

$$y = g(x) + \epsilon; \tag{1}$$

where ϵ is i.i.d. zero mean random error (noise), x is a multidimensional input and y is a scalar output. The estimation is made based on a finite number (n) of samples (training data): The training data $(x_i, y_i)_{1 \leq i \leq n}$ are independent and identically distributed (i.i.d.) generated according to some (unknown) joint probability density function,

$$\rho(x, y) = \rho_(x)\rho(y|x) \tag{2}$$

The best estimation of the unknown function in Equation 1 is the mean of the output conditional probability.

$$g(x) = \int y\rho(y|x) \tag{3}$$

A learning method selects the best model $f \in \mathcal{H}$, where $\mathcal{H}$ is some class of concepts. In general, the error of the estimator f, $\varepsilon(f)$, is written as

$$\varepsilon(f) = \int Q(x, f, y)d\rho, \tag{4}$$

where Q measures some notion of loss between $f(x)$ and the target concept y, and ρ is the distribution from which examples (x, y) are drawn to the learner. For regression tasks one usually takes $Q(x, f, y) = (y - f(x))^2$.

For a class of models $\mathcal{H}$ with finite complexity (for instance –in the case of GP– trees with bounded size or height) the model can be chosen minimizing the empirical error:

$$\varepsilon_n(f) = \frac{1}{n} \sum_{i=1}^{n} Q(x_i, f, y_i) \tag{5}$$

The problem of model selection –also called complexity control– arises when a class of models consists of models of varying complexity (for instance –in the case of Genetic Programming– trees with varying size or height). Then the problem of regression estimation requires optimal selection of model complexity (i.e., the size or the height) in addition to model estimation via minimization of empirical risk as defined in Equation (5).

Since most model selection criteria, in particular, analytic model selection and Structural Risk Minimization (SRM) are based on certain assumptions, mainly linearity and exact computation of the classification capacity of the class of concepts $\mathcal{H}$, it is important to perform empirical comparisons in order to understand their practical usefulness in settings when these assumptions may not hold, which is the case of Genetic Programming algorithms.

The aim of this paper is to study the practical usefulness of analytic model selection and SRM model selection when they are used in the framework of Genetic Programming. The paper is organized as follows. Section 2 describes classical model selection criteria (AIC and BIC) and SRM approach used for empirical comparisons. Section 3 describes the experimental setting and results. Section 4 contains some conclusive remarks.

2 Analytical Model Selection Criteria

In general, analytical estimates of error (Equation 4) as a function of empirical error (Equation 5) take one of the following forms:

$$\varepsilon(f) = \varepsilon_n(f).pen(h,n) \tag{6}$$

$$\varepsilon(f) = \varepsilon_n(f) + pen(h/n, \sigma^2), \tag{7}$$

where f is the model, pen is called the penalization factor, h is the model complexity, n is the size of the training set and σ is the standard deviation of the additive noise (Equation 1). In this paper we shall make use of three analytical estimates of error.

- Akaike Information Criterium (AIC) which is as follows (see [1]):

$$\varepsilon(f) = \varepsilon_n(f) + \frac{2h}{n}\sigma^2 \tag{8}$$

- Bayesian Information Criterium (BIC) (see [2]):

$$\varepsilon(f) = \varepsilon_n(f) + \ (ln\ n)\frac{h}{n}\sigma^2 \tag{9}$$

- Structural Risk Minimization (SRM) (see [10])

$$\varepsilon(f) = \varepsilon_n(f).\left(1 - \sqrt{p - p\ ln\ p + \ \frac{ln\ n}{2n}}\right)^{-1}, \tag{10}$$

  where $p = \frac{h}{n}$.

2.1 Estimation of σ

When using a linear estimator with parameters, the noise variance can be estimated from the training data (x_i, y_i) as:

$$\sigma^2 = \frac{n}{n-h}\frac{1}{n}\sum_{1\leq i\leq n}(y_i - \hat{y}_i), \tag{11}$$

$\hat{y}_i$ is the estimation of value y_i by model f, i.e. $\hat{y}_i = f(x_i)$. Then one can use Equation 11 in conjunction with AIC or BIC for each (fixed) model complexity.

2.2 Model Complexity of GP-trees

The above described model selection criteria are used in the framework of linear estimators and the model complexity h, in this case, is the number of free parameters of the model (for instance, in the familiar case the models are polynomials, h is the degree of the polynomial).

For GP trees we use in this paper the number of non-scalar nodes of the tree, that is, nodes which are not labeled with $\{+, -\}$ operators. This is a measure of the non-linearity of the considered model and can be seen as a generalization of the notion of degree to the case of GP-trees. This notion is related with the Vapnik-Chervonenkis (VC) dimension of the set of models given by GP-trees using a bounded number of non-scalar operators. The exact relationship between non-scalar size of a GP-tree (more generally, a computer program) and its VC dimension is showed in [4].

3 Experimentation

3.1 Experimental Settings

We consider instances of symbolic regression problem for our experimentation. We have executed the algorithms over two groups of target functions. The first group includes the following three functions that also were used in [3] for experimentation:
Discontinuous piecewise polynomial function:

$$g_1(x) = \begin{cases} 4(x^2(3-4x) & x \in [0, 0.5] \\ (4/3)x(4x^2 - 10x + 7) - 3/2 & x \in (0.5, 0.75] \\ (16/3)x(x-1)^2 & x \in (0.75, 1] \end{cases} \tag{12}$$

Sine-square function:

$$g_2(x) = sin^2(2\pi x),\ x \in [0, 1] \tag{13}$$

Two-dimensional *sin* function:

$$g_3(x) = \frac{sin\sqrt{x_1^2 + x_2^2}}{x_1^2 + x_2^2},\ x_1,\ x_2 \in [-5, 5] \tag{14}$$

The second group of functions is constituted by five functions of several classes: trigonometric functions, polynomial functions and one exponential function. These functions are the following:

$$\begin{array}{ll} f_1(x) = x^4 + x^3 + x^2 + x & x \in [-5, 5] \\ f_2(x) = e^{-sin\ 3x+2x} & x \in [-\frac{\pi}{2}, \frac{\pi}{2}] \\ f_3(x) = e\ x^2 + \pi\ x & x \in [-\pi, \pi] \\ f_4(x) = cos(2x) & x \in [-\pi, \pi] \\ f_5(x) = min\{\frac{2}{x}, sin(x) + 1\} & x \in [0, 15] \end{array} \tag{15}$$

For the first group of target functions we use the following set of functionals $F = \{+, -, *, //\}$, incremented with the *sign* operator for the target function g_1. In the above set F, "//" indicates the protected division, i.e. $x//y$ returns x/y if $y \neq 0$ and 1 otherwise. The terminal set T consists of the variables of the corresponding function and includes the set of constants $\{0, 1\}$.

For the second group of functions, the basic set of operations F is also incremented with other operators. This aspect for each function is showed in the following table.

Table 1. Function set for the second group of target functions.

Function	Function set
f_1	$F \cup \{sqrt\}$
f_2	$F \cup \{sqrt, sin, cos, exp\}$
f_3	$F \cup \{sin, cos\}$
f_4	$F \cup \{sqrt, sin\}$
f_5	$F \cup \{sin, cos\}$

We use GP-trees with height bounded by 8. As it was mentioned above, the model complexity h is measured by the number of non-scalar nodes of the tree. The rest of the parameters for the genetic training process are the following: population size $M = 100$; maximum number of generations $G = 1000$; probability of crossover $p_c = 0,9$ and probability of mutation $p_m = 0,1$. Tournament selection and the standard operators of crossover and mutation for tree-like structures are used. For all the executions, the genetic training process finishes after 10^7 operations have been computed. Observe that the number of computed operations equals the internal nodes of the trees that are visited during the process. Training sets of $n = 30$ examples are generated where the x- values follow from uniform distribution in the input domain. For the computation of the y-values, the equation 1 is used in order to corrupt the values with noise. The noise variance σ was fixed to 0.2.

The experimentation scheme is as follows: For each model selection criterium (AIC, BIC, SRM) and each target function, we use a simple competitive coevolution strategy where 10 populations of 100 individuals evolve independently, considering same training set. Then, we select the model proposed by the best of these 10 executions. The above process completes one experiment. We have performed 100 experiments and for each one a different random realization of 30 training samples was considered. Hence for each target function, we have executed each algorithm 1000 times and finally 100 models have been selected. These models are the best ones of each group of 10 populations related to same training set.

3.2 Experimental Results

When the competitive genetic training processes finishes, the best individual is selected as the proposed model for the corresponding target function. In order to measure the quality of the selected model, it makes sense to consider a new set of points generated without noise from the target function. This new set of examples is known as the test set or validation set. So, let $(x_i, y_i)_{1 \leq i \leq n_{test}}$ a validation set for the target function $g(x)$ (i.e. $y_i = g(x_i)$) and let $f(x)$ be the model estimated from the training data. Then the prediction risk $\varepsilon_{n_{test}}$ is defined by the mean square error (MSE) between the values of f and the true values of the target function g over the validation set:

$$\varepsilon_{n_{test}} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (f(x_i) - y_i)^2 \tag{16}$$

Figures 1 and 2 show comparison results for AIC, BIC and SRM, after that the 100 experiments were completed. The empirical distribution of the prediction risk for each model selection is displayed using standard box plot notation with marks at 25%, 50% and 75% of that empirical distribution. The first and last mark in each case, stands for the prediction risk of the best and the worst selected model respectively. In all cases the size n_{test} of the validation set is 200. Note that the scaling is different for each function.

A first analysis of the figures concludes that the three model selection procedures perform quite similar. Nevertheless it seems that SRM produces better models for the target functions than AIC and BIC. Note that for the most part of the target functions, the SRM strategy obtains prediction risk values for the best solution and also for the solutions ranked at positions 25 and 50, that are lower than or equal to those obtained by AIC and BIC. Motivated by this fact we display in the following tables the best prediction risk value for each target function and model selection criterium (table 2) and the mean of the prediction risk values considering the 5 and the 25 best selected models (table 3).

Table 2. Prediction risk values of the best obtained model for each target function

Function	AIC	BIC	SRM
f_1	4.05E-28	4.22E-28	**3.92E-28**
f_2	0.00E+00	0.00E+00	0.00E+00
f_3	1.72E-03	1.85E-03	**0.10E-03**
f_4	9.61E-06	9.53E-06	**1.12E-32**
f_5	3.28E-02	4.04E-02	**1.83E-02**
g_1	2.39E-02	3.09E-02	**1.38E-02**
g_2	1.21E-01	1.23E-01	**1.18E-01**
g_3	1.10E-01	1.17E-01	**0.92E-01**

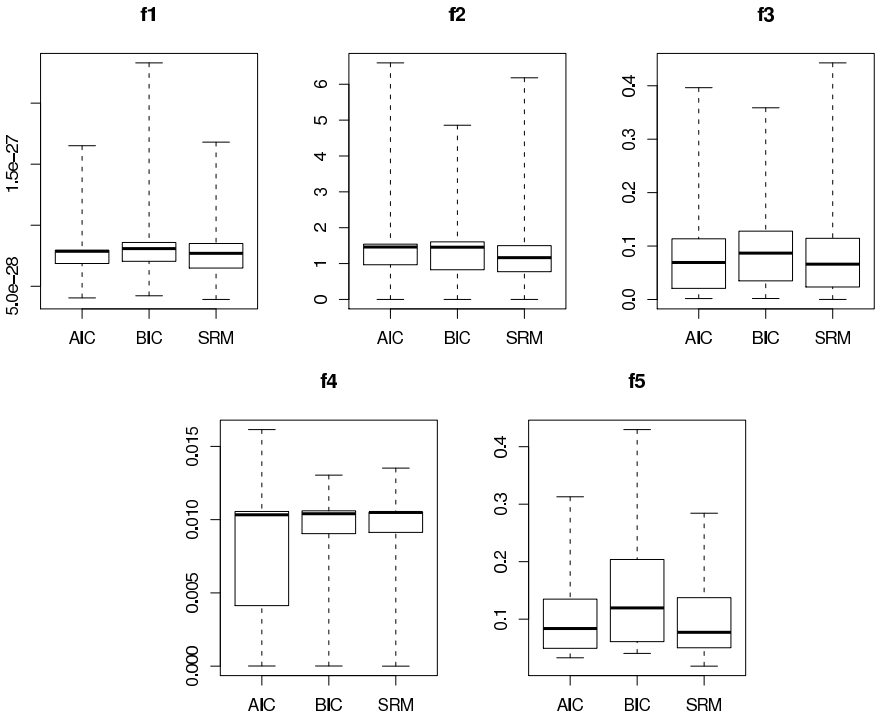

Fig. 1. Empirical distribution of the prediction risk $\varepsilon_{n_{test}}$ for target functions f_1 to f_5, over 100 experiments

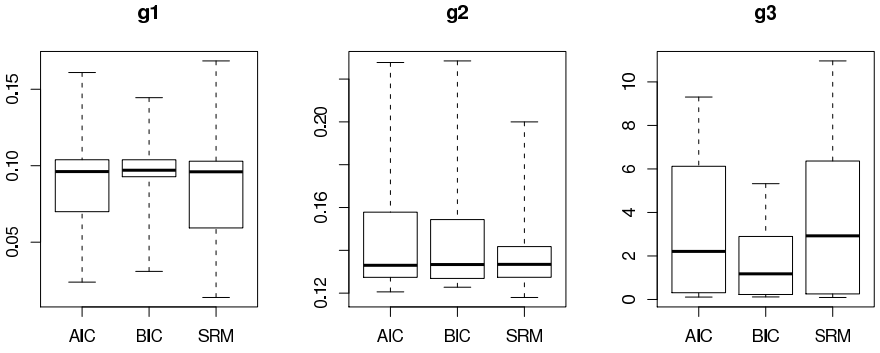

Fig. 2. Empirical distribution of the prediction risk for target functions g_1 to g_3, over 100 experiments

Table 3. Mean prediction risk of the best 5% and 25% of the models, for each target function

	5%			25%		
Function	AIC	BIC	SRM	AIC	BIC	SRM
f_1	4.92E-28	4.69E-28	**4.61E-28**	6.11E-28	5.87E-28	**5.74E-28**
f_2	8.90E-02	**8.25E-30**	3.92E-02	5.85E-01	**2.68E-01**	4.42E-01
f_3	3.26E-03	2.80E-03	**0.89E-03**	1.02E-02	1.42E-02	**0.94E-02**
f_4	9.68E-06	9.63E-06	**7.68E-06**	**0.27E-03**	1.63E-03	2.16E-03
f_5	3.84E-02	4.10E-02	**2.73E-02**	4.41E-02	4.95E-02	**3.95E-02**
g_1	2.80E-02	3.48E-02	**2.15E-02**	4.56E-02	6.32E-02	**4.01E-02**
g_2	1.23E-01	1.23E-01	**1.21E-01**	1.25E-01	1.25E-01	1.25E-01
g_3	1.30E-01	1.21E-01	**1.18E-01**	2.04E-01	**1.60E-01**	1.89E-01

As we can see in table 2, the quality of the best solutions are very similar for the three model selection criteria. Nevertheless, SRM produces the best obtained model for all target functions. In this sense, it can be also deduced from the results showed in table 3, that considering for each target function and each model selection strategy a reasonable set of the best solutions, the mean quality of the set related to the strategy SRM is better than those related to the other two studied strategies.

Finally, in table 4 we present the mean prediction risk, considering all the performed executions. Cause we have 10 executions for each experiment and 100 experiments were completed, each result presented in table 4 corresponds to a mean value over 1000 runs.

Table 4. Mean prediction risk of the 1000 executions for each pair strategy-target function

Function	AIC	BIC	SRM
f_1	**6.44E-28**	1.93E-03	7.32E-28
f_2	24.2E+00	24.3E+00	**3.23E+00**
f_3	5.58E-02	7.78E-02	**4.79E-02**
f_4	**4.15E-03**	8.48E-03	6.23E-03
f_5	9.03E-02	10.2E-02	**6.98E-02**
g_1	8.01E-02	9.44E-02	**7.42E-02**
g_2	**1.33E-01**	1.44E-01	1.47E-01
g_3	4.33E-01	11.8E-01	**3.53E-01**

The results in table 4 confirm that after the proposed genetic training process, the studied model selection criteria always obtain quite good hypotheses for the selected target functions. However, again SRM presents the best results

for the most part of the target functions. Taking into account the above experimentation we can conclude that Structural Risk Minimization method (based on VC-theory) as a model selection criterium when using genetic training from noisy data, slightly outperforms classical statistical methods as AIC or BIC. In general, SRM obtain better solutions than AIC or BIC in almost all studied cases.

4 Conclusive Remarks

In this paper we have presented an empiral comparative study of three model selection criteria for learning problems with GP-trees. The first two methods (AIC and BIC) are classical statistical methods and the third one (SRM) is a selection criterium based on VC-theory. The strategy used for the selection of the model was a genetic training method over a finite set of noisy examples. For measuring the quality of the selected model after the training process, a validation set of noise free examples was generated and a mean square error fitness of the model over the validation set is computed. An extensive experimentation over several symbolic regression problem instances, suggests that SRM selection criterium performs better than the other two considered methods. However AIC and BIC methods, that usually are employed in combination with least-squares fitting techniques, also perform quite well when using genetic training.

As final remark we note that it is impossible to draw any conclusions based on empirical comparisons unless one is sure that model selection criteria use accurate estimates of model complexity. There exist experimental methods for measuring the VC-dimension of an estimator ([9], [5]); however they are difficult to apply for general practitioners. In this paper we have used a new complexity measure for GP-trees. Essentially, under this approach we combine the known analytical form of a model selection criterium, with appropriately tuned measure of model complexity taken as a function of (some) complexity parameter (i.e. the value h that measures the non-linearity of the considered tree). This alternative practical approach is essentially to come up with empirical 'common-sense' estimates of model complexity to be used in model selection criteria.

References

1. Akaike, H.: Statistical prediction information. Ann. Inst. Statistic. Math 22 (1970) 203–217
2. Bernardo, J., Smith, A.F.M.: Bayesian theory. John Willey & Sons (1994)
3. Cherkassky, V., Yunkian, M.: Comparison of Model Selection for Regression. Neural Computation 15 (2003) 1691–1714
4. Montaña, J.L., Alonso, C.L., Borges, C.E., Crespo, J.L.: Adaptation, performance and vapnik-chervonenkis dimension of straight line programs. Proc. 12th European Conference on Genetic Programming (2009) 315–326
5. Shao, X., Cherkassky, V., Li, W.: Measuring the VC-dimension using optimized experimental design. Neural Computation 12 (8) (2000) 1969–1986

6. Teutaid, O., Gelly, S., Bredeche, N., Schoenauer, M.A.: Statistical Learning Theory Approach of Bloat. Proceedings of the 2005 conference on Genetic and Evolutionary Computation (2005) 1784–1785
7. Vapnik, V, Chervonenkis, A.: On the uniform convergence of relative frequencies of events to their probabilities. heory of Probability and its Applications 16 (1971) 264–280
8. Vapnik, V, Chervonenkis, A.: Ordered risk minimization. Automation and Remote Control 34 (1974) 1226–1235
9. Vapnik, V., Levin, E., Cun, Y.: Measuring the VC-dimension of a learning machine. Neural Computation 6 (1994) 851–876
10. Vapnik, V.: Statistical learning theory. John Willey & Sons (1998)