EUROPEAN JOURNAL OF PURE AND APPLIED MATHEMATICS
Vol. 3, No. 3, 2010, 382-405
ISSN 1307-5543 – www.ejpam.com

SPECIAL ISSUE ON
GRANGER ECONOMETRICS AND STATISTICAL MODELING
DEDICATED TO THE MEMORY OF PROF. SIR CLIVE W.J. GRANGER

Predictive Subset VAR Modeling Using the Genetic Algorithm and Information Complexity

J. Andrew Howe[1]* and Hamparsum Bozdogan[2]

[1] *Tennessee Valley Authority, Chattanooga, Tennessee, USA*
[2] *University of Tennessee, Knoxville,Department of Statistics, Operations, and Management Science, Stokely Management Center,Knoxville, Tennessee 37996, USA*

Abstract. Can we use lagged values of major stock market indices to provide useful predictions as a standard vector autoregressive model? Underlying this application, of course, is the question of finding a vector autoregressive model which makes accurate and efficient forecasts. In this paper, we use the Genetic Algorithm with information complexity criteria as the fitness function to drive subset selection and parameter estimation.
In the testing period when the target index lost more than 15%, the identified subset VAR model gained over 17%. The prediction error bands built around the forecasts are half as wide as those obtained by the saturated model.
Using both simulation and application studies, we present evidence that even when the typical regression assumptions seem to be met, the VAR model is misspecified.

2000 Mathematics Subject Classifications: 62HXX,62MXX,91BXX

Key Words and Phrases: Model Selection, Multivariate Time Series, Forecast Evaluation, Robustness, Information Criteria, Stochastic Search

1. Introduction

When considering multivariate time series in the context of dynamic vector autoregressive (VAR) modeling, how do we determine the structure of the relationships? In the simple case of 2 variables, say x and y, and few lags under consideration, the problem is relatively straightforward. For few predictors and lags, combinatorial evaluation of all possible subsets of responses and predictors is not very computationally intensive. As long as both variables are

*Corresponding author.

Email addresses: ahowe42@gmail.com (A. Howe), bozdogan@utk.edu (H. Bozdogan)

© 2010 EJPAM All rights reserved.

stationary, or integrated of order 1 ($I(0)$), we can use multivariate least squares regression. Under the usual assumption of Gaussianity, the *VAR*(q) model is given by (1):

$$\left[\begin{array}{c} y_t' \\ x_t' \end{array}\right] = \left[\begin{array}{c} b_{01} \\ b_{02} \end{array}\right] + \sum_{i=1}^{q} \left[\begin{array}{cc} \phi_{i11} & \phi_{i12} \\ \phi_{i21} & \phi_{i21} \end{array}\right] \left[\begin{array}{c} y_{t-i}' \\ x_{t-i}' \end{array}\right] + \left[\begin{array}{c} \varepsilon_{yt}' \\ \varepsilon_{xt}' \end{array}\right]. \tag{1}$$

The ε are error terms drawn from homoskedastic multivariate white noise. The principle of parsimony drives us to prefer small q. Doing so can protect against overfitting and lead to more efficient forecasts; additionally, it is generally unlikely that higher-order autoregressions are in effect for most econometric data. What can the researcher do, however, if there are many time series under consideration? For example, consider a mere $p = 2$ variables with lags from one to four; for OLS the researcher has two responses and eight potential predictors. Along with an intercept term to estimate, there are $2^{18} - 1 = 262,143$ asymmetric subset VAR models. Thus, as the size of the likely universe increases linearly, the number of combinations increases exponentially; performing complete enumerative subset analysis quickly becomes impossible. In the realm of statistical modeling and data mining, this situation is known as the “curse of dimensionality”. In most applications, *a priori* information useful for restricting terms to 0 is rare. Several approaches have been proposed to impose restrictions, in an effort to make the problem more tractable. Unfortunately, these existing attempts bring their own problems.

In this paper, we propose and present the efficacy of a stochastic search procedure known as the *genetic algorithm* (GA). Of course, the effectiveness of any search algorithm is strongly affected by the choice of the fitness function which is to be optimized. To drive the model selection process, we use the information complexity criterion $ICOMP$; in the spirit of the well known AIC and SBC criteria. $ICOMP$ was first introduced by [5]. In our numerical examples, we first demonstrate our methods with a Monte Carlo simulation study, in which we show that the estimated subset VAR model outperforms the saturated (all predictors included) model. Secondly, we apply the methods to the practical stock-market movement prediction problem. Given a subset of major stock market indices, can we use their lagged values to provide useful predictions for themselves as a standard VAR model? The indices we use are:

DJ20:	Dow Jones 20
MID:	Amex Midcap 400
NDX:	Nasdaq 100
RUT:	Russell 2000
SPX:	Standard & Poor’s S&P 500
XAU:	Amex Gold Producers

Using our methods, we obtain accurate and efficient forecasts. To set the stage, the remainder of this paper is organized as follows. In Section 2, we discuss vector autoregressive modeling, and some of the issues that must be addressed. Section 3 gives background information on the genetic algorithm. We discuss and give the derived forms of information criteria and complexity in Section 4. In Section 5, we provide numerical results on both a Monte Carlo simulation study and the aforementioned stock market prediction problem. Section 6 concludes the paper with remarks.

2. Vector Autoregressive (VAR) Modeling

The purpose of developing a vector autoregressive model is to identify the relationships between a set of linear time series so as to develop accurate and precise forecasts for the series included. In a *structural VAR* model, the time path of each variable is influenced by the lags of all included variables. For example, consider (2), with a simple *VAR*(2) model in standard form (contemporaneous effects removed).

$$\begin{array}{l} y_t = a_{10} + a_{1\_11} y_{t-1} + a_{1\_12} z_{t-1} + a_{2\_11} y_{t-2} + a_{2\_12} z_{t-2} + e_{1t} \\ z_t = a_{20} + a_{1\_21} y_{t-1} + a_{1\_22} z_{t-1} + a_{2\_21} y_{t-2} + a_{2\_22} z_{t-2} + e_{2t} \end{array} . \tag{2}$$

A *symmetric VAR model* is one in which a lag of a specific variable is included for all variables. For example, if $a_{1\_11} \neq 0$, $a_{1\_21} \neq 0$. On the other hand, an *asymmetric VAR model* does not share this restriction. In this case, we could possibly have

$$\begin{array}{l} y_t = a_{10} + a_{1\_12} z_{t-1} + a_{2\_11} y_{t-2} + e_{1t} \\ z_t = a_{20} + a_{1\_21} y_{t-1} + a_{2\_22} z_{t-2} + e_{2t} \end{array} . \tag{3}$$

Despite the value of modeling a set of autoregressive time series in parallel, there is no free lunch, and the cost of VAR modeling is that, in the absence of *a priori* restrictions, the model can easily become overparameterized. This leads to highly biased regression coefficients and large out-of-sample forecast errors.

2.1. Attempts to Make VAR Modeling More Tractable

Over the years, several suggestions have been put forth in the literature, in order to solve these issues. Both Lutkepohl [14] and Penm and Terrell [18] recommend subset VAR models which are basically saturated *VAR*(q^*) models, where $q^* <$ order of full saturated model. The *Vector Error Correction Model* has also gained popularity, though it can only be applied to time series that are *cointegrated*. Using the definition of the well-known [8] paper by Engle and Granger, the components of a vector X of T time series are said to be cointegrated of order d, b ($X \sim CI(d, b)$) if the following restrictions apply.

1. Each series exhibits the same order of integration: $x_t \sim I(d), t = 1 \ldots T$.
2. There exists a vector β (called the cointegrating vector) such that the linear combination $X\beta$ is integrated at a lower order: $X\beta \sim I(d-b), b > 0$. This is a special situation; it is generally the case that a linear combination of $I(d)$ variables remains $I(d)$, and they are not cointegrated.

Cointegrated variables share a common stochastic trend; for a set of cointegrated variables, a valid error correction can be written which will react to correct short term departures from the common trend. Additionally, *Bayesian vector autoregression* has been developed by several researchers, both for cointegrated [2, also see the unpublished manuscript of Kleibergen and van Dijk] and non-cointegrated [13] data.

Many of these attempts to deal with the curse of dimensionality come with their own shortcomings. We can use likelihood ratio hypothesis testing to determine the maximum lags to model, but the test statistic often does not follow the asymptotic chi-squared distribution under the null hypothesis. This approach also restricts the researcher to considering sequential lags $(\Delta_1, \Delta_2, \ldots, \Delta_{q^*})$, where a better model may skip certain lags. Granger causality tests [9, 23] could also be used to reduce the dimensionality of the VAR model, but this approach fails to exploit potential asymmetric relationships. [18] suggest restricting the Φ matrices to be complete - a given lag is either used for all series, or it's not used at all. This method risks including useless predictors at the expense of useful ones; leading to larger forecast errors.

2.2. Subset VAR Models with Robust Covariance Estimation

The typical Gaussian VAR model, in standard form, can be written as a multivariate regression problem: $Y = XB + E$, where $Y \in \mathbb{R}^{n\times p}$ is the matrix of p responses across n observations. Assuming k lags of Y, $X \in \mathbb{R}^{n\times(pk+1)}$ - all the appropriate lags of each response, plus a constant term. The error terms are assumed to be drawn from a multivariate Gaussian white noise process, with mean vector $\mu = \mathbf{0}$ and constant covariance matrix Σ. Finally, $B \in \mathbb{R}^{(pk+1)\times p}$ is the matrix of model coefficients. Of course, there is potentially a coefficient on each lag of each response (plus the constant), for all responses. In order to perform subsetting with asymmetric restrictions in this context, we need to rearrange the data slightly.

Following [3], the first step is to transform Y from an $(n \times p)$ matrix into an $(np \times 1)$ Y_{vec} vector using the $vec(\cdot)$ operator, which vertically catenates columns of a matrix. Secondly, we use the *Kronecker product*, which multiplies all elements of two matrices, to create
$X_{\text{sup}} = I_p \otimes X$; X_{sup} is an $(np \times p\,(pk+1))$ matrix. With these transformations, both the coefficient and error matrices become vectors. The relationship then becomes

$$\underbrace{Y_{vec}}_{np\times 1} = \underbrace{X_{\text{sup}}}_{np\times p(pk+1)} \underbrace{\beta}_{p(pk+1)\times 1} + \underbrace{\varepsilon}_{np\times 1}. \tag{4}$$

At this point, subset selection becomes simple - each column of the sparse X_{sup} matrix represents a specific predictor used for a specific response. For example, if $p = 2$ and $k = 3$, the 1^{st} & 7^{th} columns are the constant applied to the 1^{st} & 2^{nd} predictors, respectively. A binary string of length $p\,(pk+1)$, indicates the presence or absence of a specific predictor used for a specific response - exactly how the GA operates (more later). Following [14], we apply *feasible generalized least squares* (FGLS), which is asymptotically equivalent to OLS. For FGLS estimation of the subset VAR model, we can follow a simple two-step procedure:

1. Compute a consistent estimate of $\hat{\Omega}$:

- Estimate the coefficients: $\hat{\beta}_1 = \left(X'_{\text{sup}} X_{\text{sup}}\right)^{-1} X'_{\text{sup}} Y_{vec}$,
- Get the estimated residuals: $\hat{\varepsilon} = Y_{vec} - X_{\text{sup}} \hat{\beta}_1$,
- Construct an estimate of the covariance matrix after reshaping $\hat{\varepsilon}$ so that $\hat{\varepsilon} \in \mathbb{R}^{n \times p}$: $\hat{\Sigma}_1 = \frac{1}{n} \hat{\varepsilon}' \hat{\varepsilon}$,
- Compute: $\hat{\Omega} = \hat{\Sigma}_1 \otimes I_n$.

2. Compute the FGLS estimates:

$$\hat{\beta}_{FGLS} = (X'_{\text{sup}} \hat{\Omega}^{-1} X_{\text{sup}})^{-1} X'_{\text{sup}} \hat{\Omega}^{-1} Y_{vec}, \tag{5}$$

$$\hat{\varepsilon}_{FGLS} = Y_{vec} - X_{\text{sup}} \hat{\beta}_{FGLS}, \tag{6}$$

$$\hat{\Sigma}_{FGLS} = \frac{1}{n} \hat{\varepsilon}'_{FGLS} \hat{\varepsilon}_{FGLS}. \tag{7}$$

This is the estimation method proposed in [3]. Under the assumption of Gaussianity of the error terms, the FGLS estimators have the same asymptotic distribution as the traditional maximum likelihood estimators.

There is one slight modification that needs to be made to the 3rd items in each of the steps above. In many real-life problems, covariance matrices can become ill-conditioned, non-positive definite, or singular. This is especially true in cases of regression with highly collinear predictors. As can be seen in Table 1, there is a high degree of multicollinearity among the daily changes of the first five indices used in our application. The usual response to singular or ill-conditioned covariance matrix estimates is ridge regularization, which works to counteract the ill-conditionedness by adjusting the eigenvalues of $\hat{\Sigma}$. Usually, the ridge parameter α is chosen to be very small. This, of course, begs the questions

Table 1: Correlation Matrix for Indices Used.

	DJ20	MID	NDX	RUT	SPX	XAU
DJ20	1.000	0.736	0.613	0.702	0.693	−0.107
MID		1.000	0.881	0.935	0.924	−0.158
NDX			1.000	0.886	0.877	−0.231
RUT				1.000	0.869	−0.151
SPX					1.000	−0.179
XAU						1.000

- "*How large should α be?*"
- "*How small can α be?*"

The answer to ridge regularization questions is to use a robust covariance estimator that data-adaptively improves ill-conditioned and/or singular covariance matrix estimates. Many different robust, or smoothed, covariance estimators have been developed; several of them work by the same mechanism as ridge regularization - perturb the diagonals, and hence, the eigenvalues. In this paper, we use the *Maximum Likelihood / Empirical Bayes* covariance estimator

$$\hat{\Sigma}_{MLE/EB} = \hat{\Sigma} + \frac{p-1}{(n)\,tr(\hat{\Sigma}^{-1})} I_p, \tag{8}$$

which is ridge regularization, where the ridge parameter α is determined by the data - not a subjective decision. For various other robust covariance estimators we've found valuable, see [22, 19, 7, 25, 12].

The integration with FGLS estimation is simple; assuming we were using $\hat{\Sigma}_{MLE/EB}$, the final part of *step 1* would entail computing $\hat{\Omega} = \hat{\Sigma}_{MLE/EB} \otimes I_n$. Finally, after the FGLS estimates of the coefficients have been computed, we smooth the $\hat{\Sigma}^{*}_{FGLS}$ using the same estimator. In general, we prefer to not change the problem more than necessary, so we perform two tests for matrix condition before using the selected robust covariance estimator - if the answer to either question is in the affirmative, we instead use the robust estimator:

1. Is the reciprocal of the condition number small: $\kappa^{-1}(\hat{\Sigma}) \leq 1e^{-10}$?

2. Is $\hat{\Sigma}$ nonpositive definite?

2.3. Error Term Bootstrap Procedure

Runkle [21] identified several shortcomings with the ways in which econometricians computed and reported variance decompositions and impulse response functions. Chief was that confidence intervals around estimates were missing in the literature. His claims were that error bands became so large as to make inference from point estimates useless and invalid. He demonstrated, using the example from [24], both an analytical and an empirical method for computing the missing confidence intervals. We employ a variation of his bootstrap procedure here. The fundamental insight is that, since the estimated residuals from any model are assumed to be a representative sample of the true disturbances, the order in which they occur should not matter. This allows us to determine an empirical distribution (and mean) by generating many artificial observations of the data using the estimated residuals. With our modifications, the procedure iterates these four steps after estimating a subset VAR model:

1. Draw an appropriate number of bootstrapped realizations from $\hat{\varepsilon}_{FGLS}$ uniformly and with replacement. The bootstrapped matrix is called $\hat{\varepsilon}_{FGLS,B}$; the number of rows it contains is equal to the original (in-sample) sample size plus enough disturbances for all forecast observations.

2. Conditioning on the pre-sample observations, recursively simulate the data Y_B using the $\hat{\beta}_{FGLS}$ model coefficients matrix and the error terms from $\hat{\varepsilon}_{FGLS,B}$, as in (4).

3. With the simulated dependent and independent matrices, Y_B and X_B, re-estimate the FGLS parameters for both the subset model $\hat{\beta}_{SUB,B}$, and the saturated model $\hat{\beta}_{SAT,B}$.

4. These models are then used to compute point estimates and forecast errors for the out-of-sample simulated datapoints: $F_{SUB} = Y_B - \hat{\beta}_{SUB,B}X_B$ and $F_{SAT} = Y_B - \hat{\beta}_{SAT,B}X_B$. For this work, we allowed the procedure to look ahead 100 periods.

For our results reported here, we used $B = 2000$ iterations. Utilizing all bootstrapped simulations and for each time step, two *Mean Squared Errors* (MSE) are computed and stored - one for the subset VAR model, and the other for the saturated model.

$$MSE_{SUB,i} = \frac{1}{B}\left(\sum_{b=1}^{B} Y_{b,t+i} - \hat{\beta}_{SUB,b}X_b\right), MSE_{SAT,i} = \frac{1}{B}\left(\sum_{b=1}^{B} Y_{b,t+i} - \hat{\beta}_{SAT,b}X_b\right) \tag{9}$$

With this procedure we obtain a point estimate and a variance estimate for each out-of-sample forecast. This allows us to compare the precision with which models make forecasts; we can build error bands, such as $\hat{Y}_{t+i} \pm 2\hat{\sigma}_t$, around the point estimates.

An obvious question is why we should go through all this trouble to bootstrap from the estimated residuals after fitting the VAR model. After all, as long as the OLS assumptions are justified, it would be much simpler to simulate error terms from a multivariate Gaussian distribution with an appropriate scatter matrix. However, when the data exhibit non-Gaussian behavior, from what distribution would we simulate? The use of this method is justified on the basis that it is more robust.

3. Genetic Algorithm (GA)

There are many search algorithms that a researcher could apply to a subset model selection problem such as this. We could have chosen to use a gradient-based algorithm, such as the *greedy algorithm* or a *modified Newton* method. One shortcoming of this approach is that maximization of the likelihood does not consider model complexity, and will lead to suboptimal forecasts when the functional form is misspecified. Additionally, the likelihood landscape is very rugged in the high dimensions that characterize vector autoregressions. Hill-climbing algorithms have a high likelihood of getting stuck in local optima. A second approach would be to use *simulated annealing*. Simulated annealing shares the complexity short-sightedness of other methods; in its defense, it is less likely to get stuck far away from the global optimum. However, it requires several subjective decisions. If made poorly, these decisions can doom the algorithm to failure. For example, how are we to decide the range for the temperature parameter, or the cooling schedule?

Evolutionary algorithms such as the GA, popularized by [10, 11], have become useful tools for complex statistical modeling, identifying near-optimal solutions while providing computational efficiency. Published research that has used the GA for financial / econometric modeling include [17, 20, 26]. The GA is a search algorithm that borrows concepts from biological evolution. Biological chromosomes, which determine so much about organisms, are represented as binary words – these determine the composition of possible solutions to an optimization problem. For multivariate regression subsetting, each chromosome is a q-length vector such that each locus represents the presence (1) or absence (0) of a specific predictor. An example chromosome may be [10011001]; in this case, predictors 1,4,5,8 will be used for OLS while 2,3,6,7 will not. One argument leveled against the GA is that there is no artificial constraint to prevent duplication of solutions within or between iterations. On the surface, this seems wasteful, but a true understanding of the GA reveals this as a strength. This is due to the way in which solutions are considered as an ensemble, and not individually - specifically because of the crossover operator. The general procedure in the GA is simple and straightforward, and is shown here.

1. Generate initial population of chromosomes
2. Score all members of current population
3. Determine how current population is mated and represented in next generation
4. Perform chromosomal crossover and genetic mutation
5. Pass on offspring to new generation
6. Loop back to *2* until termination criteria met

Table 2: Genetic Algorithm Parameters used in Application Example.

Parameter	Setting
Number of Generations	100
Population Size	100
Generation Seeding	Relative Ranking
Crossover Probability	0.75
Mutation Probability	0.25
Objective Function	$ICOMP_{MISP\_PEU}(\hat{\mathscr{F}}^{-1})$

As seen in Table 2, there are eight major parameters used to define the operation of the genetic algorithm. Since the GA has become a fairly well-known search algorithm, we direct interested readers to many excellen sources for further details.

4. Information Criteria and Complexity

Introduced by [5], $ICOMP$ is a logical extension of Akaike's AIC [1]. AIC scores a model by penalizing a bad fit with twice the negative log-likelihood, and model complexity with twice the number of parameters estimated. For multivariate Gaussian errors,

$$\begin{aligned} AIC &= np\log(2\pi)+n\log|\hat{\Sigma}|+np+2m, \text{ where} \\ m &= \left(p\,(k+1)+\frac{p\,(p+1)}{2}\right). \end{aligned} \tag{10}$$

For m, the first term is the number of unrestricted VAR components in the model; the second is the number of variances and covariances. Schwartz's Bayesian Criteria (SBC or BIC) enforces a similar penalty, scaling the number of parameters with $\log n$. Penalizing model complexity with no more information than the number of parameters can be compared to the proverbial blind man trying to identify an elephant by only feeling it's legs. This is just not enough information to measure the information in a model. *For the same dataset, neither AIC nor SBC will be able to distinguish between two models with a similar fit and size, but different structures.* Thus, we use a form of $ICOMP$ which penalizes model complexity with a more judicious penalty term.

$ICOMP(\hat{\mathscr{F}}^{-1})$ utilizes the information in the *first order maximal entropic complexity* of the estimated *inverse Fisher information matrix* (IFIM). Because of this more intelligent penalty, the number of variables, their different structures, and their interrelationships are all simultaneously taken into consideration. In a World Title Award winning paper, [4] demonstrated the value of $ICOMP$ and information theoretic techniques to select autoregressive distributed lag models for forecasting food consumption in the Netherlands. The simplest form of $ICOMP$ that uses this penalty function is shown in (11).

$$ICOMP(\hat{\mathscr{F}}^{-1}) = np\log(2\pi) + n\log|\hat{\Sigma}| + np + 2C_1(\hat{\mathscr{F}}^{-1}) \tag{11}$$

Modeling procedures like this have high potential for overparameterization and bias; a useful form of $ICOMP$ which uses a stricter penalty,

$$ICOMP_{MISP\_PEU}(\hat{\mathscr{F}}^{-1}) = np\log(2\pi) + n\log|\hat{\Sigma}| + np + m + 2B + 2C_1(\hat{\mathscr{F}}^{-1}), \tag{12}$$

was developed by [6] as a Bayesian criterion for maximizing a posterior expected utility. The MISP_PEU indicates that this criterion is **missp**ecification-resistant, and its relationship to the **p**osterior **e**xpected **u**tility (more on the misspecification robustness soon). For the feasible generalized least squares estimates, $\hat{\mathscr{F}}^{-1}$ is shown in (13).

$$\hat{\mathscr{F}}^{-1}(\theta) = \begin{bmatrix} (X'_{\text{sup}}\hat{\Omega}^{-1}X_{\text{sup}})^{-1} & \mathbf{0} \\ \mathbf{0}' & \frac{2}{n}D_p^{+}(\hat{\Sigma}\otimes\hat{\Sigma})D_p^{+'} \end{bmatrix} \tag{13}$$

The matrix D_p is a unique $\left(p^2 \times p\left(p+1\right)/2\right)$ duplication matrix which transforms a square matrix; D_p^{+} is its Moore-Penrose Inverse:

$$D_p^{+} = (D_p^{'}D_p)^{-1}D_p'.$$

See [15] for more about the duplication matrix and the matrix calculus required for the derivations. For a given square matrix M, the complexity is defined as

$$C_1(M) = \frac{s}{2}\log\left(\frac{tr(M)}{s}\right) - \frac{1}{2}\log|M|, \tag{14}$$

where $s = rank(M)$. After some work, we have the complexity of the inverse Fisher information matrix in (15).

$$\begin{aligned} C_1(\hat{\mathscr{F}}^{-1}) \quad = \quad & \{\frac{m}{2}\log\left(\frac{tr[\left(X'_{\text{sup}}\hat{\Omega}^{-1}X_{\text{sup}}\right)^{-1}] + \frac{1}{2n}G}{m}\right)\dots \\ & -\frac{1}{2}\log|\left(X'_{\text{sup}}\hat{\Omega}^{-1}X_{\text{sup}}\right)^{-1}| - \frac{p}{2}\log(2) + \frac{p\left(p+1\right)\log n}{4}\dots \\ & -\frac{(p+1)}{2}\log|\hat{\Sigma}^{*}_{FGLS}|\}. \end{aligned} \tag{15}$$

We define G to be

$$G = tr(\hat{\Sigma}^{*2}_{FGLS}) + tr(\hat{\Sigma}^{*2}_{FGLS}) + 2\sum_{j=1}^{p}\left(\hat{\sigma}^2_{jj}\right)^2.$$

In most statistical modeling problems, we can't assume that the true model is one of those being evaluated. This can bias parameter estimates and over- or under- estimate their variances. In the context of regression, one of the most abused assumptions is that of normality. Non-normal characteristics of data, such as kurtosis and skewness, bias the coefficient estimates. To combat this, a bias estimate can be computed as $\hat{B} = tr(\hat{\mathscr{F}}^{-1}\hat{\mathscr{R}})$. When a model is correctly specified, it is well known that the covariance of the parameters is $\hat{\mathscr{F}}^{-1}$. However, in the presence of misspecification, the appropriate covariance matrix can be shown to be $\widehat{Cov}(\theta) = \hat{\mathscr{F}}^{-1}\hat{\mathscr{R}}\hat{\mathscr{F}}^{-1}$. Whereas $\hat{\mathscr{F}}$ is the inner-product (or Hessian) form of the FIM, $\hat{\mathscr{R}}$ is the outer-product form, both shown in (16) and (17).

$$\mathscr{F} = -E\left(\frac{\partial^2 \log L(\theta)}{\partial\theta\partial\theta'}\right) \tag{16}$$

$$\mathscr{R} = E\left[\frac{\partial \log L(\theta)}{\partial\theta}\cdot\frac{\partial \log L(\theta)}{\partial\theta'}\right] \tag{17}$$

Of course, we must use the observed values for these matrices: $\hat{\mathscr{F}}$ and $\hat{\mathscr{R}}$. Unfortunately, for the problem of feasible generalized least squares, computation of $\hat{\mathscr{R}}$ is a currently intractable problem, so $\hat{B}$ is not directly computable. However, it can be shown that $\hat{B} \simeq nm/(n-m-2)$. Thus, $ICOMP_{MISP\_PEU}(\hat{\mathscr{F}}^{-1})$ can still drive effective model selection, considering the non-normal characteristics of the data, despite any incorrect assumptions, is given by

$$\begin{aligned} ICOMP_{MISP\_PEU}(\hat{\mathscr{F}}^{-1}) &= np\log(2\pi) + n\log|\hat{\Sigma}_{FGLS}| + np \\ &+ m + 2\left(\frac{nm}{n-m-2}\right) + 2C_1(\hat{\mathscr{F}}^{-1}). \end{aligned} \tag{18}$$

Slightly simpler would be $ICOMP_{MISP}$, which does not have the m term, or $ICOMP_{PEU}$, missing the $2\hat{B}$.

Finally, we would mention that cross-validation based criteria are often used for model selection problems. Given the high complexity of our application, cross-validation is just not computationally feasible. This complexity raises concerns regarding the feasibility of subset selection, given existing computational power. One response would be to restrict attention to very small models. Our results, however, demonstrate that arbitrary restrictions such as this are unnecessary.

5. Numerical Results

The value of our techniques are demonstrated here using both simulated data and the real-world example using stock market indices, respectively. In both cases, we see models emerging that fit the data well, are efficient with their forecasts, and use substantially reduced parameter spaces.

5.1. Simulation

We begin by generating data from a bivariate ($p = 2$) autoregressive data generating process shown in (19).

$$\begin{bmatrix} y_t' \\ x_t' \end{bmatrix} = \phi_1 \begin{bmatrix} y_{t-1}' \\ x_{t-1}' \end{bmatrix} + \phi_2 \begin{bmatrix} y_{t-2}' \\ x_{t-2}' \end{bmatrix} + \varepsilon_t \tag{19}$$

We build the coefficient matrices as

$$\phi_1 = \begin{bmatrix} -0.2800 & 0.0215 \\ -0.5496 & 0.2854 \end{bmatrix}, \text{ and } \phi_2 = \begin{bmatrix} -0.2785 & -0.4081 \\ -0.0144 & 0.1809 \end{bmatrix},$$

and the error terms are generated from a multivariate Gaussian white noise process $\varepsilon_t \sim N_2(\mu, \Sigma)$ with parameters

$$\mu = \begin{bmatrix} 0.00 & 0.00 \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} 0.09 & 0.05 \\ 0.05 & 0.04 \end{bmatrix}.$$

Note the presence of correlation between the error terms - we do this so as to introduce some extra difficulty into the modeling process. Each simulation is allowed to run for a *burn-in* period of 1000 cycles, which are subsequently thrown away before $n = 200$ observations are saved. All modeling performed with this data was based on (1) with $q = 4$ lags of both variables; thus there are $2^{18} - 1 = 262,143$ potential nontrivial subset models. An example of one such set of simulated data can be seen in Figure 1. The simulation is clearly generating stationary data. To demonstrate that this is not necessarily an easy simulation, we attempted to fit a subset model of the correct structure to one set of observations. Hence we assume some *a priori* knowledge of the structure. The GA string for this solution is [011110000011110000]. This indicates that: all 3rd and 4th lags are restricted to 0 for each variable, the 1st and 2nd lags are not, and there is no intercept. We would like, of course, to see estimated parameter

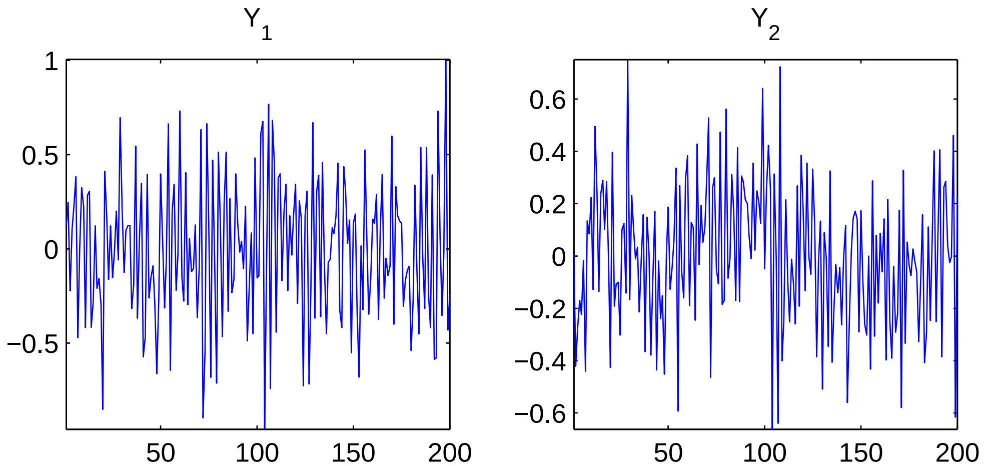

Figure 1: Example of Simulated VAR Data.

estimates reasonably close to those shown above. The fit was first performed with $n = 200$ observations, then with $n = 1000$ observations. As can be seen in Table 3, many of the parameter estimates are quite biased, even when many observations were available. Next to each estimated coefficient matrix is a matrix showing the percentage deviation from the true coefficient. With the exception of the bold element, the accuracy of the estimates improved substantially when $n = 1000$ observations are used. This is clearly a difficult environment in which to pick a good model.

Table 3: Estimated ϕ matrices and Relative Errors for Different n.

n	$\hat{\phi}_1, \frac{\lvert\hat{\phi}_1-\phi_1\rvert}{\phi_1}\%$	$\hat{\phi}_2, \frac{\lvert\hat{\phi}_2-\phi_2\rvert}{\phi_2}\%$
200	$\begin{bmatrix}-0.33 & 0.13\\ -0.62 & 0.36\end{bmatrix}, \begin{bmatrix}18.86 & 493.02\\ 12.81 & 25.33\end{bmatrix}$	$\begin{bmatrix}-0.15 & -0.50\\ 0.01 & 0.22\end{bmatrix}, \begin{bmatrix}46.14 & 22.03\\ 147.92 & 22.33\end{bmatrix}$
1000	$\begin{bmatrix}-0.32 & 0.084\\ -0.58 & 0.33\end{bmatrix}, \begin{bmatrix}13.04 & 291.16\\ 5.26 & 16.78\end{bmatrix}$	$\begin{bmatrix}-0.21 & -0.43\\ 0.05 & 0.14\end{bmatrix}, \begin{bmatrix}25.03 & 4.43\\ \mathbf{443.06} & 21.17\end{bmatrix}$

Our first set of simulation experiments with this data generating process involved fitting all possible **symmetric** subset models. We assume that a lag is included / excluded for both responses. With $q = 18$, this substantially reduces the computational burden from $262,143$ to 511 possible models. In each simulation, five information criteria are computed based on the FGLS estimators - AIC, SBC, $ICOMP$, $ICOMP_{MISP}$, and $ICOMP_{PEU}$. With no desire to "multiply hypotheses more than necessary"(William of Occam), robust covariance estimation was performed with the Maximum Likelihood / Empirical Bayes estimator in this and the next experiment. Table 4 shows correct model hit rates and the average model size for $n = 50$, $n = 200$, and $n = 500$. Clearly, the performance of all criteria is rather dismal for the smallest sample size evaluated. As n increases, we observe the consistency of AIC picking the correct model less than 50% of the time, as well as the tendency to pick overly complex models. The $ICOMP$ with neither the heavier penalty nor the misspecification adjustment is the 2$^{\text{nd}}$

Table 4: Model Hit Rates and Average Subset Model Sizes (True Model Size = 4).

	AIC	SBC	ICOMP	$ICOMP_{MISP}$	$ICOMP_{PEU}$
$n = 50$					
Hit Rate (%)	18.5	13.8	13.1	2.7	12.1
Average Size	4.24	2.96	3.96	2.45	3.27
$n = 200$					
Hit Rate (%)	49.5	87.90	59.80	84.8	77.6
Average Size	4.67	3.95	4.46	3.99	4.13
$n = 500$					
Hit Rate (%)	49.7	99.2	62.0	94.7	84.8
Average Size	4.68	4.01	4.49	4.05	4.17

worst performer and also exhibits a tendency to overfit. Both SBC and $ICOMP_{MISP}$, however, perform very well, picking the correct model with very high frequencies, and exhibiting no tendency to overfit.

The strong performance of $ICOMP_{MISP}$ is of particular interest. Recall that the error terms were generated as homoskedastic multivariate gaussian noise with only slight correlation. Thus, we would expect the model to be correctly specified. However, what we observe here suggests something very interesting: model misspecification comes **automatically** with

- even slight correlation of errors
- high dimensionality $\longrightarrow$ overparameterization
- multicollinearity inherent in time series data

More on this later.

For our second experiment with this simulation protocol, we performed 100 Monte Carlo simulations of the entire modeling process, using the simpler form of $ICOMP$, shown in (11). The best 5 subset models selected by the genetic algorithm shared remarkable similarities in structures and parsimony. None selected an intercept for the model, and all restricted most elements of $\hat{\phi}_3$ and $\hat{\phi}_4$ to be 0 - in the interest of space, only the first two terms are shown in Table 5. Out of these top five models. It is interesting to note that there was substantially more confusion in the estimates for the $\hat{\phi}_2$ matrices, despite the fact that the true variance of the error term on this component was the smaller of the two. Here we've identified, in the 4th column, the score for both the subset model and the corresponding saturated model - the latter in parentheses. The final column identifies the percentage by which the model complexity was reduced - both in terms of the parameter space and the information complexity, respectively. With a relatively modest reduction in the information

Table 5: Top Five Models as Selected By the GA with $ICOMP$.

	$\hat{\phi}_1$	$\hat{\phi}_2$	Score	% Reduction
1	$\begin{bmatrix} -0.230 & \cdot \\ -0.466 & 0.245 \end{bmatrix}$	$\begin{bmatrix} -0.247 & -0.525 \\ 0.071 & \cdot \end{bmatrix}$	$-299.83\,(-272.85)$	61.1 (9.9)
2	$\begin{bmatrix} -0.120 & \cdot \\ -0.472 & 0.327 \end{bmatrix}$	$\begin{bmatrix} -0.166 & -0.486 \\ 0.118 & \cdot \end{bmatrix}$	$-295.08\,(-272.35)$	44.5 (8.3)
3	$\begin{bmatrix} -0.281 & \cdot \\ -0.560 & 0.327 \end{bmatrix}$	$\begin{bmatrix} -0.339 & -0.369 \\ \cdot & 0.142 \end{bmatrix}$	$-286.94\,(-257.26)$	61.1 (11.5)
4	$\begin{bmatrix} -0.276 & \cdot \\ -0.511 & 0.194 \end{bmatrix}$	$\begin{bmatrix} -0.230 & -0.529 \\ \cdot & 0.167 \end{bmatrix}$	$-284.28\,(-256.56)$	61.1 (10.8)
5	$\begin{bmatrix} -0.274 & \cdot \\ -0.553 & 0.352 \end{bmatrix}$	$\begin{bmatrix} -0.223 & -0.633 \\ 0.090 & \cdot \end{bmatrix}$	$-266.46\,(-240.98)$	61.1 (10.6)

criteria score, we obtained greatly simplified models. Though not shown here, both the subset

Table 6: Mean Squared Errors for Simulated Data.

	X_1		X_2	
Step	$MSE_{SUB,i}$	$MSE_{SAT,i}$	$MSE_{SUB,i}$	$MSE_{SAT,i}$
10	0.0652*	0.0675	0.0325*	0.0337
20	0.0687*	0.0701	0.0321*	0.0330
30	0.0664*	0.0687	0.0326*	0.0339
40	0.0675*	0.0699	0.0331*	0.0341
50	0.0674*	0.0696	0.0334*	0.0346
60	0.0656*	0.0690	0.0320*	0.0335
70	0.0711*	0.0729	0.0339*	0.0349
80	0.0645*	0.0669	0.0315*	0.0327
90	0.0652*	0.0670	0.0319*	0.0329
100	0.0684*	0.0699	0.0333*	0.0340

and saturated models performed similarly when used to make out-of-sample predictions for $T = t+1 \ldots t+100$ periods ahead. Having said that, we see immense value in these methods when we consider the forecast precision of the subset VAR model. We utilized the error term bootstrapping procedure to perform this evaluation, with results in Table 6. The bootstrap procedure was executed $B = 2000$ times, in approximately 4.5 minutes, using a forecast time horizon of 100 days. For both the saturated and best subset models, the mean squared error (9) was computed across all simulations. As can be seen, for all time steps considered, the subset VAR model provided a much tighter forecast precision than the saturated model, as measured by MSE. Finally, using the best subset VAR model, the residuals were computed and

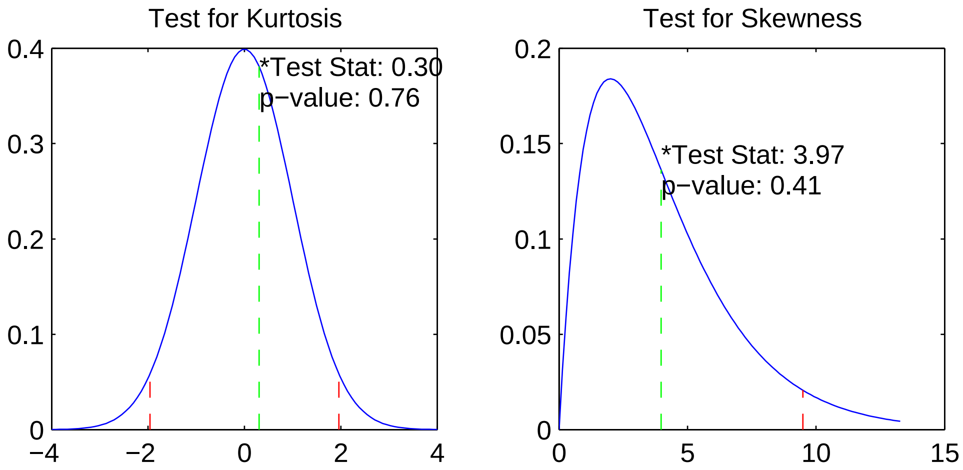

Figure 2: Multivariate Normality Test Results for Residuals.

analyzed to determine how well they met the standard OLS assumptions of Gaussianity. We employed the tests for multivariate skewness and kurtosis of [16], and found the data fit the assumption of normality very well, as can be seen in Figure 2. Additionally, we evaluated the sample autocorrelation coefficient for orders one to five; the strongest correlation was -0.0408 - virtually negligible. Figure 3 shows the time-ordered plots of the residuals from each response. It is clear that the residuals are homoskedastic. So here we see that the error terms from the best subset model show no departure from the assumption of uncorrelated homoskedastic multivariate normality. And yet, our first experiment suggested the model was misspecified. Thus, the use of stricter misspecification-resistent criteria may be justified **even when the model seems correct**. We propose that this is the case for VAR modeling, in general.

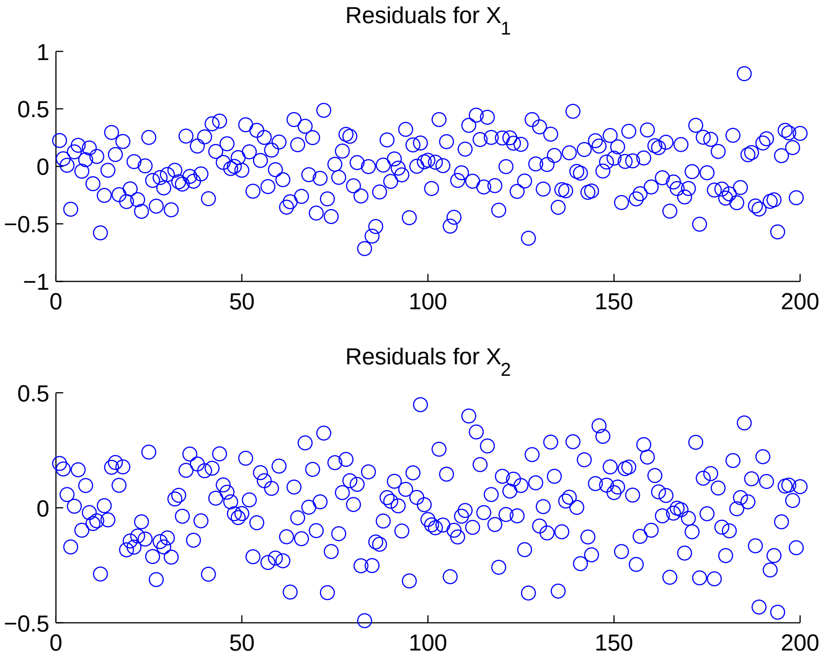

Figure 3: Time-ordered Plot of Residuals Displaying Homoskedasticity.

5.2. Selecting an Economically Valuable Trading Model

Our application example was performed using the first 3 months of 2002 (approximately 60 trading days) as the training period, and the ensuing 3 months as the trading period. Using the usual benchmark of Standard & Poor's S&P 500, the uncompounded return for the in-sample period is -0.40%. The index lost 15.28% during the testing period. Figure 4 shows the activity of the SPX over the entire period, along with its daily changes. All analysis was performed using uncompounded returns; this was accomplished by decomposing the series into its relative daily changes: $\left(y_t - y_{t-1}\right)/y_{t-1}$. A cumulative sum was then computed on these daily returns. This gives a simple measure of trading activity, and acts to more conservatively estimate the value of a given model. In preparing the predictor variables, the data used

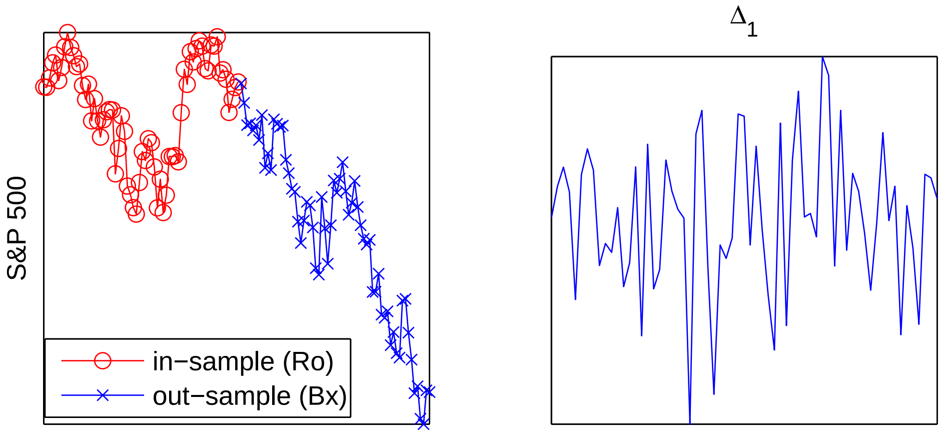

Figure 4: S&P 500 Benchmark Daily Values and Changes.

was begun six days prior to the 1st day of 2002. After all five lags were computed on the first differences, these conditional observations were dropped. This was done so as to maintain a constant number of usable observations across all models. In time series analysis, we are mostly interested in forecasting, conditional upon existing data. In the specific specialization of trading, a model that has a higher out-of-sample return is deemed to have stronger forecast power. Due to the much higher dimensionality of this dataset, the chances of overparameterization are higher. Additionally, since we are dealing with financial data, we expect heavier tails. Thus, the model selection information criteria we employ is $ICOMP_{MISP\_PEU}$ with a strict and misspecification-robust penalty term. Using the GA settings in Table 2, 100 replications of the genetic algorithm were executed. Replications that did not terminate early evaluated at most 15, 000 unique subset VAR models, with average running time approximately 15 minutes. Due to the vast number of possible subset models (9.808×10^{55}) (and inherent

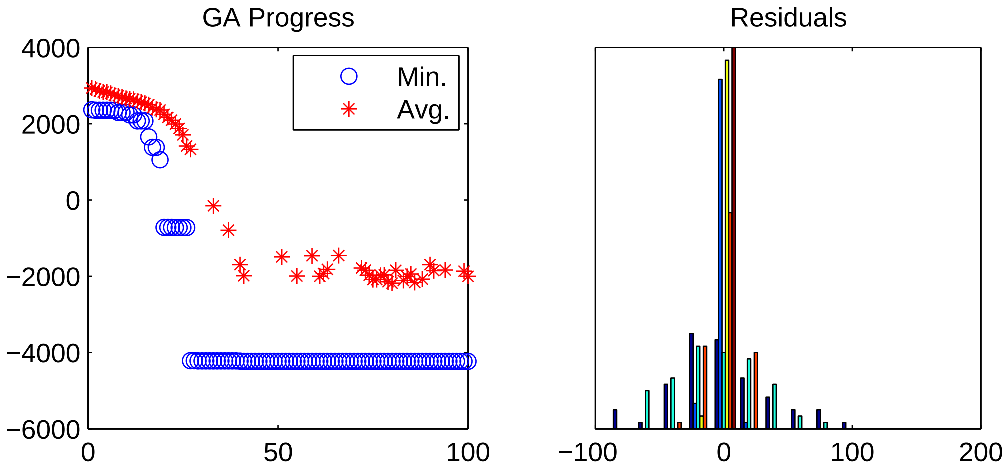

Figure 5: GA Progress and Residuals Plot from Best Subset Model.

randomness of the GA), there was much variety in results across all replications. According to the GA progress plot in Figure 5, the best replication had converged to its final solution before the 40th generation. The right pane of Figure 5 shows histograms of the residuals from the best subset model; note the general Gaussian look, though the residuals from some indices are clearly non-normal. Overall, though, there appears to be no significant departure from Gaussianity, using Mardia's tests. Additionally, we evaluated the condition of autocorrelation in the residuals when the model was used to predict the SPX daily moves. 1st through 6th order autocorrelations were -0.0635, 0.0016, 0.1242, -0.0102, -0.0193, -0.0175 - all very small. Finally, in assessing the OLS assumptions, we inspected the time-ordered plot of these residuals; there was no sign of heteroskedasticity.

The overall best model achieved a dramatic decrease in the number of parameters from the saturated model - approximately 80%. The 38 (out of 186) coefficients selected by the best model are shown in Table 7. For comparison purposes, we also performed 100 replications of the modeling process using AIC to drive the model selection. The best subset model identified by AIC used 82 predictors, more than twice as many as shown here - leading to less precise forecasts. It is interesting to note that this parsimonious model only utilized nine predictors to trade the target index, the four most important of which were:

- XAU - 5th lag - -0.393
- RUT - 2nd lag - 0.386
- SPX - 2nd lag - -0.347
- XAU - 4th lag - 0.155

According to this model, the SPX is mostly influenced by the Amex Gold Producers index, the Russell 2000 index, and itself. After the best VAR coefficients were determined by the genetic algorithm, the direction of daily moves was predicted by $signum(\widehat{SPX}_t)$. Interpreting these predictions as buy-long or sell-short signals, trading was simulated over the entire 6 month period, with uncompounded returns accumulated and shown in Figure 6. The model did not perform all that well over the in-sample period, racking up more than 3% in losses while the index was flat. In the testing period, however, while the index lost more than 15%, the model made more than 17% - far outperforming the benchmark. Who wouldn't be happy with +17% gains while the stock market lost so much ground? Out of the 100 replications, 90 of the models outperformed the index - for the training period. While only 15% lost money during the testing period, **ALL** outperformed the benchmark. In Table 8, we've provided results from the best seven models (associated with seven lowest $ICOMP$ values) across all simulations. Assuming we were performing the model training in real time, clearly we would have been more interested in the models with both low scores and high in-sample returns. Several of these models had very good out-of-sample performance. Thus, even if we were to pass over the best model according to $ICOMP$, the next several would perform admirably. In fact, the first two have such similar $ICOMP$ scores (not e^1 apart) that the models are indistinguishable. This table bolsters our confidence that out-of-sample performance of our procedure is **generally** good, despite the specific evolutionary trajectory followed by the population of chromosomes leading to this **specific** overall best solution. Finally, we address the precision of the forecast errors from this very parsimonious model, using the procedure detailed in Section 2.2. In the interest of simplified output, the entire

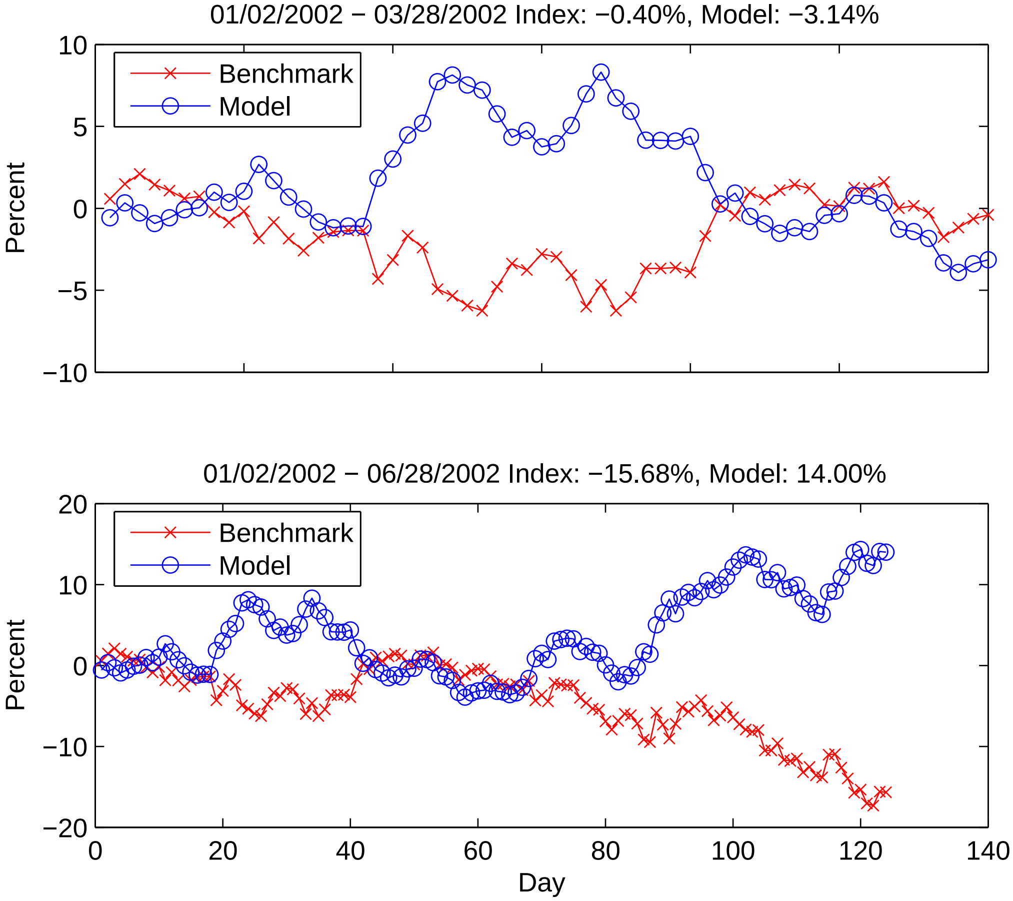

Figure 6: Trading Results from Best Subset Model.

forecast horizon was divided into increments of 10. Table 9 shows the MSE's for just the SPX index. Even as far as 100 observations into the future, we see that the subset model made substantially (order of 2 or better) more precise forecasts than the saturated model. We can also see this demonstrated graphically in Figure 7. It is interesting to note how similar the point estimates were for both models. At least on this wide scale, they overlap enough to blur the differences. We see, however, how much wider are the error bands ($\pm 2\hat{\sigma}$) when constructed using the saturated model. Thus, the methods presented have allowed us to build a model that is:

- not overly complex
- fits the data extremely well
- is very precise

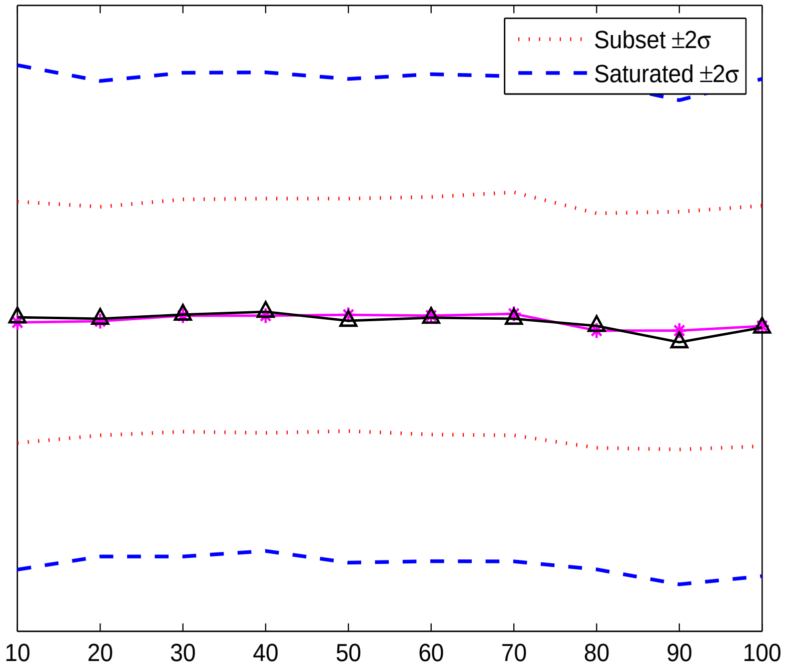

Figure 7: S&P 500 Forecast Point Estimates and Error Bands for Subset and Saturated Models.

Table 7: Estimated Coefficients from Best Subset VAR Model.

	DJ20	MID	NDX	RUT	SPX	XAU
CONST	8.130	·	·	·	·	·
DJ20(1)	·	·	·	·	·	·
MID(1)	0.459	·	·	·	−0.039	−0.008
NDX(1)	0.797	·	·	·	·	·
RUT(1)	−4.267	·	·	·	·	·
SPX(1)	·	·	·	−0.022	·	·
XAU(1)	·	·	·	·	·	·
DJ20(2)	·	·	·	·	·	·
MID(2)	·	·	·	·	·	·
NDX(2)	·	·	·	·	0.049	·
RUT(2)	·	·	·	0.133	0.386	·
SPX(2)	·	·	·	·	−0.347	·
XAU(2)	·	·	0.332	−0.022	·	·
DJ20(3)	·	·	·	·	·	·
MID(3)	·	0.157	·	·	·	·
NDX(3)	·	·	·	·	·	−0.012
RUT(3)	·	−0.192	·	·	−0.087	·
SPX(3)	·	−0.017	·	·	·	·
XAU(3)	−3.555	·	·	·	·	·
DJ20(4)	·	·	0.079	−0.007	·	·
MID(4)	4.454	·	·	·	·	·
NDX(4)	0.401	−0.014	·	·	·	·
RUT(4)	−4.938	0.025	−0.470	·	·	·
SPX(4)	−1.824	·	·	·	·	·
XAU(4)	·	0.174	−0.635	·	0.155	0.064
DJ20(5)	·	·	·	·	·	·
MID(5)	·	0.009	·	·	·	·
NDX(5)	·	·	·	·	·	·
RUT(5)	·	·	0.322	·	−0.058	·
SPX(5)	·	·	·	·	0.021	·
XAU(5)	−6.665	·	·	·	−0.393	·

Table 8: Trading Results from Best 7 Subset VAR Models.

Score	In-Sample %	Out-Sample %
−4284.11	−3.14	17.14
−4283.64	7.40	18.16
−4277.51	3.49	5.95
−4277.35	0.09	13.38
−4276.85	13.33	15.65
−4275.28	3.27	24.73
−4274.25	10.64	28.25

Table 9: Mean Squared Errors for S&P 500.

Step	$MSE_{SUB,i}$	$MSE_{SAT,i}$
10	154.2658*	322.3142
20	146.0149*	303.9236
30	148.3140*	309.1208
40	149.6718*	305.8698
50	148.4665*	309.1991
60	151.9430*	311.2008
70	155.3841*	310.0720
80	149.8774*	311.3456
90	151.9360*	309.3476
100	153.6293*	317.7458

6. Concluding Remarks

In this paper, we have set forth the idea that the genetic algorithm, along with the appropriate form of $ICOMP$, can be used to select a vector autoregressive model that exhibits accurate and efficient forecast performance. Our research suggests that, when modeling complex vector autoregressions, a strict criterion that is robust to model misspecification is required even when there is no sign of heteroskedasticity, non-Gaussianity, or autocorrelation in the model residuals.

We have demonstrated our claims by creating a valuable trading model for the Standard and Poor S&P 500 index from a universe of the first five lags of six major market indices. In our analysis, we have assumed there are no trading fees associated with implementing the model. Indeed, the best model switched positions (long/short) 34 times during the 60-day testing period, so trading fees could add up quickly. There are at least two mutual fund companies (ProFunds, Rydex) that offer market index tracking funds. They both allow daily trading, at least they used to, so this assumption is not too wildly made. Of course, in an effort to appropriately diversify, one would be interested in trading other indices as well. This method would be one voice in a portfolio of models. In Table 1, we noted the high multicollinearity between the indices modeled. A quick review of the table, however, will show that the Amex Gold Producer index XAU was only slightly negatively correlated with any of the others. Additionally, two lags of XAU were selected for SPX. This suggests that a valuable modification would be to include uncorrelated, and even negatively correlated predictors, in order to achieve a model that uses as much information as possible and is *market neutral*. To operationalize these methods, several implementation modifications would have to be made.

Past experience has shown us that over time, predictive relationships between indices change. There was a time (1998-2002) when a very naive model based on 1, 2, or 3 lags of the daily changes in the SPX, RUT, and MID could very effectively predict the indices daily moves. These relationships largely broke down after that period. Additionally, it has been seen that predictor indices can move in and out of a model quickly, even on a weekly basis. As such, this modeling procedure would need to be retrained periodically. Secondly, a question that must be answered is how long are the optimal training / testing periods. In this empirical work, we demonstrated our ideas using 3 months for each. It seems unlikely that the same indices would consistently retain their predictive power over entire quarters. It is anticipated that better results would be achieved by retraining the model more frequently. The appropriate training period is also an important consideration. If it is too short, the model may be unduly influenced by unstable short-term parameter shifts; however, if the period is too long, the model will miss trend reversals and react too slowly. We have seen economically valuable trading models with training periods as long as a year, and as short as two weeks. These two settings can be determined empirically using historical back-testing.

Lastly, there are computational issues to be considered with this type of model. The complexity of our problem requires consideration of the feasibility of subset selection, given existing computational power. While one simulation took at most 15 minutes, obtaining results across many simulations is recommended, due to the fact that there are 9.808×10^{55} possible subset VAR models. The 100 replications executed in approximately 24-25 hours - at most an estimated $1,500,000$ unique models were evaluated. While this seems large, it is insignificant compared to the number possible. Accelerating or distributing the computations so as to better search the model space would be a valuable contribution.

References

[1] H Akaike. Information theory and an extension of the maximum likelihood principle. In B.N. Petrox and F. Csaki, editors, *Second International Symposium on Information Theory.*, pages 267–281, Budapest, 1973. Academiai Kiado.

[2] L Bauwens and M Lubrano. Identification restrictions and posterior densities in cointegrated gaussian var systems. In T.M. Fomby and R. Carter Hill, editors, *Advances in Econometrics.*, volume 11b. JAI Press, Conneticut, USA, 1993.

[3] P Bearse and H Bozdogan. Subset selection in vector autoregressive models using the genetic algorithm with informational complexity as the fitness function. *Systems Analysis Modelling Simulation*, 31:61–91, 1998.

[4] P Bearse, H Bozdogan, and A Scholottman. Empirical econometric modelling of food consumption using a new informational complexity approach. *Journal of Applied Econometrics*, 12:563–592, 1997.

[5] H Bozdogan. Icomp: A new model-selection criteria. In H.H Bock, editor, *Classification and Related Methods of Data Analysis.* North-Holland, 1988.

[6] H Bozdogan and D Haughton. Informational complexity criteria for regression models. *Computational Statistics and Data Analysis*, 28:51–76, 1998.

[7] M Chen. Estimation of covariance matrices under a quadratic loss function. Research Report S-46, Department of Mathematics, SUNY at Albany, 1976.

[8] R Engle and C Granger. Cointegraion and error-correction: Representation, estimation and testing. *Econometrica*, 55:251–276, 1987.

[9] C Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometric*, 37:424–438, 1969.

[10] J Holland. *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, Michigan, 1975.

[11] J Holland. Genetic Algorithms. *Scientific American*, pages 66–72, 1992.

[12] O Ledoit and M Wolf. Honey, I Shrunk the Sample Covariance Matrix. Technical report, Universitat Pompeu Fabra, 2003.

[13] R Litterman. Forecasting with bayesian vector autoregressions - five years of experience. *Journal of Business and Economic Statistics*, 4:25–38, 1986.

[14] H Lütkepohl. *Introduction to Multiple Time Series Analysis.* Springer-Verlag, 1993.

[15] J Magnus and H Neudecker. *Matrix Differential Calculus with Applications in Statistis and Econometrics.* Wiley, 1988.

[16] K Mardia. Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies. *Sankhya*, B36:115–128, 1974.

[17] C Neely, P Weller, and R Dittmar. Is Technical Analysis in the Foreighn Exchange Market Profitable? A Genetic Programming Approach. *Journal of Financial and Quantitative Analysis*, 32(4):405–426, 1997.

[18] J Penm and R Terrell. On the recursive fitting of subset autoregressions. *Journal of Time Series Analysis*, 3:43–59, 1982.

[19] S Press. Estimation of a normal covariance matrix. Technical report, University of British Columbia, 1975.

[20] B Routledge. Adaptive Learning in Financial Markets. In *The Review of Financial Studies*, volume 12, pages 1165–1202. Oxford University Press, 1999.

[21] D Runkle. Vector autoregressions and reality. *Journal of Business and Economic Statistics*, 5:437–442, 1987.

[22] A Shurygin. The linear combination of the simplest discriminator and fisher's one. In Nauka, editor, *Applied Statistics.* Moscow, Russia, 1983.

[23] C Sims. Income and causality. *American Economic Review.*, 62:540–552, 1972.

[24] C Sims. Macroeconomics and reality. *Econometrica*, 48:1–48, 1980.

[25] C Thomaz. *Maximum Entropy Covariance Estimate for Statistical Pattern Recognition.* PhD thesis, University of London and for the Diploma of the Imperial College (D.I.C.), 2004.

[26] J West and Linster. The Evolution of Fuzzy Rules in Two-Player Games. *Southern Economic Journal*, 69(3):705–717, 2003.