University of Kent

School of Economics Discussion Papers

A Practical, Universal, Information Criterion over N^{th} Order Markov Processes

Sylvain Barde

January 2015

KDPE 1504

A Practical, Universal, Information Criterion over Nth Order Markov Processes*

Sylvain Barde†‡

January 2015

Abstract

The recent increase in the breath of computational methodologies has been matched with a corresponding increase in the difficulty of comparing the relative explanatory power of models from different methodological lineages. In order to help address this problem a universal information criterion (UIC) is developed that is analogous to the Akaike information criterion (AIC) in its theoretical derivation and yet can be applied to any model able to generate simulated or predicted data, regardless of its methodology. Both the AIC and proposed UIC rely on the Kullback-Leibler (KL) distance between model predictions and real data as a measure of prediction accuracy. Instead of using the maximum likelihood approach like the AIC, the proposed UIC relies instead on the literal interpretation of the KL distance as the inefficiency of compressing real data using modelled probabilities, and therefore uses the output of a universal compression algorithm to obtain an estimate of the KL distance. Several Monte Carlo tests are carried out in order to (a) confirm the performance of the algorithm and (b) evaluate the ability of the UIC to identify the true data-generating process from a set of alternative models.

JEL classification: B41, C15, C52, C63.

Keywords: AIC, Minimum description length, Model selection.

*The author is grateful to WEHIA 2013, CFE 2013 and CEF 2014 participants for their comments and suggestions on earlier versions of this paper. The author would like to thank particularly Stefano Grassi, Sandrine Jacob-Léal, Miguel León-Ledesma, Jagjit Chadha, Mauro Napoletano and Lionel Nesta. Finally, a special thanks to James Holdsworth for his invaluable help in setting up the computer cluster on which the Monte Carlo analysis was run. Any errors in the manuscript remain of course the author's.

†School of Economics, Keynes College, University of Kent, Canterbury, CT2 7NP, UK
tel : +44 (0)1 227 824 092, email: s.barde@kent.ac.uk

‡affiliate, Observatoire Français des Conjonctures Economiques

Non-Technical Summary

The recent increase in the breadth of computational methodologies has been matched with a corresponding increase in the difficulty of comparing the relative explanatory power of models from different methodological lineages, particularly simulations. The traditional statistical and econometric methods that researchers rely on to evaluate the relative explanatory power of different models requires that these models possess a specific formal structure of equations and parameters. This is no longer the case for many of the modelling techniques used nowadays, making the problem of comparing the predictions of such models an important open question in the field.

In order to help address this problem the paper develops an information criterion that is analogous to the traditional Akaike information criterion (AIC) in its theoretical derivation and yet can be applied much more widely, as it can be used to compare the explanatory power of any model able to generate simulated data, regardless of its formal structure.

Both the proposed criterion and the AIC are grounded in the same information theoretical concept of using the Kullback-Leibler (KL) distance between model predictions and real data as a measure of prediction accuracy. However instead of using the standard maximum likelihood approach, like the AIC, the proposed criterion relies on the original computer science interpretation of the KL distance as the inefficiency of compressing data using a model that imperfectly approximates the true process that generated the data.

While this may seem like an unnecessary complication, it is what enables the comparison of very different formal models, as the algorithm chosen for the procedure simply maps all the models to a standardised representation (formally, their Markov transition matrices), at which point their predictions can be compared easily. The specific algorithm used in the paper is the Context Tree Weighting (CTW) algorithm. The paper establishes that this algorithm is chosen because it provides the proposed criterion with three desirable properties:

- The criterion is *optimal*, which essentially guarantees that the measurements produced by the algorithm reach the maximum theoretical precision.
- It is also *universal* i.e. the optimal performance mentioned previously is proven for all Markov processes. Markov processes are a very wide class of data-generating processes that englobe nearly all the modelling methodologies in existence, from regression models to simulations. This property underpins the claim that the proposed criterion to compare the predictions of any model capable of producing simulated data.
- Finally, it is *sequential*: the criterion can measure the relative prediction accuracy of different models observation by observation. This means that when comparing the predictive power of different models on a given set of data, statistical testing can be performed to ensure the measurements obtained are statistically significant.

Two Monte Carlo exercises are carried out validate the proposed methodology. The first of these is to check that these theoretical properties are realised in practice, which is shown to be the case, confirming that the algorithm behaves the way information theory predicts.

The second Monte Carlo exercise tests the effectiveness of the methodology at ranking models according to their accuracy. Seven models (one "true" model and six alternate models) are simulated and passed through the CTW algorithm. The result of this test confirms that the methodology can identify the true model from the others and rank all the models according to their predictive power.

1 Introduction

The rapid growth in computing power over the last couple of decades, coupled with the development of user-friendly programming languages and an improvement of fundamental statistical and algorithmic knowledge have lead to a widening of the range of the computational methods available to researchers, from formal modelling to estimation, calibration or simulation methodologies. While this multiplication of available methods has offered a greater modelling flexibility, allowing for the investigation of richer dynamics, complex systems, model switching, time varying parameters, etc., it has come at the cost of complicating the problem of comparing the predictions or performance of models from radically different methodological classes. Two recent examples of this, which are by no means exclusive, are the development of the dynamic stochastic general equilibrium (DSGE) approach in economics, and the increase in the popularity of what is generally referred to as agent-based modelling (ABM), which uses agent-level simulations as a method of modelling complex systems and for which even the issue of bringing models to the empirical data can prove to be a problem.

Within the DSGE literature on model validation and comparison, one of the first to identify and address this problem in a direct and systematic manner is Schorfheide (2000), who introduces a loss function-based method for evaluating DSGE models. This is then complemented by the DSGE-VAR procedure of Del Negro and Schorfheide (2006); Del Negro et al. (2007), which explicitly sets out to answer the question 'How good is my DSGE model?' (p.28). The procedures gradually developed over time in this literature are summarised in the section on DSGE model evaluation of Del Negro and Schorfheide (2011), which outlines several methods for evaluating DGSE performance, such as posterior odds ratios, predictive checks and the use of VAR benchmarking.

Similar concerns relating to model evaluation and comparison also exist in the ABM literature, and in recent years two special journal issues have been published in order to identify and address them. Fagiolo et al. (2007), as part of the special issue on empirical validation in ABM of *Computational Economics*, provide a very good review of the existing practices and provide advice as to how to approach the problem of validating an agent-based simulation model. Nevertheless, as outlined by Dawid and Fagiolo (2008) in the introduction of the special issue of the *JEBO* on adapting ABM for policy design, finding effective procedures for empirical testing,

validation and comparison of such models are still very much an open question.

This paper attempts to address this general issue of comparing different lineages of models by providing a proof-of-concept for a universal information criterion (UIC) that generalises the Akaike (1974) information criterion (AIC) to any class of model able to generate simulated data. Like the AIC, the proposed criterion is fundamentally an estimate of the Kullback and Leibler (1951) (KL) distance between two sets of probability densities. The AIC uses the maximised value of the likelihood function as an indirect estimate of the KL distance, however, this obviously requires the model to have a parametric likelihood function which is no longer straightforward for many classes of modelling methodologies. The proposed criterion overcomes this problem by relying instead on the original data compression interpretation of the KL distance as the inefficiency resulting from compressing a data series using conditional probabilities that are an estimate or approximation of the true data generating process. This fundamental equivalence between data compression and information criteria has led to the emergence of what is know as the *Minimum Description Length* (MDL) principle, which relies the efficiency of data compression as a measure of the accuracy of a model's prediction. Grünewald (2007) provides a good introduction to the MDL principle and its general relation to more traditional information criteria, while Hansen and Yu (2001) explore the use of MDL within a model selection framework, concluding that "MDL provides an objective umbrella under which rather disparate approaches to statistical modeling can coexist and be compared" (Hansen and Yu, 2001, page 772).

A critical motivation for the proposed criterion is that compression algorithm used to calculate its value is universal, i.e. it provides a guaranteed optimal performance over the widest possible range of models. The procedure places all models on an equal footing, regardless of numerical methodology or structure, by treating the simulated data they produce as the result of a N^{th} order Markov process, where the number of lags is chosen to capture the time dependency of the data. As pointed out by Rissanen (1986), Markov process of arbitrary order form a large subclass (denoted FSMX) of finite-state machines (FSM), i.e. systems where transitions are governed by a fixed, finite transition table. By mapping every model to be compared its FSM representation and comparing the transition table probabilities themselves, the UIC is able to overcome differences in modelling methodologies and produce a standardised criterion for any model reducible to a Markov process.

On top of its connections to traditional information criteria and data compression methodologies, the proposed UIC is related to two additional strands of literature. The first is the indirect inference approach initiated by Smith (1993), Gouriéroux and Monfort (1993) and Gouriéroux et al. (1993) which, as will become obvious in the following sections, is similar both in spirit and in practice to the UIC methodology. In cases where a model has a structure that is too complicated to estimate directly an auxiliary model which is tractable but known to be misspecified can be estimated using simulated data series generated by this model for various parametrisations. By comparing this set estimates to the estimation of auxiliary model obtained with the real data, one is able to identify the best parameter values in the initial model. Similarities and differences of the UIC approach with indirect inference will be discussed further below, however for a good general discussion of the procedures involved, the reader is referred to chapter 4 of Gouriéroux and Monfort (1996).

Because purpose of the approach is to compare a set of models $\{M_1, M_2, ..., M_m\}$ against a fixed-size data set, the second related strand of literature is the data snooping problem identified by White (2000) and the reality check procedures that must be carried out to avoid it. Essentially, because statistical tests always have a probability of type I error, repeated testing of a large (and possibly increasing) set of models on a fixed amount of data creates the risk of incorrectly selecting a model that is not truly the best model in the set. White (2000) therefore proposes a procedure that takes into account the size of the model comparison set $\{M_1, M_2, ..., M_m\}$ when testing for performance against a benchmark model M_0. A recent development in this literature is the model confidence set (MCS) methodology of Hansen et al. (2011), which differs from White's reality check in that it does not test against a benchmark model, but instead identifies the subset $\hat{\mathcal{M}}_{1-\alpha}$ of the model comparison set that cannot be distinguished from each other at significance level α. This is well suited to the model-specific scores produced by the UIC, therefore the MCS was be included in the Monte Carlo analyses presented below.

The remainder of the paper is organised as follows. Section 2 first discusses the use of universal data compression as an empirical tool for evaluating prediction accuracy and details the theoretical properties of the UIC. A Monte Carlo analysis is then performed in section 3 in order to compare the UIC against the AIC benchmark in an ARMA-ARCH setting and evaluate the criterion's practical usefulness. Section 4 discusses the findings and concludes.

2 The UIC: motivation and theoretical properties

Before examining the information-theoretical motivation for the UIC methodology and the core properties that justify the choice of algorithms, it is important to first briefly clarify the terminology and notation that will be used throughout the paper.

First of all, we define a *prediction* as a conditional probability mass function over the states a system can occupy, given knowledge of the system's history. A *model* is very loosely defined as any device that can produce a complete set of predictions, i.e. a prediction for every acceptable history. This is very similar to the loose definition adopted by Hansen et al. (2011) for their MCS procedure. Furthermore, no assumption is made on the quality of the predictions: a uniform distribution over the system's states is an acceptable prediction. Conceptually, this set of predictions corresponds to the state transition table of a FSMX, or equivalently, as the transition matrix of a Markov process. This definition of a model is intended to be very general as anything from personal belief systems to formal analytical models, as well as calibrated simulations or fitted econometric specifications are reducible to this class of processes. The *accuracy* of a model relates to how its predictions compare to the true, but unobserved, transitions probabilities. This is done through the *encoding* operation, which refers to the process of compressing a string of data into a shorter string. The *decoding* operation will refer to the reverse process, where the original data is recovered from the compressed data.

Regarding notation, the binary logarithm will be clearly identified as "$\log_2$", while the natural logarithm simply be will be "log". X_t is an unobserved, real-valued random variable describing the state of a system at time t and x_t its observed realisation. Data series are denoted as $x_1^t = \{x_1, x_2, \ldots, x_t\}$. $\underline{X}_t$, $\underline{x}_t$ and $\underline{x}_1^t$ are the discretised versions of the same variables, with r being the number of bits of resolution used for the discretisation and $\Omega = 2^r$ the resulting number of discrete states the system can occupy. Because of the binary discretisation used, $\underline{x}_t$ will refer to both the value of the observation and the corresponding r-length binary string describing it. When necessary, the k^{th} bit of a given observation $\underline{x}_t$ will be identified as $\underline{x}_t\{k\}$. $P_{dgp}\left(\underline{X}_t | \underline{x}_1^{t-1}\right)$ is the true probability distribution over the Ω states at time t conditional on the past realisations of the variable. Let $P_{M_i}\left(\underline{X}_t | \underline{x}_{t-L}^{t-1}\right)$ be a corresponding conditional probability distribution predicted by a model M_i at the same time and over the same state space, perhaps using a limited number of lags L. Using the chain rule for conditional probabilities,

$P\left(\underline{x}_1^t\right) = P\left(\underline{X}_t | \underline{x}_1^{t-1}\right) P\left(\underline{x}_1^{t-1}\right)$, the model predictions and true conditional probabilities can be used recursively to build the overall probabilities for the series $P_{M_i}\left(\underline{x}_1^t\right)$ and $P_{dgp}\left(\underline{x}_1^t\right)$.

2.1 Information criteria and Minimum Description Length

Given a model M_i, a reasonable metric for evaluating the accuracy of the overall prediction $P_{M_i}\left(x_1^t\right)$ with respect to $P_{dgp}\left(x_1^t\right)$ is the Kullback and Leibler (1951) (KL) distance measure between the two distributions, which was developed as an extension of the fundamental concept of information entropy introduced in Shannon (1948).

$$D_{KL}\left(P_{M_i}\left(x_1^t\right) \middle\| P_{dgp}\left(x_1^t\right)\right) = E_{dgp}\left[\log \frac{P_{dgp}\left(x_1^t\right)}{P_{M_i}\left(x_1^t\right)}\right] \tag{1}$$

In terms of notation, the $E_{dgp}[\dots]$ operator indicates that the expectation is taken with respect to the true distribution $P_{dgp}\left(x_1^t\right)$. The first obvious consequence of (1) is that the KL divergence D_{KL} is zero whenever $P_{M_i}\left(x_1^t\right) = P_{dgp}\left(x_1^t\right)$. As shown by Cover and Thomas (1991), by taking into account the strict concavity of the logarithm and applying Jensen's inequality to the expectation term in (1) one can show that the KL distance is strictly positive for $P_{M_i}\left(x_1^t\right) \neq P_{dgp}\left(x_1^t\right)$, making it a *strictly proper* scoring rule in the sense of Gneiting and Raftery (2007). This property underpins the use of the KL distance as a conceptual criterion for determining the accuracy of a model, as minimising the KL distance with respect to the choice of prediction model should theoretically lead to the identification of the true model.

While the KL distance is a desirable measure of accuracy in theory, it suffers from not being directly computable in practice, as this would require knowledge of P_{dgp}. The key insight of Akaike (1974) was to identify that it is possible to use the maximum likelihood estimation of the model M_i, to obtain an estimate of the following cross entropy, without requiring knowledge of the true distribution P_{dgp}:

$$E_{dgp}\left[\log \frac{1}{P_{M_i}\left(x_1^t\right)}\right] = D_{KL}\left(P_{M_i}\left(x_1^t\right) \middle\| P_{dgp}\left(x_1^t\right)\right) + E_{dgp}\left[\log \frac{1}{P_{dgp}\left(x_1^t\right)}\right] \tag{2}$$

Assuming that the model M_i uses a vector of κ_i parameters θ_i, and that $\hat{\theta}_i$ are the parameter values that maximise the likelihood $\mathcal{L}\left(\theta_i | x_0^T\right)$, Akaike (1974) showed the cross entropy between the data and the model can be estimated asymptotically by the following relation, directly leading to the classical definition of the AIC for a set of models:

$$\frac{\text{AIC}_i}{2} = E_{dgp}\left[\log \frac{1}{P_{\hat{M}_i}\left(x_1^T\right)}\right] = -\log\left[\mathcal{L}\left(\hat{\theta}_i \middle| x_1^T\right)\right] + \kappa_i \tag{3}$$

The fact that (3) is not directly an estimate of the KL distance (1) but instead of the cross entropy (2) explains why Akaike (1974) recommends looking at the AIC differences between models, $\Delta\text{AIC}_{i,j} = \text{AIC}_i - \text{AIC}_j$, as this removes the model-independent Shannon entropy terms and keep only the relative KL distance $\Delta D_{KL}\left(P_{M_i}\left(x_1^t\right) \middle\| P_{dgp}\left(x_1^t\right)\right)_{i,j}$.

As emphasised by the MDL literature, the originally interpretation of the KL distance (1) relates to the fundamental theoretical limits to compressibility of data. Given a discretised data series $\underset{\sim}{x}_1^t$, the binary Shannon entropy $-E_{dgp}\left[\log_2 P_{dgp}\left(\underset{\sim}{x}_1^t\right)\right]$ gives the number of bits below which the data series cannot be compressed without loss. Because the true probability over states of nature P_{dgp} is unknown, practical data compression has to rely on a predetermined model of how the data is distributed, P_{M_i}. Intuitively this should introduce some inefficiency, thus increasing the theoretical limit below which the data cannot be compressed. This higher limit, measured by the cross entropy (2), is the sum of the Shannon entropy and the KL distance between M and the true data generating process. In other words, on top of the number of bits required to encode the true information content of the data, one has to add extra bits to account for the fact that the model distribution P_{M_i} does not exactly match the true distribution P_{dgp}.

The MDL principle is at the core of the proposed UIC precisely because of the flexibility it offers, enabling practical model comparison on the basis of simulated data alone. However, as pointed out by Grünewald (2007), MDL only provides a guiding principle for analysis and does not prescribe a specific methodology. It is therefore important choose any implementation carefully and verify its efficiency. The context tree weighting (CTW) proposed by Willems et al. (1995) and used as the basis of the proposed UIC is chosen specifically because of its desirable theoretical properties, discussed below. Another implication of the MDL as a guiding principle rather than a prescriptive approach is that the universality of the specific methodology proposed here refers to its ability to perform optimally on all FSM sources, and does not imply that it is unique, or even the 'best' with regards to other objective criteria.[1]

[1] As an illustration, ongoing work by Lamperti (2015) also explores the possibility of comparing models on the basis of simulated data alone. The approach chosen, however, is very different from the one used here and does not offer the same theoretical guarantees, focusing instead on lower computational requirements.

2.2 Theoretical properties of the UIC procedure

Discounting a preliminary data preparation step required to convert the data series x_1^t to a discretised vector $\underline{x}_1^t$, the methodology uses a two stage procedure to obtain the UIC, outlined in appendix A. In the first stage the CTW algorithm scans the simulated series generated by each candidate model M_i and produces a set of tree structures containing model-specific conditional probabilities $P_{M_i}\left(\underline{X}_t|\underline{x}_{t-d}^{t-1}\right)$. In the second stage, the real data is compressed using these CTW probabilities, providing the required cross entropy measure (2). [2]

Letting $\lambda_i\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$ be the length (in bits) of the code string produced by the Elias arithmetic encoder using the probabilities $P_{M_i}\left(\underline{X}_t|\underline{x}_{t-d}^{t-1}\right)$ to encode an observation $\underline{x}_t$, the UIC for model M_i will primarily be based on the output length of the arithmetic encoder on the entire sequence $\underline{x}_1^t$, i.e.:

$$\lambda_i\left(\underline{x}_1^t\right) = \sum\nolimits_t \lambda_i\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right) \tag{4}$$

This two-stage procedure may seem cumbersome and questions may legitimately be raised about the potential inefficiencies generated by this very indirect method. In fact, the choice of the specific algorithms used in both stages rests precisely on the fact that they endow the UIC with three key properties of interest.

- The measurement of cross-entropy (2) is *optimal*, which guarantees that the inefficiency in measurement attains the theoretical minimum and can therefore be controlled for.
- This measurement is *universal* over finite state machines, i.e. the optimal performance is proven for all Markov processes of arbitrary order. This key property underpins the name of the criterion.
- The measurement is *sequential*, providing a cross entropy measurement for each observation and allowing for confidence testing of the aggregate UIC value.

Regarding optimality, both stages 1 and 2 in appendix A have a tight bound on measurement error. Stage 2, which provides the actual measurement, relies on arithmetic encoding, a simple, elegant and efficient approach to data compression initially outlined by Elias (1975), and further

[2] Because the aim of the paper is to present the desirable theoretical properties of the proposed criterion and assess their usefulness in practice, the more technical aspects relating to the algorithmic implementation are detailed in a technical manual available from the author on request.

developed by Rissanen (1976) and Rissanen and Langdon (1979) into a practical algorithm. Its most important property is that the compression efficiency it achieves approaches the theoretical limit given by Shannon's source coding theorem, as by construction the length of the its output is designed to be equal to the binary log score of the data. One of the key contributions of Elias (1975) is a proof that the inefficiency of the encoder's output over the entire length of the data $\underline{x}_1^t$ (4) is guaranteed to be less than 2 bits when compared to the theoretical cross entropy (2).

$$\lambda_i\left(\underline{x}_1^t\right) - E_{dgp}\left[\log_2 \frac{1}{P_{M_i}\left(\underline{x}\, x_1^t\right)}\right] \quad \leq \quad 2 \tag{5}$$

Using the expression for cross entropy (2) and the inefficiency bound on arithmetic encoding (5) one can see that the measurement error induced by the algorithm when comparing the candidate models M_i and M_j with the relative score $\Delta\lambda_{i,j}\left(\underline{x}_1^t\right) = \lambda_i\left(\underline{x}_1^t\right) - \lambda_j\left(\underline{x}_1^t\right)$ is very tightly bound:

$$-2 \quad \leq \quad \Delta\lambda_{i,j}\left(\underline{x}_1^t\right) - \Delta D_{KL}\left(P_{M_i}\left(\underline{x}_1^t\right) \| \, P_{dgp}\left(\underline{x}_1^t\right)\right)_{i,j} \quad \leq \quad 2 \tag{6}$$

The inefficiency incurred in the first stage by using the CTW to determine the transition probabilities can similarly be bounded. The general intuition behind CTW is that each $\{0,1\}$ bit in the binary training series is treated as the result of a Bernoulli trial. More precisely, all the bits in the series that have the same past historical context $\underline{x}_{t-L}^{t-1}$, identified by the binary string s, and the same initial observation bits $\underline{x}_t\{1,2,\ldots,k\}$, identified by string o, are governed by the same Bernoulli process with with unknown parameter $\theta_{s,o}$. As the training series is processed, each node in the tree maintains a set of counters $(a_{s,o}, b_{s,o})$ for the number of times it has respectively observed a '0' or a '1' after having seen both context s and the first o bits of the current observation. Given these $(a_{s,o}, b_{s,o})$ counters, the estimator for Bernoulli processes developed by Krichevsky and Trofimov (1981) (henceforth referred to as the KT estimator) can be used to estimate the probability of observing an additional '1', based on the following recursion:

$$P_e\left(a_{s,o}, b_{s,o}+1\right) = \frac{b_{s,o}+\frac{1}{2}}{a_{s,o}+b_{s,o}+1} P_e\left(a_{s,o}, b_{s,o}\right) \tag{7}$$

As one would intuitively expect, such a learning process has an efficiency cost. In this case,

the inefficiency of compressing the $a_{s,o}$ zeros and $b_{s,o}$ ones using probabilities obtained with the KT estimator (7) compared to the true Bernoulli process with parameter $\theta_{s,o}$ is measured by the following term:

$$\chi\left(a_{s,o}, b_{s,o}\right) = \log_2 \frac{\left(1-\theta_{s,o}\right)^{a_{s,o}} \theta_{s,o}^{b_{s,o}}}{P_e\left(a_{s,o}, b_{s,o}\right)} \tag{8}$$

The key contribution of Willems et al. (1995) is to prove that χ, the inefficiency cost incurred by estimating probabilities using the CTW algorithm (8), is bounded above by the following term:

$$\chi\left(a_{s,o}, b_{s,o}\right) \quad \leq \quad \frac{1}{2}\log_2\left(a_{s,o}+b_{s,o}\right)+1 \tag{9}$$

Importantly, information theory also provides us with a lower bound for the learning cost χ. In a series of key contributions, Rissanen (1978, 1984) shows that if the probabilities P_{M_i} used to encode data come from a model M_i with a parameter vector θ_i that first has to be estimated from the data, then the effective lower bound on compression is higher than the Shannon entropy alone. Intuitively, there is necessarily an amount of inefficiency to be expected when using estimated rather than known parameter values. This larger lower bound, referred to as the Rissanen bound, includes a cost term which depends on the number of parameters κ_i used in model M_i and the number of data N used to estimate the parameters:

$$E_{dgp}\left[\log_2 \frac{1}{P_{M_i}\left(x_1^t\right)}\right] \geq E_{dgp}\left[\log_2 \frac{1}{P_{dgp}\left(x_1^t\right)}\right] + \frac{1}{2}\kappa_i \log_2\left(N\right) \tag{10}$$

This result bears a strong resemblance to the Bayesian Information Criterion (BIC) developed during the same time period by Schwarz (1978), particularly if one considers that, as shown by Akaike (1974), the maximised likelihood function is a good estimator of the cross entropy (2). Rissanen (1984) is very aware of the similarity of the bound (10) with the BIC, and refers several time to the lineage of his work with Akaike (1974).

$$\frac{\mathrm{BIC}_i}{2} = -\log\left[L\left(\hat{\theta}_i \middle| x_0^T\right)\right] + \frac{1}{2}\kappa_i \log\left(N\right) \tag{11}$$

The Rissanen bound (10) can be used as a lower bound for (9). Because the CTW estimates the parameter of a Bernoulli source, the number of ones and zeros observed must give the total

number of observations, so $N_{s,o} = a_{s,o}+b_{s,o}$, and there is only a single parameter $\theta_{s,o}$, so $\kappa_{s,o} = 1$. One can see below that for each tree node estimator, the inefficiency (8) introduced by having to estimate the transition probabilities in the first stage of the UIC procedure is very tightly bound above the theoretical maximum efficiency.

$$\frac{1}{2}\log_2\left(a_{s,o}+b_{s,o}\right) \quad \leq \quad \chi\left(a_{s,o},b_{s,o}\right) \quad \leq \quad \frac{1}{2}\log_2\left(a_{s,o}+b_{s,o}\right)+1 \tag{12}$$

As pointed out by Willems et al. (1995), what makes this result optimal is that it is not possible to obtain tighter bounds: the lower bound (10) is a fundamental information-theoretic limit, similar to Shannon entropy, and the upper bound (9) is simply the lower bound plus the smallest possible increment, i.e. one bit. Another crucial aspect is that Willems et al. (1995) prove that these bounds (12) hold for all FSMX sources, in particular all Markov process of arbitrary order L. This proven optimal performance over a very general class of processes corresponds to the second important property mentioned above, universality, and justifies the choice of this algorithm for the proposed information criterion.

The third and final property of interest is the fact that the cross entropy measurement is sequential: observations are encoded one after the other, generating an observation-specific encoding $\lambda_{M_i}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$, which sums up to the total length of the code string (4). This has three important implications. First of all it allows for a local version of the UIC and therefore enables the assessment of the relative accuracy of models both at the global and local level, which will be illustrated in section 3.3. The second important implication is that the availability of an observation-by-observation log score $\lambda_{M_i}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$ provides the basis for statistical analysis and confidence testing of the UIC measurements obtained. In fact, as will be shown in section 3.1, this makes the UIC methodology naturally suited to model confidence set procedures.

The last and most important implication of the sequential nature of the methodology is that when calculating the transition probabilities (7) with the node counters $(a_{s,o}, b_{s,o})$, one can simultaneously obtain a set of error bounds, enabling the correction of CTW inefficiency at the observation level. One slight problem is that one cannot directly use the bounds (12), as they measure the theoretical CTW inefficiency cost of compressing the training data in a single pass. This is not appropriate here as the real and training data are different, and processed in two stages. What is required instead is the inefficiency related to the specific real observation

being compressed, ignoring the past training observations. This can be obtained from (12) by calculating the marginal bound, i.e. the increase in overall inefficiency in the KT estimator (7) incurred by adding an extra observation, in our case taken from the real data.

$$\frac{\Delta\chi\left(a_{s,o},b_{s,o}\right)}{\Delta a_{s,o}}=\frac{\Delta\chi\left(a_{s,o},b_{s,o}\right)}{\Delta b_{s,o}}=\frac{1}{2}\log_2\frac{a_{s,o}+b_{s,o}+1}{a_{s,o}+b_{s,o}} \tag{13}$$

Because the KT estimator only predicts the value of a single bit, the overall inefficiency for a given data observation, identified by its context string s is calculated by summing (13) over those internal counters of node s corresponding to the observation string o.

$$\epsilon_i\left(\underline{X}_t|\,\underline{x}_{t-L}^{t-1}\right)=\frac{1}{2}\sum_{k=\varnothing}^{o}\log_2\frac{a_{s,k}+b_{s,k}+1}{a_{s,k}+b_{s,k}} \tag{14}$$

Subtracting the observation-level error vector (14) from the raw score vector (4) results in an error-corrected score $\lambda_i^{ec}\left(\underline{X}_t|\,\underline{x}_{t-L}^{t-1}\right)$ which accounts for the inefficiency cost of using the CTW algorithm to learn the model probabilities in stage 1 and sums to the overall UIC.

$$\mathrm{UIC}_i=\sum\nolimits_t\lambda_i^{ec}\left(\underline{X}_t|\,\underline{x}_{t-L}^{t-1}\right)=\sum\nolimits_t\left[\lambda_i\left(\underline{X}_t|\,\underline{x}_{t-L}^{t-1}\right)-\epsilon_i\left(\underline{X}_t|\,\underline{x}_{t-L}^{t-1}\right)\right] \tag{15}$$

2.3 Verification of the UIC's theoretical efficiency

The UIC (15) aims provide a reliable and statistically testable measurement of the cross entropy (2) between data and a model, for all models that are reducible to a Markov process. What makes this proposition feasible are the desirable theoretical properties of the two algorithms used to generate the UIC, i.e. the fact that the output length of the Elias (1975) algorithm $\lambda_i\left(\underline{x}_1^t\right)$ relative to the true cross entropy over an entire sequence of data is tightly bounded by (6) and that the marginal CTW bound (14) provides an effective correction for the efficiency cost $\chi\left(a_{s,o},b_{s,o}\right)$ of learning the probabilities in the first stage. Because this theoretical performance is central to the procedure, it is important to check, as a first step, that it is achieved in practice by the implementation.

The testing strategy chosen is to run the UIC procedure on a stream of data with known distribution, and therefore known entropy, and to compare the performance achieved in practice to the one expected in theory. In order to provide a reliable test of performance, 1000 random

Table 1: Algorithm performance on 1000 8-bit beta distributed data series

	Shannon benchmark	Elias, fixed probability	Elias, CTW probability
λ/N, mean	6.9839	6.9839	6.9828
λ/N, std. dev.	0.0013	0.0013	0.0460
λ/N, $P_{2.5}$	6.9815	6.9815	6.8989
λ/N, $P_{97.5}$	6.9864	6.9864	7.0711
$\Delta\lambda/N$, mean	-	1.907×10^{-6}	3.536×10^{-4}
$\Delta\lambda/N$, std. dev.	-	1.327×10^{-12}	0.0015
$\Delta\lambda/N$, $P_{2.5}$	-	1.907×10^{-6}	-0.0026
$\Delta\lambda/N$, $P_{97.5}$	-	1.907×10^{-6}	0.0037
Theoretical bound	-	Elias (5)	Marginal (14)
Value	-	3.815×10^{-6}	3.295×10^{-4}

λ/N: Measured cross-entropy per observation.
$\Delta\lambda/N$: Measured cross-entropy per observation, relative to Shannon benchmark.

data series of length $N = 2^{19}$ are generated from the following beta distribution.[3]

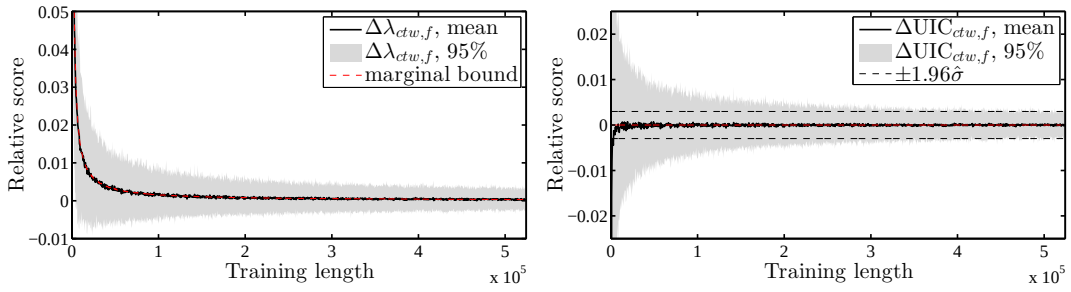
$$X_t \underset{iid}{\sim} \text{Beta}\,(2, 7) \tag{16}$$

The beta distribution is chosen for its $[0, 1]$ support, which simplifies the process of discretising the observations, and for its asymmetry under the chosen parameters, in order to test the CTW's ability to identify asymmetric distribution shapes. Furthermore, the i.i.d. assumption means that each data series is memoryless, ensuring that the best possible compression performance per observation is simply the Shannon entropy of the overall distribution. For the purpose of the analysis, the 1000 series are quantised to an 8-bit level, i.e. $r = 8$.

For each of the 1000 data series, the 8-bit theoretical entropy is calculated by collecting the N observations into a 256 bin histogram on the $[0, 1]$ support, which when normalised by N provides an empirical frequency vector f from which the Shannon entropy $S = -\sum_{i=1}^{256} f_i \log_2 f_i$ can be calculated. This provides the theoretical lower bound for compression and serves as the performance benchmark, the descriptive statistics of which are presented in the first column of table 1.

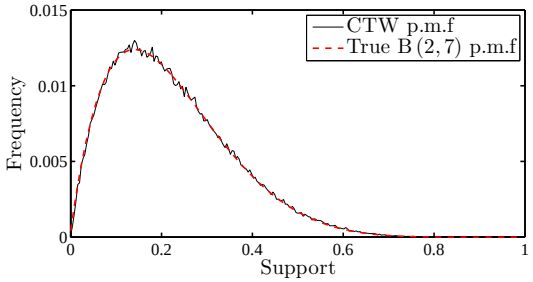
The bound on the Elias algorithm (5) is tested by compressing each data series using its corresponding frequency vector f, and comparing the result against the theoretical benchmark. The result, displayed the second column of table 1, confirms that the difference in performance

[3]Because of the binary nature of the data and algorithm, the data lengths used in the analysis are powers of two as this simplifies calculations requiring the binary logarithm $\log_2$.



(a) Relative score per observation

(b) Relative UIC per observation



(c) CTW probabilities at $T = 2^{19}$

Figure 1: Effective vs. theoretical performance of CTW algorithm

between arithmetic encoding using the fixed probabilities f and the theoretical benchmark is vanishingly small (on the order of $1/N$) and remains below the Elias bound (5) of $2/N$. One can conclude from this result that the Elias algorithm, as implemented in the UIC toolbox, provides extremely reliable measures of cross-entropy (2).

A second test evaluates the learning cost of the CTW algorithm by training it with an additional, independent, stream of beta distributed data (16) and using the resulting CTW probabilities to compress the 1000 Monte Carlo data series. In order to provide an illustration of the literal learning curve of the CTW algorithm, the Monte Carlo analysis is run for varying amounts of training data, from from $T = 1$ to $T = 2^{19} = N$, and the result is illustrated in figure 1.

Along with the third column of table 1, figure 1(a) provides two key conclusions. The first is that, as expected, there is a learning curve: the performance of the algorithm is poor at very low levels of training but quickly starts converging to the benchmark as the amount of training data is increased. The second important element is that the theoretical learning cost (14) tracks the mean inefficiency very closely. The result, as shown in figure 1(b), is that even at low levels of training the expected difference between the UIC score with CTW probabilities and the score

for fixed probability f is near zero. A final confirmation of the CTW algorithms good learning properties is brought by 1(c), which shows that probability distribution learnt by the algorithm closely follows the beta distributed probability mass function (16).

This exercise suggests that the numerical implementations of both algorithms behaves in line with the theoretical properties presented in section 2.2, and provides confidence that the suggested methodology is reliable from a numerical point of view.

3 Monte Carlo validation on ARMA-ARCH models

Having established the theoretical properties of the UIC implementation, its usefulness as a practical information criterion is tested by running the methodology on a series of ARMA-ARCH models, and evaluating its ability to identify the true model as well as rank the alternative models. This will also illustrate the UIC's performance on subsets of data, by attempting to identify portions of the data where the relative explanatory power of two models switches over.

3.1 The ARMA-ARCH model specification and Monte Carlo analysis

Because the UIC aims to generalise the AIC to all FSMX models, the analysis uses a set of ARMA models with ARCH errors, as it is possible to obtain the AIC for each of the the models and use this as a basis for comparing the rankings produced by the UIC. The general structure for the set of models, presented in equation (17), allows for two autoregressive lags, two moving average lags and two ARCH lags.

$$\begin{cases} X_t = a_0 + a_1 X_{t-1} + a_2 X_{t-2} + b_1 \sigma_{t-1}\varepsilon_{t-1} + b_2 \sigma_{t-2}\varepsilon_{t-2} + \sigma_t \varepsilon_t \\ \sigma_t^2 = c_0 + c_1 \varepsilon_{t-1}^2 + c_2 \varepsilon_{t-2}^2 \end{cases} \tag{17}$$

The various specifications used in the analysis only differ in their parameters, shown in Table 2. Only the parameters for the true model M_0 are chosen *ex-ante* with an aim to generate persistence in the auto-regressive components. The parameters for the alternate models M_1-M_6 are estimated using the following procedure. Firstly, a training data series with $T = 2^{19}$ observations is generated using the parameters for the the true model and random draws $\varepsilon_t \underset{iid}{\sim} \mathrm{N}(0,1)$. The parameters for the alternate models are then obtained by using this data series to estimate

Table 2: ARMA-ARCH model structures, parameter estimates and AIC rankings

	M_0 True	M_1 No AR	M_2 No AR-2	M_3 No MA	M_4 No MA-2	M_5 No ARCH	M_6 No ARCH-2
a_0	0	-0.048	0	0	0	0	0
a_1	0.7	-	0.957	0.874	0.299	0.694	0.690
a_2	0.25	-	-	0.087	0.642	0.254	0.256
b_1	0.2	0.916	-0.038	-	0.534	0.205	0.212
b_2	0.2	0.544	0.211	-	-	0.215	0.219
c_0	0.25	0.643	0.252	0.265	0.258	1.234	0.405
c_1	0.5	0.275	0.492	0.470	0.486	-	0.665
c_2	0.3	0.552	0.307	0.332	0.315	-	-
$E(\Delta\text{AIC}_{i,0})$	-	9510.25	38.32	424.95	245.40	4608.41	875.53
$std(\Delta\text{AIC}_{i,0})$	-	447.90	12.62	40.60	56.65	1438.50	160.10
AIC rank ρ_i^*	1	7	2	4	3	6	5

the ARMA-ARCH equation (17) with the corresponding lag(s) omitted. [4]

Once the parameters are obtained, the various data series required for the Monte Carlo analysis of the UIC can be generated using equation (17) parameterised with the relevant column from Table 2 and further random draws $\varepsilon_t \underset{iid}{\sim} \text{N}(0,1)$. T-length training series of are generated for each of the six alternate specifications, and used in stage 1 to train the CTW algorithm. Similarly, 1000 'real' data series with $N = 2^{13} = 8192$ observations are generated using the parameters for M_0. These are used in a Monte Carlo analysis of the UIC's ability to correctly rank the set of models.

The test benchmark is obtained by estimating the set of models using the 1000 N-length data series and calculating the respective AIC for each model and estimation. The descriptive AIC statistics are shown at the bottom of Table 2 and, as expected, the true model is consistently ranked first. An important point to keep in mind for the following section is that because the two AR parameters for M_0 are chosen so as to approach a unit-root behaviour, the AR(1) model M_2 is ranked an extremely close second. In fact, given the average $\Delta\text{AIC}_{2,0} = 38.32$, normalising by the number of observations N gives a mean AIC gap per observation of 4.7×10^{-3}, making those models difficult to distinguish in practice.

Finally, for the purpose of illustrating the local version of the UIC explored in section 3.3, two additional sets of 1000 N-length data series are generated using model switching. In the first

[4]This was done in STATA using the 'arch' routine. As a robustness check, the true data series was also re-estimated, and the chosen parameters for the true model all fall within the 95% confidence interval for the estimates.

Table 3: UIC performance on ARMA models, $T = 2^{22}$

$r = 7$ $d = 21$	M_0 TRUE	M_1 No AR	M_2 No AR-2	M_3 No MA	M_4 No MA-2	M_5 No ARCH	M_6 No ARCH-2
UIC_i, mean	24104.57	30723.17	24106.96	24248.67	24191.26	26362.64	24358.57
$\text{P}(\rho_i = \rho_i^*)$	0.561	1	0.560	0.989	0.992	1	0.995
$\Delta\text{UIC}_{i,0}$, mean	-	6618.60	2.39	144.10	86.69	2258.08	254.00
$\text{P}(\Delta\text{UIC}_{i,0} > 0)$	-	1	0.561	1	0.999	1	1
$\text{P}(M_i \in \hat{\mathcal{M}}_{0.95})$	0.980	0	0.969	0.004	0.085	0	0
$\text{P}(M_i \in \hat{\mathcal{M}}_{0.9})$	0.965	0	0.942	0.002	0.043	0	0

$\text{P}(\rho_i = \rho_i^*)$: Monte Carlo probabilities of UIC rank ρ being equal to AIC rank ρ^*.
$\Delta\text{UIC}_{i,0}$: Mean UIC difference between M_i and the true model M_0, with Monte Carlo probability of being positive
$\text{P}(M_i \in \hat{\mathcal{M}}_{1-\alpha})$: Monte Carlo probability of M_i being included in the Model Confidence Set at α% confidence.

case, the data generating process uses M_0 for the first half of the observations before switching to M_2. In the second case, the data generating process starts with M_5 for half the observations before switching to M_0 for $N/4$ observations and then switching back to M_5 for the remainder of the series.

3.2 UIC performance on ARMA-ARCH models

The Monte Carlo analysis carried out on the ARMA-ARCH models follows the protocol outlined appendix A. As a preliminary step, the data and training series are all discretised to a resolution $r = 7$. In stage 1, the CTW algorithm was run on varying lengths of training series with a chosen tree depth of $d = 21$ bits, which corresponds to 3 observation lags L if one accounts for the 7-bit resolution. Finally, in stage two, the trees are used to compress the 1000 data series, providing a UIC value for each of the models on each of the series.

Table 3 summarises the three main tests that were carried out to evaluate the UIC performance.[5] The first section examines whether the ranking assigned by the UIC to each model, ρ_i, matches the AIC ranking ρ_i^* in table 2. This is a relatively strict test because the ranking for a given model i is affected by the performance of the UIC on all the other models, making this a global test of performance on the full model comparison set. Nevertheless, at training lengths $T = 2^{22}$ the probability $\text{P}(\rho_i = \rho_i^*)$ of correctly obtaining replicating the AIC relative ranking is high for most models, except for M_0 and M_2, where the UIC does little better than random chance. The second test is less strict, as it instead looks at the probability of correctly

[5]More detailed tables, which present the results for the UIC (15) at several values of the training length T and including upper/lower tail critical values for 95% significance levels, are available in appendix B.

selecting the best model in a simple head-to-head competition against the true model M_0. The $\mathrm{P}(\Delta\mathrm{UIC}_{i,0} > 0)$ values reveal that the UIC performs as expected, with a high probability of identifying the true model M_0, in all cases except M_2, where again the UIC does little better than a coin flip.

Both these tests rely on frequencies obtained through the Monte Carlo analysis to evaluate the UIC's ability to rank models. While this provides a useful illustration, it is not sufficient as real-life applications will have to rely on a single real data set with N observations rather than the 1000 series available here. This is where the availability of an N-length observation-level score $\lambda_i^{ec}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$ which sums to the UIC proves useful, as it allows for statistical testing of the overall UIC measurement. As stated previously, availability of this score means that the most natural and rigourous testing approach for the UIC is the reality check proposed White (2000) or the MCS of Hansen et al. (2011). The last part of table 3 reveals the percentage of series for which the i^{th} model is included in the model confidence set $\hat{\mathcal{M}}_{1-\alpha}$ at the 5% and 10% confidence levels. The MCS procedure relies on 1000 iterations of the Politis and Romano (1994) stationary bootstrap for each of the Monte Carlo series, the block length for the bootstrap being determined using the optimal block length procedure of Politis and White (2004). Even accounting for the conservative nature of the MCS test, the procedure is able to effectively narrow down the confidence set to the subset of models that have the lowest UIC ranking. The MCS procedure also confirms the UIC's inability to distinguish M_0 and M_2, as they are almost always included in the confidence set $\hat{\mathcal{M}}_{1-\alpha}$.

Table 4: Performance of the rule of thumb on head-to-head comparisons

	$\lvert\Delta\mathrm{UIC}_{i,j}\rvert$	$> \lvert\tau_{i,j}\rvert$	$< \lvert\tau_{i,j}\rvert$
	N^o correct selections	20046	503
$\alpha =$	N^o incorrect selections	122	329
0.05	P(incorrect$\lvert\ \lvert\Delta\mathrm{UIC}_{i,j}\rvert > \lvert\tau_{i,j}\rvert$)	0.006	-
	P(incorrect$\lvert\ \lvert\Delta\mathrm{UIC}_{i,j}\rvert < \lvert\tau_{i,j}\rvert$)	-	0.395
	N^o correct selections	20226	323
$\alpha =$	N^o incorrect selections	209	242
0.1	P(incorrect$\lvert\ \lvert\Delta\mathrm{UIC}_{i,j}\rvert > \lvert\tau_{i,j}\rvert$)	0.010	-
	P(incorrect$\lvert\ \lvert\Delta\mathrm{UIC}_{i,j}\rvert < \lvert\tau_{i,j}\rvert$)	-	0.428

Because the MCS procedure can be cumbersome to carry out and conservative in its results, we also suggest a faster rule of thumb to test the reliability of a head-to-head comparison between two models, $\Delta\mathrm{UIC}_{i,j}$. This is done though a one-tailed test, aiming to establish whether

$|\Delta\text{UIC}_{i,j}|$ is above a certain reliability threshold. This threshold $\tau_{i,j}$ depends on the sign of $\Delta\text{UIC}_{i,j}$ and for a given level of significance α is given by:

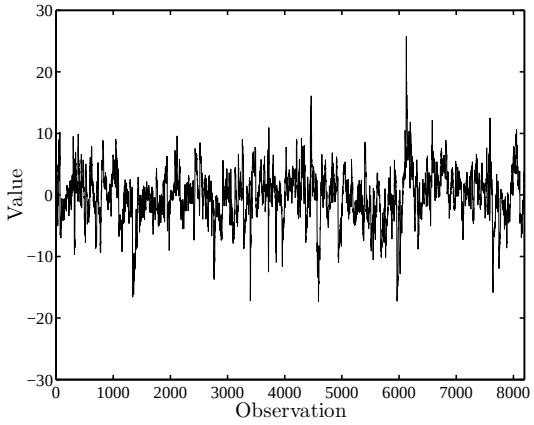
$$\begin{cases} \tau_{i,j} = P_{\alpha}\left(\Delta\lambda_{i,j}^{ec}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)\right)\sqrt{N} & \text{if} \quad \Delta\text{UIC}_{i,j} < 0 \\ \tau_{i,j} = P_{1-\alpha}\left(\Delta\lambda_{i,j}^{ec}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)\right)\sqrt{N} & \text{if} \quad \Delta\text{UIC}_{i,j} > 0 \end{cases} \tag{18}$$

Like the MCS procedure used above, this rule of thumb takes advantage of the availability of the observation-level score $\lambda_i^{ec}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$, meaning that the $\Delta\text{UIC}_{i,j}$ is treated as the sum of N random variables with mean $\Delta\text{UIC}_{i,j}/N$. The thresholds (18) use the α and $1-\alpha$ percentiles of the relative observation level score $\Delta\lambda_{i,j}^{ec}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$ rather than the standard deviation over the series, which would lead to the more traditional thresholds based on $\sigma\sqrt{N}$. What makes this more of a rule of thumb than a fully-fledged test is that the $\sqrt{N}$ terms in (18) implicity rely on a central limit argument to ensure convergence, and this is not explicitly checked for.

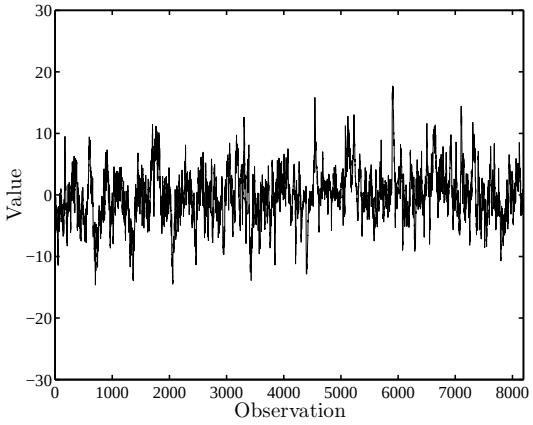
Whenever $|\Delta\text{UIC}_{i,j}| > |\tau_{i,j}|$ one can consider the comparison to be reliable and choose the model with the lowest UIC value. If the reverse is true and $|\Delta\text{UIC}_{i,j}| < |\tau_{i,j}|$, then the UIC measurement must be considered unreliable. Table 4 provides an illustration of the effectiveness of this rule for the 5% and 10% significance levels on the 21000 distinct bilateral comparisons available with the 7 models and 1000 data series of the Monte Carlo analysis. As for the MCS analysis, this is carried out on the bound-corrected criterion with training length $T = 2^{22}$. Table 4 suggests that as long as the rule of thumb is followed, the probability of incorrectly selecting the worse of the two models is very low. Conversely, were one to rely on the $\Delta\text{UIC}_{i,j}$ measurement to identify the best model when it fails the rule of thumb, the probability of an incorrect choice increases greatly and tends towards the worst possible performance where $\Delta\text{UIC}_{i,j}$ is positive or negative with probability 0.5 making $\Delta\text{UIC}_{i,j}$ uninformative with respect to either model. Such a case is illustrated by the M_2, M_0 comparison in table 3 and is the reason the rule of thumb is designed to test the reliability of the measurement.

3.3 Localised UIC performance on ARMA-ARCH models

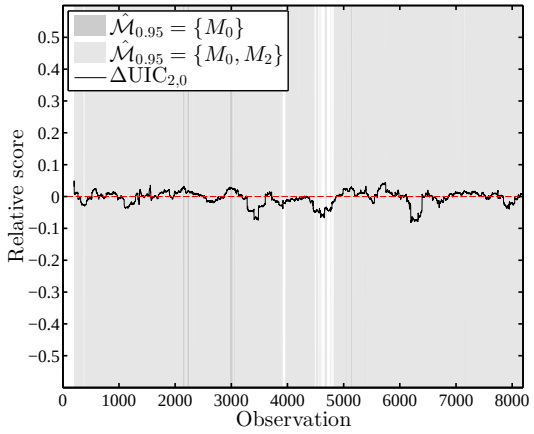
The availability of the observation-level score $\lambda_i^{ec}\left(\underline{X}_t|\underline{x}_{t-L}^{t-1}\right)$ also enables the calculation of a local version of the UIC, allowing models to be compared over subsets of the data. This is illustrated by running the procedure on the two sets of 1000 model-switching series mentioned in section 3.1 using the CTW trees obtained for $T = 2^{22}$ training observations in the previous



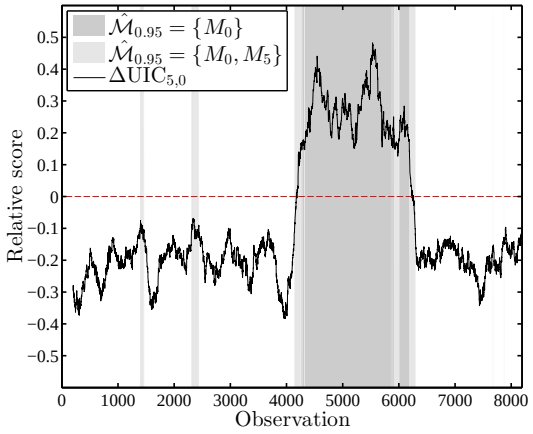
(a) Single data series, M_0 & M_2 switching



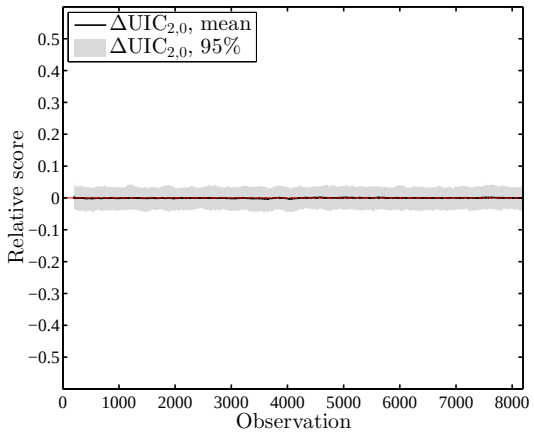
(b) Single data series, M_0 & M_5 switching



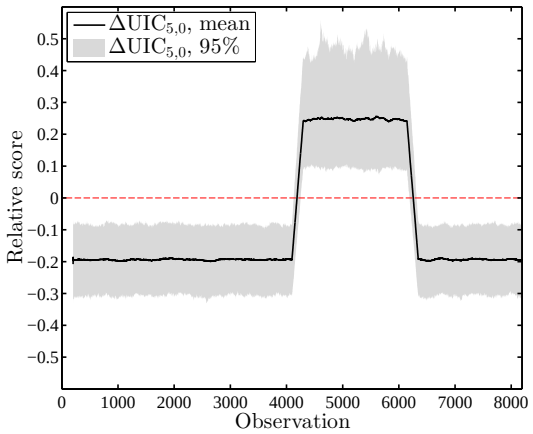
(c) Corresponding $\Delta\text{UIC}_{2,0}$



(d) Corresponding $\Delta\text{UIC}_{5,0}$



(e) $\Delta\text{UIC}_{2,0}$ - Monte Carlo average



(f) $\Delta\text{UIC}_{5,0}$ - Monte Carlo average

Figure 2: Localised UIC performance

section. The results, which are presented in figure 2, have been smoothed using a 200 observation-wide moving average window. In order to also illustrate the small-sample properties of the UIC, the MCS procedure is run on the resulting 200-observation averages. These are shown in figures 2(c) and 2(d), where the dark gray areas indicate observations for which the confidence set $\hat{\mathcal{M}}_{0.95}$ only includes M_0, and the lighter gray cases where the procedure is unable to separate M_0 from the alternate model (M_2 or M_5 depending on the case) and both remain in the confidence set $\hat{\mathcal{M}}_{0.95}$. The areas in white implicitly identify those data points where the confidence set $\hat{\mathcal{M}}_{0.95}$ is reduced to the alternate model.

In the first case, the localised version of the algorithm is unable to detect the transition from M_0 to M_2, for both the individual data series and the Monte Carlo average. This is not surprising given that the UIC is unable to reliably distinguish between M_0 and M_2 at the aggregate level in table 3. In the second case, however, the temporary switch from M_5 to M_0 is clearly visible in the Monte Carlo average and is detected by the MCS procedure on the individual data series. This localised version of the UIC may prove to be as useful for comparing model performance as the aggregate version presented above. It is indeed reasonable to expect that in practical situations one model may outperform others on aggregate yet may be beaten on some specific features of the data, as is the case in figure 2.

4 Discussion and conclusion

This paper develops a methodology which follows the MDL principle and aims provide an information criterion with practically no formal requirement on model structure, other that it be able to generate a simulated data series. While the MDL-inspired algorithms might be unfamiliar, it is important to emphasise that this methodology nevertheless follows the same logic as the AIC, which is that one can measure the cross entropy between some data and a model without knowing the true conditional probability distributions for the events observed. Because these measurements contain an unobservable constant (the true information entropy of the data generating process), one then has to take differences across models to obtain the difference in Kullback-Leibler distance, which is the desired indicator of relative model accuracy. The difference between the AIC and the methodology suggested here rests simply in the choice of method used to measure the cross entropy.

It is important to spell out what can be gained by relying on data compression as the estimation method for cross entropy and by specifically choosing the CTW and Elias algorithms as the means of achieving that data compression. Indeed, the protocol outlined in appendix A requires three distinct steps, each of them unfamiliar, and there seems little purpose to this added complexity if all one wishes to do is calculate the AIC on a regression model, as is done here. The main benefit of the methodology is that it compares models on the basis of the predictive data they produce, by estimating the transition tables of the underlying finite state machines corresponding to the candidate models. This mapping of models to a general class of finite state machines explains why there is no requirement that the candidate models have a specific functional form or estimation methodology, only that they be able to generate a predicted data series. It is this specific aspect which, while unfamiliar, enables the information criterion to claim universality across classes of models, from regression to simulation.

As explained in the introduction, this is similar in spirit to indirect inference therefore some clarification of the relation between the two is required. The UIC is a primarily a fitness criterion, not an estimation methodology. The candidate models compared by the UIC must be already calibrated or fitted in order to produce the required training data. One could of course use the UIC as part of a calibration exercise, where a given model is evaluated with different parameter values, with the best performing parameter configuration being the one which produces the lowest UIC value. While this might seem at first to replicate indirect inference, it would have to rely on a brute-force method, for example an exhaustive grid search, as there is no updating function able to guide the search in the parameter space, as is the case with indirect inference. On the other hand, while indirect inference may be used to compare different models on a given dataset, it is primarily an estimation methodology. Indeed, several features of indirect inference, notably relating to the specification of the auxiliary model, complicate the problem of comparing across models. As pointed out by Gouriéroux and Monfort (1996), for instance, identification considerations mean the parameter dimension in the auxiliary model is determined by the parameter dimension in the initial model, so comparing two very different initial models requires a careful choice of auxiliary model. This is not a problem for the UIC, as the criterion calculated in the second stage uses probabilities extracted from standardised transition tables produced in the first stage. The result of this is that while UIC and indirect inference clearly share the same philosophy, one is designed for estimation and the other for model comparison

and should therefore be considered to be more complements than substitutes.

The Monte Carlo analyses in sections 2.3 and 3 provide several validations of the methodology by demonstrating that given enough training data the UIC procedure provides model ranking information that is comparable to the AIC, and also illustrates a bound correction procedure that can be used to increase the reliability of the UIC. One additional feature of interest illustrated by the Monte Carlo analysis is the possibility of using a local version of the UIC which can compare model performance on subsets of the data, thus detecting data locations where the relative performance of models switches over.

One of the limits of the procedure is that some inefficiency is incurred by the CTW algorithm having to learn the transition table of a model from the training data. A large part of this can be corrected using the known theoretical bounds of the CTW algorithm, however the residual variation creates a "blind spot" which somewhat limits the UIC's ability to distinguish similar models. The Monte Carlo analysis shows that this blind spot is quite narrow, only confusing models that have extremely similar performance, and its size can be established reliably, providing both a rule-of-thumb warning on the reliability of a measurement as well as a rigourous statistical test using the MCS approach.

This paper only provides a proof-of-concept, however, and it is important to point how one might extend this methodology to more common settings. Indeed, the work presented here focuses on a univariate time-series specification, where the candidate models attempt to predict the value of a single random outcome conditional on its past realisations, i.e. $P\left(X_t | \, x_{t-L}^{t-1}\right)$. While this is reasonable as a starting point for establishing that the methodology works on small-scale problems, it is important to outline how it can be scaled up to larger settings.

First of all, extending the approach to multivariate models poses no conceptual problem, as the current state of a FSM does not have to be restricted to a single variable. Supposing that X_t represents instead a state vector made up of several variables $\{A_t, B_t, C_t, \dots\}$, one could use the preliminary step of the protocol in appendix A to discretise each variable at its required resolution $r(a), r(b), r(c), \dots$. The binary string for an observation $\underline{x}_t$ is then simply the concatenation of the individual observations $\underline{a}_t, \underline{b}_t, \underline{c}_t$, and its resolution r is the sum of the individual variable resolutions, i.e. $\Sigma_i r(i)$.

Secondly, in the time-series setting presented here, the observations in x_1^t are used both as the outcome to be predicted (in the case of the current observation) and the conditioning

information for the prediction (for past observations). However, there no reason why the steams of outcome and conditioning data could not be separated. Keeping x_1^t as the conditioning data, the methodology can also generate predictions about the state of a separate outcome variable or vector Y_t, i.e. probability distributions of the type $P\left(Y_t | \, x_{t-L}^t\right)$, which can be used to extend the UIC approach beyond time-series analysis.

While both these extensions are conceptually feasible, they create implementation challenges. The main one is that the larger resolution $\Sigma_i r\left(i\right)$ of a multivariate setting implies a correspondingly larger depth d of the binary context tree for any given number of time lags L, which creates a larger memory requirement for storing the tree nodes. Ongoing work on the implementation of the CTW algorithm is specifically directed at improving the memory efficiency of the algorithm in order to address this last point and turn the proposed methodology into a practical tool.

References

Akaike, Hirotugu (1974) "A new look at the Statistical Model Identification," *IEEE Transactions on Automatic Control*, Vol. AC-19, pp. 716–723.

Cover, Thomas M. and Joy A. Thomas (1991) *Elements of Information Theory*: John Wiley & Sons.

Dawid, Herbert and Giorgio Fagiolo (2008) "Agent-based models for economic policy design: Introduction to the special issue," *Journal of Economic Behavior and Organization*, Vol. 67, pp. 351–354.

Del Negro, Marco and Frank Schorfheide (2006) "How Good Is What You've Got? DGSE-VAR as a Toolkit for Evaluating DSGE Models," *Economic Review-Federal Reserve Bank of Atlanta*, Vol. 91, pp. 21–337.

——— (2011) *The Oxford Handbook of Bayesian Econometrics*, Chap. Bayesian Macroeconometrics: Oxford University Press.

Del Negro, Marco, Frank Schorfheide, Frank Smets, and Rafael Wouters (2007) "On the Fit of New Keynesian Models," *Journal of Business and Economic Statistics*, Vol. 25, pp. 123–143.

Elias, Peter (1975) "Universal Codeword Sets and Representations of the Integers," *IEEE Transactions on Information Theory*, Vol. IT-21, pp. 194–203.

Fagiolo, Giorgio, Alessio Moneta, and Paul Windrum (2007) "A Critical Guide to Empirical Validation of Agent-Based Models in Economics: Methodologies, Procedures, and Open Problems," *Computational Economics*, Vol. 30, pp. 195–226.

Gneiting, Tilmann and Adrian E. Raftery (2007) "Strictly Proper Scoring Rules, Prediction, and Estimation," *Journal of the American Statistical Association*, Vol. 102, pp. 359–378.

Gouriéroux, Christian and Alain Monfort (1993) "Simulation based inference : a survey with special reference to panel data models," *Journal of Econometrics*, Vol. 59, pp. 5–33.

——— (1996) *Simulation-based Econometric Methods*: Oxford University Press.

Gouriéroux, Christian, Alain Monfort, and Eric Renault (1993) "Indirect Inference," *Journal of Applied Econometrics*, Vol. 8, pp. S85–S118.

Grünewald, Peter D. (2007) *The Minimum Description Length Principle*: MIT Press.

Hansen, Mark M. and Bin Yu (2001) "Model Selection and the Principle of Minimum Description Length," *Journal of the American Statistical Association*, Vol. 96, pp. 746–774.

Hansen, Peter R., Asger Lunde, and James M. Nason (2011) "The Model Confidence Set," *Econometrica*, Vol. 79, pp. 453–497.

Krichevsky, Raphail E. and Victor K. Trofimov (1981) "The Performance of Universal Encoding," *IEEE Transactions on Information Theory*, Vol. IT-27, pp. 629–636.

Kullback, S. and R. A. Leibler (1951) "On Information and Sufficiency," *Annals of Mathematical Statistics*, Vol. 22, pp. 79–86.

Lamperti, F. (2015) "An Information Theoretic Criterion for Empirical Validation of Time Series Models," *LEM working paper series*, Vol. 2015/02.

Politis, Dimitris N. and Joseph P. Romano (1994) "The Stationary Bootstrap," *Journal of the American Statistical Association*, Vol. 89, pp. 1303–1313.

Politis, Dimitris N. and Halbert White (2004) "Automatic Block-Length Selection for the Dependent Bootstrap," *Econometric Reviews*, Vol. 23, pp. 53–70.

Rissanen, Jorma (1976) "Generalized Kraft Inequality and Arithmetic Coding," *IBM Journal of Research and Development*, Vol. 20, pp. 198–203.

——— (1978) "Modeling by shortest data description," *Automatica*, Vol. 14, pp. 465–471.

——— (1984) "Universal Coding, Information, Prediction and Estimation," *IEEE Transactions on Information Theory*, Vol. IT-30, pp. 629–636.

——— (1986) "Complexity of Strings in the Class of Markov Sources," *IEEE Transactions on Information Theory*, Vol. IT-32, pp. 526–532.

Rissanen, Jorma and Glen G. Jr Langdon (1979) "Modeling by shortest data description," *IBM Journal of Research and Development*, Vol. 28, pp. 149–162.

Schorfheide, Frank (2000) "Loss function-based evaluation of DSGE models," *Journal of Applied Econometrics*, Vol. 15, pp. 645–670.

Schwarz, Gideon (1978) "Estimating the Dimension of a Model," *The Annals of Statistics*, Vol. 6, pp. 461–464.

Shannon, Claude E. (1948) "A Mathematical Theory of Communication," *The Bell System Technical Journal*, Vol. 27, pp. 379–423.

Smith, Anthony A. Jr. (1993) "Estimating Nonlinear Time-Series Models Using Simulated Vector Autoregressions," *Journal of Applied Econometrics*, Vol. 8, pp. S63–S84.

White, Halbert (2000) "A Reality Check For Data Snooping," *Econometrica*, Vol. 68, pp. 1097–1126.

Willems, Frans M. J., Yuri M. Shtarkov, and Tjalling J. Tjalkens (1995) "The Context-tree Weighting Method: Basic Properties," *IEEE Transactions on Information Theory*, Vol. IT-41, pp. 653–664.

A Outline of the UIC protocol

Only two inputs are required in order to run the protocol. The first is a $N \times 1$ data series that provides the empirical benchmark for model comparison. The second input is a set of training series produced from the various fitted, calibrated or simulated models in the comparison set $\{M_1, M_2, ..., M_m\}$, etc., which should be $T >> N$ observations in length. The sequence of stages is as follows:

1. In stage 0, the data and training series are discretised and converted into binary strings. This process is controlled by the choice of a resolution parameter r, which determines the number of bits used to describe an observation, as well as the the number states an observation can occupy, $\Omega = 2^r$. Several tests are available and should be run at this stage to verify that the chosen resolution parameter r is sufficient to capture the variation in the data.

2. Stage 1 builds a set of transition probabilities $P_{M_i}\left(\underline{X}_t | \underline{x}_{t-L}^{t-1}\right)$ from the training series of each of the models $\{M_1, M_2, ..., M_m\}$ that can used to compress the data in the following stage. The CTW algorithm stores these transition probabilities within a set of binary trees, where each leaf corresponds to the transition probability following a given history. While this is not done in practice, the full transition matrix can in principle be reconstructed from the information in the tree.

3. In stage 2 the $N \times 1$ data series is compressed using the transition probabilities obtained in stage 1. For each model M_i, the algorithm provides three outputs. The first is the binary code string produced by the Elias (1975) encoder, the length of which directly measures the UIC for each model. The second output is a $N \times 1$ vector, $\lambda_i\left(\underline{X}_t | \underline{x}_{t-L}^{t-1}\right)$, containing the number of bits required to encode each observation and sums to the UIC, and the third is $N \times 1$ vector, $\epsilon_i\left(\underline{X}_t | \underline{x}_{t-L}^{t-1}\right)$, containing the CTW error for each observation, which can be used to correct the UIC measure for the inefficiency of having to learn the transition probabilities from a training series in stage 1.

4. Stage 3 is optional, and involves running the code string obtained in stage 2 back through the Elias (1975) algorithm and verifying that the original data can be recovered. This last step is not directly useful in comparing the models, however because the UIC is given by

the length of the compressed string, any error in the encoding process could potentially invalidate the use of this measure and would furthermore be difficult to detect directly. When doing development work in particular, it can be useful to check that no such error has occurred during stage 2.

B Extended Monte Carlo results

Table 5: Monte Carlo analysis of UIC performance on ARMA models

	$r = 7$ $d = 21$	M_0 TRUE	M_1 No AR	M_2 No AR-2	M_3 No MA	M_4 No MA-2	M_5 No ARCH	M_6 No ARCH-2
	UIC_i, mean	24532.95	30744.29	24547.17	24638.04	24610.50	26484.86	24685.19
	$\text{P}(\rho_i = \rho_i^*)$	0.683	1	0.662	0.764	0.812	1	0.909
	$\text{P}(\rho_i > \rho_i^*)$	0.317	-	0.023	0.084	0.162	0	0
	$\text{P}(\rho_i < \rho_i^*)$	-	0	0.315	0.152	0.026	0	0.091
$T =$	$\Delta\text{UIC}_{i,0}$, mean	-	6211.34	14.23	105.10	77.55	1951.91	152.25
2^{19}	$\Delta\text{UIC}_{i,0}$, $P_{2.5}$	-	5624.20	-40.77	42.09	17.99	1547.03	83.94
	$\Delta\text{UIC}_{i,0}$, $P_{97.5}$	-	6781.65	67.70	170.09	136.76	2566.08	221.48
	$\text{P}(\Delta\text{UIC}_{i,0} > 0)$	-	1	0.683	1	0.996	1	1
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.95})$	0.992	0	0.938	0.137	0.375	0	0.024
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.9})$	0.985	0	0.87	0.085	0.252	0	0.012
	UIC_i, mean	24352.74	30779.62	24361.00	24485.07	24439.73	26412.56	24541.52
	$\text{P}(\rho_i = \rho_i^*)$	0.624	1	0.622	0.892	0.956	1	0.931
	$\text{P}(\rho_i > \rho_i^*)$	0.376	0	0.003	0.069	0.039	0	0
	$\text{P}(\rho_i < \rho_i^*)$	-	0	0.375	0.039	0.005	0	0.069
$T =$	$\Delta\text{UIC}_{i,0}$, mean	-	6426.88	8.27	132.33	86.99	2059.83	188.78
2^{20}	$\Delta\text{UIC}_{i,0}$, $P_{2.5}$	-	5825.29	-39.54	70.68	28.10	1651.70	119.88
	$\Delta\text{UIC}_{i,0}$, $P_{97.5}$	-	7042.44	56.33	192.40	146.45	2701.07	258.16
	$\text{P}(\Delta\text{UIC}_{i,0} > 0)$	-	1	0.624	1	0.998	1	1
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.95})$	0.991	0	0.957	0.014	0.189	0	0
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.9})$	0.975	0	0.912	0.007	0.117	0	0
	UIC_i, mean	24210.90	30733.95	24220.41	24350.40	24301.29	26366.04	24436.91
	$\text{P}(\rho_i = \rho_i^*)$	0.670	1	0.669	0.973	0.984	1	0.988
	$\text{P}(\rho_i > \rho_i^*)$	0.330	0	0.001	0.012	0.015	0	0
	$\text{P}(\rho_i < \rho_i^*)$	0	0	0.33	0.015	0.001	0	0.012
$T =$	$\Delta\text{UIC}_{i,0}$, mean	-	6523.05	9.51	139.50	90.40	2155.14	226.02
2^{21}	$\Delta\text{UIC}_{i,0}$, $P_{2.5}$	-	5934.88	-34.98	82.81	36.88	1730.65	159.77
	$\Delta\text{UIC}_{i,0}$, $P_{97.5}$	-	7155.49	55.39	200.28	141.80	2842.82	295.35
	$\text{P}(\Delta\text{UIC}_{i,0} > 0)$	-	1	0.67	1	1	1	1
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.95})$	0.991	0	0.933	0.006	0.103	0	0
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.9})$	0.980	0	0.881	0.003	0.056	0	0
	UIC_i, mean	24104.57	30723.17	24106.96	24248.67	24191.26	26362.64	24358.57
	$\text{P}(\rho_i = \rho_i^*)$	0.561	1	0.56	0.989	0.992	1	0.995
	$\text{P}(\rho_i > \rho_i^*)$	0.439	0	0.001	0.005	0.006	0	0
	$\text{P}(\rho_i < \rho_i^*)$	0	0	0.439	0.006	0.002	0	0.005
$T =$	$\Delta\text{UIC}_{i,0}$, mean	-	6618.60	2.39	144.10	86.69	2258.08	254.00
2^{22}	$\Delta\text{UIC}_{i,0}$, $P_{2.5}$	-	6004.30	-36.90	89.39	38.85	1787.03	183.50
	$\Delta\text{UIC}_{i,0}$, $P_{97.5}$	-	7258.09	40.49	199.07	137.31	3010.67	332.12
	$\text{P}(\Delta\text{UIC}_{i,0} > 0)$	-	1	0.561	1	0.999	1	1
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.95})$	0.980	0	0.969	0.004	0.085	0	0
	$\text{P}(M_i \in \hat{\mathcal{M}}_{0.9})$	0.965	0	0.942	0.002	0.043	0	0

$\text{P}(\rho_i \lessgtr \rho_i^*)$: Monte Carlo probabilities of UIC rank ρ being equal to, greater or smaller than AIC rank ρ^*.
$\Delta\widehat{\text{UIC}}_{i,0}$: Mean UIC difference between M_i and the true model M_0, with 2.5% and 97.5% percentiles.
$\text{P}(M_i \in \hat{\mathcal{M}}_{1-\alpha})$: Monte Carlo probability of M_i being included in the Model Confidence Set at α% confidence.

Recent Kent Discussion Papers in Economics

15/03: 'Public Good Provision in Indian Rural Areas: the Returns to Collective Action by Microfinance Groups', Paolo Casini, Lore Vandewalle and Zaki Wahhaj

15/02: 'Fiscal multipliers in a two-sector search and matching model', Konstantinos Angelopoulos, Wei Jiang and James Malley

15/01: 'Military Aid, Direct Intervention and Counterterrorism', María D.C. García-Alonso, Paul Levine and Ron Smith

14/18: 'Applying a Macro-Finance Yield Curve to UK Quantitative Easing', Jagjit S. Chadha and Alex Waters

14/17: 'On the Interaction Between Economic Growth and Business Cycles', Ivan Mendieta-Muñoz

14/16: 'Is there any relationship between the rates of interest and profit in the U.S. economy?', Ivan Mendieta-Muñoz

14/15: 'Group Lending and Endogenous Social Sanctions', Jean-Marie Baland, Rohini Somanathan and Zaki Wahhaj

14/14: 'Tax Reforms in Search-and-Matching Models with Heterogeneous Agents', Wei Jiang

14/13: 'A Fair Wage Explanation of Labour Market Volatility', Robert Jump

14/12: 'Explaining Differences in the Productivity of Capital Across Countries in the Context of 'New' Growth Theory', Kevin S. Nell and A.P. Thirlwall

14/11: 'The Organic Food Premium: A Canterbury Tale', Adelina Gschwandtner

14/10: 'Contrasting the Perception and Response of Domestic Manufacturing Firms to FDI in Sub-Saharan Africa', Penélope Pacheco-López

14/09: 'Euro- US Real Exchange Rate Dynamics: How Far Can We Push General Equilibrium Models? ', Aydan Dogan

14/08: 'The Role of Conferences on the Pathway to Academic Impact: Evidence from a Natural Experiment', Fernanda L. L. de Leon and Ben McQuillin

14/07: 'Optimal taxation and labour wedge in models with equilibrium unemployment', Wei Jiang

14/06: 'EuroMInd-C: a Disaggregate Monthly Indicator of Economic Activity for the Euro Area and member countries', Stefano Grassi, Tommaso Proietti, Cecilia Frale, Massimiliano Marcellino and Gianluigi Mazzi

14/05: 'Forecasting with the Standardized Self-Perturbed Kalman Filter', Stefano Grassi, Nima Nonejad and Paolo Santucci de Magistris

14/04: 'It's all about volatility of volatility: evidence from a two-factor stochastic volatility model', Stefano Grassi and Paolo Santucci de Magistris