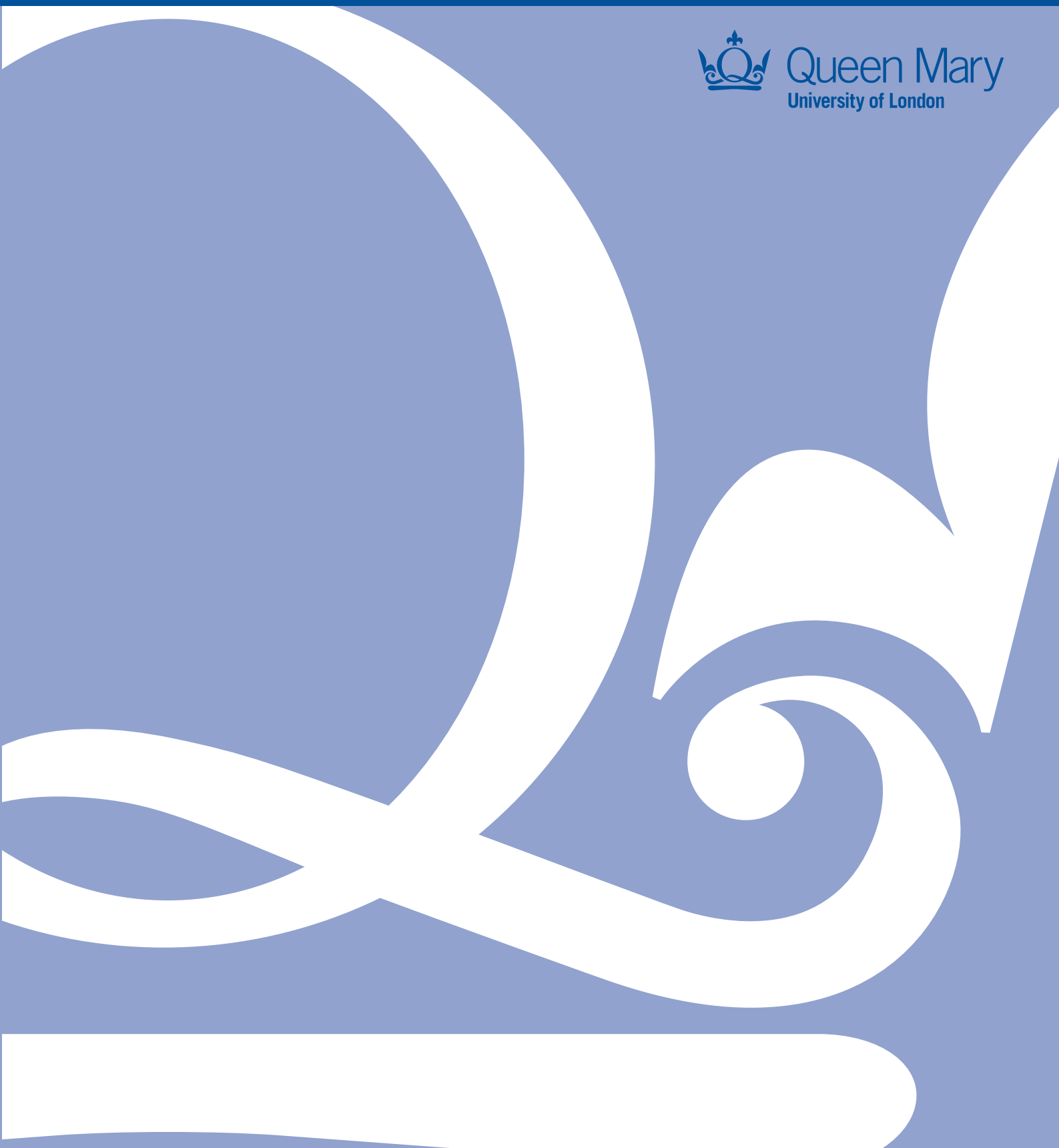
Department of Economics

Variable Selection Using Non-Standard Optimisation of Information Criteria

George Kapetanios

Working Paper No. 533 May 2005 ISSN 1473-0278



Variable Selection using Non-Standard Optimisation of Information Criteria

George Kapetanios*
Queen Mary, University of London

April 8, 2005

Abstract

The question of variable selection in a regression model is a major open research topic in econometrics. Traditionally two broad classes of methods have been used. One is sequential testing and the other is information criteria. The advent of large datasets used by institutions such as central banks has exacerbated this model selection problem. This paper provides a new solution in the context of information criteria. The solution rests on the judicious selection of a subset of models for consideration using nonstandard optimisation algorithms for information criterion minimisation. In particular, simulated annealing and genetic algorithms are considered. Both a Monte Carlo study and an empirical forecasting application to UK CPI infation suggest that the new methods are worthy of further consideration.
Keywords: Simulated Annealing, Genetic Algorithms, Information Criteria, Model Selection, Forecasting, Inflation
JEL: C110, C150, C530

1 Introduction

The question of variable selection in a regression model is a major open research topic in econometrics. Traditionally, model selection in regression models has been addressed using two broad classes of tools. The first such class is based on sequential testing. This idea underlies the widely used 'general-to-specific' approach, developed and popularised in a number of papers by David Hendry and his co-authors, such as Krolzig and Hendry (2001). Briefly summarised, this approach involves starting from a general dynamic statistical model which captures the characteristics of the data and via sequential testing reducing the complexity of this model while retaining the congruence of the resulting model.

*Department of Economics, Queen Mary, University of London, Mile End Road, London E1 4NS. email: G.Kapetanios@qmul.ac.uk

The second class of tools considers information theoretic ideas to model selection. Such ideas have given rise to a variety of tools generically known as information criteria. These criteria provide a score for every model considered which is a combination of the fit of the model and a penalty term for model complexity. The model that optimises the criterion is chosen to be the best representation of the data.

Recently, the problem of model selection has taken on increased significance due to the emergence and increased usage of large datasets. A major impetus for the use of large datasets, has been provided by work on forecasting carried out in institutions such as central banks who need macroeconomic forecasts to conduct monetary policy. In this context model selection faces two new challenges. Firstly, it faces competition from approaches that do not require an initial whittling down of the amount of available data, but are specifically designed to use all of them. One such approach which is increasingly popular is factor analysis (see Stock and Watson (2002)). The second challenge is perhaps the most obvious one. It relates to the fact that the performance of either sequential testing or information criterion optimisation will inevitably decline monotonically with respect to an increasing set of models.

This paper tries to address the problems that large datasets pose to information criterion optimisation. The most pressing problem is the sheer number of models that need to be evaluated as the number of available variables, denoted by N, increases. To state the problem starkly, the number of models that needs to be evaluated is equal to 2^N. Setting N to 30 or 40 indicates the extent of the problem. Of course such numbers of variables are commonly considered when statistical forecasting models are built.

The only possible solution to this problem is to evaluate information criteria only for a subset of the models in the model set. This paper suggests a possible answer to the selection of this subset. Insight into the solution we suggest may be obtained by viewing the problem as one of optimising a function (the information criterion) over a domain. Unfortunately, standard optimisation techniques cannot be applied since the domain is discrete. Nevertheless, there are techniques which can address this issue, such as simulated annealing and genetic algorithms. We investigate these in detail. Comparing the new method we suggest with a sequential testing alternative in a Monte Carlo study illustrates the potential of the new approach. Given the importance of good forecasting performance for selected models we also consider model selection for forecasting. In this case we do not optimise the penalised in sample fit but the out-of-sample forecast RMSE of the model during a short period prior

to the forecast period. Once again the new method outperforms the standard forecasting AR model.

The paper is organised as follows: Section 2 presents the main idea of the paper. Section 3 outlines the non-standard optimisation algorithms we consider. Section 4 presents a Monte Carlo study. Section 5 presents an empirical forecasting application. Finally, Section 6 concludes.

2 Theory

Let us consider the following regression model

$$y_t = \alpha + \beta^{0'} x_t^0 + \epsilon_t, \quad j = 1, \ldots, N, \quad t = 1, \ldots, T \tag{1}$$

where x_t^0 is a k-dimensional vector of predetermined variables. The superscript ^{0} denotes the true regression model. Denote, the set of all available variables at time t by $x_t = (x_{1,t}, \ldots x_{N,t})'$, where it is currently assumed that $x_t^0 \in x_t$. x_t is an N-dimensional vector. The aim of the analysis is to determine x_t^0. Formally, let $\boldsymbol{\mathcal{J}} = (\boldsymbol{\mathcal{J}}_1, \ldots, \boldsymbol{\mathcal{J}}_N)'$ denote a vector of zeros and ones (which we will refer to as string). Let $\boldsymbol{\mathcal{J}}^0$ be the string for which $\boldsymbol{\mathcal{J}}_i^0 = 1$, if $x_{i,t} \in x_t^0$ and zero otherwise. We wish to estimate $\boldsymbol{\mathcal{J}}^0$.

To do this we consider the use of information criteria to select the variables that go in (1). The generic form of such criteria is usually

$$IC(\boldsymbol{\mathcal{J}}) = -2L(\boldsymbol{\mathcal{J}}) + C_T(\boldsymbol{\mathcal{J}}) \tag{2}$$

where $L(\boldsymbol{\mathcal{J}})$ is the log-likelihood of the model associated with string $\boldsymbol{\mathcal{J}}$ and $C_T(\boldsymbol{\mathcal{J}})$ is the penalty term associated with the string $\boldsymbol{\mathcal{J}}$. The three most usual penalty terms are $2\tilde{m}(\boldsymbol{\mathcal{J}})$, $ln(T)\tilde{m}(\boldsymbol{\mathcal{J}})$ and $2ln(ln(T))\tilde{m}(\boldsymbol{\mathcal{J}})$ associated with the Akaike, Schwartz (Schwarz (1978)) and Hannan-Quinn (Hannan and Quinn (1979)) information criteria. $\tilde{m}(\boldsymbol{\mathcal{J}})$ is the number of free parameters associated with the modelling of the dataset associated with $\boldsymbol{\mathcal{J}}$. Note that, in this case, $\tilde{m}(\boldsymbol{\mathcal{J}}) = \boldsymbol{\mathcal{J}}'\boldsymbol{\mathcal{J}}$. It is straightforward under relatively weak conditions on $x_{j,t}$ and $\epsilon_{j,t}$, and using the results of say, Sin and White (1996), to show that the string which minimises $IC(.)$ will converge to $\boldsymbol{\mathcal{J}}^0$ with probability approaching one as $T \to \infty$ as long as $C_T(\boldsymbol{\mathcal{J}}) \to \infty$ and $C_T(\boldsymbol{\mathcal{J}})/T \to 0$.

More specifically, the assumptions needed for the results of Sin and White (1996) to hold are mild and can be summarised as follows, assuming estimation of the models is un-

dertaken in the context of Gaussian or pseudo maximum likelihood (which in the simplest case, of spherical errors, is equivalent to OLS): (i) Assumption A of Sin and White (1996) requires measurability, continuity and twice differentiability of the log-likelihood function and a standard identifiability assumption; (ii) A uniform weak law of large numbers for the log-likelihood of each observation and its second derivative; (iii) A central limit theorem for the first derivative of the log-likelihood of each observation. (ii) and (iii) above can be obtained by assuming, e.g., that $x_{j,t}$ are weakly dependent, say, near epoque dependent, processes and $\epsilon_{j,t}$ are martingale difference processes. Hence, it is clear that consistency of model selection as long as the penalty related conditions hold is straightforwardly obtained.

The problem is of course how to minimise the information criterion. For small dimensional x_t, evaluating the information criterion for all strings may be feasible, as, e.g., in lag order selection. In the case of lag selection the problem is made easier by the fact that there exists a natural ordering of the variables. But, in the general variable selection case, as soon as N exceeds say 30 or 40 units, this strategy is bound to fail. Since $\boldsymbol{\mathcal{J}}$ is a binary sequence there exist 2^N strings to be evaluated. For example, when $N = 50$ and optimistically assuming that 100000 strings can be evaluated per second, we still need about 357 years for an evaluation of all strings. Clearly this is infeasible.

We may alternatively treat this as a maximisation problem. Nevertheless, clearly standard maximisation algorithms do not apply due to the discreteness of the domain over which the objective function (information criterion) needs to be optimised. We resort to two powerful non-standard maximisation algorithm classes: simulated annealing and genetic algorithms. These are discussed in the next section.

3 Nonstandard Optimisation Algorithms

In the previous section we reviewed the translation of the problem of model selection to a problem of maximising an information criterion. On the one hand the space where the information criterion is defined is discrete and hence standard optimisation methods cannot be applied. On the other hand, standard grid search which is usually implemented to maximise the information criterion, as in, e.g., lag selection, is clearly infeasible due to the computational burden of the problem. One alternative is to resort to nonstandard optimisation algorithms that do not require neither smoothness nor continuity for the algorithm to

converge.

3.1 Simulated Annealing

Simulated annealing is a generic term used to refer to a family of powerful optimisation algorithms. In essence, it is a method that uses the objective function to create a non-homogeneous Markov chain that asymptotically converges to the optimum of the objective function. It is especially well suited for functions defined in discrete spaces like the information criteria considered here. Below, we give a description of the algorithm together with the necessary arguments that illustrate its validity in our context. We describe the operation of the algorithm when the domain of the function (information criterion) is the set of binary strings i.e. $\{\boldsymbol{\mathcal{J}} = (\boldsymbol{\mathcal{J}}_1, \ldots, \boldsymbol{\mathcal{J}}_N)' | \boldsymbol{\mathcal{J}}_i \in \{0, 1\}\}$.

Each step of the algorithm works as follows starting from an initial string $\boldsymbol{\mathcal{J}}_0$.

1. Using $\boldsymbol{\mathcal{J}}_i$ choose a neighboring string at random, denoted $\boldsymbol{\mathcal{J}}^*_{i+1}$. We discuss the definition of a neighborhood below.

2. If $\hat{S}(\boldsymbol{\mathcal{J}}_i) > \hat{S}(\boldsymbol{\mathcal{J}}^*_{i+1})$, set $\boldsymbol{\mathcal{J}}_{i+1} = \boldsymbol{\mathcal{J}}^*_{i+1}$. Else, set $\boldsymbol{\mathcal{J}}_{i+1} = \boldsymbol{\mathcal{J}}^*_{i+1}$ with probability $e^{-(\hat{S}(\boldsymbol{\mathcal{J}}^*_{i+1}) - \hat{S}(\boldsymbol{\mathcal{J}}_i))/T_i}$ or set $\boldsymbol{\mathcal{J}}_{i+1} = \boldsymbol{\mathcal{J}}_i$ with probability $1 - e^{-(\hat{S}(\boldsymbol{\mathcal{J}}^*_{i+1}) - \hat{S}(\boldsymbol{\mathcal{J}}_i))/T_i}$.

Heuristically, the term T_i gets smaller making it more difficult, as the algorithm proceeds, to choose a point that does not decrease $\hat{S}(.)$. The issue of the neighborhood is extremely relevant. What is the neighborhood? Intuitively, the neighborhood could be the set of strings that differ from the current string by one element of the string. But this may be too restrictive. We can allow the algorithm to choose at random, up to some maximum integer (say h), the number of string elements at which the string at steps i and $i+1$ will differ. So the neighborhood is all strings with up to h different bits from the current string. Another issue is when to stop the algorithm. There are a number of alternatives in the literature. We have chosen to stop the algorithm if it has not visited a string with lower $\hat{S}(.)$ than the current minimum for a prespecified number of steps (B_v) (Steps which stay at the same string do not count) or if the number of overall steps exceeds some other prespecified number (B_s). All strings visited by the algorithm are stored and the best chosen at the end rather than the final one.

The simulated annealing algorithm has been proven by Hajek (1988) (see also Del Moral and Miclo (1999)) to converge asymptotically, i.e. as $i \to \infty$, to the minimum of the function

almost surely as long as $T_i = T_0/ln(i)$ for some T_0 for sufficiently large T_0. In particular, for almost sure convergence to the minimum it is required that $T_0 > d^*$. d^* denotes the maximum depth of all local minima of the function $\hat{S}(.)$. Heuristically, the depth of a local minimum, $\boldsymbol{\mathcal{J}}_1$, is defined as the smallest number $E > 0$, over all trajectories, such that the function never exceeds $\hat{S}(\boldsymbol{\mathcal{J}}_1)+E$ during a trajectory from[1] this minimum to any other local minimum, $\boldsymbol{\mathcal{J}}_2$, for which $\hat{S}(\boldsymbol{\mathcal{J}}_1) > \hat{S}(\boldsymbol{\mathcal{J}}_2)$.

3.2 The genetic algorithm (GA)

Once again, we describe the operation of the algorithm when the domain of the function is the set of binary strings. The motivating idea of genetic algorithms is to start with a population of binary strings which then evolve and recombine to produce new populations with 'better' characteristics, i.e. lower values for the MSE function. We start with an initial population represented by a $N \times m$ matrix made up of 0's and 1's. Columns represent strings. m is the chosen size of the population. Denote this population (matrix) by $\mathbf{P}_0$. The genetic algorithm involves defining a transition from $\mathbf{P}_i$ to $\mathbf{P}_{i+1}$. The algorithm has the following steps:

1. For $\mathbf{P}_i$ create a $m \times 1$ 'fitness' vector, $\mathbf{p}_i$, by calculating for each column of $\mathbf{P}_i$ its 'fitness'. The choice of the 'fitness' function is completely open and depends on the problem. For our purposes it is the opposite of the MSE function. Normalise $\mathbf{p}_i$, such that its elements lie in $(0, 1)$ and add up to 1. Denote this vector by $\mathbf{p}_i^*$. Treat $\mathbf{p}_i^*$ as a vector of probabilities and resample m times out of $\mathbf{P}_i$ with replacement, using the vector $\mathbf{p}_i^*$ as the probabilities with which each string with be sampled. So 'fit' strings are more likely to be chosen. Denote the resampled population matrix by $\mathbf{P}_{i+1}^1$.

2. Perform cross over on $\mathbf{P}_{i+1}^1$. For cross over we do the following: Arrange all strings in $\mathbf{P}_{i+1}^1$, in pairs (assume that m is even). Denote a generic pair by $(a_1^\alpha, a_2^\alpha, \ldots, a_n^\alpha)$, $(a_1^\beta, a_2^\beta, \ldots, a_n^\beta)$. Choose a random integer between 2 and $n-1$. Denote this by j. Replace the pair by the following pair: $(a_1^\alpha, a_2^\alpha, \ldots, a_j^\alpha, a_{j+1}^\beta, \ldots, a_n^\beta)$, $(a_1^\beta, a_2^\beta, \ldots, a_j^\beta, a_{j+1}^\alpha, \ldots, a_n^\alpha)$. Perform cross over on each pair with probability p_c. Denote the new population by $\mathbf{P}_{i+1}^2$. Usually p_c is set to some number around 0.5-0.6.

3. Perform mutation on $\mathbf{P}_{i+1}^2$. This amounts to flipping the bits (0 or 1) of $\mathbf{P}_{i+1}^2$ with

[1] A trajectory from $\boldsymbol{\mathcal{J}}_1$ to $\boldsymbol{\mathcal{J}}_2$ is a set of strings, $\boldsymbol{\mathcal{J}}_{11}, \boldsymbol{\mathcal{J}}_{12}, \ldots, \boldsymbol{\mathcal{J}}_{1p}$, such that (i) $\boldsymbol{\mathcal{J}}_{11} \in N(\boldsymbol{\mathcal{J}}_1)$, (ii) $\boldsymbol{\mathcal{J}}_{1p} \in N(\boldsymbol{\mathcal{J}}_2)$ and (iii) $\boldsymbol{\mathcal{J}}_{1i+1} \in N(\boldsymbol{\mathcal{J}}_{1i})$ for all $i = 1, \ldots, p$, where $N(\boldsymbol{\mathcal{J}})$ denotes the set of strings that make up the neighborhood of $\boldsymbol{\mathcal{J}}$.

probability p_m. p_m is usually set to a small number, say 0.01. After mutation the resulting population is $\mathbf{P}_{i+1}$.

These steps are repeated a prespecified number of times (B_g). Each set of steps is referred to as generation in the genetic literature. If a string is to be chosen this is the one with maximum fitness. For every generation we store the identity of the string with maximum 'fitness'. At the end of the algorithm the string with the lowest MSE value over all members of the populations and all generations is chosen. One can think of the transition from one string of maximum fitness to another as a Markov Chain. So this is a Markov Chain algorithm. In fact, the Markov chain defined over all possible strings is time invariant but not irreducible as at least the $m-1$ least fit strings will never be picked. To see this note that in any population there will be a string with more fitness than that of the $m-1$ worst strings. There has been considerable work on the theoretical properties of genetic algorithms. Hartl and Belew (1990) and Del Moral and Miclo (1999) have shown that with probability approaching one, the population at the n-th generation will contain the global maximum as $n \to \infty$. For more details see also Del Moral, Kallel, and Rowe (2001).

4 Monte Carlo Study

In order to evaluate the performance of the suggested methods we carry out an Monte Carlo study. Rather than concentrate on simulated data and an inevitably arbitrary data generation process, we carry out our Monte Carlo study on a well known dataset. We utilise the dataset put together by Stock and Watson (2002). This comprises of 147 US macroeconomic variables spanning from 1959M1 to 1998M12. Each series is normalised to have mean zero and variance one.

In the experiments we want to control for a number of parameters such as N, T and k. Rather than fix k we fix the probability, p_k, that a given series out of the N series in x_t will be in x_t^0. We define an experiment as a set of replications for a triplet (p_k, N, T). For every experiment we carry out 500 replications. For every replication we do the following. We take the first $T+36$ observations of a random, without replacement, selection of N series in the dataset. We apply the stationary block bootstrap on that set with block length equal to $[T^{1/4}]$. This forms the set of x_t for this replication. From this we construct x_t^0. Each variable in x_t has probability p_k of being in x_t^0. We then construct y_t where each element of β is equal to one. To get y_t we also add i.i.d. $N(0, \sigma^2)$ noise where σ^2 is fixed to be a given multiple, q, of the variance of $\beta^{0\prime} x_t^0$. In our current experiments we set $q = 1$ giving an R^2

of 50%. We apply our variable selection algorithms to the first T observations of the sample keeping the last 36 for an out-of-sample forecasting exercise.

We use the simulated annealing and genetic algorithms discussed in the previous section. In particular, for simulated annealing we set $h = 1$, $T_0 = 10$, $B_v = 500$, $B_s = 5000$. For the genetic algorithm we set $m = 200$, $B_g = 200$, $p_c = 0.6$,$p_m = 0.01$. All of these parameter values are standard in the literature and we have not experimented with their effects on the performance of the algorithms. Given the adequate performance of the algorithms, as documented below, we believe that these choices are reasonable.

We also use two alternative algorithms currently available in the literature. The first is based on a Bayesian approach. For this we borrow heavily from the work of Fernandez, Ley, and Steel (2001). In that paper model uncertainty is tackled by averaging over a subset of the available models in the spirit of Bayesian model averaging. Nevertheless, the ideas in the paper can be easily adapted to the context of model selection. A vehicle for doing this is the MC^3 algorithm. This algorithm is similar to simulated annealing for the construction of its steps. In particular it defines a search path in the model space just like the simulated annealing algorithm we considered in the previous section. As a result we refer to the setup of the previous section to minimise duplication for the exposition. The difference between SA and MC^3 is the criterion used to move from one string to the other at step i. Here, the Bayes factor for string (model) $i+1$ versus string (model) i is used. This is denoted by $B_{i+1,i}$. The chain moves to the $i+1$ string with probability $min(1, B_{i+1,i})$. This is again a Metropolis-Hastings type algorithm. The Bayes factor we use following Fernandez, Ley, and Steel (2001) is given by

$$B_{i+1,i} = \left(\frac{g_{0i+1}}{g_{0i+1}+1}\right)^{k_{i+1}/2} \left(\frac{g_{0i}+1}{g_{0i}}\right)^{k_i/2} \left(\frac{\frac{1}{g_{0i}+1}RSS_i + \frac{g_{0i}}{g_{0i}+1}RSS}{\frac{1}{g_{0i+1}+1}RSS_{i+1} + \frac{g_{0i+1}}{g_{0i+1}+1}RSS}\right) \tag{3}$$

where RSS_i is the sum of squared residuals of the i-th model, RSS is the sum of the squared deviations from the mean for the dependent variable, k_i is the number of variables in model i and g_{0i} is a model specific constant relating to the prior relative precision. The results of Fernandez, Ley, and Steel (2001) suggest that for consistent model selection g_{0i} should be set to $1/T$. More details may be found in Fernandez, Ley, and Steel (2001).

The second extant algorithm is the one used in Hoover and Perez (1999). The only modifications to the algorithm, as described in pages 175-176 of the paper are as follows: (i) All possible paths, rather than only 10, are considered. (ii) In B(d) we use $CUSUM^2$ instead

of Chow as stability test. (iii) No out-of-sample evaluation is undertaken, since this would change the information set for the other algorithms. We try two versions of this algorithm for two different significance levels for all the tests involved (5% and 1%). We consider 2 information criteria based methods, i.e. AIC, and BIC denoted by (A) and (B) in the Tables.

We report results in Tables 1-3. Denoting a generic estimated string by $\hat{\boldsymbol{\mathcal{J}}}$, Table 1 reports $(\hat{\boldsymbol{\mathcal{J}}} - \boldsymbol{\mathcal{J}}^0)'(\hat{\boldsymbol{\mathcal{J}}} - \boldsymbol{\mathcal{J}}^0)$ for all algorithms. In words, the average number of variables which should be included but are not and which should not be included but are, is reported. Clearly, the lower this is the better the algorithm performs.

Looking at the results several interesting features emerge. Firstly, we see that algorithms using BIC perform better than algorithms using AIC in most cases. The differences in many cases are dramatic. For example, for $N = 25$, $T = 100$ and $p_k = 0.1$ we see that whereas simulated annealing with BIC deviates from the true model by an average of 1.4 variables, this number is increased to about 5 when AIC is used. The second finding relates to the relative performance of simulated annealing and the genetic algorithm. It appears that simulated annealing works better in most cases as well. However, the difference in performance is not very large. The third finding relates to the relative performance of simulated annealing and the MC^3 algorithm. It appears that MC^3 works very well but is narrowly beaten in most of the cases by simulated annealing. Finally, we compare the performance of the sequential testing algorithm and the information criteria methods. Once again the information criteria methods and especially simulated annealing work better. Overall, the conclusion is pretty clear. Combining simulated annealing with BIC works very well.

When using AIC we note that MC^3 significantly outperforms simulated annealing with AIC in a number of cases. This is slightly puzzling since the result is reversed for BIC. For this reason we examine the average value of the information criterion for AIC in Table 2. We see that both the simulated annealing and the genetic algorithm manage to obtain a smaller average value for the criterion than MC^3. However, this does not translate to better performance when variable selection is considered. We conclude that AIC does not seem to be performing very well in this respect in that models that have low value for AIC are not necessarily close to the true model.

Table 3 presents the results of the forecasting exercise. We estimate the parameters using data available in the estimation period. Then, we use the parameter estimates and the selected variables, according to each algorithm, to forecast the dependent variable over

36 periods. We average the relative RMSFE compared to the case where the true model is used, over all replications and report results in Table 3. As we see results are pretty similar across algorithms but the conclusions reached from the results of Table 1 still hold.

5 Forecasting Inflation using a Large Dataset

Up to this point we have proposed methods for model selection that rely on in sample evaluation of the models concerned. However, the litmus test for any model is its forecasting ability. It is also well known that, in many cases, models which fit well during the estimation period will not necessarily produce good forecasts. As a result we consider explicitly adapting our methods to a forecasting context in this section.

Given that in-sample fit is a poor guide to out-of-sample performance, an attractive alternative is to consider the out-of-sample performance of a model in the recent past in order to decide whether it is a good forecasting model. A formalisation of this idea is to consider a set of models and choose as the preferred forecasting model the one that minimises the root mean square forecast error (RMSFE) over the recent past. Once again if the set of models contains a large number of models, evaluation of all of them may be computationally impossible.

We apply this methodology to forecast quarterly UK CPI inflation, denoted by π_t. Our baseline model used for comparative purposes is an $AR(4)$ model constructed using π_t. The set of models over which the optimal forecasting model is obtained is made up of models of the form

$$\pi_{t+\tilde{h}} = \alpha_0 + \sum_{i=1}^{4} \alpha_i \pi_{t-i+1} + \sum_{j=1}^{k} \beta_j x_{j,t} + e_t \tag{4}$$

where $\tilde{h}$ is the forecast horizon. We select both k and the identity of the variables, $x_{j,t}$ by minimising the RMSFE during a window of s periods, over the model space. Clearly we choose a different model for every horizon, $\tilde{h}$. We need to set the size of the window s. We suggest that the details relating to the empirical application under consideration, form the basis of this choice. For example, in the case of UK CPI inflation, we note that an important determinant of the behaviour of this series is the fact that the Bank of England has an inflation targetting monetary policy. This monetary regime dates from 1997Q2. The target horizon is currently 2 years. It is reasonable to suggest that the window be longer than that but not much longer given the frequency of the data. We therefore set s to 12 (three years).

Our data span 1980Q2-2004Q1. The dataset we use to select $x_{j,t}$ is made up of 58 variables and contains a wide variety of macroeconomic variables. Details are given in the data appendix. We evaluate this forecasting strategy over the period that the current monetary regime has been fully in operation. Given the 2 year horizon we drop the first year of data for the current regime and start our evaluation period in 1998Q3. We use simulated annealing to minimise the RMSFE over the window. We set $B_v = 2000$ and $B_s = 10000$, $h = 1$ and $T_0 = 10$. We experimented initially with values for the first two parameters. It appears that the values used in the previous section are too low for this application. Thus, we suggest that one errs on the side of caution and uses these higher values instead. The cost of these choices is only an expected moderate rise in the computational time of the algorithm.

The relative RMSFEs of the optimal models as selected by simulated annealing for $\tilde{h} = 4, 8, 12$, i.e. for one, two and three year ahead forecasts are 0.96, 0.93 and 0.92 respectively compared to the $AR(4)$ model. Given the widespread inability in the literature to beat the forecasts of simple autoregressive models this result is extremely encouraging for our method.

6 Conclusion

The question of variable selection in a regression model is a major open research topic in econometrics. Traditionally two broad classes of methods have been used. One is sequential testing and the other is information criteria. The advent of large datasets used by institutions such as central banks has exacerbated this model selection problem. This paper provides a new solution in the context of information criteria.

The main idea is to note that information criteria optimisation is a nonstandard optimisation problem because the domain of the objective function is discrete. However, it is possible to define a neighborhood in this space as we do and then optimisation algorithms for discrete domains may be applied.

We consider two of the most popular classes of algorithms: simulated annealing and genetic algorithms. Our Monte Carlo study indicates that optimising information criteria using these algorithms provides very promising results. A further application of the basic idea to forecasting, where the RMSFE of a model is minimised, over an out-of-sample forecast evaluation period, indicates a wider potential for these methods in model selection.

References

Del Moral, P., L. Kallel, and J. Rowe (2001): "Modelling Genetic Algorithms with Interacting Partical Systems," *Revista de Matematica, Teoria y Aplicaciones*, 8(2).

Del Moral, P., and L. Miclo (1999): "On the Convergence and the Applications of Generalised Simulated Annealing," *SIAM Journal of Control and Optimisation*, 37(4), 1222–1250.

Fernandez, C., E. Ley, and M. F. J. Steel (2001): "Benchmark Priors for Bayesian Model Averaging," *Journal of Econometrics*, 100, 381–427.

Hajek, B. (1988): "Cooling Schedules for Optimal Annealing," *Mathematics of Operations Research*, 13(2), 311–331.

Hannan, E. J., and B. G. Quinn (1979): "The Determination of the Order of an Autoregression," *Journal of the Royal Statistical Society (Series B)*, 41, 190–195.

Hartl, H. R. F., and R. K. Belew (1990): "A Global Convergence Proof for a Class of Genentic Algorithms," *Technical Report, Technical University of Vienna*.

Hoover, K. D., and S. J. Perez (1999): "Data Mining Reconsidered: Encompassing and the General-To-Specific Approach to Specification Search," *Econometrics Journal*, 2, 167–191.

Krolzig, H. M., and D. F. Hendry (2001): "Computer Automation of General-to-Specific Model Selection Procedures," *Journal of Economic Dynamics and Control*, 25(6–7), 831–866.

Schwarz, G. (1978): "Estimating the Dimension of a Model," *Annals of Statistics*, 6, 461–464.

Sin, C. Y., and H. White (1996): "Information Criteria for Selecting Possibly Misspecified Parametric Models," *Journal of Econometrics*, 71(1–2), 207–225.

Stock, J. H., and M. W. Watson (2002): "Macroeconomic Forecasting Using Diffusion Indices," *Journal of Business and Economic Statistics*, 20, 147–162.

Table 1: Average number of false positives and negatives						
p_k	T	N	$MC^3(B)$	SA(B)	PC(5%)	GA(B)
		25	$1.418_{(1.530)}$	$1.428_{(1.514)}$	$5.652_{(5.725)}$	$1.430_{(1.504)}$
	100	50	$5.082_{(3.209)}$	$4.586_{(3.199)}$	$18.766_{(12.891)}$	$5.154_{(2.943)}$
0.1		75	$11.914_{(4.471)}$	$10.206_{(4.730)}$	$61.282_{(14.537)}$	$11.852_{(4.194)}$
		25	$0.766_{(1.153)}$	$0.756_{(1.131)}$	$5.658_{(6.608)}$	$0.778_{(1.113)}$
	200	50	$3.050_{(2.596)}$	$2.836_{(2.649)}$	$12.346_{(10.701)}$	$4.036_{(2.581)}$
		75	$7.228_{(3.572)}$	$5.966_{(3.842)}$	$23.004_{(15.523)}$	$9.294_{(3.245)}$
		25	$6.812_{(3.010)}$	$6.820_{(3.032)}$	$7.990_{(3.702)}$	$6.824_{(3.013)}$
	100	50	$18.452_{(3.760)}$	$18.382_{(3.823)}$	$22.420_{(5.217)}$	$18.404_{(3.806)}$
0.4		75	$29.906_{(4.308)}$	$29.458_{(4.498)}$	$43.506_{(5.589)}$	$29.562_{(4.559)}$
		25	$4.944_{(3.050)}$	$4.944_{(3.025)}$	$6.310_{(3.973)}$	$4.912_{(3.011)}$
	200	50	$16.394_{(3.995)}$	$16.236_{(4.076)}$	$19.344_{(5.742)}$	$16.514_{(3.914)}$
		75	$28.052_{(4.737)}$	$27.578_{(4.877)}$	$33.300_{(6.478)}$	$28.162_{(4.721)}$

Table 1 (cont.): Average number of false positives and negatives						
p_k	T	N	PC(1%)	$MC^3(A)$	SA(A)	GA(A)
		25	$2.576_{(4.673)}$	$4.200_{(2.201)}$	$5.048_{(2.564)}$	$5.048_{(2.469)}$
	100	50	$9.644_{(12.050)}$	$9.772_{(3.980)}$	$16.074_{(5.039)}$	$14.932_{(4.828)}$
0.1		75	$52.048_{(25.099)}$	$18.148_{(5.367)}$	$33.396_{(7.661)}$	$30.078_{(7.615)}$
		25	$3.140_{(6.195)}$	$3.310_{(1.859)}$	$4.272_{(2.359)}$	$4.466_{(2.232)}$
	200	50	$6.412_{(10.056)}$	$6.942_{(3.200)}$	$12.422_{(3.753)}$	$12.442_{(3.631)}$
		75	$11.916_{(13.726)}$	$12.112_{(4.426)}$	$23.002_{(5.338)}$	$22.748_{(5.348)}$
		25	$7.564_{(3.482)}$	$7.224_{(2.832)}$	$7.436_{(2.851)}$	$7.540_{(2.881)}$
	100	50	$20.154_{(5.026)}$	$19.634_{(3.902)}$	$21.508_{(3.891)}$	$20.782_{(3.845)}$
0.4		75	$41.848_{(7.310)}$	$31.540_{(4.400)}$	$36.004_{(4.514)}$	$34.882_{(4.569)}$
		25	$5.844_{(3.937)}$	$5.232_{(2.718)}$	$5.546_{(2.735)}$	$5.590_{(2.769)}$
	200	50	$17.742_{(5.340)}$	$16.906_{(4.177)}$	$18.464_{(3.917)}$	$18.264_{(3.934)}$
		75	$30.258_{(6.422)}$	$28.664_{(4.718)}$	$31.878_{(4.737)}$	$31.410_{(4.777)}$

Table 2: Average Value of Akaike Information Criterion					
p_k	T	N	$MC^3(A)$	SA(A)	GA(A)
		25	57.14	56.79	56.97
	100	50	131.18	129.11	130.39
0.1		75	192.22	178.75	187.56
		25	174.37	174.00	174.63
	200	50	343.22	343.56	345.22
		75	441.54	440.16	443.68
		25	247.44	247.04	247.01
	100	50	351.56	348.37	348.51
0.4		75	412.36	399.03	406.44
		25	560.00	559.59	559.73
	200	50	767.90	766.68	766.35
		75	908.98	906.26	907.17

Table 3: Relative Forecast RMSE													
p_k	T	N	$MC^3(B)$	SA(B)	PC(5%)	GA(B)	PC(1%)	$MC^3(A)$	SA(A)	GA(A)	$MC^3(P)$	SA(P)	GA(P)
		25	1.05	1.05	1.10	1.05	1.06	1.08	1.10	1.09	1.07	1.09	1.09
	100	50	1.12	1.12	1.32	1.12	1.20	1.17	1.25	1.22	1.19	1.26	1.27
		75	1.18	1.18	5.39	1.17	4.78	1.26	1.58	1.42	1.28	1.45	1.49
0.1		25	1.01	1.01	1.05	1.01	1.04	1.03	1.04	1.04	1.02	1.03	1.02
	200	50	1.04	1.04	1.10	1.04	1.06	1.07	1.09	1.09	1.06	1.09	1.09
		75	1.06	1.06	1.17	1.07	1.10	1.09	1.15	1.14	1.09	1.16	1.17
		25	1.09	1.09	1.09	1.09	1.10	1.08	1.09	1.09	1.09	1.10	1.10
	100	50	1.09	1.10	1.23	1.09	1.15	1.12	1.19	1.17	1.15	1.21	1.24
0.4		75	1.06	1.06	5.37	1.04	5.00	1.10	1.38	1.24	1.12	1.27	1.31
		25	1.05	1.05	1.05	1.05	1.05	1.04	1.04	1.04	1.04	1.05	1.05
	200	50	1.07	1.07	1.09	1.07	1.09	1.07	1.08	1.08	1.07	1.08	1.09
		75	1.05	1.05	1.13	1.04	1.08	1.05	1.11	1.09	1.07	1.12	1.12

Data Appendix

In this appendix, we provide a list of the series used in section 5 to forecast U.K. inflation. These series come from a data set which has been constructed to match the set used by Stock and Watson (2002). In total, this data set has 131 series, comprising 20 output series, 25 labour market series, 9 retail and trade series, 6 consumption series, 6 series on housing starts, 12 series on inventories and sales, 8 series on orders, 7 stock price series, 5 exchange rate series, 7 interest rate series and 6 monetary aggregates, 19 price indices and an economic sentiment index. We retained the 58 series with at least 90 observations. For each series used in section 5 the list gives the FAME alias, a brief description, seasonal adjustment (SA), the transformation applied to the series to ensure stationarity and the first available observation. The transformations applied to the series are: 1 = no transformation; 2 = first difference; 3 = second difference; 4 = logarithm; 5 = first difference of logarithm; 6 = second difference of logarithm. Series 3, 4, 5, 10, 11, 12, 13, 21 and 32 are derived series, described below. The series are grouped under 10 categories.

Series 1 to 8: Real output and income.

- S1: ABMI: Gross Domestic Product: chained volume measures: SA 5 Q1:1955
- S2: CKYY IOP: Manufacturing SA 5 Q1:1948
- S3: IOP: Durable Manufacturing SA 5 Q1:1948
- S4: IOP: Semi-durable Manufacturing SA 5 Q1:1948; constructed as CKZB (IOP: Industry DB: Manuf of textile & textile products) plus CKZC (IOP: Industry DC: Manuf of leather & leather products) plus CKZG (IOP: Industry DG: Manuf of chemicals & man-made fibres) plus CKZH (IOP: Industry DH: Manuf of rubber & plastic products)
- S5: IOP: Non-durable Manufacturing SA 5 Q1:1948; constructed as CKZA (IOP: Industry DA: Manuf of food, drink & tobacco) plus CKZE (IOP: Industry DE: Pulp/paper/printing/publishing industries) plus CKZF (IOP: Industry DF: Manuf coke/petroleum prod/nuclear fuels)
- S6: CKYX IOP: Mining & quarrying SA 5 Q1:1948
- S7: CKYZ IOP: Electricity, gas and water supply SA 5 Q1:1948
- S8: NRJR Real households disposable income SA 5 Q1:1955

Series 9 to 21: Employment and hours.

- S9: DYDC UK Workforce jobs: Total SA 5 Q2:1959
- S10: Employed, Nonagric. Industries SA 5 Q2:1978; constructed as DYDC (UK Workforce jobs (SA) : Total) minus LOLI (UK Workforce jobs (SA): Total - A,B Agriculture & fishing) minus LOMJ (UK Workforce jobs (SA): Total - G-Q Total services)
- S11: Employment Rate: All NSA 1 Q1:1971; concatenate MGRZ and MGRZ_EXP (LFS: In employment: UK: All: Aged 16), concatenate MGSL and MGSL_EXP (LFS: Population aged 16+: UK: All), then compute 1-MGRZ/MGSL
- S12: Employees on nonag. Payrolls: Total SA 5 Q2:1978; constructed as BCAJ (UK Employee jobs: Total (SA)) minus YEHU (UK Employee jobs (SA): All jobs Agriculture,hunting,forestry & fishing)
- S13: Employees nonag. Payrolls: Total: private SA 5 Q2:1978; constructed as S12 minus LOKS (UK Employee jobs (SA): Public admin. & defence)
- S14: YEJF Employee jobs: All jobs: Production Inds. SA 5 Q2:1978
- S15: YEHX Employee jobs: All jobs - Construction SA 5 Q2:1978
- S16: YEHW Employee jobs: All jobs – Manufacturing SA 5 Q2:1978
- S17: LOKL Employee jobs: Wholesale & retail trade SA 5 Q2:1978
- S18: YEIA Employee jobs: Banking, finance & ins. SA 5 Q2:1978
- S19: YEID Employee jobs: Total services SA 5 Q2:1978
- S20: LOKS Employee jobs Public admin. & defence SA 5 Q2:1978

- S21: Avg. weekly hrs. prod. wkrs.: manuf. SA 1 Q1:1971; constructed from YBUS and YBUS_EXP (LFS: Total actual weekly hours worked (millions): UK: All), MGRZ and MGRZ_EXP (LFS: In employment: UK: All: Aged 16+ SA), as YBUS/MGRZ

Series 22 to 23: Trades.

- S22: BOKI BOP: Balance: Total Trade in Goods SA 5 Q1:1955
- S23: ELBJ BOP: Balance: Manufactures SA 5 Q1:1970

Series 24 to 29: Consumption.

- S24: ABJR Household final consumption expenditure SA 5 Q1:1955
- S25: UTID Durable goods: Total SA 5 Q1:1964
- S26: UTIT Semi-durable goods: Total SA 5 Q1:1964
- S27: UTIL Non-durable goods: Total SA 5 Q1:1964
- S28: UTIP Services: Total SA 5 Q1:1964
- S29: TMMI Purchase of vehicles SA 5 Q1:1964

Series 30 to 35: Real inventories and inventories sales.

- S30: CDQN Change in Inventories: Manufacturing SA 5 Q4:1954
- S31: CDQZ Change in Inv: Manuf: Textiles & Leather SA 5 Q4:1954
- S32: Manuf & Trade Invent: Nondurable Goods SA 5 Q4:1954; constructed as CDQP (Change in Inventories: Manufacturing: Fuels) plus CDQX (Change in Inventories: Manufacturing: Food, Drink & Tobacco) plus CDQT (Change in Inventories: Manufacturing: Chemicals)
- S33: FAJX Change in Inventories: Wholesale SA 5 Q1:1959
- S34: FBYN Change in Inventories: Retail SA 5 Q1:1955
- S35: FAPF Ratio for Mfg & Trade: Inventory/Output SA 2 Q1:1955

Series 36 to 38: Stock prices.

- S36: FTALLSH_PI FTSE All Share Price Index 5 Q1:1980
- S37: FTSE100_PI FTSE 100 5 Q1:1980
- S38: FTALLSH_DY FTSE All Share Dividend Yield 1 Q1:1980

Series 39 to 43: Exchange rates.

- S39: A_GBG Sterling - Effective SA 5 Q1:1979
- S40: A_ERS EURO / £ SA 5 Q1:1979; constructed from A_DMS (MTH AVE - DEUTSCHEMARK /£) and fixed conversion rate of 1.95583
- S41: A_SFS SWISS FRANC /£ SA 5 Q1:1979
- S42: A_JYS JAPANESE YEN /£ SA 5 Q1:1979
- S43: A_USS UNITED STATES DOLLAR /£ SA 5 Q1:1979

Series 44 to 47: Interest rates.

- S44: Spread 6-months 1
- S45: Spread 1-year 1
- S46: Spread 5-years 1
- S47: Spread 10-years 1

Series 48 to 50: Monetary and quantity credit aggregates.

- S48: AUYN Money stock: M4 SA 6 Q2:1963

- S49: AVAE M0 wide monetary base SA 6 Q2:1969
- S50: AEFI BOE: reserves & other accounts outstanding NSA 6 Q1:1975

Series 51 to 57: Price indices.

- S51: PLLU PPI: Output of manufactured products NSA 6 Q1:1974
- S52: LCPI Long Run CPI NSA 6 Q1:1975
- S53: ABJS Implicit Price Deflator: H'old final cons exp SA 6 Q1:1955
- S54: UTKT Durable goods: Total IDEF SA 6 Q1:1964
- S55: UTLB Semi-durable goods: Total IDEF SA 6 Q1:1964
- S56: UTKX Non-durable goods: Total IDEF SA 6 Q1:1964
- S57: UTKZ Services: Total IDEF SA 6 Q1:1964

Series 58: Surveys.

- S58: MORI MORI General Economic Optimism index SA 1 Q3:1979

This working paper has been produced by
the Department of Economics at
Queen Mary, University of London

Department of Economics
Queen Mary, University of London
Mile End Road
London E1 4NS
Tel: +44 (0)20 7882 5096
Fax: +44 (0)20 8983 3580
Web: www.econ.qmul.ac.uk/papers/wp.htm