Ensemble modeling approach for rainfall/groundwater balancing

D. Laucelli, V. Babovic, M. Keijzer and O. Giustolisi

ABSTRACT

This paper introduces an application of machine learning, on real data. It deals with Ensemble Modeling, a simple averaging method for obtaining more reliable approximations using symbolic regression. Considerations on the contribution of bias and variance to the total error, and ensemble methods to reduce errors due to variance, have been tackled together with a specific application of ensemble modeling to hydrological forecasts. This work provides empirical evidence that genetic programming can greatly benefit from this approach in forecasting and simulating physical phenomena. Further considerations have been taken into account, such as the influence of Genetic Programming parameter settings on the model's performance.

Key words | ensemble modeling, genetic programming, groundwater, hydrology

D. Laucelli (corresponding author)
O. Giustolisi
Department of Civil and Environmental Engineering,
Technical University of Bari,
Viale del Turismo 8, Taranto74100,
Italy
Tel.: +39 099 473 3210
Fax: +39 099 473 3230
E-mail: *d.laucelli@poliba.it*;
laucelli@libero.it

V. Babovic
M. Keijzer
Department for Strategic Research and Development,
WL Delft Hydraulics,
Rotterdamseweg, 185, Delft HD 2629,
The Netherlands

NOTATION

y	output variable
$\mathbf{x}$	set of input variables
ε	random variable distributed according to some law (residuals)
$f(\mathbf{x})$	deterministic function
D	data sample time series of size N
$\hat{f}(\mathbf{x}\|D)$	estimation of f ($\mathbf{x}$) using a particular training data set D
$\bar{f}(\mathbf{x})$	the average model
M	independent subsets randomly resampled (the number of bootstrap resampling)
D_{train}	training set
D_{test}	testing set
G	number of generations in GP runs
P	size of population in GP runs
R_t	rainfall at time sample t
T_t	air temperature at time sample t
H_t	average monthly groundwater head at time sample t
Δ_t	head information at time sample t ($\Delta_t = H_{t+1} - H_t$)
m_j	weight of the jth GP model within the EM

doi: 10.2166/hydro.2007.102

INTRODUCTION

In the last few years, the global climate changes are gradually increasing their influence on regional water resource management policies as well as the hydrological stability of the Southern Mediterranean areas (i.e. south of Spain, Greece, Tunisia, Turkey, etc.) (Simeone 2001). In the particular case of southern Puglia (Italy), rainfall decrease is emphasized by the particular water circulation path in which infiltration phenomena prevail over surface runoff (i.e. rivers, lakes, etc.), the area being deeply karstified. This situation is causing a decrease in reservoir recharge from both groundwater systems and surface catchment areas, which are particularly necessary to satisfy the agricultural water demand (i.e. agriculture is the main business of the region).

As a consequence, groundwater super-exploitation causes a lot of problems, such as seawater intrusion and groundwater salt contamination, leading to a loss of soil fertility due to the lower quality water (Grassi & Tadolini 1992). The latest Italian regulations (i.e. L. n. 152/06 on Landscape and Water Resources Management and Protection, Italian National Program against Desertification) encourage water resources management on a large scale

(i.e. groundwater), with particular emphasis on long time planning, monitoring and safeguarding. Thus, water managers need to know how much water will recharge the aquifers in the near future (i.e. up to 6 months ahead), because this will affect the volume of water that can be withdrawn ('abstracted') safely from the groundwater resources. Unfortunately, recharge can't be measured directly. An estimation method is required to be used to support decisions about the future assets of water resources within a particular area. The first step can be modeling the rainfall/infiltration phenomena, which is a nonlinear process due to a certain number of extra inputs which are very expensive to check. Such a model should be (relatively) easy to build and to update, as soon as new data become available, and preferably simple to use by decision-makers. In this paper, Ensemble Modeling (Breiman 1996*a*) has been tested, using Genetic Programming for building single models. Genetic Programming (Koza 1992) is a general purpose search technique that can be applied to both regression and classification problems. In contrast to linear and nonlinear regressions, this technique does not assume a concrete functional form, since it generally starts from a definition of low level building blocks (the function set) from which a functional form is induced. Here, Genetic Programming (GP) was used as an induction engine on which the Ensemble Modeling (EM) approach is based. Starting from field data, a comparison between EM and traditional GP Single Model technique was performed both in one-step-ahead prediction and *k*-step-ahead prediction. Sensitivity analyses were briefly performed in order to understand the correlations between user-defined parameters of GP and statistical coefficients used to evaluate the performance of the models.

ENSEMBLE MODELING THEORY

In model induction both the bias and the variance of the method should be minimized as they both contribute to the total error. These two terms contradict each other, thus implying a clearly unavoidable trade-off. Previous works indicate that GP applied to symbolic regression is a low bias, but high variance induction technique (Babovic & Keijzer 2000).

Bias, variance and estimation error

In a statistical learning theory scenario, GP can be applied to the problem of function estimation. In a standard function estimation problem, one assumes that an output variable y is somehow related to a set of input variables $\mathbf{x}$ as

$$y = f(\mathbf{x}) + \varepsilon \tag{1}$$

where $f(\mathbf{x})$ is a deterministic function and ε is a random variable distributed according to some law. If the residuals ε are expected to be (near) Gaussian (i.e. $E(\varepsilon|\mathbf{x}) = 0\ \forall\mathbf{x}$), it is possible to state that

$$f(\mathbf{x}) = E(y|\mathbf{x}) \tag{2}$$

so that the goal of supervised learning is to obtain an estimate

$$\hat{f}(\mathbf{x}|D) = \hat{E}(y|\mathbf{x}, D) \tag{3}$$

where D is the training data sample of size N (Babovic & Kejizer 2000). Therefore, inaccuracy of estimation can be measured using the mean squared error statistic:

$$MSE[\hat{f}(\mathbf{x}|D)] = E_D[(y - \hat{f}(\mathbf{x}|D))^2] \tag{4}$$

where the expected value in Equation (4) is calculated with respect to the distribution in Equation (1). The estimated probability $\hat{f}(\mathbf{x}|D)$ depends on the training data set D. Differently sampled D generally results in the change of probability estimate: y_i, which are stochastic due to random variable ε, and $\mathbf{x}_i$ themselves are susceptible to observational sampling. Therefore, estimation of $f(\mathbf{x})$ using a particular training data set D in principle results in a particular random realization of $\hat{f}(\mathbf{x}|D)$. Following Babovic & Kejizer (2000), the prediction error in Equation (4) can be decomposed as follows:

$$E_D[(y - \hat{f}(\mathbf{x}|D))^2] = E_D[(f(\mathbf{x}) - \hat{f}(\mathbf{x}|D))^2] + E_\varepsilon(\varepsilon|\mathbf{x}). \tag{5}$$

Equation (5) represents the squared prediction error averaged over all data sets D drawn from the same population. The second term in (5) is independent of both the target function and the training data as it originates in the random nature of the output variable, as defined in Equation (1). The first term in Equation (5) represents the squared 'estimation error' in the target function $f(\mathbf{x})$

averaged over the training samples D. It depends on $f(\mathbf{x})$ and on the method to obtain $\hat{f}(\mathbf{x}|D)$. Thus, it can be expanded as

$$E_D[(f(\mathbf{x}) - \hat{f}(\mathbf{x}|D))^2] = [f(\mathbf{x}) - E_D(\hat{f}(\mathbf{x}|D))]^2 + [E_D[\hat{f}(\mathbf{x}|D) - E_D(\hat{f}(\mathbf{x}|D))]^2 \quad (6)$$

It is rather obvious from Equation (6) that the squared 'estimation error' depends on the statistical properties of the distribution of $\hat{f}(\mathbf{x}|D)$ (i.e. its mean and its variance). The first term on the right in square brackets in Equation (6) is the square of the 'bias':

$$\text{bias} = f(\mathbf{x}) - E_D(\hat{f}(\mathbf{x}|D)) \quad (7)$$

The bias term (7) reflects the sensitivity of an estimate $\hat{f}(\mathbf{x}|D)$ to the target function $f(\mathbf{x})$. It represents how good an estimate is able to approximate the target on the average. The bias (7) describes the generic ability of a learning method as well as the properties of kernel functions used by the learning method to approximate the target function. The second term in Equation (6) is simply the variance:

$$\text{variance} = E_D[\hat{f}(\mathbf{x}|D) - E_D(\hat{f}(\mathbf{x}|D))]^2. \quad (8)$$

Variance does not depend on the target distribution directly (Heskes 1998). Furthermore, it is non-negative and zero if and only if all estimators are equivalent. The bias depends only on the target distribution and the average model, which is defined as the model that minimizes variance. For a given bias (7), the variance (8) generally decreases with increasing training sample size N. It can be expected that, for training samples with large N, the bias remains the main contributor to estimation error. This paper investigates the circumstances where the availability of data is limited. Equation (6) clearly illustrates that it is desirable to have both low bias and low variance since both terms contribute to the squared estimation error to the same extent. However, these two objectives are contradicting each other. The purpose of training is in approximating a target function. Sensitivity to the training data is essential, implying lower bias. At the same time, large sensitivity increases variance. The contradictory nature of these two objectives gives rise to a rather natural bias/variance trade-off. This trade-off has been an objective of investigation in many machine learning areas (Geman *et al.* 1992; Friedman 1997; Breiman 1999).

Ensemble method

Under a statistical viewpoint, the bias/variance trade-off is often referred to as a dilemma. Geman *et al.* (1992) go so far as to state that the dilemma can be circumvented only if one is willing to sacrifice generality, that is, purposefully introduce bias, in order to reduce the variance. Thus, when this bias is attuned to the problem domain, such a method will reliably give good results. In GP several methods have been investigated for introducing such a bias (Montana 1995; Whigham 1996; Martin *et al.* 1999). The background knowledge about the problem domain is introduced to increase the level of bias and to decrease the magnitude of variance, due to the trade-off. When the bias so introduced does not increase the bias error, the method will produce better solutions. However, there are methods of reducing the variance without increasing the bias. One such method is selecting a single best model by using cross-validation data. Another method for obtaining better performance makes explicit use of the fact that the error due to variance is caused by deviation from the average model:

$$\bar{f}(\mathbf{x}) = \frac{1}{M}\sum_{1}^{M}\hat{f}_j(\mathbf{x}_i) \Rightarrow i = 1 \ldots N. \quad (9)$$

As this average model can be easily calculated, it can be used instead of a single best model. Therefore, given a data set D, firstly a disjoint training D_{train} and testing set D_{test} are created. D_{train} is further subdivided by randomly drawing cases to form M independent subsets. These M sets are then used as the training sets for M independent runs of the GP algorithm. Given these M models $\hat{f}_1(x), \ldots, \hat{f}_M(x)$ and testing data $D_{\text{test}} = (x_1, y_1) \ldots (x_N, y_N)$, the ensemble mean squared error is defined as

$$\frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(y_i - \hat{f}_j(\mathbf{x}_i))^2. \quad (10)$$

By introducing the average model the mean squared error can be decomposed into bias and variance by a

straightforward rearrangement of terms in (10):

$$\text{bias} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{f}(\mathbf{x}_i))^2. \quad (11)$$

$$\text{variance} = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(\bar{f}(\mathbf{x}_i) - \hat{f}_j(\mathbf{x}_i))^2. \quad (12)$$

Again the bias term depends on the target distribution (y), while the variance term does not. A more elaborate formulation would further decompose the bias term into true bias and noise (see Equation (6)) but, as in practice the inherent noise is often unknown, here the current definition is used. Note that the average model is calculated without utilizing testing data. By using this average model as the resulting model of many runs, the error due to the variance is effectively eliminated. The remaining error contribution is due to the bias alone, which represents the generic ability of the method to deal with the problem. Therefore, using the average model instead of a single model leads to a new bias/variance trade-off. The difference is that a new decomposition over different runs producing different average models can be performed. The expected improvement originates in the fact that the associated variance of these average models is lower than the variance of the original setup. As was shown in Babovic & Keijzer (2000), GP exhibits low bias. Therefore this offers a simple and feasible approach to reducing the generalization error in symbolic regression. The technique of combining multiple models into a single one is referred to as Ensemble Modeling. When the process of averaging is used in conjunction with bootstrapping to obtain the training sets, the technique is referred to as bootstrap aggregating (bagging) (Breiman 1996*a*). Other Ensemble methods are boosting (Freund & Scahpire 1996) and stacking (Breiman 1996*b*). Moreover, Iba (1999) applied the ensemble methods of boosting and bagging to genetic programming and obtained encouraging results.

Resampling techniques

Generally speaking, observed datasets are often poor in the quantitative and qualitative sense and it is not easy to get an accurate, manageable, or even analytical description of the distribution of these data. For these reasons it is usually not possible to get analytical expressions for the statistics on the desired uncertainties. Analytical approaches seem to be restricted to very special cases only. Moreover, a proper analytical description of the probability distribution of the observed data is not easy to get, and an alternative approach must be followed to obtain an expression for the model uncertainties. In particular, techniques that are generic or insensitive for the statistical properties of the data are desired. The so-called resampling techniques form a group of such statistical methods (van den Boogaard *et al.* 2000). As a result of this procedure an ensemble of estimates is available for both the uncertain model parameters and for the corresponding model output. Depending on the used form of resampling these multiple estimates must be combined in some way to obtain the desired statistics (distribution, mean, spread, confidence interval, etc.) of the involved quantities. The most common form of resampling are jack-knifing (Wu 1986) and bootstrapping (Heskes 1997). Bootstrapping is used in this paper.

Bootstrapping

A bootstrap resample is a random selection of N data out of the N original data. The N individual draws within such resample are independent but with replacement, so that every time there is a probability of $1/N$ that a particular sample of the original set is selected. At the end some samples are then selected more than once while other samples are absent in the resample. Note that the probability that an original sample is not present in the resample is then $(1-1/N)^N$ which for large N is close to $1/e$ ($\cong$ 37%). In this way an ensemble of L of such resamples is generated. This L should be sufficiently large and in practice it is typically of the order of a hundred or a few hundred, somewhat depending on the statistics to be computed. This form of bootstrapping is often called naive bootstrapping. There are alternative or adapted forms for the bootstrap such as the weighted bootstrap, bootstrapping of normalized residuals, the smoothed bootstrap, the parametric bootstrap (Efron 1982; Wu 1986; Efron & Tibshirani 1993) or 'bagging' (Breiman 1996*a*).

CASE STUDY: BRINDISI SURFACE AQUIFER LEVELS MODELING

The particular case study is from the southern zone of Puglia (Italy). In the last few years, this region has shown a gradual decreasing of rainfall events with consequences on the hydrological water surplus and the infiltration process (Grassi & Tulipano 1983). In fact, in this region the infiltration flows are prevalent with respect to the surface organized ones, such as rivers or lakes, because of its particular hydrogeological features (i.e. karst area) (Simeone 2001). As consequence of this low availability of surface water, the groundwater overexploitation has become a widespread habit in agriculture, inducing a self-increasing negative feedback between natural and human-induced phenomena, whose final effect is not of easy management and control. One of the most significant consequences is seawater intrusion and groundwater salt contamination leading to a severe loss of soil fertility due to the lower quality of water (Grassi & Tadolini 1992). The paper introduces an application of bias/variance decomposition of mean squared error to real data from the Brindisi area, as well as a presentation of experimental results on the application of GP Ensemble Model. A concise description of geographical and geological situation follows.

Background to hydrogeological information

This area, also called *Piana di Brindisi*, is characterized by terraced surfaces gently sloping caused over the centuries by the regression of sea levels. There are many natural streams with small depth, but only a few of these have an uninterrupted flow over the months. The *Piana di Brindisi* consists of a wide structural depression of the Apulian karst block, open towards the Adriatic coast. Over the centuries, drifts have settled on this depression forming the actual surface aquifer body. This water resource is fed by infiltration because of its high permeability. Two different hydrogeological systems can be distinguished: the first is the local phreatic sandy surface aquifer, lying on an impermeable bed of Sub-Apennines clay; the second, is the regional deep karst and fractured aquifer, underlying the first one, whose waters, locally, can flow pressurized (Ricchetti & Polemio 1996). During the year, there is usually a drought period around July and rainfall peaks in November and December. Observing Figure 1, it is clear that the aquifer bed is tilted towards the north-east with a slight concavity close to the coast line. The aquifer thickness has a modest local variability showing a general increase in the central area of the aquifer, along a NE–SW direction. On the basis of piezometric information and direct measurements, it was argued that the sea has a low drainage action on groundwater. The gradients of the bottom and upper surfaces are parallel to the global piezometric gradient. Moreover, the aquifer thickness is smaller than the gap between its upstream and downstream heads and groundwater is drained through its bottom towards the deep aquifer.

Available data

In order to outline the hydrogeological regimes of the surface aquifer near the town of Brindisi many available data have been considered from several phreatimetric and rainfall gauges belonging to regional boards like Hydrographical Service Office (Bari) and Meteorological Observatory (Taranto). The two time series used in this paper are those considered the most reliable ones. Both measurement

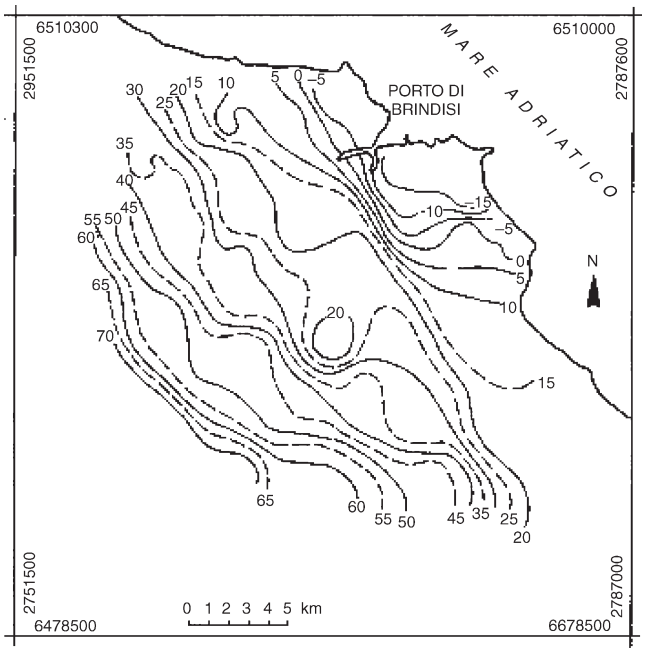

Figure 1 | Hydraulic heads of Brindisi's aquifer.

stations were chosen because of their continuous and reliable series of data, since they are located outside urbanized areas. The available data are the total monthly rainfall (mm) and the monthly average temperature (°C) and the average monthly groundwater head (m above mean sea level). A period of time of 44 years, from 1953 to 1996, has been considered.

Modeling strategies

With the constraints of data availability, GP is able to approximate the target function reliably, with an unconstrained program size. At the same time, small programs appear to be strongly biased, exhibiting high variance. This leads straightforwardly to the conclusion that the GP algorithm needs many nodes (more than 15) (Babovic & Keijzer 2000). Moreover, many techniques exist that can approximate a target function to any degree of accuracy with much less effort than GP, for example Artificial Neural Networks (see Giustolisi & Laucelli 2004). Conversely, when data are sparse, the situation is more complex. Small programs are strongly biased again and highly variant, while large programs exhibit high variance as well. In these situations, bias/variance decomposition helps in selecting the optimal program size. Examination of the contribution of bias and variance to the total error helps in indicating the location of a low bias/reasonable variance region where reliable models might be found, while a reasonable bias and low variance region is of less interest due to the strong influence of bias in defining the squared estimation error, see Equation (6).

Application of trimming

EM provides an alternative approach to obtaining a single model. It combines the results of many GP runs in a single average model. This averaging process could be destabilized due to GP's ability to produce extremely poor solutions (outliers) that destabilize bias, also producing high variance (Babovic & Kejizer 2000). Since EM reduces the total error to the bias error alone, the question of destabilized bias needs to be addressed with care. Thus, some post-processing needs to be applied in order to obtain more insightful results. The predictions will be trimmed before calculating error due to bias (11) and variance (12). The main purpose for trimming is to remove extremely poor programs and thus stabilize bias. Enlarging the trimming percentage even more would effectively stabilize variance as well (Babovic & Kejizer 2000). As is clear from the remainder, an optimal trimming scenario should stabilize bias while leaving most of the variance intact, as a high degree of variance is instrumental in establishing an ensemble model (Babovic & Keijzer 2000). As the calculation of variance is independent of the target function, in this paper trimming has been applied after the EMs have been created.

Moreover, a sensitivity analysis has been performed to study the relationships between the trimming percentage and the bias/variance performance of the EMs. Other kinds of sensitivity analyses were carried out between the statistical parameters used to identify the goodness of the models (CoD, Bias and Variance) and, in order, the number of bootstrap resampling (M), the number of generations in GP runs (G) and the size of populations in GP runs (P). It is noteworthy that these last analyses show no relevant variation in the global performance of the EMs, even if a low number of bootstrap resampling ($M < 15$) carries to low performance models. The conclusion of trimming analyses will be reported in the next paragraphs.

Experimental setup

This paper describes an approach using GP as the induction engine for EM.

The case study starts from field data of rainfall, temperature and groundwater heads from the Brindisi surface aquifer. A comparison between EM and traditional GP Single Model was performed in one-step-ahead prediction. Moreover, EM was tested in k-step-ahead forecasting. Starting from the considerations of Babovic & Keijzer (2000) about the contributions of bias and variance to the total error and from the prior physical knowledge about the aquifer (Ricchetti & Polemio 1996) the following setup for applying GP to the case study was build up:

- Building the input data matrix involving rainfall till four samples (R_t, R_{t-1}, R_{t-2} and R_{t-3}), air temperature at the present sample (T_t) and head information at the present sample (Δ_t). It has been used that $\Delta_t = (H_{t+1} - H_t)$ because of the strong correlation among head data, in order to perform bootstrapping with uncorrelated data points.

- Splitting the input data set into two main sets, the training and the test sets, 75% and 25%, respectively, of the whole input set. The training set was completely used for performing GP runs and defining all the single models, while the test set was split into two further sets, one (calibration set) for determining the coefficients of the single models in the EM and the other (evaluation set) for really testing the performance of the whole EM. For the single GP model, the calibration set was used in selecting the best model, which was evaluated on the evaluation set. Usually the calibration set was 25% of the whole test set.
- Performing M different runs for every EM determination, using bootstrap method as the resampling strategy, creating unique datasets for different runs. M was determined as a consequence of the above mentioned sensitivity analyses. A value of 30 seems to be good also according to the time efficiency of the computations. Bootstrapping was performed both for EM and for the single GP model.
- Fixing the maximum size of GP programs to three different values: 20, 140 and 260 nodes, for every experiment. For both the EM and the single GP model M different runs with all three maximum sizes were performed. For the single GP model only the best one was selected, while the EM was built up with models having different sizes. The other user-defined GP parameters were fixed according to the above mentioned sensitivity analyses. In particular, a number of generations (50) and a population of 100 individuals were chosen, thus obtaining a very fast algorithm. Finally, note that the function set consisted of addition, multiplication, difference, square root, division and power two, according to the authors' GP insight (Babovic & Kejizer 2000; Giustolisi & Laucelli 2002).
- Combining the results of all the runs in a single ensemble model. Indeed, Equation (10) can be rewritten as

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} m_j(y_i - \hat{f}_j(\mathbf{x}_i))^2. \quad (13)$$

In this way coefficients m_j can be determined performing a ridge regression on data (calibration set), being $\Sigma m_j = 1$ and $\forall \Delta j = 1 \ldots M$: $m_j \geq 0$. As a consequence of this variation

Table 1 | One-step-ahead statistics

	CoD	Bias	Variance
Full_EM	0.953 08	0.019 06	0.004 07
Avg EM	0.952 95	0.019 11	0.003 78
Trimm EM	0.959 03	0.016 64	0.002 61
Single Model	0.950 68	0.020 04	–
Naive	0.942 38	–	–

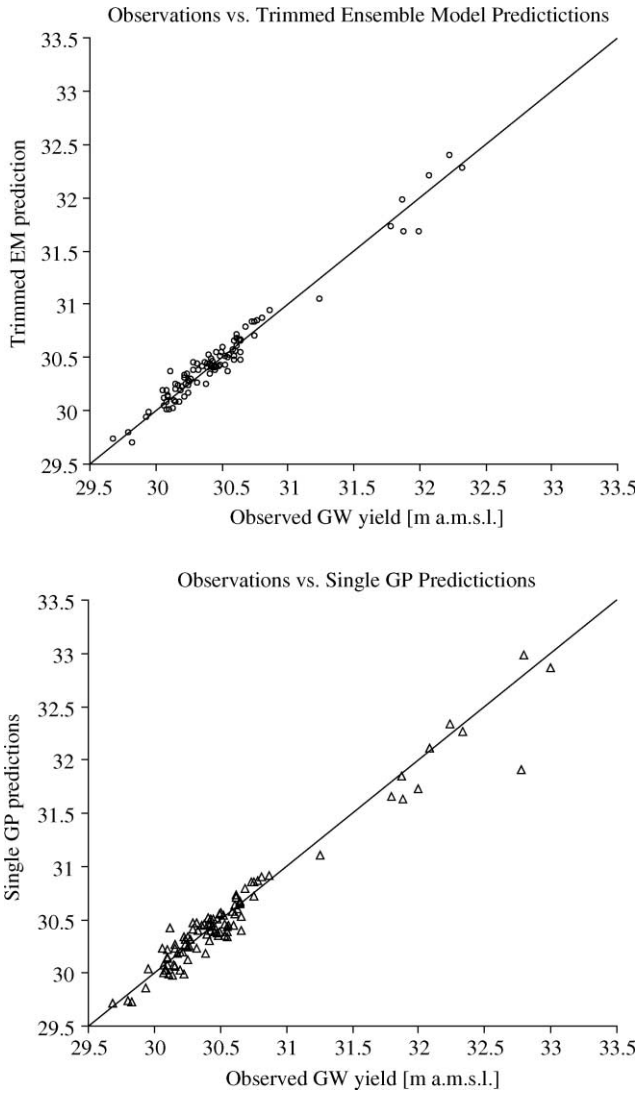

Figure 2 | Trimmed EM vs. Single GP model on one-step-ahead prediction.

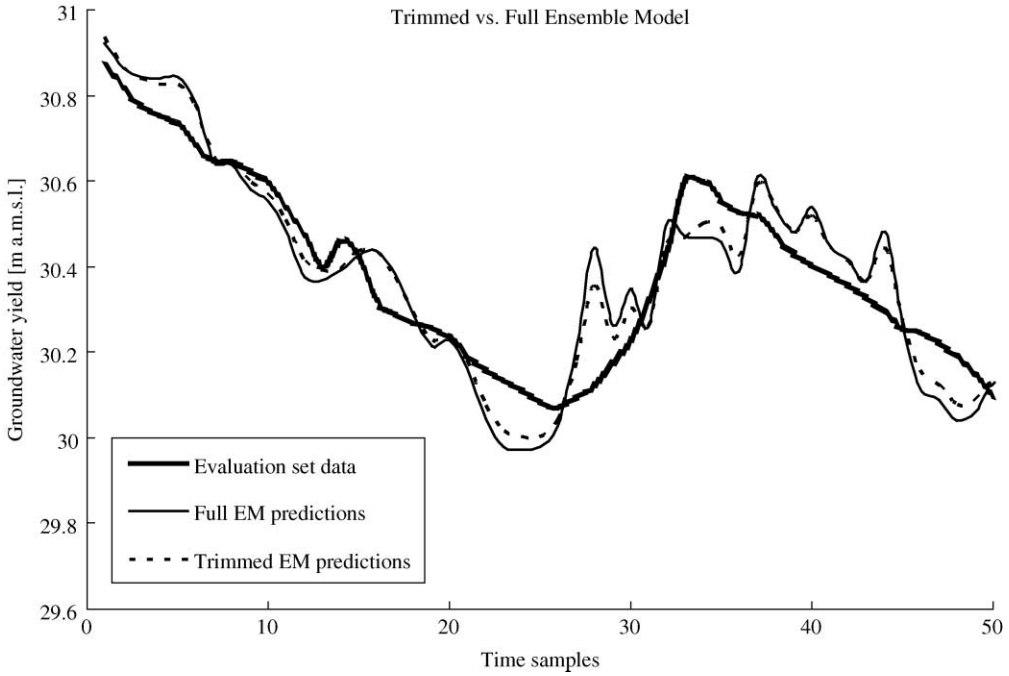

Figure 3 | Highlight of trimmed EM vs. full EM prediction diagrams.

Equation (12) also has to be rewritten as

$$\text{variance} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} m_j(\bar{f}(\mathbf{x}_i) - \hat{f}_j(\mathbf{x}_i))^2 \quad (14)$$

The authors did not evaluate the mean value of variance (useful for general considerations) but the variance value for every test point, erasing $1/N$ in Equation (14). This recipe refers to the one-step-ahead prediction performances, and following it a calibrated EM and a Single GP model are obtained. In post-processing, the authors applied a trimming scenario to the EM, aiming at deleting predictions that destabilize its performances. According to the sensitivity analyses a trimming percentage of 20% has been chosen. At the same time, it was also to evaluate the real average model in Equation (6). From the fitness point of view, no strong differences were found between untrimmed EM and average EM.

RESULTS AND DISCUSSION

Here are concisely reported the results with relative diagrams, starting from one-step-ahead prediction. In Table 1 there are all the statistics calculated in order to

Table 2 | k-step-ahead statistics, with $k = 2, 3, 4, 6$

k		CoD	Bias	Variance
2	Full_EM	0.894 85	0.043 05	0.001 15
	Trimm EM	0.893 45	0.043 63	0.002 63
	Naive	0.828 30	–	–
3	Full_EM	0.819 32	0.074 63	0.001 24
	Trimm EM	0.814 35	0.076 68	0.002 65
	Naive	0.687 73	–	–
4	Full_EM	0.778 32	0.092 40	0.001 41
	Trimm EM	0.770 97	0.095 47	0.002 66
	Naive	0.588 87	–	–
6	Full_EM	0.766 42	0.099 26	0.001 33
	Trimm EM	0.757 82	0.102 92	0.002 69
	Naive	0.467 10	–	–

evaluate performances of both EMs and the single GP model. Coefficient of Determination (CoD), Bias and Variance values are reported. In Table 1 the notation 'Full_EM' indicates the untrimmed EM, while the authors used as common reference the performance of the naive model, $H_{t+1} = H_t$. The improvement, given the naive performances, is not particularly relevant, because of the average monthly data of groundwater level.

A large water resource, such as an aquifer, represents a system with a high inertial component (i.e. persistency) in its behavior during days, weeks, etc. and this means that H_{t+1} is not much different from H_t. The one-step-ahead comparison between EM and single GP model aims at demonstrating the efficiency of the ensemble approach in hydrological problems. Looking at Table 1, the trimmed EM model seems to be the better performing one, in one-step-ahead forecasting. Thus in Figure 2, it is compared with the 'traditional' single GP model. Figure 2 shows the predictions of the trimmed EM model and of the single GP model, plotted against the experimental data in the evaluation set.

In Figure 3, the untrimmed (full EM) model and the trimmed EM model performances are compared for better understanding of the efficacy of trimming. Figure 3 focuses on a subarea of the relatives prediction plots. Trimmed EM model appears to be more stable and smooth, especially at

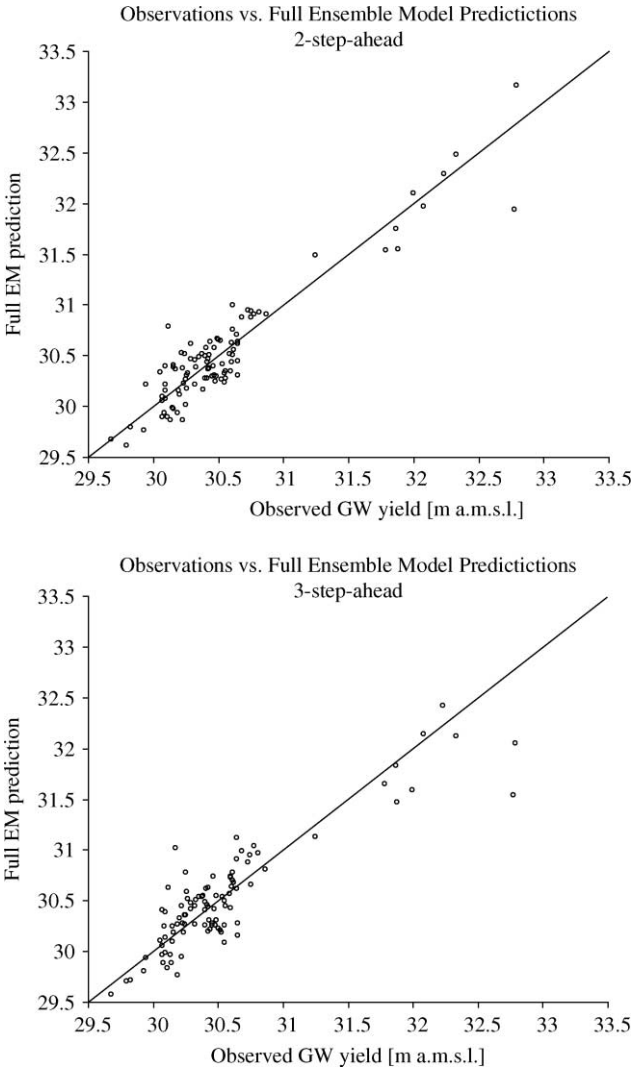

Figure 4 | Full EM two-step-ahead and three-step-ahead predictions.

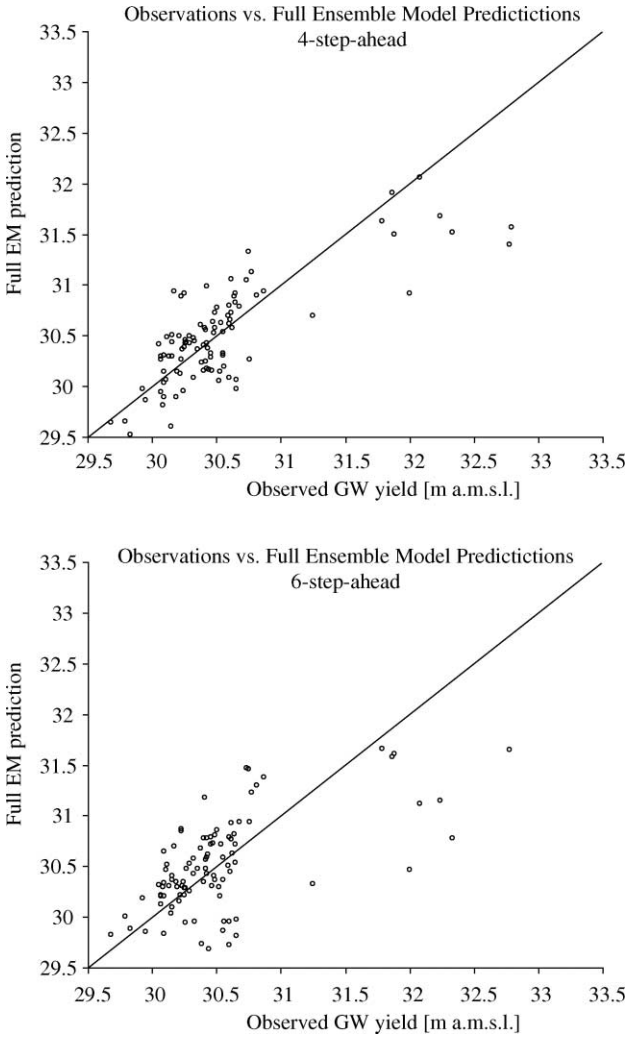

Figure 5 | Full EM four-step-ahead and six-step-ahead predictions.

low values. Moreover, looking at the whole evaluation test it has a lower point variance on average (see Table 1).

Analyzing now the *k*-step-ahead forecasting, Table 2 reports the statistics for time horizons up to 6 months ahead. These analyses are due to the aforementioned discussion on the planning issues for the aquifer at stake. Looking at Table 2, statistics are quite good, also considering that the sample rate is monthly, and they are comparable with those of a well trained artificial neural network (Giustolisi & Laucelli 2004). It is noteworthy that this approach returns low variance models, overcoming GP problems about high variance in symbolic regression (Babovic & Keijzer 2000). Moreover, for time horizons longer than one month, trimming seem to be no longer necessary in order to improve both bias and variance contribution to the total mean error. Thus, Figures 4 and 5 report the full EM model performances plotted against the experimental data in the evaluation set.

Some considerations have been done aimed at understanding how the performance of the EM model could change by modifying some relevant user-defined parameters in the GP algorithm. The analyzed parameters were (i) the dimension of population, *P*, ranging between 100 and 1000; (ii) the number of generations for each GP run, *G*, varying between 50 and 1050; and (iii) the number of resampling, *M*, ranging between 5 and 100. Moreover, the sensitivity of bias and variance to trimming percentage was studied, using percentages between 0 and 50%.

For the first three parameters the analyses were performed for both the trimmed and untrimmed EM, returning that they are not influential on the performance of EM models, relying on bias and variance values and on MSE and CoD values (Laucelli 2004). About the trimming scenarios, as in Figure 6, it can be stated that trimming percentage positively influences decreasing the bias values and can give good reduction of variance using a particular range (between 5% and 20%). Therefore, trimming percentage has to be chosen carefully, aiming at reducing both bias and variance values.

Finally, in one-step-ahead prediction, beyond the numbers, which are undoubtedly good, the results appear to be satisfying because there is not a 'phase error' in forecasts, see Figure 7. This error makes the naive model unreliable for real uses. In the case study, the naive model is a high performance reference because the phenomenon dynamics are persistent. However, the statistical parameters chosen for performance analyses show that both Ensemble Model and single GP model have better performance than the naive one, having also no 'phase error'. The sensitivity analyses of bias/variance values with trimming percentage show that a percentage of 20% guarantees a good trade-off between variance and bias, taking both on low values. These facts demonstrate that trimming is useful in one-step-ahead prediction eliminating those 'outliners' (Babovic & Keijzer 2000), which make EM performance worse, in particular decreasing a lot of variance values. Conversely, trimming

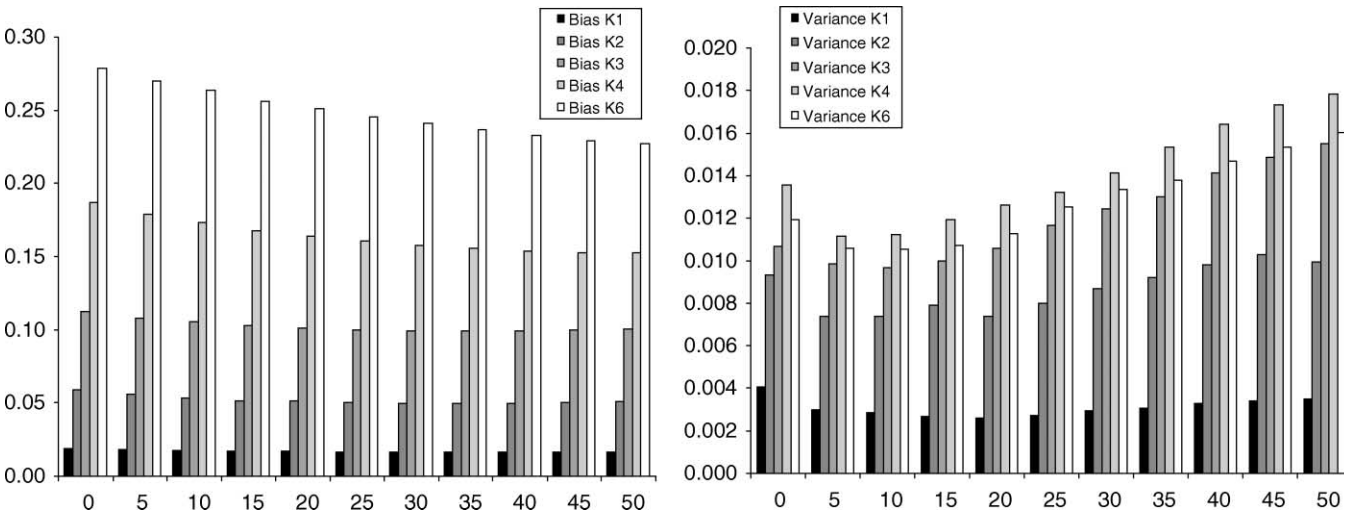

Figure 6 | Bias and variance diagrams, for different values of trimming percentage.

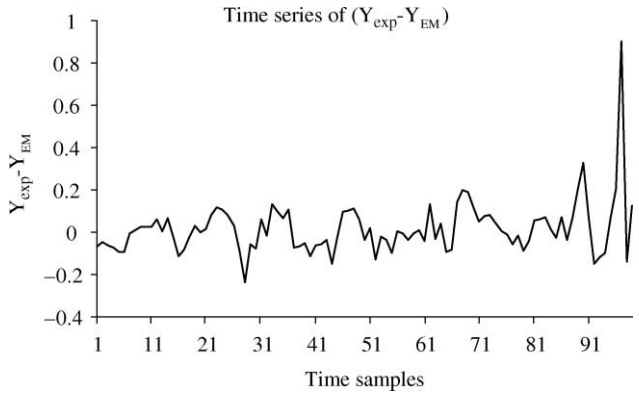

Figure 7 | Time series of (Y_{exp}-Y_{EM}) plot (for one-step ahead prediction).

seems non relevant in k-step-ahead forecasting, with $k > 1$ (see Table 2), likely due to the fact that, with a longer prediction horizon, GP produce a lower number of outliers. In particular, the no-trimming scenario in k-step-ahead predictions implies lower variance models. Finally, also in k-step-ahead prediction the results are good. Due to available data, which are under-sampled (monthly sampled), the approach used by going on step by step, month by month in the case study, better relies on a longer time horizon, showing performances which are comparable with well-trained neural networks, such as those in Giustolisi & Laucelli (2004).

CONCLUSIONS

In a statistical learning theory scenario, GP can be applied to the problem of function estimation. Then, in the limits of data abundance, it is able to approximate the target function reliably, starting from an unconstrained program size. Otherwise, when data are sparse, small programs are strongly biased again and highly variant, while large programs exhibit high variance as well. In these situations, bias/variance decomposition helps in selecting the optimal program size. Aiming to obtain a single model, Ensemble Modeling provides an alternative approach. On these premises, this paper tests Ensemble Modeling using a Genetic Programming numerical engine. Starting from field data of rainfall, temperature and groundwater heads, a comparison between EM and traditional GP Single Model was performed in one-step-ahead prediction. The EM model has been tested also in k-step-ahead prediction. Ensemble methods, together with trimming, have substantially deleted the contribution of variance to the total error, also improving the absolute value of the bias and of the Coefficient of Determination. Conversely, the single GP model based on cross-validation did not guarantee the same performance, because it is based on a single model, so there are higher possibilities that undesired behavior occurs. In one-step-ahead, the trimmed EM model results were more stable along with the entire evaluation test, especially for low target values, with respect to the traditional single GP model. Otherwise, trimmed EM has a lower point variance than the untrimmed EM model, and this could be a constraint in using the model for planning. In k-step-ahead prediction, the EM approach shows good performances carrying to lower variance models especially without trimming scenarios. Trimming is useful in one-step-ahead prediction eliminating those 'outliners' that cause worse EM performances, strongly decreasing the variance values.

ACKNOWLEDGEMENTS

The authors want to acknowledge Prof. Vincenzo Simeone and Dr. Rossana Racioppi for their help in providing, understanding and processing the field data used in this work. Finally, the authors are particularly grateful to Dr. Angelo Doglioni and Mr. Davide Mancarella for their valuable help in reviewing the paper both from a technical and linguistic point of view.

REFERENCES

Babovic, V. & Keijzer, M. 2000 Genetic programming and the bias/variance trade-off. In *Proc. of the European Genetic Programming Conference, Edinburgh, UK*, pp. 76–90. Springer-Verlag, Berlin.

Breiman, L. 1996*a* Bagging predictors. *Machine Learning* **26**, 123–140.

Breiman, L. 1996*b* Stacked regressions. *Machine Learning* **24**, 51–64.

Breiman, L. 1999 Prediction games and arching algorithms. *Neural Comput.* **11**, 1493–1517.

Efron, B. 1982 *The Jackknife, the Bootstrap and Other Resampling Plans*. SIAM, Philadelphia, PA.

Efron, B. & Tibshirani, R. 1993 *An Introduction to the Bootstrap*. Chapman & Hall, London.

Freund, Y. & Scahpire, R. H. 1996 Experiments with a new boosting algorithm. In *Machine Learning: Proc. of the 13th International Conference*, vol. 1, pp. 148–156. Morgan Kaufmann, San Francisco.

Friedman, J. H. 1997 On bias, variance 0/1 loss and the curse-of-dimensionality. *Data Mining Knowledge Discovery* **1**, 55–77.

Geman, S., Bienenstock, E. & Doursat, R. 1992 Neural networks and the bias/variance dilemma. *Neural Comput.* **4**, 1–58.

Giustolisi, O. & Laucelli, D. 2002 Modelling rainfall-runoff by genetic programming: results from two experimental urban basins. In *Proc. 5th International Conference on Hydroinformatics, Cardiff*, vol. 2, pp. 1484–1491. IWA Publishing, London.

Giustolisi, O. & Laucelli, D. 2004 A new method to train multi-layer perceptrons as support vector machines. In *Proc. 6th International Conference on Hydroinformatics, Singapore*, vol. 2, pp. 1605–1612. IWA Publishing, London.

Grassi, D. & Tadolini, T. 1992 Predisposition of the Apulian Mesozoic carbonate platform to anthropic and marine pollution. In *29th International Geological Congress, Kyoto*, vol. 3, p. 756. VSP. Leiden, Abstracts volume.

Grassi, D. & Tulipano, L. 1983 Connessioni fra l'assetto morfo-strutturale della Murgia (Puglia) caratteri idrogeologici della falda profonda verificati anche mediante analisi della temperatura delle acque sotterranee. *Geologia Applicata ed Idrogeologia* **18** (1), 135–153.

Heskes, T. 1997 Practical confidence and prediction intervals. In *Advance in Neural Information Processing Systems*, vol. 9 (ed. M. C. Mozer, M. I. Jordan & T. Petsche). MIT Press, Cambridge, pp. 176–182.

Heskes, T. 1998 Bias/variance decompositions for likehood-based estimators. *Neural Comput.* **10**, 1425–1433.

Iba, H. 1999 Bagging, boosting and bloating in genetic programming. In *Proc. of the Genetic and Evolutionary Computation Conference, Orlando, FL*, vol. 2 (ed. W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela & R. E. Smith), pp. 1053–1060. Morgan Kaufman, San Francisco.

Koza, J. R. 1992 *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge.

Laucelli, D. 2004 *Ensemble Modeling Approach Applied to Rainfall/Groundwater Balance. Technical Report no. X0280.40*. WL Delft Hydraulics, Delft, The Netherlands.

Martin, L., Moal, F. & Vrain, C. 1999 Declarative expression of biases in genetic programming. In *Proc. of the Genetic and Evolutionary Computation Conference, Orlando, FL* (ed. W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela & R. E. Smith), pp. 401–408. Morgan Kaufman, San Francisco.

Montana, D. J. 1995 Strongly typed genetic programming. *Evol. Comput.* **3** (2), 199–230.

Ricchetti, E. & Polemio, M. 1996 L'acquifero superficiale del territorio di Brindisi: dati idrogeologici diretti e immagini radar da satellite. *Memorie Della Società Geologica Italiana* **51** (2), 1059–1074.

Simeone, V. 2001 Variazioni climatiche e rischi di depauperamento delle falde e di desertificazione in provincia di Taranto. *Geologia Tecnica and Ambientale* **2**, 23–32.

van den Boogaard, H. F. P., Mynett, A. & Heskes, T. 2000 Resampling techniques for the assessment of uncertainties in parameters and predictions of calibrated models. In *Proc. of the 4th International Conference on Hydroinformatics, Cedar Rapids, Iowa, USA*, p. 313. IAHR, Delft, Abstracts volume, (CD-ROM UA-2:342.pdf).

Whigham, P. A. 1996 *Grammatical Bias for Evolutionary Learning* PhD thesis, School of Computer Science, University College, University of New South Wales, Australian Defence Force Academy.

Wu, C. F. J. 1986 Jackknife, bootstrap and other resampling methods in regression analyses. *Ann. Stat.* **14**, 1261.