Time Series Modeling with Genetic Programming
Relative to ARIMA Models

Miroslav Klúčík¹, Jana Juriová², Marian Klúčík³
¹INFOSTAT, Slovakia, klucik@infostat.sk
²INFOSTAT, Slovakia, juriova@infostat.sk
³Slovak University of Technology, Slovakia, marian.klucik@stuba.sk

Abstract

INFOSTAT, the research institution of the Statistical Office of the Slovak Republic, is intending to supplement its model tools (ECM, ARIMA) with modern heuristic methods for analyzing and forecasting the macroeconomic reality. The initial research is concentrated on time series modeling using genetic programming and comparing the results with a more conventional ARIMA model. Genetic programming tool based on evolutionary computation technique can find not only optimal parameters of a searched function but also its structure. Our experiments deal with modeling and forecasting of the industrial production for Slovakia and European Monetary Union. For our purpose the genetic programming tool is kept as simple as possible. The predicted variables are estimated by the concept of symbolic regression. The solutions of symbolic regression are expressed in a tree-type structure. Concerning the ARIMA approach, we have used seasonal ARIMA models that satisfied all the quality model conditions. Both methods’ performance was tested in a twelve-month forecasting. The second experiment involves the simulation of shocks for each model. The GP model manages to compete with ARIMA models in all cases. Finally we show a way to depict a complicated nonlinear solution in a simply understandable form. The continually changing and hardly predictable environment of contemporary and future global economy will require a multidisciplinary approach to approximate the complex reality. The GP instrument with its flexibility and efficiency manages to confront these challenges with promising results.

Keywords: Genetic programming, symbolic regression, ARIMA

1. Introduction

INFOSTAT (Institute of Informatics and Statistics) as a research institution of the Statistical Office of the Slovak Republic is engaged on a long-term basis in socio-economic analyses referring to time series analysis and macroeconomic modeling and forecasting. The econometric models include the ARIMA model approach and Error Correction Models (ECM). Nowadays, INFOSTAT is intending to supplement these model tools with heuristic methods for analyzing and forecasting the macroeconomic reality. Current global economy evolvement implies coming structural changes that the conventional econometric model tools will probably fail to capture. The volatile character of time series in dynamic environment can be more successfully analyzed using nonlinear methods. In this paper we apply a heuristic multidisciplinary method – Genetic
programming (GP) to estimate models for industrial production of Slovakia and European Monetary Union (EMU) and compare it with ARIMA models. GP using symbolic regression is kept as simple as possible and following the most recent experiences in the field (e.g. Poli, Langdon, McPhee, 2008). Concerning the ARIMA approach, we are using seasonal ARIMA models that satisfy all the quality model conditions. We compare a conventional linear model with a nonlinear GP model from two different aspects: real data and simulated data. Simulated data are presented as permanent shocks – a fall of industrial production for a period of approximately one year. This way we can compare the performance of both methods in a real-time period (last observations showing already the effects of the economic financial crisis) and in a future-time period with a significant shock (simulated continuing crisis).

2. Genetic programming model description

GP is a method based on genetic algorithms (evolutionary algorithms). Genetic algorithm is using the analogy of evolutionary process and genetic operations in order to solve a problem. During a process of evolution it transforms a set of population (mathematical objects) into a new population using the principles of reproduction and survival of the fittest. Generally, genetic programming enlarges the concept of evolutionary algorithms so that the population is not composed of mathematical objects but computer programs. GP is a stochastic method that explores the space of potential solutions (computer programs) in order to find the global optimum (Koza, 1992). The solutions are meant to be approximations of the searched near-optimum, not exact solutions. The computation process is connected with probabilities and random variables; i.e. the results are different by each run of the program.

Typically, the individuals (solutions) are represented as trees contrary to the line-code, e.g. the term \((x + y) * 3\) is depicted as a tree (the Picture 1 above). The advantages of symbolic regression lie in the creation of complex models with no assumptions about the structure (e.g. stationarity, cointegration) and no need for economic interpretation. On the other hand the classical regression prefers the linear (or quadratic) functions. In addition researchers are forced to mechanically combine numerous variables to find the optimal one, which is very time consuming.

In a nutshell the GP run contains following steps: the execution of the GP run begins with creation of random initial solutions (population). Depending on the fitness the best solutions are chosen to be parents. Next generations of solutions (offspring) are formed by means of genetic operations. This process iterates until the final termination criteria are met and final solution found.\(^1\)

\(^1\) For GP analysis a genetic programming system (under development) was used, written in C++ programming language and developed by Marian Kľúčik, ICII FEI STU (Slovak University of Technology), Bratislava, Slovakia
At the beginning a GP run requires setting up a number of initial parameters. The variables and constants \((x, y, 3)\) are the **terminal set**. The **function set** includes the operators (functions) – nodes of the trees - used in the transformation process \((+, *, /)\). The **measure of fitness** is the crucial parameter for searching the near-optimal solutions, e.g. absolute mean error, root mean squared error etc. The **GP run parameters** are very much depending on the problem domain. The results of the run depend on the chosen initial population size, initial population depth, probabilities of crossover and mutation, maximum size of programs, number of runs and other chosen control parameters. Also the **termination criterion** has to be set to limit the search for near-optimal solutions. Finally there is to decide about the interpretation or design of the **final result**.

In the mentioned steps of the GP run it is necessary to determine the following properties:

1. **Setting up the initial population** - The random population is set by grow method, full method or so-called “ramped half and half” method (combination of first two methods). In the first method the trees are formed as incomplete (like the individual on the Picture 1 - the number 3 (leaf) is different from the other side of the tree - branch \((x + y)\)). The full method contains subtrees (branches) of the same form (e.g. the same branch on the right side \((x + y)\)). Also the form of the random population creation is to be decided between uniform initialization, seeding and others. The chosen form is connected with the bias of the initial population, e.g. the seeding method allows putting individuals with better fitness among the random population. These individuals have better chance to transmit into next generations. Also the probability of better fitness of the final solution is higher.

2. **Selection of the best individuals** – Each random individual from the first step is evaluated by the chosen fitness measure. A method (and another parameter set) has to be chosen to ensure that the best fitting individuals have higher chance to survive and to enter the process of crossover (e.g. the tournament selection method or elitism method).

3. **Crossover and mutation** – Individuals with better fitness (parents) are recombined depending on the probability of crossover. Some properties of new individuals (offspring) enter the next generation with changed characteristics (mutation). Regarding the position and size of crossover and mutation specific method has to be chosen, e.g. subtree-, uniform-, content preserving- and size-fair crossover or subtree mutation, point mutation etc.

Given the details of general character of genetic programming settings, we decided to use the following properties for our problem domain: The first population (random) is set by means of the grow method and fully random initialization. The **grow method** is very simple and ensures the variety of the initial individual trees. The **depth of the initial population** is set to 3. The depth of the individual on the Picture 1 is 2 (the root is at depth zero). The tree depth varies usually between 2 and 6. For our purposes depth 3 is small enough to ensure a chance for simple solutions to be successful in the first few generations. The **terminal set** contains the variables \(x_1, x_2, ...., x_n\) (with different lags) and constants. The **function set** includes four basic operators \((+, -, *, /)\). The lag and moving average is applied directly to the time series and so they act as terminal set variables. The lag and moving average is limited to 12. In our case (time series forecasting) the chosen set is reasonable because it offers possibilities to predict variables on the basis of previous year and on the other side a higher lag would shorten the time series too much. For the **fitness measure** we used the mean square error (MSE), i.e. the MSE of predicted values against real (actual) values. The **population size** influences the diversity of population. A greater diversity of the population decreases the probability of getting stuck in a local optimum (not near-optimal solution). A very high number of the population size can be for
the GP program very time demanding. So the upper limit for the parameter set-up is limitation of time. In our GP tests we are using the population of 5000, which is a relatively high number regarding the input variables and number of operators. For the best individual selection we used the tournament selection method. The best-fitted individuals from a number of randomly chosen individuals are becoming the parents. The tournament method does not consider the difference between the values of fitness among individuals. It chooses only the best one. This ensures the fitness rescaling and non-decreasing diversity. The number of individuals to enter the tournament (tournament size) is another important GP parameter. The higher the number is the more quickly next generations are created. Where the number is high, there is also a higher probability that we will miss the global optimum. On the other side a low number can cause slowing down the program. We decided to set this value to 15, which could be an optimal value regarding the initial population size.

The process of crossover and mutation of the parents is demanding to set up the occurrence probability for each of the two operations. Among the researchers it is common to set the crossover probability to a very high level and the mutation probability to a very low level. We are using the probabilities of 0.9 and 0.01 respectively. A high crossover rate guarantees that the most of the parent’s good characteristics are transmitted to the offspring. On the other side a high mutation rate can cause a big loss of good genes. In our analysis the size-fair subtree crossover method was used. The mutation of genes was done by the simple point mutation method. The maximum length of the program is set as a control variable. In our case we determined the size to the maximum of 100 (the size of the program on Picture 1 is 5). This way we get results which are not too much computation demanding and otherwise also the individuals are protected from the so-called bloat process (excessive growth of the program size). Our termination criterion is the maximal number of generations. Commonly the number varies from 20 to 50 generations and the solutions tend to increase their fitness only during these early generations. After that the improvement is always smaller. Our initial tests with time series showed that after the 50th generation only small improvement could be observed. But we used the 200th generation as the limitation criterion, mostly because we could get better results without oversize the solutions (maximum program size criterion). The last criterion is number of runs. Each run gives one best-fitted individual as a result. So we tested the initial parameters with 100 runs to find the most suitable ones. The number of runs is closely associated with the parameter of initial population size. The size of population can be set as large as possible, but as to the time restrictions, the number of runs has to be then chosen very small. The aim is to find balance between these two parameters.

3. ARIMA model description

ARIMA models count among the most often used time series models in the forecasting practice. This time series approach relates current values of an economic variable only to its past (historical) values and to the values of current and past random disturbances. As a starting point, these time-series models are based on the assumption that the analyzed time series is stationary. Most of economic time series do not satisfy this condition, as well as the time series of industrial production index for Slovakia and EMU. So firstly the original time series have to become stationary. Thereafter the stationary time series serve for identification of model parameters.
3.1 ARIMA model for industrial production index of Slovakia

The time series of industrial production index (IPI) for Slovakia is non-stationary with respect to the mean level and also to the variance (as it can be seen on the Graph 1, next page). In this case the logarithmic transformation is fitting for original time series to become variance stationary. The first unseasonal differences are sufficient to make time series stationary also with respect to the unseasonal mean. However, the seasonal time series should be stationary also with respect to the seasonal mean level. The autocorrelation function (ACF) of transformed time series (after logarithmic transformation and first differences) shows statistically significant gradually decreasing coefficients in the positions with seasonal lag, i.e. with lag of 1 year. From this reason there were used first seasonal differences (i.e. differences with lag of 12 months). Such re-expressed time series has already been stationary also from the seasonal and unseasonal view and can be used for the selection of suitable ARIMA model. The pattern of ACF and partial autocorrelation function (PACF) for stationary time series serve for identification of parameters of a suitable model. As first a seasonal model has been found. Negative coefficients of ACF in positions with seasonal lag indicate the suitability of SMA process (seasonal moving average process). The exponentially declining ACF and first 2 statistically significant partial autocorrelation coefficients are typical of AR(2) process.

3.2 ARIMA model for industrial production index of EMU

The performance of industrial production index for EMU has a little different shape than for Slovakia. As it is seen on the Graph 2 (on the next page), the time series is non-stationary in level (like in the case of IPI for Slovakia), but stationary in variance. The second apparent difference is that the seasonality is much stronger for IPI of EMU. Using first unseasonal and seasonal differences is a sufficient transformation for the original time series to become stationary. On the basis of analyzed ACF and PACF for these stationary time series several models could be proposed, but as a seasonal model also the SMA process is suitable.

4. The GP - ARIMA comparison

In this section we compare the performance of both methods in modeling and forecasting of industrial production of Slovakia and EMU. Both time series enter the analysis in a seasonal unadjusted form. Firstly, the sample contains the real monthly data from January 1998 to December 2007 for industrial production of Slovakia and from January 1990 to December 2007 for industrial production of EMU. The resulting relations are used to predict the values from January 2008 to the last known real data of November 2008. The model performance is compared by their residuals behavior. The GP presents two versions of estimates – the first one (GP_1) estimates only on the basis of historical values (the same as ARIMA). The second one (GP_2) uses also other explanatory variables (business surveys). The ex post model forecast errors were evaluated and compared by a forecasting accuracy measure, namely the Root Mean Square Error (RMSE), given in the Formula 1 (1).
\[ RMSE = \sqrt{\frac{\sum_{i=1}^{l} (x_i - \text{est } x_i)^2}{l}} \] (1)

- \( l \) denotes the number of forecasts and \( \text{est } x_i \) the forecasting value of \( x_i \).

Secondly, simulated data of a permanent shock from December 2008 till December 2009 are added to the real data. Again both model residuals are compared. That way we can compare the performance of both methods in an environment with a changing trend.

Both methods have different approach to time series analysis. In ARIMA model the time series components – e.g. seasonal, white noise – are treated in a specific way. The GP does not distinguish between these components and take the time series as a whole. Both methods can predict the considered time series by their historical values, i.e. shifted (lagged) observations. The GP can also make a forecast on the basis of other time series. Both methods are not based on any economic interpretation. The advantage of GP modeling is that compared to ARIMA models it does not need any kind of assumptions (e.g. stationarity, invertibility). On the other hand, ARIMA concept is deeply reworked and detailed analyzed. Furthermore, experiences of GP users are, that GP does better mostly in areas that are not yet well understood (Poli, Langdon, McPhee, 2008).

4.1 Industrial production of Slovakia

Real data sample (January 1998 – December 2007)

For the modeling purposes monthly time series of industrial production index are used from January 1998 to December 2007, 120 observations totally. It is an index with the base of average month of 2000=100 (IPI00); its performance can be seen on the next picture (Graph 1).

On the basis of the analysis in Chapter 3 several ARIMA models have been proposed and estimated, but the most suitable seems the seasonal model ARIMA(2,1,0)(0,1,1)\(_{12}\) without constant (see Model 1, next page). The stationarity and invertibility conditions of the estimated ARIMA model are satisfied, all its parameters are statistically significant, its residuals are more or less white noise and the model explains almost 63% of variance for the analyzed variable. The GP model was designed in both versions with the following parameters: population size (5000), initial depth size (3), tournament size (15), maximum size of program (100), number of generations (200), crossover probability (0.90), mutation probability (0.01) and number of constants (100). The GP’s second version (GP_2) uses for the estimation in addition the time series from business surveys in industry (without lag and moving average transformation): production expectations, order books, production expectations (trend) and employment expectations.
The residuals of all three models are depicted on the Graph 3 (below). The forecast was performed for the sample January 2008 – November 2008. The results (RMSE) are compared in the Table 1. The ARIMA outperforms the GP_1 model. Although in Graph 3 we do not observe any distinct differences in the residuals between those models.

### Table 1

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIMA</td>
<td>9.32</td>
</tr>
<tr>
<td>GP_1</td>
<td>10.48</td>
</tr>
<tr>
<td>GP_2</td>
<td>9.26</td>
</tr>
</tbody>
</table>

### Simulation data sample (January 1998 – December 2009)

The simulated data are added to the real monthly data sample. The permanent shock is applied as a stable 10%-fall of monthly industrial production against the previous year. The economic crisis will probably cause volatile behavior and deeper falls but this paper’s purpose is to show only how the particular methods manage to estimate a significant break of trend in time series. The GP_2 model was not used, because the business survey time series would also need an extrapolation to the future. The ARIMA model’s parameters changed slightly – the break of the trend caused a fall in the variability explained by the model and also Durbin-Watson statistics (by
0.1 point). The addition of the permanent shock had effect on the GP computation process. Much more generations were needed to increase the fitness of the individuals than in the real data sample. However both methods managed to capture the change of trend after few months as it is seen on the Graph 4 (previous page - the shadowed part are the simulated data).

4.2 Industrial production of EMU

Real data sample (January 1990 – December 2007)
The data of industrial production for EMU are in the form of base index 2000 and are working day adjusted (WDA). The time series of industrial production index for EMU shows a significantly different seasonal pattern (Graph 2, page 6) compared to the industrial production of Slovakia. The GP should be able to explain such patterns with the set of variables included in the terminal set, mostly because it contains the twelve-month lagged variable that is able to eliminate most of the seasonality. The most suitable ARIMA model for this sample is the model ARIMA(0,1,3)(0,1,1)_12 without constant (see Model 3, next page). Again all the quality conditions of the model were satisfied (stationarity, invertibility). The parameters are significant at the 5% level. The model explains about 42% of the variance for the predicted variable.
The residuals for both ARIMA and GP models have problems about absorbing the shocks (trend breaks) in 1994/1995, 1997/1998, 2001/2002 and 2006. As in the Slovak IPI case, it is difficult to distinguish any considerable differences regarding the quality of the compared models. The 2008 forecasts show better results for GP models (Table 2 on the right).

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIMA</td>
<td>3.40</td>
</tr>
<tr>
<td>GP_1</td>
<td>2.44</td>
</tr>
<tr>
<td>GP_2</td>
<td>1.53</td>
</tr>
</tbody>
</table>

Simulation data sample (January 1990 – December 2009)
The simulation data are added again to the real data sample. The shock is represented as permanent 5% year-over-year fall of the EMU industrial production index (shadowed region in the Graph 6). As seen on ARIMA Model 4 (next page) the R-squared fell down and Durbin-Watson statistics has not changed. The GP model’s performance from the residual point of view is very similar to the ARIMA model. Again the breaking points are seen in the residuals series,
but none of the method has done distinctly better than the other one. Evidently the volatile period began already in the early 2008 and our shock-extrapolation is its continuation. The residuals have adapted more or less to the change of trend approximately after 3 quarters.

### 4.3 GP result interpretation

In this section we interpret graphically the final equation of industrial production of EMU from the GP_2 model based on real sample data. The tree structure of the final result is transformed into the following equation:

\[
\text{IPIEMU} = \frac{\text{PRTR} \cdot (\text{IPIEMU}(-4) \cdot \text{PRTR} \cdot \text{STPR}) \cdot \text{OB} \cdot \text{IPIEMU}(-10)) + (\text{IPIEMU}(-3) \cdot \text{EXOB} \cdot \text{IPIEMU}(-1))}{(\text{PRTR} \cdot \text{IPIEMU}(-7) \cdot \text{IPIEMU}(-1) \cdot \text{IPIEMU}(-4)) + (\text{EXOB} \cdot \text{STPR}) + ((\text{IPIEMU}(-12) \cdot \text{OB} \cdot \text{STPR})) + (\text{IPIEMU}(-12) \cdot \text{OB} \cdot \text{STPR}) + (\text{OB} \cdot \text{STPR}) + (\text{STPR}) + (\text{OB} \cdot \text{STPR}) + (\text{EXOB} \cdot \text{STPR}) + (\text{STPR}) + (\text{OB} \cdot \text{STPR}) + (\text{PRTR} \cdot \text{IPIEMU}(-3)) + (\text{PRTR} \cdot \text{IPIEMU}(-5) \cdot \text{EXOB} \cdot \text{PRTR} \cdot \text{STPR}))} + \text{RES}
\]

We computed the sum of each subtree of the final best-fitted individual for each observation in the sample 1991–2008 (including our forecast) without changing the tree’s structure (Picture 2, next page). Each subtree was given a letter from the alphabet beginning with the root. The root — letter A — is the final solution, i.e. the sum of the subtrees below (the whole equation). As we move further down, the equation is decomposed into its parts. The letter B and C are the two main trees forming the final result. As we can see on the picture, the C subtree markedly resembles the subtrees below — E, I, M and as we arrive at the bottom we find the IPIEMU(-12) variable. The other main subtree (B) resembles the time series of PRTR (Production trend). Remarkable is the character of the other subtrees regarding their distinct complexity. Except the IPIEMU(-12) and PRTR variables we can find cyclical time series (subtrees F, J, L, N), time series with extreme outliers (D, G, K) or seasonal behavior time series (H — a simple equation: IPIEMU(-12)/IPIEMU(-4)). To sum up, the IPIEMU dependent variable is explained mostly by the IPIEMU(-12) and PRTR explanatory variables and its performance adapted through variety of cyclical time series, time series with outliers, seasonal-like time series etc. This kind of analysis offers an insight into the structure of results without losing complexity of the final solution.

---

1. PRTR – Production trend, OB – Order books, EXOB – Export order books, PREX – Production expectation, STPR – Stock of finished products, RES – residual, IPIEMU(lag)
5. Conclusions

The GP analysis provided a new insight into the time series analysis. It yielded not only forecasting results at the level of ARIMA models, but also extended the view at the complexity of nonlinear modeling. The future analysis should be oriented at extension of the function set (sine, cosine) and further analysis of the results interpretation as given in the last section. The contemporary volatile economic environment will require analysis based on nonlinear models. The GP approach with its flexibility manages to confront these challenges with promising results.

References


