Dipartimento di Statistica
"Giuseppe Parenti"

Dipartimento di Statistica "G. Parenti" – Viale Morgagni 59 – 50134 Firenze - www.ds.unifi.it

WORKING PAPER 2003/04

A Flexible Tool for Model Building: the Relevant Transformation of the Inputs Network Approach (RETINA)

Teodosio Perez-Amaral,
Giampiero M. Gallo, Hal White

Università degli Studi
di Firenze

Econometrics

This version: MARCH 14, 2003

A Flexible Tool for Model Building: the Relevant Transformation of the Inputs Network Approach (RETINA)

Teodosio Pérez-Amaral,
Departamento de Analisis Economico
Universidad Complutense de Madrid,
teodosio@ccee.ucm.es

Giampiero M. Gallo,
Dipartimento di Statistica "G.Parenti"
University of Florence, Italy
gallog@ds.unifi.it

Halbert White,
Department of Economics
University of California, San Diego,
hwhite@weber.ucsd.edu

ABSTRACT

A new method, called relevant transformation of the inputs network approach (RETINA) is proposed as a tool for model building and selection. It is designed to improve on some of the shortcomings of neural networks.

RETINA has the flexibility of neural network models, the concavity of the likelihood in the weights of the usual linear models, and the ability to identify a parsimonious set of attributes that are likely to be relevant for predicting out of sample outcomes. It achieves flexibility by considering transformations of the original inputs; it splits the sample into three disjoint subsamples, sorts the candidate regressors by a saliency feature, chooses the models in subsample 1, uses subsample 2 for parameter estimation, and uses subsample 3 for cross-validation. It is modular, can be used as a data exploratory tool, and is computationally feasible in personal computers.

In tests on simulated data, it achieves high rates of successes when the sample size or the R^2 are large enough. As our experiments show, it is superior to alternative procedures such as the non-negative garrote and backward stepwise regression.

1. Introduction

Model building and selection are crucial in statistical analysis and at some point in the effort, a decision must be made as to which among several specifications (possibly belonging to different classes of models) should be chosen to represent a relationship between a dependent variable and other variables of interest. Among these, one may prefer a parametric specification (either linear or nonlinear) where some interpretation of parameter values may be retained, or else suggest the adoption of flexible functional forms where the relationship among the variables is guided by other criteria of explanatory power. In this respect, the search for flexibility may be guided by the inadequacy of a linear model with Gaussian errors to represent data in a suitable way. Generalized Linear Models (McCullogh and Nelder, 1989) and Generalized Additive Models (Hastie and Tibshirani, 1990) for specific classes of problems and Artificial Neural Networks (White, 1989) provide leading examples of such a strategy.

Within each class of models, the problem of selecting the specification is far from trivial. Approaches to model selection are numerous: not only do they differ between one another, but they present peculiarities which reveal the importance given by each to different aspects of modelling itself.

Some methods focus on the relationship between a model and its interpretability according to some theory, others are based on hypothesis testing between competing models; some depend upon the trade-off between explanatory power and parsimony in the retained specification, others are based on the performance of a model in explaining a set of data not used for estimation, especially when the flexibility of the in-sample specification may lead to overparameterization; and so on. One popular approach in econometrics is the so-called general-to-specific methodology, whereby from a specification with a certain degree of complexity one seeks, more parsimonious representations of the data which retain the same information in a simpler form. For a recent debate on this approach and its capability to recover the traits of the DGP, see Hoover and Perez, 2000, and the discussion contained therein, especially the somewhat skeptical view by Granger and Timmerman. This method involves a battery of diagnostic tests on estimated coefficients and residuals to achieve its goals.

No approach is perfect, especially when misspecification of a model relative to the process which generated the data is always a possibility; hence all approaches to model selection have their particular limitations. Hypothesis testing in support of model choice is well-reputed as potentially dangerous (cf. Granger et al., 1995) given the implicit advantage attributed to the model under the null hypothesis in a nested framework or the possible ambiguity of results in a non-nested context. Moreover, one of the undesirable aspects of such an approach is the need to resort to pairwise comparisons.

In frameworks in which a penalty function for the number of parameters modifies the value of the likelihood function to provide a number which can be used to select a model (as in the Akaike's AIC or the Schwartz's BIC) there is always the issue of which form such a function should take, especially given certain undesirable properties of such information criteria to systematically choose over- or under-parameterized models in some circumstances.

Model selection based on out-of-sample performance is also prone to problems and, in fact, after the pioneering work by Granger and Newbold in the early 1970s (Granger and Newbold, 1973), it has become standard practice only in recent years to adopt testing procedures for predictive ability whereby some measure of performance (such as the Mean Squared Prediction Error, but again the choice of the criterion is not neutral) is used in a formal hypothesis testing framework (cf. Diebold and Mariano, 1995, West, 1996, White, 2000, Giacomini and White, 2003).

In general, researchers are aware that model selection via a specification search (led, for example, by a forecasting performance criterion) may translate into the choice of a good model just due to luck: the pervasive investigation of the same data either individually or collectively may distort the view on the "right" model to work with.

In this paper we present a tool that may be useful for model building and selection: we suggest a novel approach to investigating a data structure with the purpose of achieving a flexible and parsimonious representation of the mean of a variable, conditional on a set of variables deemed of interest for the phenomenon at hand. Our approach may prove useful in investigating phenomena where one can think of a (potentially large) list of variables that may be informative in describing the conditional mean (behavior) of a variable, but where one does not have strong priors as to the form of the relevant function, or of the relevance of individual variables for the data at hand. The procedure may be even more useful when the data generating process involves a relatively small number of relatively large parameters. Customer credit scoring and demand for telecommunication services by firms using individual data are two examples of such contexts.

Our approach, called the Relevant Transformation of the Inputs Network Approach (RETINA) is based on earlier work by White (1998). It has the flexibility of neural network models in that it accommodates non linearities and interaction effects (through non linear transformations of the potentially useful variables in the conditioning set), the concavity of the likelihood in the weights of the usual linear models (which avoids numerical complexity in estimation), and the ability to straightforwardly identify a set of attributes that are likely to be truly valuable for predicting performance evaluation outcomes (which corresponds to a principle of parsimony). Moreover, it is computationally not demanding and has good finite sample properties. When compared to the related method of White (1998), RETINA has higher rates of success and better finite sample properties at a slightly higher computational cost.

In selecting the relevant inputs, the approach has certain elements of similarity to the subset regression literature in statistics (Miller, 1990). To highlight the differences, we will report the results of comparisons of RETINA to some of these methods, namely, stepwise regression – where some variables are eliminated according to the significance of their parameters – and, the non-negative Garrote – where some parameters are set to zero while others are shrunk toward zero (Breiman, 1995).

In performing model selection, our approach relies on a cross-validation scheme which is aimed at limiting the possibilities that good performance is due to sheer luck. In particular, we divide the observations into three homogeneous sub-samples and implement a selection procedure in which possible models are selected in the first sub-sample. After a "candidate" is selected, a similar procedure is performed, this time estimating various models derived from the "candidate" in the second sub-sample and cross-validating them on the third sub-sample by means of an information criterion. We do not provide a theoretical justification for the division of the overall sample in three sub-samples or for the choice of two different measures for evaluation purposes, except heuristically: the evidence of good performance of the procedure in the various simulations we performed. While we are sure that worse choices exist, the existence of better criteria for specific types of DGP's which would further improve the procedure is an open question.

There are several issues that our approach does not address: first and foremost, we are aware of the fact that any model specification exercise is intended to be one of finding a good approximation according to some criteria and not one of finding the "true" model. Accordingly, we are not addressing the issue of finding the "true" model. Second, in our simulations we treat the case in which the data generating process (DGP) is i.i.d., although extensions to heterogeneous and/or dependent processes (including the important case of non stationary variables) along these lines can be envisaged. Third, since we are not concerned about retrieving the form of the function linking the variables in the conditioning set to the dependent variable**.**

This approach, also does not solve the problem of choosing which class of models is best-suited to represent certain data (e.g., in a time series context, linear, bilinear, ARCH, Threshold Autoregressive, and so on) assuming that the true DGP is among them.

The structure of the paper is as follows: in section 2 we define RETINA and we justify the adopted steps in the procedure. In section 3 we give a brief description of the main differences with similar existing approaches. Section 4 contains a description of the simulation design where we consider a number of situations in which the DGP may contain elements of noise for model selection (presence of outliers, of structural breaks, of sparse data, etc.). We limit the presentation of the results to a few leading cases, transferring the main bulk of the detailed results to an Appendix. Concluding remarks follow.

2. The RETINA procedure.

As mentioned in the introduction, the method presented here, RElevant Transformation of the Inputs Network Approach (RETINA) shares some characteristics of earlier work by White (1998), in that it has the flexibility of neural network models, the computational simplicity of the usual likelihood-based methods for which the likelihood function is concave in the weights, and the ability to straightforwardly identify a parsimonious set of attributes that are likely to be truly valuable for predicting outcomes by way of model selection criterion based on a cross-validation scheme. The “relevant-input” network model described below is not computationally demanding and has good finite sample properties. RETINA has higher rates of success and better finite sample properties at a slightly higher computational cost than the original proposal by White (1998).

Let us start by considering a (potentially large) number of variables *X* of potential relevance in a relationship describing the behavior of the mean of a dependent variable *Y*. Given a lack of information on the form of such a relationship (as is common in economics), in order to maintain a certain degree of flexibility one may want to use a nonlinear transformation (e.g. squares, ratios, cross-products, etc.) of the input variables, say $\zeta(X)$*.* In considering these transformations, we will keep in mind our goal of identifying a parsimonious set of (transformed) attributes that are likely to be truly relevant for predicting out-of-sample outcomes for Y. Hence, we need to be careful that the transformations we choose are not highly correlated with one another, as highly correlated transforms will not provide a great deal of independent predictive information.

Concavity of the likelihood in the parameters can be achieved by allowing the effects of the $\zeta(X)$'s on Y to be exerted in a linear fashion, providing a model of the form:

$$E(Y/X) \approx \zeta(X)'\beta$$

in the regression case or, more generally

$$E(Y/X) \approx F(\zeta(X)'\beta)$$

where F is a suitable link function (e.g. the logistic cdf for binary classification problems). We will rule out the appearance of further parameters inside ζ because that may result in non-concavity.

An important feature of White (1998) which we will exploit here is to avoid the evaluation of all the 2^m possible models when we have m candidate regressors in the set of transformed variables $\zeta(X)$, and then applying some form of model selection. Rather, in the present formulation, the approach envisages the selection of a number (of order proportional to m) of candidate models, inserting new explanatory variables on the basis of their relevance for the problem at hand (for instance, ranking the candidate regressors according to their correlation in absolute value with the dependent variable). At the same time the degree of dependency of the new information added is controlled for by keeping the amount of collinearity among the regressors under a threshold parameter λ chosen by the experimenter (as λ approaches 0 new regressors approach orthogonality; as λ approaches 1 new regressors may be highly collinear).

As with all flexible modelling, the issue then becomes one of not favoring the model which performs the best in-sample, in order to avoid overparameterization. Thus, an important features of such procedures is the use of disjoint sub-samples for cross-validation and of an out-of-sample forecastability criterion for model selection. We achieve these goals essentially in two phases: in the first phase we select relevant inputs by examining a number of candidate models (according to a procedure which we will detail below), estimate them in a first sub-sample and cross-validating them on a second sub-sample. The outcome is considered a possible "best model". In the second phase, we explore whether it is possible to achieve a more parsimonious representation by deleting the selected inputs one by one starting from the last included variable and judging the performance of these sub-models by cross-validating them on a third sub-sample. The outcome of this phase is considered the best model to retain.

The data generating process may not be within the class of models considered by the researcher. However s/he may use a parametric model of some aspect of the phenomenon that is the “preferred model”, a useful approximation for a particular purpose, e.g., estimation of a conditional mean, hypothesis testing or out of sample forecasting.

The “preferred model”, in general, may be different depending on its intended use and the data available. For example, economists some times use different models of consumption depending on whether they wish to estimate a given parameter, choosing among competing theories, or to obtain out-of-sample forecasts. The type of data available (e.g., cross section, time series or panel) or the level of aggregation (e.g., individual, family, city, region or country) are also relevant elements influencing the choice of the model.

RETINA´s recommended model should be taken as a suggestion for a useful approximation to some unknown relationship. It is the researcher’s responsibility to assess its coherence and the rationale for the suggested transformations. S/he can add variables, delete others, or introduce restrictions based on prior knowledge, or theoretical or empirical considerations.

We now describe the steps to be followed more in detail. Assume that for each observation i (i=1, …, n) we observe a value of the response variable Y_i and we have available candidate predictor attributes X_{ih}, h = 1, …, k, where k is potentially a very large number. From the original attributes X_{ij} we can form a collection W_{ij}, j = 1, …, m of transformations of the original attributes by including the original X_{ij}´s (which we refer to as the “level 0 transforms”), their squares and cross products, and their inverses and cross-ratios (taking care to avoid perfect multicollinearity and divisions by zero). We call the result of this transformations the “level 1 transforms”. If desired, a set of “level 2 transforms” can be calculated by appending the level 1 transforms to the original level 0 transforms and then performing level 1 transforms again. The process can be continued to any desired level, but to keep things simple, let us limit ourselves to the level 1 transforms: that is transforms of the form $X_{ih}{}^{\alpha} X_{ij}{}^{\beta}$, α, β = -1, 0, 1, for simplicity, but there are other choices possible (square roots, logarithms, linear combinations, etc.). These transforms W_{ij} fulfill our requirements of providing a rich set of univariate predictors that embody both nonlinearities and interactions.

To avoid evaluation of all 2^m possible models, we seek to extract a relatively parsimonious subset that may provide a useful basis for predicting Y_i . Given that we need to avoid overparameterization and that predictive performance is the selection criterion, we divide the entire sample into three sub-samples. We want these to be disjoint so that the information and the statistics we compute are independent across samples, and we would like them to be as similar to one another as possible to limit the dangers of unaccounted for heterogeneity. Choosing three sub-samples allows us to refine the process of model selection as discussed below.

We can use the observations in sub-sample 1 to provide a ranking among the candidate predictors W_j's according to (the absolute value of) a relevance measure of the relationship between Y and W_j, for example the sample correlation $\hat{\rho}_j$, j = 1, …, m. Let us denote these ranked W_j's as $W_{(j)}$, j = 1, …, m, where $W_{(1)}$ has the highest (absolute) relevance with Y in subsample 1 and $W_{(m)}$ the lowest.

The next step is to create a candidate subset of predictors. Apart from a constant term, the first variable to be included is $W_{(1)}$; the list of candidates becomes longer by proceeding through the list of $W_{(j)}$, and by including in the subset of predictors any $W_{(j)}$ so long as the new variable is likely to add relevant information that is not included in the subset. One way to achieve this is to select a $W_{(j)}$ only if the R^2 of the regression of that $W_{(j)}$ on the current subset is below λ, where λ is a prespecified threshold value, $0 \leq \lambda \leq 1$, which controls for the level of collinearity among candidate predictors. When all variables have been checked, we will have a candidate set of predictors which is a function of the specific value choose for λ, say λ_1: we will denote such a set of transforms as $\zeta_1(X)$. By making the selection depend on λ, we control the correlation among predictors: thus, if we repeat our candidate subset of predictors selection process for a grid of values for λ, say $\lambda_1, ..., \lambda_\nu$, we obtain corresponding candidate transformations $\zeta_p(X)$, $p = 1, …, \nu$.

We want estimate the models corresponding to the different $\zeta_p(X)$, $p = 1, …, \nu$, in sub-sample 1 and compute an out-of-sample prediction criterion (e.g.: the cross-validated mean square prediction error) by using the observations in sub-sample 2. We will choose as the candidate "best model" (corresponding to λ_p and involving m_p variables) the one optimizing this criterion in sub-sample 2. This completes the first phase.

As mentioned earlier, the second phase starts with a search over other possible specifications involving the same set of variables which may lead to a more parsimonious model. Among various possibilities, for a given value of λ_p we choose the simplest one by proceeding in a step-wise fashion, estimating m_p sub-models in sub-sample 1, the first sub-model of which includes only the first element of ζ_p, $W_{(1)}$, the second including the first two elements of ζ_p, and so on. For each sub-model estimated, we compute the forecast performance criterion in sub-sample 3, and we choose the sub-model with the best value. This criterion is then compared across different values of λ to select λ^*, the best choice for

λ among $\lambda_1, \ldots ,\lambda_v$. In practice we may want to repeat this last step of the procedure by reordering the elements of ζ_p by their univariate correlation measure with Y, computed this time in sub-sample 2. This may allow the consideration of a wider range of candidate models.

The procedure just described can be repeated changing the order of the sub-samples (with a choice of three sub-samples this would involve a total of six repetitions). At this point, we may have more than one candidate model. The recommended model (and optimal λ) could be the one that has the best performance in an appropriately defined sense using the whole sample.

We are now in a position to provide some further comments and justification for the choice of three sub-samples as opposed to the customary two, when using cross-validation techniques. We use the first sub-sample for selecting the variables which are most promising in terms of their relationship with the target *Y* and controlling for the degree of collinearity among variables. The models are selected in sub-sample 1 and cross-validated and estimated in sub-sample 2. We keep in mind Miller's (1990) result that in sub-sample 1 parameters and standard deviations are biased away from zero, and therefore we estimate parameters again in the second sub-sample. It is advisable therefore to perform the cross validation once work, by using unused information included in the third sub-sample. In view of this, using more than three sub-samples would seem to have diminishing returns. Code for running RETINA can be downloaded from www.ds.unifi.it/fedra.

3. Related approaches.

RETINA has features similar to other methods in the literature: it is worth pointing out some similarities and differences between RETINA and other model building and selection approaches such as neural networks (White, 1989), stepwise regression (Miller, 1990), the London School of Economics methodology (Hoover and Pérez, 2000) and the non negative garrote (Breiman, 1995). We also comment on the relationship with other types of models such as generalised linear models and generalised additive models, e.g., Hastie and Tibshirani (1990). We will not comment on other model building methods such as ACE, Breiman and Friedman (1985) and CART, Breiman et al. (1984) and Chipman et al. (1998), because they are not as closely related to RETINA.

RETINA has some common features with artificial neural networks (White, 1989), such as flexibility, which is gained here by using nonlinear transformations of the inputs, while maintaining linearity in the parameters within the link function. At the same time, it is designed to overcome some of the drawbacks of neural networks models such as the presence of nonlinearities in the parameters, which makes estimation cumbersome. With respect to the objective function, RETINA uses an out-of- sample predictive criteria, while neural nets, often use an in-sample goodness of fit criterion.

RETINA has features in common also with stepwise regression (Miller, 1990), such as the ability to search for a subset of relevant regressors, in a non-exhaustive fashion. In particular, RETINA performs a selective search guided by a saliency feature of the regressors. In this regard we should stress that the main difference between the two procedures is the use of out-of-sample (RETINA) versus in-sample model selection criterion.

The London School of Economics (LSE, Hoover and Pérez, 2000), "general-to-specific" approach to model building and selection starts with a reasonably general specification of a model and through parameter and residual tests, selects a parsimonious model that is intended to adequately represent the relationship under consideration. RETINA can be considered a general-to-more general-to-specific methodology. First, it expands the range of possible regressors by including the transformations of the inputs, then it considers models that include both the inputs and their transforms, then it narrows the search to the most promising models using a selective search criterion (saliency). An important difference is that the LSE methodology uses in-sample hypothesis tests while RETINA uses out-of-sample predictive criteria.

An alternative model building and selection approach is the non-negative garrote (Breiman, 1995). this is a method for doing subset regression. It starts with a linear regression including all the possible explanatory variables and selects subsets by zeroing and/or shrinking coefficient estimates. It works well in experimental data compared to subset selection when there is a small number of large coefficients. The non-negative garrote uses cross validation in model selection but does not take into explicit consideration transformations of the original inputs.

Generalised linear models and generalised additive models (Hastie and Tibshirani, 1990) are linear in the parameters. Like our procedure, they can incorporate nonlinear link functions. Since RETINA also considers models that involve interactions of the original inputs, the models considered by RETINA are broader than generalised linear and generalised additive models. Further enrichment of RETINA to consider other assumptions for the distribution of the error terms can be easily devised.

As is apparent, many ingredients of RETINA are already present in the literature. Some have well established roots, like those in the generalised linear models, the out-of-sample forecasting criteria, and the selective search. The use of the λ parameter for controlling collinearity, this particular saliency feature, and the partition into three subsamples may be less common. All of them are simple and have intuitive appeal.

4. Simulations.

Because analytic results are difficult to come by in this area, the major proving ground is testing the procedure on simulated data. Here we explore the capabilities of RETINA to select a model corresponding to the DGP and compare its performance with the results achievable by backward stepwise regression (Miller, 1990; since it is a widely used model selection procedure) and the non-negative garrote (Breiman, 1995). The reasons for the latter comparison are that the non-negative garrote procedure is focused on the forecasting ability of the model, it is more stable than subset regression and it is superior both to subset selection and ridge regression when the number of relevant regressors is small (like in the present situation). However, the non-negative garrote is computationally more demanding than our procedure.

RETINA is designed to select the model that forecasts better out of sample. Asking it also to correctly identify the DGP when it is within the range of models considered may be asking too much, but even so, it performs quite well vis a vis its competitors. Here we investigate how well RETINA selects the right model along the following directions:

1. The frequency by which RETINA chooses a model that coincides with the data generation process (DGP) when the variables in the DGP are among the candidate predictors. We consider two cases, one of linearity in the X´s and one of nonlinearity in the X´s;
2. The impact on the procedure performance when the DGP includes:
    a. discrete explanatory variables;
    b. explanatory variables with sparse data;
3. The sensitivity of the procedure to the presence
    a. of outliers in the DGP;

   b. of a structural break in the DGP;

4. The impact of the refinement of the grid for λ (the parameter that controls for collinearity) on accuracy and computation time.

4.1. Design of the experiments.

The data were generated using the data generation processes (DGP):

DGP1 (linear): $y_i = \alpha_0 + \alpha_1 x_{1i} + \alpha_2 x_{2i} + \delta u_i,$ $i = 1, \ldots, n,$

where $\alpha_0 = \alpha_1 = \alpha_2 = 1$, x_{1i} and x_{2i} are jointly normal with correlations $\rho = 0.5$ or 0.9. The error term u_i is iid N(0,1), δ is a parameter that controls the ratio between variance of the dependent variable and variance of the error term so as to allow values of the R^2 for estimations in the experiment to be on average around 0.75, 0.50 and 0.25 respectively. The sample sizes are n=100, 200 and 1000. Other sample sizes were used with results consistent with the ones reported here. The number of replications for each run of each experiment was 1000.

DGP2 (ratio): $y_i = \alpha_0 + \alpha_1 x_{1i} / x_{2i} + \delta u_{i,}$ $i = 1, \ldots, n,$

where the second term depends on the ratio x_{1i} / x_{2i} and everything else is as in DGP1 except that $\rho = 0.5$ only. Analogously, a different type of nonlinearity is

DGP3 (product): $y_i = \alpha_0 + \alpha_1 x_{1i} x_{2i} + \delta u_{i,}$ $i = 1, \ldots, n,$

For the case of a discrete explanatory variable we use the same setup and parameter values as DGP1 except that now x_{1i}' is a dummy variable that takes the value 1 with probability 0.5 and 0 otherwise:

DGP4 (dummy): $y_i = \alpha_0 + \alpha_1 x_{1i}' + \alpha_2 x_{2i} + \delta u_{i\,.}$ $i = 1, \ldots, n,$

For the case of sparse data we use the same set up as in DGP1, except that x_{1i}'' is iid N(0,1) with probability 0.2 and zero otherwise.

DGP5 (sparse data): $y_i = \alpha_0 + \alpha_1 x_{1i}'' + \alpha_2 x_{2i} + \delta u_i$ $i = 1, \ldots, n,$

To check for the sensitivity to outliers we go back to DGP1 and use u_i', which is the same as u_i except that when the absolute value of u_i is larger than 1.96 standard deviations, it is multiplied by 5 (and alternatively 2.5 or 10). That is, we expect 5% outliers.

DGP6 (outliers): $y_i = \alpha_0 + \alpha_1 x_{1i} + \alpha_2 x_{2i} + \delta u_i'$ $i = 1, \ldots, n.$

We explore the ability of RETINA to recover the right regressors when a structural break in the parameters has occurred. The DGP is linear, as in DGP1, but now

DGP7 (struct. break): $y_i = \alpha_0 + \alpha^*_1 x_{1i} + \alpha^*_2 x_{2i} + \delta u_i,$ $i = 1, \ldots, n,$

where $\alpha^*_1 = \alpha^*_2 = 1$ for the first half of the sample and $\alpha^*_1 = 0.5$, $\alpha^*_2 = 2$ for the second half.

4.2. Results.

To perform our simulations we developed a program written in GAUSS. For Experiment 1, the data were generated using DGP1 with regressors jointly normal with correlation equal to ρ = 0.5. RETINA was used with the level one transforms of x_{1i} , x_{2i} and x_{3i}, and the constant, where x_{3i} is an irrelevant regressor which has the same distribution and correlations as above. The parameter λ varies from 0 to 1 by increments of 0.1. The maximum number of candidate regressors (W_j's) is 25, of which only three are relevant. The total number of possible candidate models to consider is 2^{24}, since the constant is always in the candidate model. RETINA, however, evaluates around 2x24 different candidate models.

We count a "success" when RETINA chooses a model which coincides with the DGP (a model that includes irrelevant variables is not a success). The percentages of successes and two standard deviations (represented by the vertical bars at each point) are displayed in Figure 1. This suggests that if we have either a large R^2 or sample size, the percentage of successes is close to 100%.

Figure 1. Dgp1, linear case, ρ=0.5

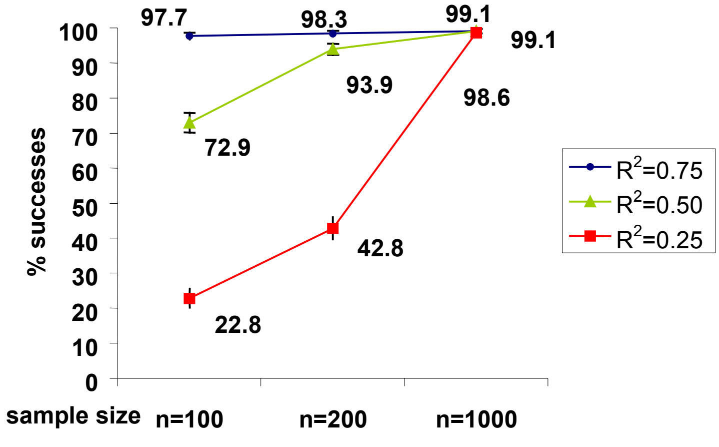


Table 1. Percentages of successes of RETINA for different DGPs and R^2 =0.5

	Sample size		
DGP	n = 100	n = 200	n = 1000
DGP1 linear	73	94	99
DGP2 ratio	73	82	95
DGP3 product	96	99	99
DGP4 dummy	43	73	96
DGP5 sparse data	-	33	93
DGP6 outliers	65	93	99
DGP7 struct. break	39	67	99

Table 1 present an overview of the results of the use of RETINA with different DGPs. For simplicity, we round up the percentages to the closest integer and show only the leading case of R^2 = 0.5. Other results and details are reported in the Appendix. RETINA works well when the DGP includes transformations of the inputs, discrete explanatory variables, sparse data, outliers or structural breaks. These results suggest that when the DGP is among the models considered by RETINA, there is a high probability, in some cases close to one, that it is recovered. The probability of success increases with the sample size. Figure 1 and further results shown in the Appendix also suggest that this probability increases with R^2.

Table 2. Percentages of successes of RETINA versus other procedures for DGP1 and R^2=0.5

R^2 = 0.5	Sample size		
DGP1	n = 100	n = 200	n = 1000
RETINA	73	94	99
Non negative garrote	5	11	54
Stepwise regression	9	8	9
Simpler RETINA	60	68	76

Table 2 shows the performance of RETINA *vis-à-vis* its competitors, such as the non-negative garrote, backward stepwise regression, and a simpler version of RETINA based just on two subsamples. The simulation results suggest that RETINA outperforms its rivals when the objective is to recover the DGP. Some further evidence is provided in the Appendix.

Finally, we find that the grid for λ, the parameter that controls collinearity among regressors included in the models to be evaluated, need not be too fine. In our experiments, using λ between 0 and 1 by increments of 0.1 works well and keeps the computational burden low. A finer grid does not deliver substantially better results which would overcome the additional computational costs entailed.

5. Concluding remarks.

A new method, an extension of White (1998), called relevant transformation of the inputs network approach, RETINA, is proposed for model building and selection. It is designed to have the flexibility of neural network models, the concavity of the likelihood in the weights of the usual linear models and the ability to identify a parsimonious set of attributes that are likely to be relevant for predicting performance evaluation outcomes.

RETINA may be a useful tool for model building and selection. It can be used as a data exploratory tool to suggest possible models and transformations of the explanatory variables.

One of the characteristics of RETINA is that it is designed as a modular procedure, which allows for the addition of features designed by the user. One can also substitute some of the ingredients, that is, for example, use further levels of transformations, or different types of transformations (we mentioned square roots and logarithms for positive-valued variables and linear combinations with arbitrary weights), a different saliency feature or cross-validation criteria.

One of its greatest advantages is that RETINA can reduce the search over a potential number of $2^m–1$ models to a modest multiple of m, where m is the total number of candidate regressors after the level one transforms are computed. For instance, when we have a constant and two varying inputs the potential number of models if we always include a constant is $2^{12} = 4096$; if the number of varying inputs becomes three we would have a number of potential models equal to a dazzling 16,777,216. Thus, a major advantage of the selective search suggested in RETINA (guided by a saliency feature) is to reduce the number of models actually evaluated.

RETINA allows for likelihood-type estimation techniques other than regression, and, with modifications, for the use of dependent observations. It may be more appropriate for cases in which there are a few large nonzero parameters, while other methods such as ridge regression may be more appropriate when there are many nonzero, but possibly small, parameters (Breiman, 1995).

The idea of splitting the sample in three disjoint sub-samples seems to be rewarding since the use of sub-sample 1 aims at an initial reduction of possible models to be evaluated, while sub-sample 2 is used for model selection and parameter re-estimation and sub-sample 3 for cross-validation purposes.

To assess the finite sample performance of RETINA we performed simulations in which we examine the ability of the procedure to recover the DGP. In general, the results are encouraging, except when the sample size is small or the variance of the error term is fairly big. Otherwise, RETINA seems to perform well for DGP's either linear or nonlinear in the inputs, and with dummies, sparse data, outliers, and structural breaks.

The procedure is computationally feasible on any desktop computer and the rates of success are better than some competing criteria, such as the non-negative garrote and backward stepwise regression. This suggests that RETINA can be useful for applied researchers. In the present context RETINA is applicable to independent identically distributed observations. It may be worth considering the applicability of RETINA to other types of data, such as time series data.

The search for the "true" model may be frustrated by the absence of the variables in the DGP from the set of transformed variables *W.* In this respect, one should always keep in mind that the goal is one of an adequate representation of the available data and that some form of approximation is unavoidable. One also cannot exclude cases in which the data are generated in a very peculiar fashion: in such a case a prediction-based criterion would signal the desirability of a specific model over other possible ones. The results that we have obtained here show that other model selection criteria are unlikely to perform better.

When working with real data, all sorts of problems may arise which we do not considered here: the presence of outliers beyond what we have assumed in our experiments, the possibility of error terms the distribution of which may have substantial departures from the normal distribution, the need to rebalance the order of magnitude of some variables to avoid numerical estimation problems, and the treatment of residual heterogeneity in the model which is not adequately captured by the relationship of the dependent variable with the transformed variables. Some preliminary investigations performed by Perez-Amaral and Marinucci (2002) show that some ad-hoc modifications of RETINA may be advisable, in line with the modular spirit of the procedure.

References.

Akaike, H. (1973) “Information Theory and an Extension of the Likelihood Principle”, in B. N. Petrov and F. Csaki (eds.), *Proceedings of the Second International Symposium on Information Theory*. Budapest: Akademiai Kiado, 267-281.

Breiman, L. (1992) “The Little Bootstrap and Other Methods for Dimensionality Selection in Regression: X-fixed Prediction Error,” *Journal of the American Statistical Association*, 87, 419, 738-754.

Breiman, L. (1995) “Better Subset Regression Using the Nonnegative Garrote”, *Technometrics*, Vol. 37, 4, 373-384.

Breiman, L. and H. Friedman (1985) “Estimating Optimal Transformations for Multiple Regression and Correlation” *Journal of the American Statistical Association*, 80, 580-619.

Breiman, L., Friedman, H., Olshen, R. and C. Stone (1984) *Classification and Regression Trees,* Wadsworh Statistics/Probability Series, Belmont, California.

Burnham, K. and D. Anderson (1998) *Model Selection and Inference: A Practical Information-Theoretic Approach*, Springer-Verlag, New York.

Chipman, H., E. George and R. McCulloch (1998) “Bayesian CART Model Search”, *Journal of the American Statistical Association*, Vol. 93, 935-960.

Diebold, Francis X., and Roberto S. Mariano, Comparing Predictive Accuracy, *Journal of Business and Economic Statistics*, v.13, no.3 (July 1995), pp. 253-63.

Giacomini, R. and H. White (2003) “Tests of conditional predictive ability”, Working Paper, UCSD.

Granger, C.W.J., M. King and H. White, (1995) , Comments on Testing Economic Theories and the Use of Model Selection Criteria, *Journal of Econometrics*, 67, 173-187.

Hastie, T. J. and R. J. Tibshirani (1990) *Generalized Additive Models*, Monographs on Statistics and Applied Probability 43, Chapman and Hall, London.

Hoover, K. and J. Perez (1999) “Data Mining Reconsidered: Encompassing and the General-to-Specific Approach to Specification Search”, *Econometrics Journal,* 2, p.

Miller, A. J. (1990) *Subset Selection in Regression*, Monographs on Statistics and Applied Probability 40, Chapman and Hall, London.

Shao, J. (1993) “Linear Model Selection by Cross-Validation”, *Journal of the American Statistical Association*, Vol. 88, No. 422, 486-494.

Shao, J. (1996) “Bootstrap Model Selection”, *Journal of the American Statistical Association*, Vol. 91, No. 434, 655-665.

White, H., (1989), Learning in artificial neural networks: a statistical perspective, Neural Computation, 1, pp. 425—464 (reprinted in H.White (1992) *Artificial Neural Networks: Approximation and Learning Theory* Oxford: Blackwell).

White, H. (1998) *Artificial Neural Network and Alternative Methods for Assessing Naval Readiness*. Technical Report, NRDA, San Diego.

White, H. (2000), “A Reality Check for Data Snooping,” *Econometrica*, 68, 1097-1126.

West, K. D. (1996). “Asymptotic inference about predictive ability”, *Econometrica*, 64, 1067–84.

Zhang, P. (1992) “On the Distributional Properties of Model Selection Criteria”, *Journal of the American Statistical Association*, Vol. 87, No. 419, 732-737.

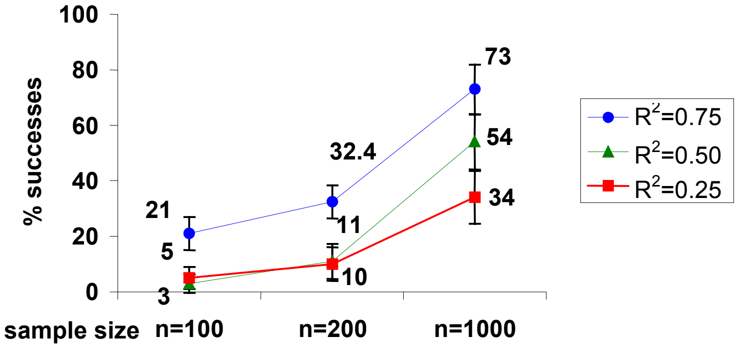
Appendix

This appendix presents the outcomes of the experiments in more detail.

1. DGP1. For comparison with RETINA, we present in Figure 2 the results of another experiment, with the same DGP and parameter values, but using Breiman's (1995) non negative garrote instead of RETINA. We use tenfold cross validation, make available to the non-negative garrote all the Wij's and set to zero those coefficients whose estimated absolute values are below 0.01.

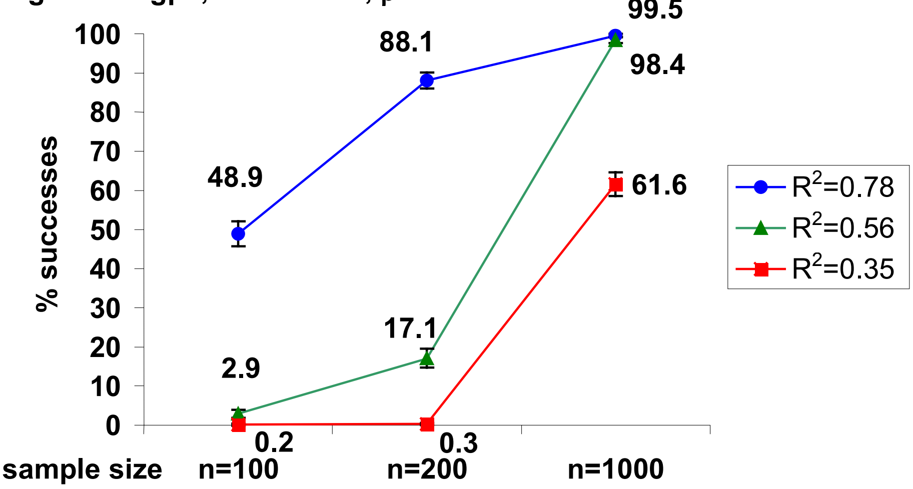
The rates of successes follow the same patterns as those of Figure 1, increasing with the sample size and the R^2**;** however, they are uniformly lower than those of RETINA. The execution time for the non-negative garrote is more than two hundred times that of RETINA, due to the complicated optimizations and the tenfold cross-validation used by this method (this is why the numbers of replications used for the non-negative garrote are smaller). We have also limited the maximum value of s (the garrote parameter) to 6 or 4 for faster convergence and execution. On these grounds, RETINA is superior to non-negative garrote as a model selection approach.

Figure 2. Dgp1, garrote, linear case, ρ=0.5, 100 repl.



In Experiment 2, we analyze the same DGP1 (including the same values of δ) as in Experiment 1. The main difference is that here we allow for more collinearity among the candidate regressors, ρ= 0.9, and therefore, the R^2's are higher than in Experiment 1. The results are summarized in Figure 3, which represents the percentage of successes for RETINA in recovering the regressors of the DGP. The vertical bars represent two standard deviations. The results seem good for high R^2 and large sample sizes, but are generally poor for samples of sizes 100 and 200. This suggests that collinearity can be challenging for model selection.

Figure 3. Dgp1, linear case, ρ= 0.9



In Experiment 3 we analyze the performance of RETINA when the data are generated in a nonlinear fashion by DGP2. The results are summarized in Figure 4. The rates of success are again reasonably high if the R^2 or the sample size are large enough.

In the next Experiment we use the multiplicative DGP3. The results are summarized in Figure 5. In this case RETINA works well in all cases except for n=100 and R^2=0.25.

Figure 4. Dgp2: ratio x1/x2

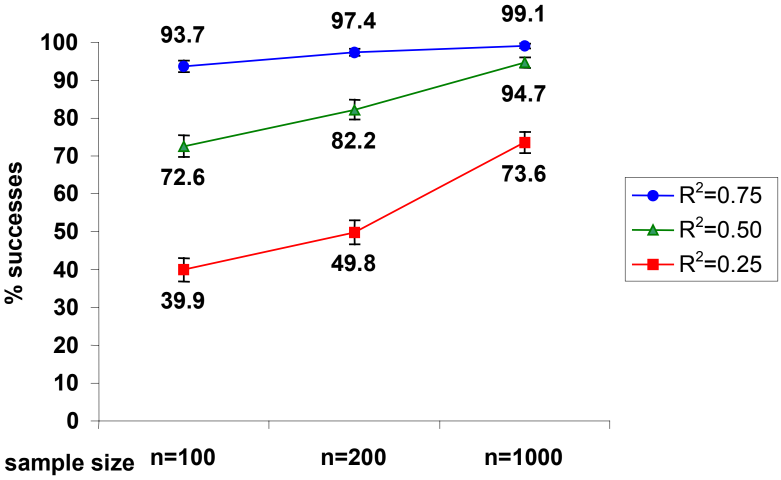
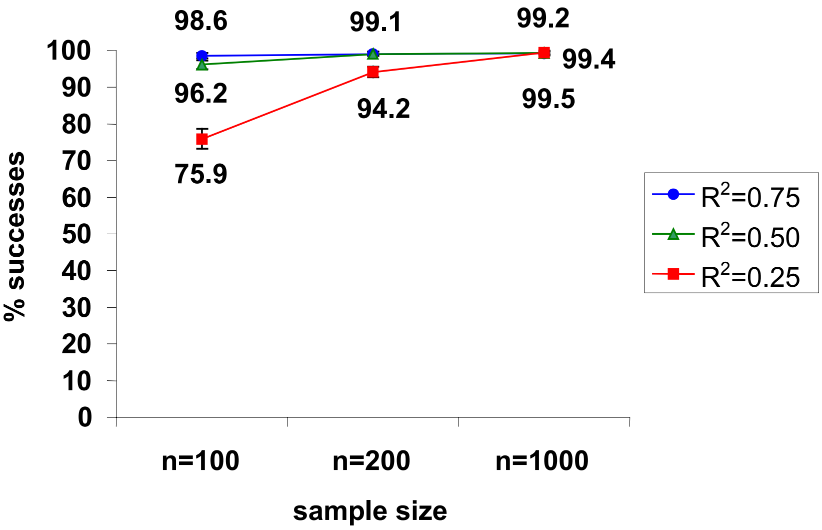


Figure 5. Dgp3, product x1*x2



In the next Experiment, with DGP4, one of the regressors is a dummy variable that randomly takes the value 1 or 0 with probability 0.5. The results are summarized in Figure 6. Here, again, a large sample or a high R^2 are required for the procedure to yield a reasonable percentage of successes.

Figure 6. Dgp4, linear with dummy

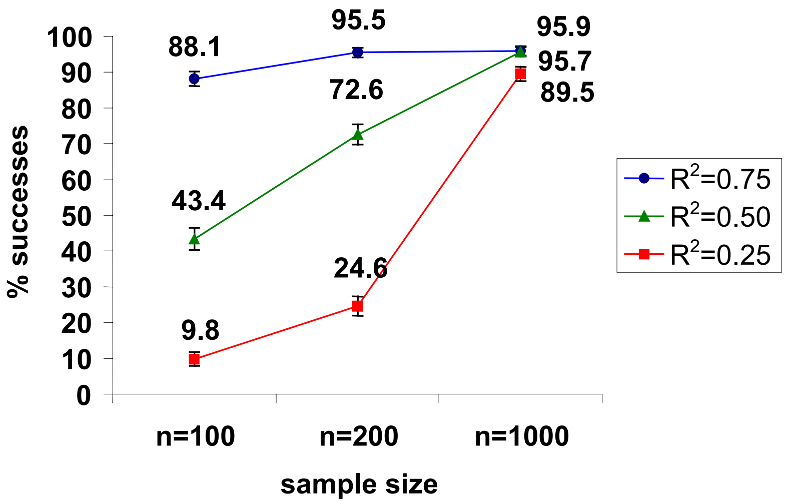


Figure 7. Dgp5, linear dgp, sparse data

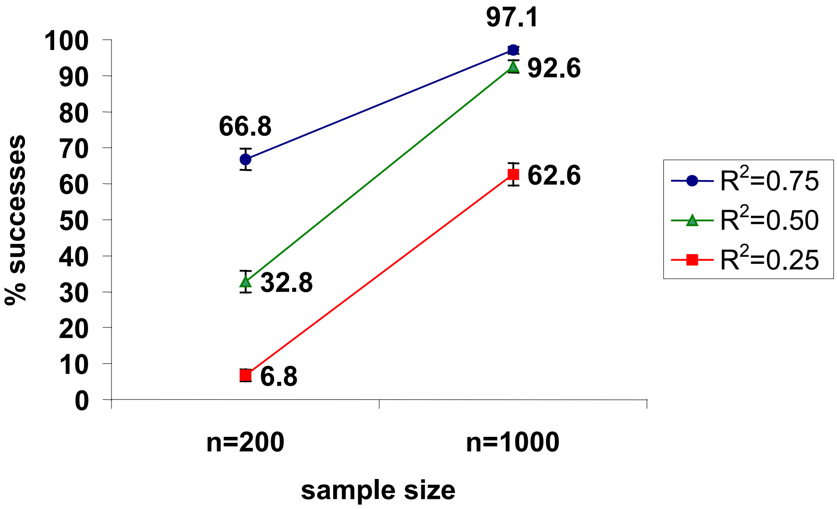
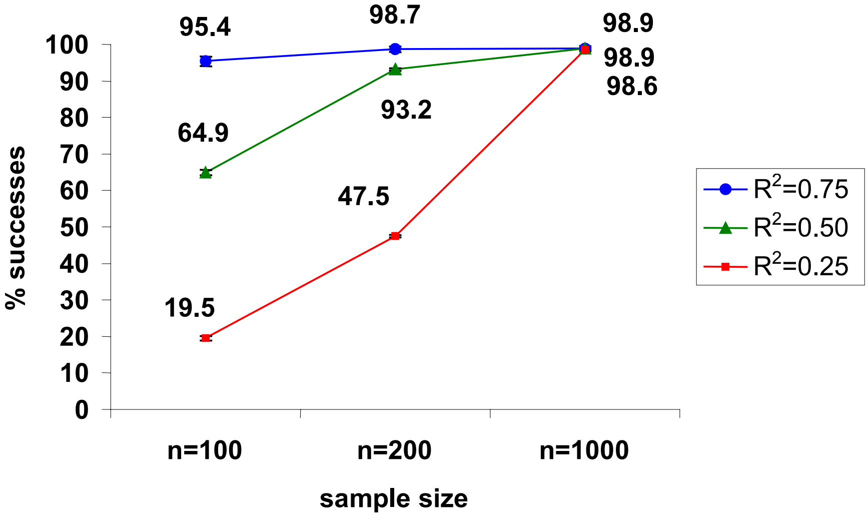


Figure 7 summarizes the results of applying RETINA to data generated by DGP5, that is, a linear model in which one of the regressors takes the value zero with probability 0.8 and is iid N(0,1) otherwise. This is the case of sparse data. Here, no samples of size 100 were drawn because, frequently, after splitting the sample in three, all the observations for one variable in one of the subsamples were zero. If the sample size is large enough and the R^2 not too low the procedure works reasonably well.

Figure 8 summarizes the results of the experiments that use dpg6 to generate the data. Here, the values of the error term greater in absolute value than 1.96 are multiplied times 5. In this case, we do not expect RETINA to recover the dpg; instead we consider a success when it correctly identifies the regressors of the DGP. Again, when the sample size is large enough or the R^2 is high enough, the procedure can recover the regressors of the DGP with high frequency. Otherwise, it may fail to do so.

Figure 8. Dgp4, linear with 5% outliers in error term



Next, we analyze how the presence of outliers affects the ability of RETINA to select the correct model. This is presented in Figure 9. Here we use the n=100, R^2=0.50 case and use different values of the parameter that multiplies each of the outliers. For the last case (X10), each realization of the error larger in absolute value than 1.96 is multiplied times 10. The baseline case of no outliers is X1, for which the values of the error are not altered. The results suggest that RETINA may be robust to the presence of outliers. However, the estimates of the coefficients and their precision are adversely affected.

Figure 9. Dgp4, linear, 5% outliers, varying their size

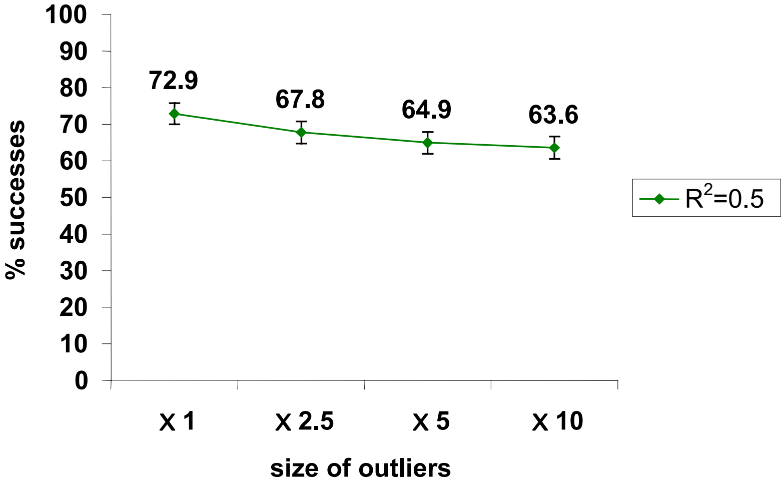
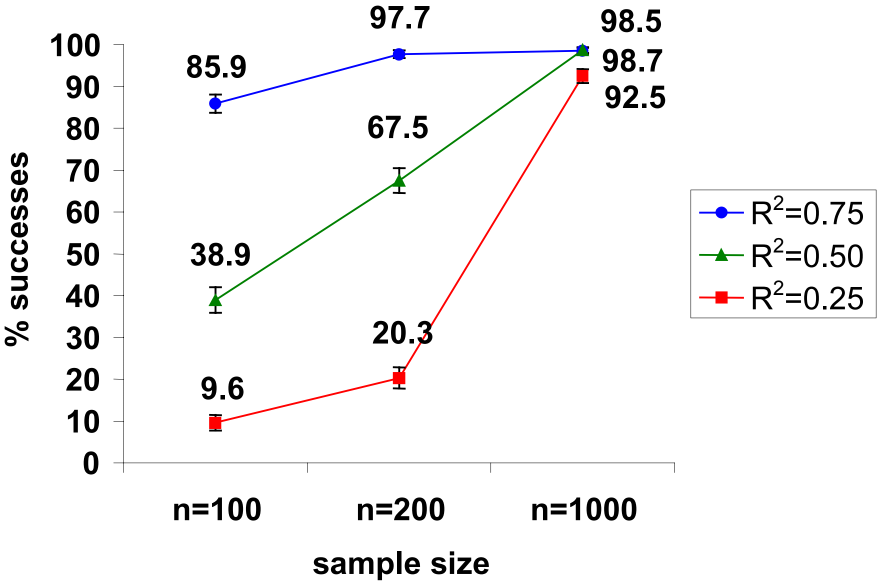


Figure 10 summarizes the results of RETINA when there is a structural break in the middle of the sample. In that case, it still achieves high rates of success for recovering the right regressors, except in the cases with low R^2 or relatively small samples. However it does not recover the DGP. Subsequent tests for structural change or residual analysis may detect the existence and location of the structural breaks.

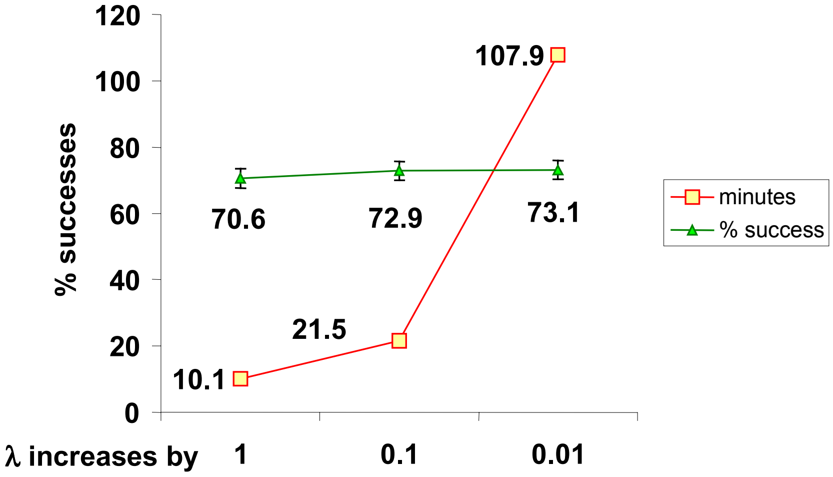
In our experiments, a reasonable grid for the parameter λ is between 0 and 1 by increments of 0.1. A finer grid increases the success rate marginally while the execution

Figure 10. Dgp7, linear, structural break



time increases substantially. On the other hand, a less fine grid decreases the success rate somewhat while saving little computer time. This is suggested by Figure 11, where we show the rates of success and the minutes of execution for each of these values for the λ grid, obtained by repeating one run of Experiment 1, with DGP1, for R^2=0.50 and n=100 varying the increments of λ between 1 (only zero or one are considered), 0.1 and 0.01.

Figure 11. What grid for λ? (R^2=0.5, n=100)



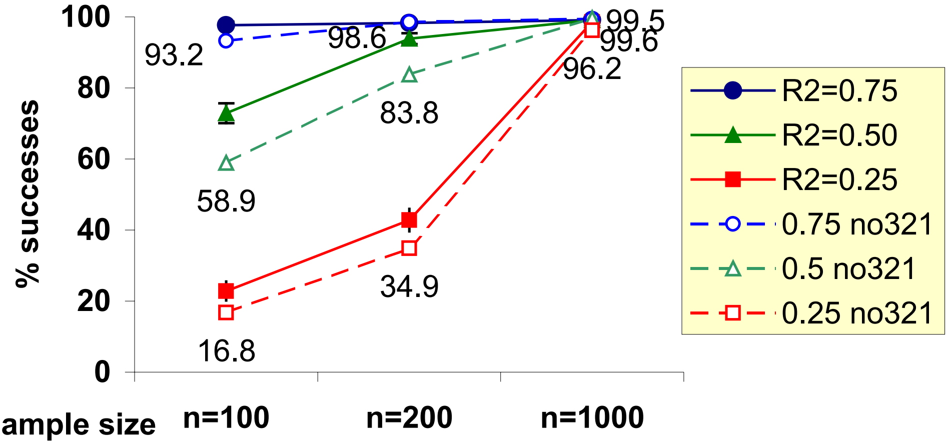
2. RETINA vs. simpler versions of RETINA.

In this section we compare RETINA with some simpler versions of RETINA. In Table 3 we compare RETINA and a simpler version of RETINA, in which we do not perform the resorting of the three subsamples and therefore do not repeat the model selection procedure with the subsamples resorted as 321.

Table 3: Dgp1, linear case ρ =0.5
RETINA without 321 resorting

	n=100	n=200	n=1000
R2=0.75	97.7	98.3	99.1
st dev	0.47	0.41	0.3
W/out 321	93.2	98.6	99.5
R2=0.50	72.9	93.9	99.1
st dev	1.41	0.76	0.3
W/out 321	58.9	83.8	99.6
R2=0.25	22.8	42.8	98.6
st dev	1.33	1.56	0.37
W/out 321	16.8	34.9	96.2

Figure 12. Dgp1, linear =0.5, RETINA w/o 321



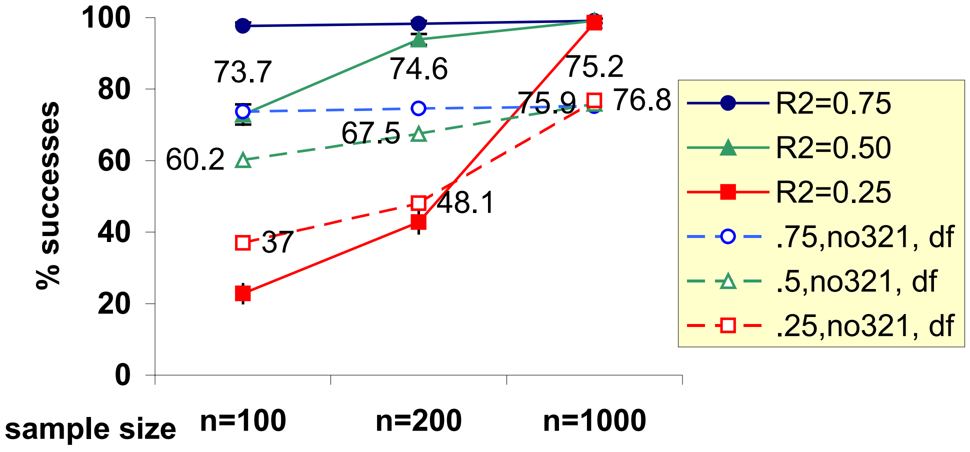
In Figure 12, lines in dashes correspond to RETINA without 321 resorting, while solid lines correspond to RETINA. The percentages of successes are generally better for RETINA, suggesting that the 321 resorting increases the ability to select the DGP especially when R^2 and n are not large.

2.1. In Table 4 we present the comparison of RETINA with a simpler version of RETINA which does not resort the subsamples as 321, and that does not use a degrees of freedom correction to compare the performance of the models that use different number of parameters.

Table 4: Dgp1, linear case ρ =0.5
RETINA without 321 resorting and
without dof correction

	n=100	n=200	n=1000
R2=0.75	97.7	98.3	99.1
st dev	0.47	0.41	0.3
no 321 and dof	73.7	74.6	75.2
R2=0.50	72.9	93.9	99.1
st dev	1.41	0.76	0.3
no 321 and dof	60.2	67.5	75.9
R2=0.25	22.8	42.8	98.6
st dev	1.33	1.56	0.37
no 321 and dof	37.0	48.1	76.8

Fig 13. Dgp1, linear =0.5, RETINA w/o 321, w/o dof



The elimination of the correction for degrees of freedom seems to hurt the performance of the procedure in most cases, especially when the R^2 is high or the sample size is large (cf Table 4 and Figure 13). When the sample size is large, the rates of success stabilize around 76% and do not approach 100%, as in RETINA.

However, when the R^2 is low, the rate of success of the procedure without degrees of freedom correction is larger than the one of RETINA. This suggests that in this case, the degrees of freedom correction may lead to underparameterization.

2.2 In Table 5 we present the first comparison of RETINA and a simpler version of RETINA corresponding to the procedure of White (1998), denoted “W98”.

Table 5: Dgp1, linear case ρ =0.5
RETINA – W98 comparison

	n=100	n=200	n=1000
R2=0.75	97.7	98.3	99.1
st dev	0.47	0.41	0.3
W98	43.1	40.6	39.3
R2=0.50	72.9	93.9	99.1
st dev	1.41	0.76	0.3
W98	41.9	43.8	43
R2=0.25	22.8	42.8	98.6
st dev	1.33	1.56	0.37
W98	37.2	42.4	47.3

Fig 14. Dgp1, linear ρ =0.5, RETINA vs W98

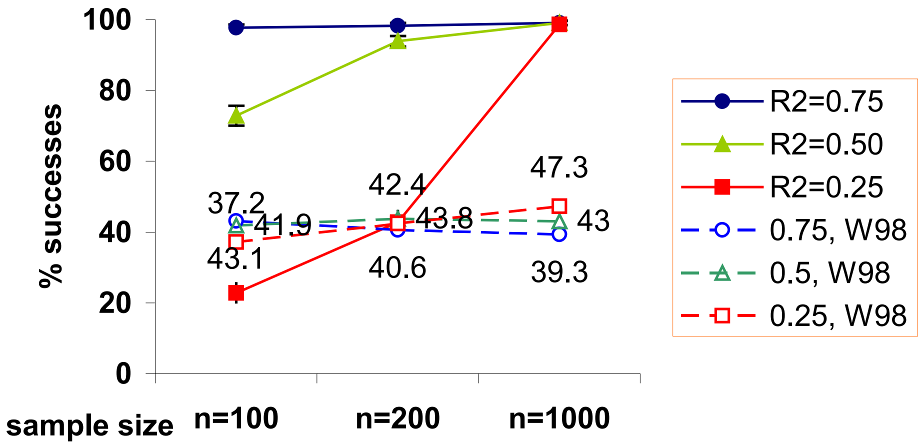


Table 6. Average number of excess regressors in overparameterized models by W98 in Table 5.
Dgp1, ρ=0.5.

	n=100	n=200	n=1000
R2=0.75	2.92	2.77	2.77
R2=0.50	2.93	2.64	2.67
R2=0.25	3.04	2.71	2.61

In this case, RETINA is better than W98 except for R^2=0.25 and n=100. The rates of success of W98 are low, do not approach 100 when n increases and for R^2=0.75 they decrease with n. The average number of excess regressors for W98 is between 2 and 3, which means that the model selected by W98 has on average around twice as many variables as the DGP. This shows no tendency to decrease with n or R^2.

2.3. RETINA vs. W98, linear case and ρ=0.9.

In this experiment the percentages of successes of W98 are often lower than those of RETINA and do not increase towards 100 with n (Table 7 and Figure 15). However, several are higher than RETINA, e.g. those for R^2=0.25 and 0.50 and n=100 and 200. This suggests the use of a criterion that employs a degrees of freedom correction less drastic than the one used by RETINA (Table 8).

Table 7: Dgp1, linear case ρ =0.9
RETINA – W98 comparison

	n=100	n=200	n=1000
R2=0.75	48.9	88.1	99.5
st dev	1.58	1.02	0.22
W98	38.6	41.9	48.5
R2=0.50	2.9	17.1	98.4
st dev	0.53	1.19	0.4
.5, RIPNET	27.8	33.4	49
R2=0.25	0.2	0.3	61.6
st dev	0.14	0.17	1.54
.25, RIPNET	18.4	23.2	45.2

Fig 15. Dgp1, linear ρ=0.9, RETINA vs W98

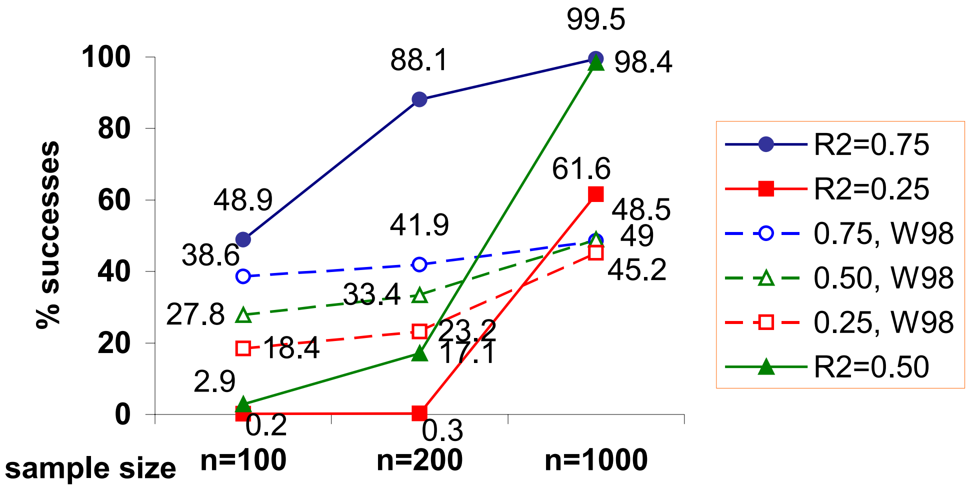


Table 8. Average number of excess regressors in overparameterized models by W98 in Table 7.
Dgp1, ρ=0.9.

	n=100	n=200	n=1000
R2=0.75	2.92	3.03	3.19
R2=0.50	1.82	2.96	2.98
R2=0.25	2.14	3.14	2.73

2.4. RETINA vs. W98 when the DGP includes the product x_1*x_2. In this experiment W98 is uniformly worse than RETINA. The percentages of successes of W98 are low and remain low for large n. The rates of successes do not increase and often decrease with n.

W98 overparameterizes substantially. The DGP has 2 variables and the average number of variables in the models selected by W98 is between 4.5 and 4.9 (cf. Table 9 and Figure 16, together with some diagnostics on overparameterization in Table 10).

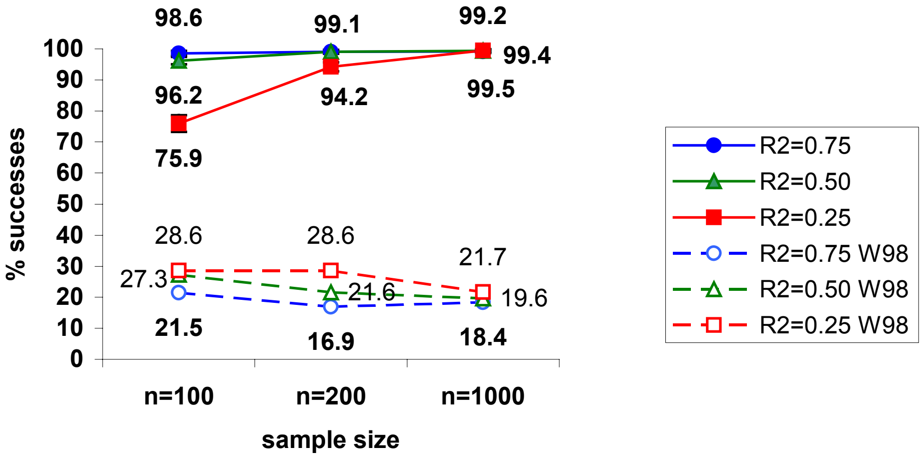
Table 9: Dgp3, product x1*x2
RETINA – W98 comparison

	n=100	n=200	n=1000
R2=0.75	98.6	99.1	99.2
st dev	0.37	0.3	0.28
W98	21.5	16.9	18.4
R2=0.50	96.2	99.1	99.4
st dev	0.6	0.3	0.24
W98	27.3	21.6	19.6
R2=0.25	75.9	94.2	99.5
st dev	1.35	0.74	0.22
W98	28.6	28.6	21.7

Table 10: Average number of excess regressors in overparameterized models by W98 in Table 9.
Dgp 3, x1*x2.

	n=100	n=200	n=1000
R2=0.75	2.65	2.73	2.89
R2=0.50	2.66	2.67	2.88
R2=0.25	2.9	2.58	2.87

Figure 16. Dgp3, product x1*x2, W98



2.5. RETINA vs. W98 with outliers.

In this experiment W98 is uniformly worse than RETINA, except for the case of R^2=.25 and n=100. The percentages of successes of W98 are low and remain low for large n. The rates of successes do not increase with n, and often decrease with n (Table 11 and Figure 17). In summary, RETINA is generally superior to W98, which has a strong tendency to overparameterize (Tab. 12).

Table 11: Dgp6, linear, 5% outliers in error term
RETINA – W98 comparison

	n=100	n=200	n=1000
R2=0.75	95.4	98.7	98.9
st dev	0.66	0.36	0.33
W98	46.7	42.3	39.8
R2=0.50	64.9	93.2	98.9
st dev	0.75	0.4	0.17
W98	44.7	45.9	42.2
R2=0.25	19.5	47.5	98.6
st dev	0.63	0.79	0.18
W98	29.7	43.4	21.7

Figure 17. Dgp6, linear with 5% outliers in u, RETINA and W98

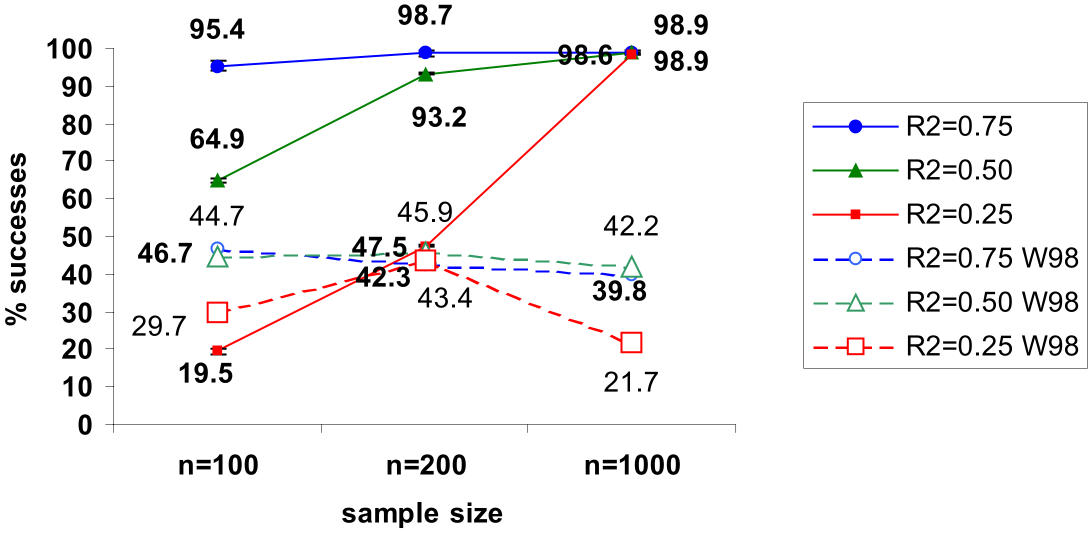


Table 12: Average number of excess regressors in overparameterized models by W98 in Table 11. Dgp 6, linear with 5% outliers in error term.

	n=100	n=200	n=1000
R2=0.75	2.93	3.14	3.21
R2=0.50	3.51	2.94	3.03
R2=0.25	3.68	3.05	2.87

3. RETINA vs. stepwise regression. In this section we compare the performance of RETINA with stepwise regression. Stepwise regression is a popular model selection technique, implemented in some commonly used software packages. We follow Miller (1990, p. 48). Stepwise regression is often used to mean an algorithm proposed by Efroymson (1960), which is a variation on forward selection. After each variable (other than the first) is added to the set of selected variables, a test is made to see if any of the previously selected variables can be deleted without appreciably increasing the residual sum of squares. Efroymson's algorithm incorporates criteria for the addition and deletion of variables as follows.

a. Addition Let RSS_p denote the residual sum of squares with p variables and a constant in the model. Suppose the smallest RSS which can be obtained by adding another variable to the present set is RSS_{p+1}. The ratio

$$R = \frac{RSS_p - RSS_{p+1}}{RSS_{p+1}/(n-p-2)}$$

is calculated and compared with and 'F-to-enter' value, say F_e. If R is greater than F_e the variable is added to the selected set.

b. Deletion With p variables and a constant in the selected subset, let RSS_{p-1} be the smallest RSS which can be obtained after deleting any variable from the previously selected variables. The ratio

$$R = \frac{RSS_{p-1} - RSS_p}{RSS_p/(n-p-1)}$$

is calculated and compared with an 'F-to-delete (or drop)' value, say F_d. If R is less than $F_{d,}$ the variable is deleted from the selected set.

The optimum F-to-enter for minimizing the mean square error of prediction in the case of random and correlated regressors is $F_e \leq 2n/n\text{-}p$, or a little less than 2 if n»p (Miller, p. 183). The F-to-delete statistic has a value not greater than 1 (Miller, p. 207).

3.1 Linear case with ρ=0.5.

First we compare the performance of RETINA with stepwise regression in the same setting as Experiment 1 using DGP1 with correlations between regressors ρ=0.5. The data are the same both for RETINA and stepwise regression. All the transformations of the original regressors (W's) of RETINA are made available to stepwise.

Table 13: Dgp1, linear, ρ=0.5
RETINA – Stepwise comparison

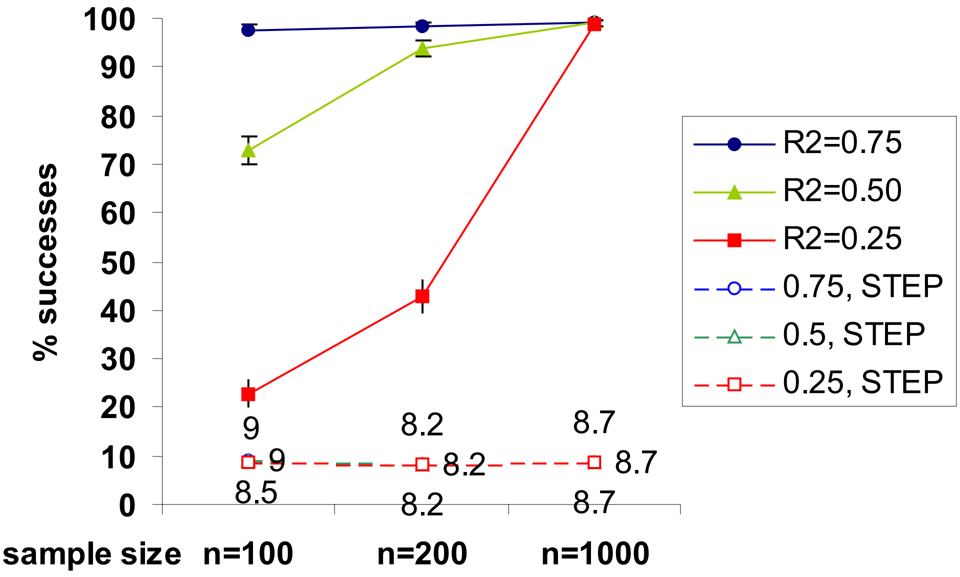
	n=100	n=200	n=1000
R2=0.75	97.7	98.3	99.1
st dev	0.47	0.41	0.3
Stepwise	9	8.2	8.7
R2=0.50	72.9	93.9	99.1
st dev	1.41	0.76	0.3
Stepwise	9	8.2	8.7
R2=0.25	22.8	42.8	98.6
st dev	1.33	1.56	0.37
Stepwise	8.5	8.2	8.7

Table 13 and Figure 18 summarize the percentages of successes for RETINA and stepwise regression. The solid lines join the points corresponding to RETINA and the broken lines link those of stepwise. The percentages of success of RETINA are considerably higher than those of stepwise regression across sample sizes and coefficients of determination. Stepwise regression overparameterizes around 90% of the time, choosing models that on average have around 2.7 more parameters than the original 3 of the DGP.

Table 14: Average number of excess regressors in overparameterized models by Stepwise in Table 13.
Dgp 1, linear with ρ=0.5

	n=100	n=200	n=1000
R2=0.75	2.62	2.69	2.67
R2=0.50	2.63	2.69	2.67
R2=0.25	2.78	2.69	2.67

Fig 18. Dgp1, linear ρ =0.5, RETINA vs STEPWISE



3.2 Linear case with ρ=0.9.

In this case we use DGP1 as before, but generate the x's with ρ=0.9. The results are summarized in Table 15 and Figure 19. The percentages of success are generally higher for RETINA than for stepwise regression except for the cases of low R^2 and samples of 100 or 200. In general, there is a strong tendency of stepwise towards overparameterization, which occurs between 24 and 85% of the time, with an average number of excess regressors in overparameterized models between 5.51 and 2.43 (Table 20).

Table 15: Dgp1, linear, ρ=0.9
RETINA – Stepwise comparison

	n=100	n=200	n=1000
R2=0.75	48.9	88.1	99.5
st dev	1.58	1.02	0.22
Stepwise	14.9	13.7	14.2
R2=0.50	2.9	17.1	98.4
st dev	0.53	1.19	0.4
Stepwise	10.5	12.5	14.2
R2=0.25	0.2	0.3	61.6
st dev	0.14	0.17	1.54
Stepwise	4	6.1	14.2

Fig 19. Dgp1, linear ρ=0.9, STEPWISE

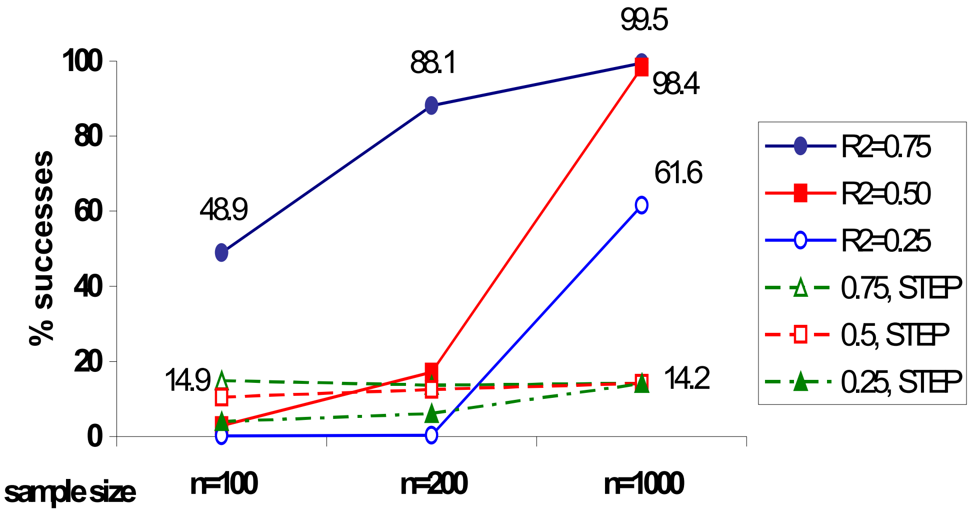


Table 16: Average number of excess regressors in overparameterized models by Stepwise in Table 15. Dgp 1, linear with ρ=0.9

	n=100	n=200	n=1000
R2=0.75	2.92	3.03	3.19
R2=0.50	1.82	2.96	2.98
R2=0.25	2.14	3.14	2.73

3.3 Nonlinear DGP3 with x_1*$x_{2.}$

In this experiment we compare the performance of RETINA and stepwise regression as model selection criteria. The data are generated by DGP3, that is a constant and the product of x_1 times $x_{2.}$ The results are summarized in Table 17 and Figure 20. The rates of success of stepwise regression are around 6.5 % while those of RETINA are always above 75%. Stepwise regression shows a strong tendency to overparameterize which occurs over 92% of the time (Table 18). When it overparameterizes, stepwise uses on average between 2.93 and 2.83 extra regressors when only 2 of them belong in DGP3.

Table 17: Dgp3, product, x1*x2
RETINA – Stepwise comparison

	n=100	n=200	n=1000
R2=0.75	98.6	99.1	99.2
st dev	0.37	0.3	0.28
Stepwise	21.5	16.9	18.4
R2=0.50	96.2	99.1	99.4
st dev	0.6	0.3	0.24
Stepwise	27.3	21.6	19.6
R2=0.25	75.9	94.2	99.5
st dev	1.35	0.74	0.22
Stepwise	28.6	28.6	21.7

Figure 20. Dgp3, x1*x2, RETINA vs STEPWISE

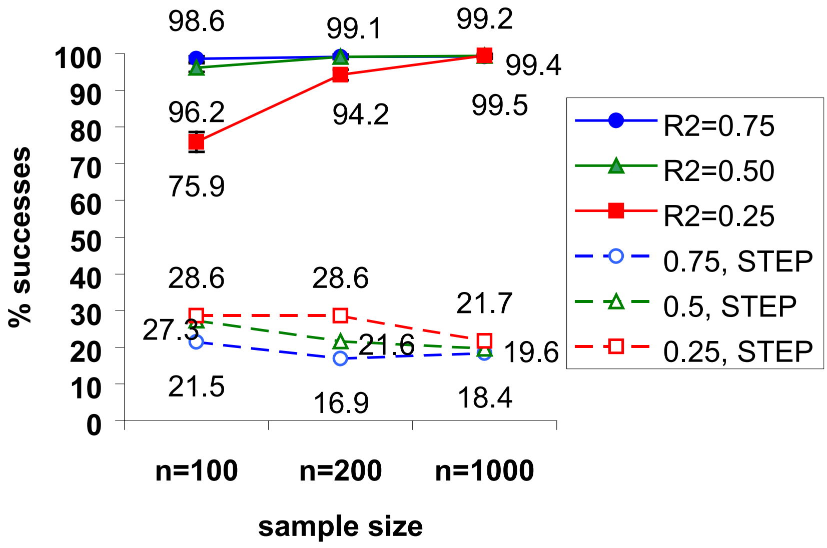


Table 18: Average number of excess regressors in overparameterized models by Stepwise in Table 17. Dgp 3, product, x1*x2

	n=100	n=200	n=1000
R2=0.75	2.65	2.73	2.89
R2=0.50	2.66	2.67	2.88
R2=0.25	2.9	2.58	2.87

3.4 RETINA vs. stepwise regression, DGP6, 5% outliers.

In this experiment we use DGP6, which incorporates outliers. Using exactly the same data for RETINA and for stepwise regression, we find that RETINA outperforms stepwise regression for all combinations of sample size and R^2, except for n=100 and R^2=0.25 (cf. Table 19 and Figure 21). The percentage of successes of stepwise is decreasing with n and does not improve with R^2. Stepwise consistently overparameterizes between 58 and 76% of the time, adding on average between 2.44 and 3.12 extra regressors (cf. Table 20).

Table 19: Dgp6, linear with 5% outlier in error RETINA – Stepwise comparison

	n=100	n=200	n=1000
R2=0.75	95.4	98.7	98.9
St dev	0.66	0.36	0.33
Stepwise	29.5	24.3	23.3
R2=0.50	64.9	93.2	98.9
St dev	0.75	0.4	0.17
Stepwise	29.1	24.3	23.3
R2=0.25	19.5	47.5	98.6
St dev	0.63	0.79	0.18
Stepwise	25.9	24.1	23.3

Figure 21. Dgp6, linear 5% outliers in u, RETINA vs STEPWISE

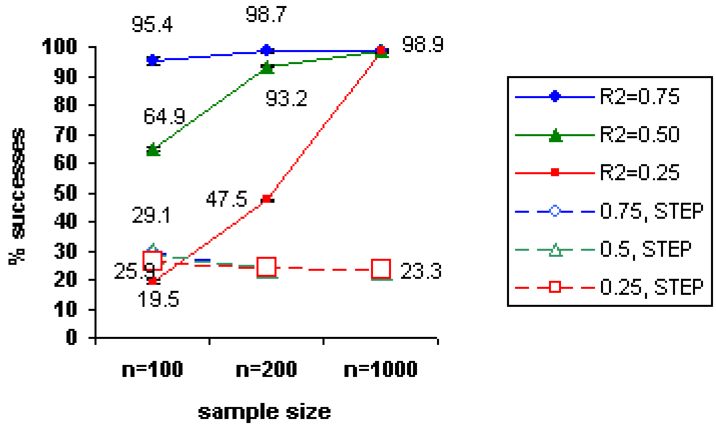


Table 20: Average number of excess regressors in overparameterized models by Stepwise in Table 19. Dgp 6, linear with 5% outliers in error term

	n=100	n=200	n=1000
R2=0.75	2.93	3.14	3.21
R2=0.50	3.51	2.94	3.03
R2=0.25	3.68	3.05	2.87

Copyright © 2003

Teodosio Perez-Amaral,

Giampiero M. Gallo, Hal White