Warsaw School of Economics–SGH
Institute of Econometrics
Department of Applied Econometrics

ISSN 2084-4573

Department of Applied Econometrics Working Papers
Warsaw School of Economics–SGH
ul. Madalinskiego 6/8
02-513 Warszawa, Poland

Working Paper No. 4-12

Inflation forecasting using dynamic factor analysis. SAS 4GL programming approach

Adam Jędrzejczyk
Warsaw School of Economics–SGH, Poland

This paper is available at the Warsaw School of Economics
Department of Applied Econometrics website at: http://www.sgh.waw.pl/instytuty/zes/wp/

Inflation forecasting using dynamic factor analysis. SAS 4GL programming approach

Adam Jędrzejczyk

August 2012

Abstract

The purpose of this article is to introduce an original macro code written in SAS 4GL. This macro is used to automate the process of forecasting with dynamic factor analysis. Automation of the process helps to save significant amounts of time and effort for the researcher. It also enables to compare different model specifications directly and, hence, to make conclusions that would be imperceptible without such automation, which is shown on the empirical study example.

Keywords: statistical programming, forecasting, factor models, inflation

JEL codes: C22, C53, C80, E31

1. Introduction

Ongoing monitoring and inflation forecasting are among the main tasks of central banks. The predictive models currently used, however, commonly deliver forecasts of unsatisfying quality. For this reason, analytical methods to find more accurate predictors are being developed. One of the intensively researched forecasting methods is dynamic factor analysis. The method is gaining more supporters because of its good prognostic properties. Many articles comparing predictive ability confirmed the usefulness of dynamic factor analysis in forecasting.

In spite of the growing popularity of dynamic factor models, there continues to be a lack of dedicated effort-saving procedures in most statistical packages. This leads to the necessity of strenuous manual search for the best fitted model. Since this method usually requires using large data sets and estimating many models with different specifications, forecasting with dynamic factor analysis takes considerable amounts of time and effort. These prerequisites indicate the need to investigate whether there is a possibility of at least partially automating the research process.

This article shows that a significant part of the forecasting process with factor models can be automated. In section 2. the general idea of dynamic factor analysis, its history, the advantages and disadvantages of the method are discussed. Section 3. presents a theoretical approach to dynamic factor analysis. Issues like the idea behind and outline of the method, factor estimation, selecting the number of factors, and forecasting are described. The next section of the article is divided into two subsections. At first, an empirical study based on dynamic factor analysis is shown. Then, the construction of the macro code, its assumptions and functions are described. Section 5. summarizes the article.

2. Dynamic factor analysis

Due to the considerable dynamism and complexity of the determinants of economic processes, price levels are inconstant in time. Therefore, central banks perform an ongoing monitoring of the economy, constantly expanding the range of factors that could potentially affect the inflation rate. Among many economic indicators one should consider world trade indexes, the prices of goods and raw materials on international markets, monetary aggregates, labor market and property market indicators, indicators for manufacturing, interest rates, exchange rates, stock market indexes.

Direct usage of all available data (comprising hundreds or even thousands of time series) is usually not possible due to statistical limitations, particularly too many observations in relation to the length of the time series.

On the other hand, excessive or improper aggregation or omission of many relevant variables can result in inaccurate and biased forecasts. For this reason, methods to incorporate as many desired variables as possible into the model are investigated and developed, while assuring the stability and accuracy of forecasts.

One of the most commonly used methods that meets the aforementioned criteria is the dynamic factor analysis model (also called dynamic factor model-DFM). This method was first described by Geweke (1977) and Sims and Sargent (1977). These articles presented an innovative way of analyzing macroeconomic time series. Their starting point was the abandonment of the traditionally used macroeconomic structural models in favor of methods that use as little a priori information as possible. This approach was combined with an attempt to find unobservable factors responsible for the overall pattern of change in the analyzed phenomenon.

Geweke and Singleton (1981) contributed to the further development of the method. They showed that analytical methods appropriate for the classical hidden variables models also apply to hidden variables models based on time series. Engle and Watson conducted research on methods of estimation for such models. In (Engle, Watson 1981) they presented a DFM estimation algorithm using maximum likelihood estimation with the Kalman filter. Then, in (Engle, Watson 1983) two additional estimation algorithms were shown. Further considerations in this area were led by Stock and Watson (1991).

An important contribution was the article of Stock and Watson (1998), in which dynamic factor model was presented in the static form. This enabled to use the principal components method (well-known from static factor analysis) to determine the unobservable factors and model parameters. This approach to DFM estimation is currently very popular and was also applied in this study.

Dynamic factor analysis is based on the idea of patterns of change (called factors), which are common to all variables. Factors are unobservable variables determined from the covariance matrix of a set of predictors; they reflect the variability of this set in a synthetic way. From an economic point of view, factors have an atheoretical construction. This lets one avoid many assumptions concerning the shape of economic relations and structure of model (Kotłowski 2008). On the other hand, these unobservable variables do not have clear economic interpretations. In the literature they are usually identified as the driving forces of the economy (Stock, and Watson 1998).

There are many applications of dynamic factor models. In macroeconomic studies, this method is often used for acquiring indicators (indexes) and concluding from them directly. For example, Forni and Lippi (1997) pointed out the inadequacy of using excessively aggregated data in macroeconomic models. The authors conducted a study of relations occurring in the U.S. economy by extracting unobservable factors using factor analysis. Del Negro and Otrok (2007) used Bayesian DFM to infer about the synthetic state of the U.S. housing market. The article Altissimo et al. (2001), in which a business indicator for the Eurozone was constructed, is also worth mentioning. Similarly, Eickmeier (2004) researched

the impact of U.S. macroeconomic shocks on the German economy. In this study, a very large number of different economic values was used to build a multi-dimensional structural DFM.

Recapitulating, the biggest advantage of dynamic factor analysis is undoubtedly the possibility of including information from a large number of predictors without entering them into the model. This increases the number of degrees of freedom considerably and enables to avoid having to determine which variables should be rejected from the set of regressors in order to achieve traceability. Due to the atheoretical construction of factors, it is possible to avoid a priori assumptions about the shape of economic relations. An undeniable advantage of this method is also good forecasting performance of models based on the extracted factors, especially in the case of inflation forecasting (Kotłowski 2008; Baranowski, Leszczyńska, Szafrański 2010). Moreover, the results of such analysis are transparent (in terms of the significant reduction in size of the analyzed phenomenon). Another significant benefit is the considerable time and effort saved compared to traditional methods. The versatility of this method has contributed to its wide range of application.

Again, the biggest drawback of the method- the lack of a clear interpretation of the factors- results from its atheoretical character. Although sometimes the extracted factors can be economically interpreted, they are usually considered as unobservable driving forces. Another consequence of the concept of synthesizing information that underlies the DFM is the inability to clearly examine the impact of a specific regressor on the dependent variable, which is a big disadvantage compared to alternative methods, such as multiple regression. It seems that, due to the reasons described above, dynamic factor analysis is worse suited for examining the structure of phenomena. It may also be less transparent for people not acquainted with quantitative methods of analysis. Furthermore, there are relatively few sources describing the DFM method.

3. Dynamic factor analysis- a theoretical approach

This section describes the assumptions, structure, and method of estimation of the dynamic factor analysis model. At first, the overall concept of dynamic factor analysis is presented. The next subsection describes the DFM estimation method. Then, issues connected with the further specification of the model and building forecasts are discussed.

3.1. The basic concept

Dynamic factor analysis is an extension and generalization of the idea of static factor analysis. Above the static form, the dynamic approach assumes as well that any linear combination of factors can be extended by lags of common factors up to a maximum of order q. Such an approach allows for time series analysis. This is a significant extension of the applicability of factor analysis, as many economic values (especially in macroeconomics) are analyzed dynamically. Also, from the macroeconomic point of view extracting factors is reasonable because some macroeconomic theories imply the existence of unobservable

driving forces in the economy. In addition, extending the analysis in the dimension of time allows for forecasting, which is very important from the point of view of macroeconomics.

The structure of the dynamic factor analysis model is analogous to the static factor model. As already mentioned, the difference lies in including the time dimension. Consequently, the *i*-th variable has different forms depending on the analyzed period. The value of the *i*-th variable at moment *t* (*t* = 1, 2, ..., *T*) can be expressed as follows (Kotłowski 2008):

$$x_{i(t)} = \boldsymbol{f}_{(t)} \cdot \boldsymbol{\lambda}_{i(t)}(L) + e_{i(t)} \tag{1}$$

where:

$\boldsymbol{f}_{(t)} = [f_{1(t)} \quad f_{2(t)} \quad \cdots \quad f_{R(t)}]_{1\times R}$ – vector of common factors in period *t*,

$\boldsymbol{\lambda}_{i(t)}(L) = \begin{bmatrix} \lambda^1_{i(t)} + \lambda^1_{i(t-1)}L + \lambda^1_{i(t-2)}L^2 + \ \ldots + \lambda^1_{i(t-q)}L^q \\ \lambda^2_{i(t)} + \lambda^2_{i(t-1)}L + \lambda^2_{i(t-2)}L^2 + \ \ldots + \lambda^2_{i(t-q)}L^q \\ \vdots \\ \lambda^R_{i(t)} + \lambda^R_{i(t-1)}L + \lambda^R_{i(t-2)}L^2 + \ \ldots + \lambda^R_{i(t-q)}L^q \end{bmatrix}_{R\times 1}$ - lag polynomials with factor loadings vector,

e_{it} - idiosyncratic error of the *i*-th variable in period *t*,

q – lag order.

Equation (1) is called a dynamic factor model. It is also possible to present this model in the static form (without the lag polynomial), as long as the lag order is less than infinity ($q < \infty$). Let:

$$F_{(t)} = [\boldsymbol{f}_{(t)} \quad \boldsymbol{f}_{(t-1)} \quad \cdots \quad \boldsymbol{f}_{(t-q)}]'_{s\times 1} = \begin{bmatrix} f_{1(t)} \\ f_{2(t)} \\ \vdots \\ f_{R(t)} \\ f_{1(t-1)} \\ f_{2(t-1)} \\ \vdots \\ f_{R(t-1)} \\ \vdots \\ f_{1(t-q)} \\ f_{2(t-q)} \\ \vdots \\ f_{R(t-q)} \end{bmatrix}_{s\times 1}$$

where $s = R \times (q+1)$ - vector of common factors in periods *t*, *t*-1, ..., *t*-*q*. Furthermore, let:

$$\Lambda_i = [\lambda_{i(t)} \quad \lambda_{i(t-1)} \quad \dots \quad \lambda_{i(t-q)}]'_{s\times 1} = \begin{bmatrix} \lambda^1_{i(t)} \\ \lambda^2_{i(t)} \\ \vdots \\ \lambda^R_{i(t)} \\ \lambda^1_{i(t-1)} \\ \lambda^2_{i(t-1)} \\ \vdots \\ \lambda^R_{i(t-1)} \\ \vdots \\ \lambda^1_{i(t-q)} \\ \lambda^2_{i(t-q)} \\ \vdots \\ \lambda^R_{i(t-q)} \end{bmatrix}_{s\times 1}$$

- the vector of factor loadings at moments *t*, *t*-1, ..., *t*-*q*. Then:

$$x_{i(t)} = \Lambda_i' F_{(t)} + e_{i(t)} \tag{2}$$

Equation (2) is a dynamic factor model in the static from (Stock, Watson 1998). Similarly to static factor analysis, if there is no correlation between specific factors, i.e. if:

$$cov\left(\varepsilon_{i(t)}, \varepsilon_{j(q)}\right) = 0 \text{ for all } i \neq j \neq t \neq q, \tag{3}$$

model (2) is called an exact (or strict) factor analysis model (Baranowski, Leszczyńska and Szafrański 2010). If the condition (3) is not satisfied, then we define an approximate factor model (Chamberlain, Rotschild 1983).

To discuss the estimation of the dynamic factor model, it is convenient to use matrix notation for all *N* variables and *T* time periods. Let:

$$X_i = [x_{i(1)} \quad x_{i(2)} \quad \dots \quad x_{i(T)}]'_{T\times 1} \quad ,$$

$$F = [F_{(1)} \quad F_{(2)} \quad \dots \quad F_{(T)}]'_{T\times s} \quad ,$$

$$e_i = [e_{i(1)} \quad e_{i(2)} \quad \dots \quad e_{i(T)}]'_{T\times 1} \quad .$$

Now, equation (2) can be presented in vector form for the *i*-th variable:

$$X_i = F\Lambda_i + e_i \tag{4}$$

Introducing further notation:

$$X = [X_1 \quad X_2 \quad \dots \quad X_N]_{T\times N} \quad ,$$

$$\Lambda = [\Lambda_1 \quad \Lambda_2 \quad \dots \quad \Lambda_N]'_{N\times s} \quad ,$$

$$e = [e_i \quad e_i \quad \dots \quad e_i]_{T \times N} \quad ,$$

the full factor model for all *N* variables and *T* periods may be presented in the following form (Kotłowski 2008):

$$X = F\Lambda' + e. \tag{5}$$

To sum up, equation (5) defines the factor model in the static form. The matrix $F_{T \times s}$ contains *s* unobservable factors which are common to all predictors. The matrix $\Lambda_{N \times s}$ consists of factor loadings (constant in time), which are the coefficients of linear combinations of these factors. Idiosyncratic components which are elements of $e_{T \times N}$ represent specific variability (variability unexplained by common factors). Thus defined, the model will be subject to further considerations.

3.2. Factor estimation

The most common method of DFM model estimation is the principal components method. Due to its computational simplicity and well-known asymptotic properties of its estimators under general assumptions, (Stock, and Watson 1998) this method is very popular. Stock and Watson (1998) first applied this method to estimate a dynamic factor model in the static form.

The principal components method allows one to find the two unknown matrices of equation (9), i.e. the factor loadings matrix $\Lambda = [\lambda^r_{i(t-q)}]$ and the factors matrix $F = [f_{r(t-q)}]$ in a single step.

Equivalently, model (5) can be presented as:

$$X = F\mathrm{H}\mathrm{H}^{-1}\Lambda' + e \tag{6}$$

where $\mathrm{H}_{s \times s} \neq \mathrm{I}$. In order to obtain the factor matrix *F*, Stock and Watson (1998) propose the imposition of the condition $\frac{\Lambda'\Lambda}{N} = \mathrm{I}_s$ to ensure matrix H orthonormality.

The estimation of matrices *F* and Λ using the principal components method consists in finding such $\hat{F}$ and $\hat{\Lambda}$ estimators that minimize the sum of squared residuals of the equation (5) (Bai & Ng 2002):

$$V(F, \Lambda) = \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(x_{it} - \Lambda_i' F_t)^2 \tag{7}$$

To obtain estimates of the matrix, the first step is to minimize the expression (7) with respect to *F* under the assumption that the matrix Λ is fixed. As a result of this operation, the estimate $\hat{F}$ as a function of Λ will be obtained. Then, the estimate $\hat{\mathrm{F}}$ is reinserted into equation (7), which has only one unknown matrix Λ after this

operation. In the next step, we minimize the modified equation (7) with respect to Λ under the condition $\frac{\Lambda'\Lambda}{N} = I_s$, obtaining $\hat{\Lambda}$.

Subsequent columns of matrix $\hat{\Lambda}$ are composed of eigenvectors of *X'X* multiplied by $\sqrt{N}$. These vectors correspond to *s* highest eigenvalues of *X'X* matrix. To obtain the estimated values of *F*, we use the following formula:

$$\hat{F} = \frac{(X\hat{\Lambda})}{N}. \tag{8}$$

The condition $\frac{\Lambda'\Lambda}{N} = I_s$, which is applied to the function (7) in the second step of estimation, is a statistical condition imposed arbitrarily. For this reason, factors found in equation (8) cannot be interpreted economically.

It is worth noticing that Stock and Watson (1998) point out that, if the number of variables is larger than the number of observations (*N*>*T*), the more efficient algorithm is to solve (7) with respect to the matrix Λ under the assumption that the matrix *F* is fixed. However, in terms of forecasting on the basis of the dynamic factor model, it is not relevant which of the methods described will be applied (Kotłowski 2008).

What should also be emphasized is the fact that there are methods of estimating DFM models other than principal components. Most notably, principal component analysis with spectral estimation of the lag order and maximization of the likelihood function using the Kalman filter should be mentioned. As emphasized by Baranowski, Leszczyńska and Szafrański (2010), the choice of the factor estimation method does not significantly affect the quality of predictions in models with the same number of factors. For this reason, this article concentrates only on the principal components method.

3.3. Selecting the number of factors

The literature on the subject points to many ways to solve the problem of selecting the number of factors in the model. The most common include:

1. The explanation level of input variables' variance. The researcher arbitrarily sets a level of variance (usually 60%-70%), and selects the smallest number of factors explaining at least that percentage of the variance.

2. Including into the analysis only those factors, whose corresponding eigenvalues of the covariance matrix are greater than 1.

3. Rejection of all factors which explain a smaller part of the input variables' variance than the established level.

4. The a priori assumption of a specified number of factors.

5. Producing a graph with the consecutive eigenvalues and finding the inflection point that separates high eigenvalues from low ones (scree test).

6. Information criteria proposed by Bai and Ng (2002). The criteria allow not only for the selection of the number of factors, but also the lag order in the dynamic factor model.

As suggested by Baranowski, Leszczyńska and Szafrański (2010), methods for selecting the number of factors give the optimal number of factors in terms of the description of the variability of a set of predictors (*X*). This, however, may not be optimal in terms of predicting the dependent variable (*y*).

3.4. Forecasting

The idea of forecasting based on the dynamic factor model was proposed by Stock and Watson (1998) in a diffusion index model. This model introduced diffusion indexes, which presented the common variation of a set of predictors. The method used to extract the unobservable diffusion indexes was factor analysis. Given the current notation, the diffusion index model can be represented as follows (see Baranowski, Leszczyńska and Szafrański 2010):

$$y_{(t+h)} = \alpha_0 + \sum_{p=0}^{P} \gamma_{(t-p)}\, y_{(t-p)} + \\ + \sum_{q=0}^{Q} \sum_{r=1}^{R} \beta^{r}_{(t-q)} f_{r(t-q)} + u_{(t+h)} \qquad (9)$$

where:

y – forecasted variable,

h – forecast horizon,

p – autoregressive lag order, $p = 1, ..., P$,

$u_{(t+h)}$ – idiosyncratic error.

In equation (9), in addition to the part containing the diffusion indexes (double sum in equation (9)), there is also an autoregressive part with lag order *p* (single sum in equation (9)). In this equation, the forecast for period $t + h$ was obtained directly from the information available in period *t*. Thus, this forecast was obtained in a non-iterative way (directly). Empirical research for DFM models indicate that such way of prediction has better performance (Breitung, Eickmeier 2006). As emphasized by Baranowski, Leszczyńska and Szafrański (2010), models with small number of factors have good prognostic properties. Also, the authors draw attention to the fact that none of the DFM models containing only

variables in the real categories, nominal categories, and both nominal and real have no clear prognostic advantage over the others (Angelini, Henry, Mestre 2001).

4. DFM model-based forecasting automation

In section 1. of this study the need to construct an algorithm that automates the process of building DFM models and forecasting with them has been indicated. Due to the large number of variables used in DFM modeling and the lack of existing procedures dedicated to this class of models, forecasting based on dynamic factor analysis is currently not very efficient. There are significant opportunities for increasing efficiency by automating most of the research activities through statistical programming.

This section is divided into two subsections. At first, an empirical study based on dynamic factor analysis is shown. The analysis has been performed using an original macro for forecasting with DFA. Then, the construction of this macro code, its assumptions and functions are described.

4.1. Empirical study

The data used in the study are monthly data for the period from January 2000 to October 2010. In total, 52 time series (each 130 observations long) were used. They represent the basic macroeconomic values: prices, social benefits, foreign trade, exchange rates, monetary aggregates, money supply, stock indexes, labor market, interest rates, indicators of economic activity. The full list of variables used in the study is presented in Appendix 1.

The collected data have been appropriately transformed before their inclusion into the analysis. Variables were divided into three groups. The first group is time series characterized by seasonal fluctuations. These fluctuations have been removed by the procedure of ARIMA-X12[1]. Then, the variables from this group were logarithmized and first order differenced. This group includes variables such as price indexes, the volume of exports and imports in constant prices, money supply aggregates M1 and M3. The second group of variables consists of time series that are not characterized by seasonal fluctuations, but required logarithmizing and taking first order differences (in order to obtain stationary series). This group includes exchange rates and stock indexes. The third group consists of variables that do not show seasonality and do not require logarithmizing. These are, among others, variables presented in the form of fractions (economic indicators, interest rates). For this group of variables, the only transformation made is taking first differences. As in the Kotłowski (2008) study, it was arbitrarily assumed that all variables are integrated in the first

[1]ARIMA-X12 is the algorithm proposed by the U.S. Census Bureau used to decompose time series. Extracting the components of time series is based on a class of moving average filters, also known as X-11 filters. The term ARIMA (AutoRegressive Integrated Moving Average) occurring in the name arises from the application of this class of models to calculate the theoretical values. Obtaining theoretical values is necessary to extend the series, which is essential to obtain the values needed to determine the moving averages at the ends of the series. The X12-ARIMA algorithm was also used in Kotłowski (2008).

order. For this reason, to ensure stationarity of the analyzed time series, the variables from all groups were only first order differenced.

The diffusion index model (9) was used to produce forecasts. Static factor analysis was used to extract factors from the analyzed data set (which contains also the forecasted variable). Then, the model (9) was created. As a comparative model the autoregressive model (AR) was used:

$$y_{(t+h)} = \alpha_0 + \sum_{p=0}^{p} \gamma_{(t-p)} y_{(t-p)} + u_{(t+h)} \qquad (10)$$

Due to the fact that the article's purpose is not examining the predictive capabilities of DFM, and that there are many studies raising this issue (Kapeniatos, Labhard, Price 2008; Kotłowski 2008; Baranowski, Leszczyńska, Szafrański 2010), only one comparative model was used (10).

In this example, an inflation-related variable, i.e. the procurement price of wheat was used as the explanatory variable. The out-of-sample forecast has been made for the forecast horizons h = 1 and h = 3. The criterion for selecting the best model was mean square error (MSE). The maximum number of extracted factors is 12.

The estimation results are as follows: 12 factors have been extracted from the analyzed data set. Nine models with lag order up to 8 have been estimated. The notation of the models is as follows: F_Lag [], where [] is the maximum lag order. For example, the notation F_lag4 indicates a model with 12 factors with lag order up to the fourth (inclusive). Thus, this model has
5 * 12 = 60 explanatory variables.

Regarding to comparative models, the best fitted model of the autoregressive models class was AR (1). Further lags were not statistically significant.

Table 4.1.1. presents MSE values for each model with the forecast horizon h = 1.

Table 4.1.1. The values of the models' mean square error with the forecast horizon *h* = 1

Model	MSE	%MSE (F_lag7)
F_lag0	0,007488964	266%
F_lag1	0,008047651	286%
F_lag2	0,006619311	235%
F_lag3	0,00580022	206%
F_lag4	0,005879009	209%
F_lag5	0,004612012	164%
F_lag6	0,004536832	161%
F_lag7	**0,002813814**	**100%**
F_lag8	0,004289446	152%
A_lag1	0,007827712	278%

Source: Own calculations

The results presented in Table 4.1.1. indicate that the best predictive model is F_lag7, i.e. model with 12 factors and lag order up to 7. Mean square error of the model is much lower than the MSEs of the other models, in particular than the MSE of the comparative AR(1) model. Thus, the described DFM has much better predictive properties than the autoregressive model. This confirms the results of previous studies (Kotłowski 2008, Baranowski, Leszczyńska, Szafrański 2010). Nevertheless, in the illustrated case, the model with a large number of lags is much better than models with a reduced number, which is contrary to what is suggested by, among others, Baranowski, Leszczyńska, Szafrański (2010). At this point, the advantage of the automation approach is visible. It would be very effort-demanding to make the above conclusions without an automation macro.

Because F_lag7 model is the best prognostic model, it will be further discussed. Model F_lag7 contains 12 * 8 = 96 variables. None of the autoregressive parameters prove to be statistically significant[2]. Therefore, this model requires the estimation of 97 parameters (intercept and 96 variables). R-squared is 94%. However, it should be emphasized that this value is overestimated by a large number of explanatory variables in the model. The value of the mean square error is 0.0028 and is significantly lower than in the competitive models (see table 4.1.1.).

Due to the complex form of the model and the large number of factors it is not possible to interpret the factors. However, as stated in section 2., in terms of forecasting, the factors' interpretability is not required.

[2] This result is not surprising, since the requirement of the presented macro is that the dependent variable is included in the analyzed data set. Information contained in this variable is therefore included (at least partly) in the extracted factors. For this reason, it can be expected that relatively often the autoregressive part in the diffusion index models is negligible. However, it seems that the arbitrary a priori exclusion of the autoregressive is too strong an assumption.

Figure 4.1.1. presents a fragment of the outcome data set for F_lag7 (see next section) that contains information about the model fit statistics and estimates of parameters. Since no estimate of the autoregressive parameters proved to be statistically significant, the values of these parameters in the set are marked as missing data (symbol '.', not shown in Figure 4.1.1.).

Figure 4.1.1. Fragment of the F_lag7 data set

VIEWTABLE: Parameter Estimates and Statistics

	Label of Model	Convergence Status	Name of Dependent Variable	Estimate of Variance	Sum of Squares Error	Intercept Parameter	Parameter Estimate for Factor1_lag0	Parameter Estimate for Factor2_lag0	Parameter Estimate for Factor3_lag0
1	F_lag7	0 Converged	CENY_PSZENICA_D11	0.0028138136	0.0675315264	0.0064074326	-0.017910723	0.0166124207	-0.022256577

Source: Own calculations

Figure 4.1.2. shows a fragment of the output set F_lag7_pred containing all the variables of the model, as well as the theoretical values (column ‘yhat’) together with the lower and upper confidence limits (columns ‘lcl’ and ‘ucl’ respectively). The theoretical values cover the range of modelled data and the additional observations which are forecasted. In this case the forecast horizon h = 1, so in the output data set F_lag7_pred there is one additional observation containing the forecasted value (row 131 in column "yhat").

Figure 4.1.2. Fragment of the F_lag7_pred data set

VIEWTABLE: Work.F_lag7_pred

	yhat	lcl	ucl	CENY_PSZENICA_D11	Factor1_lag0	Factor2_lag0	Factor3_lag0
120	-0.037497639	-0.178514034	0.1035187568	-0.000738301	1.5801060059	-0.378106955	0.4190180463
121	0.084207412	-0.060125356	0.2285401798	0.0425105989	0.4375978775	-0.083225001	0.6085728675
122	-0.108368499	-0.254577583	0.0378405854	-0.110233493	-0.568562463	1.0525839641	-0.002164027
123	-0.031363621	-0.177250104	0.1145228614	-0.010238976	0.61041306	-1.310891696	0.7994653225
124	0.0450158746	-0.10256737	0.1925991191	0.0310034257	1.3729949785	-0.510509201	0.9630866452
125	0.0656793838	-0.082173817	0.213532585	0.0380453975	-0.004010046	0.9360439381	0.1925855897
126	0.0284254834	-0.120729583	0.1775805494	0.0380383449	-1.904516389	0.3895011394	-0.149238566
127	0.0721716438	-0.075728887	0.2200721748	0.0671262552	-0.721690268	0.1619420448	-0.115180245
128	0.2764919873	0.1304260238	0.4225579508	0.2752860284	1.2678953442	0.4711738223	-0.746502517
129	0.0042226996	-0.143214818	0.151660217	-0.019572998	-0.446588308	0.7858625304	1.1564567022
130	0.0077830323	-0.139014935	0.1545809994	0.0198181292	1.4569451574	0.01838125	-1.242711599
131	-0.079510335	-0.262153586	0.1031329158	.	0.5394798507	0.2171127086	-0.809173144

Source: Own calculations

The next step of the analysis is to forecast with horizon greater than h = 1. Table 4.1.2. presents results for the forecast horizon h = 3 analogous to those presented in Table 4.1.1.

Table 4.1.2. The values of models' mean square error in the forecast horizon h = 3

Model	MSE	%MSE (F_lag6)
F_lag0	0,006802523	136%
F_lag1	0,005664764	113%
F_lag2	0,006467925	130%
F_lag3	0,005947402	119%
F_lag4	0,005452566	109%
F_lag5	0,005518686	111%
F_lag6	**0,004993473**	**100%**
F_lag7	0,006033056	121%
F_lag8	0,006187312	124%
A_lag1	0,007827712	157%

Source: Own calculations

For the forecast horizon h = 3 the best prognostic model is F_lag6, but its advantage is not as overwhelming as it was in the case of the model F_lag7 for h = 1. Models with fewer lags have mean square errors that are not considerably higher. It seems that a model with only a first order lag can be successfully applied.

The above example shows that even when predicting one economic value for only two forecast horizons, at least twenty models have to be estimated. Considering the common practice to make forecasts for a larger number of variables with multiple forecast horizons, the previously indicated need to automate at least part of the research activities seems to be indispensable.

4.2. The DFA_Forecast macro

The main purpose of this study is to create a computer program (a macro) that automates the process of obtaining forecasts from diffusion index models. This program is written in the SAS 4GL language, which is one of the most common statistical systems in the world.

The main task of the DFA_Forecast macro is to produce forecasts for the indicated variable within a particular forecast horizon. To ensure full macro automation, autonomous modules that process the intermediate results were designed. It was also necessary to ensure the compatibility of the consecutive modules. For these reasons, in order to increase the readability of the code, the macro level structure was separated by functionality.

The description of the consecutive modules of the macro with regard to their functionality is presented below. In the program code the modules are described in comments (see Appendix 2).

0. Data transformation

In the described empirical study, which is an example of the macro's functioning, the variables used have been divided into groups according to their type of transformation (see section 4.1.). Point 0 of the below description does not directly belong to the macro code, it is an example of transformation of input data.

0.1. Variables with the transformations: seasonal adjusting, logarithmising, first order differentiation

0.1.1. Seasonal adjusting

0.1.2. Logarithmising, first order differentiation

0.2. Variables with the transformations: logarithmising, first order differentiation

0.3. Variables with the transformations: first order differentiation

0.4. Merging the results

1. Dynamic factor analysis

Extracts the factors from the indicated data set. Three criteria for selecting the number of factors have been arbitrarily assumed: the eigenvalue of each isolated factor must be at least 1, the cumulative explanation degree of the input data set's variance must be at least 0.6, the maximum number of separate factors is "Nfactors" where "NFactors" is a parameter whose value is given by the user. The procedure selects a number of factors, which is the minimum of the values indicated by the above eligibility criteria. In practice, the value of the last of the formal criteria determines the final number of factors most often, therefore only this criterion is parameterized. The more advanced user can manually change the value of the other criteria in the code for the macro.

In this part of the code the variable that specifies the maximum lag order taken into account is also automatically declared. It has been arbitrarily assumed that this value is [0.15*n], where [] denotes the integer part, and n is the number of observations in the analyzed data set. It seems that this value is large enough to take into account all lag distributions possible to obtain (in a statistical sense) or theoretically justified. For example, in the analysis described in section 4.1., it was possible to estimate models with lag order not greater than eight (this resulted from the large number of extracted factors- each addition of a subsequent lag order reduced the number of degrees of freedom by 12).

1.1. Transforming the results

At this point, essential transformations are performed on the intermediate outcomes to ensure further modules' compatibility.

1.1.1. Factors

Among others, lags of extracted factors are calculated. They will be used further to build the diffusion index model.

1.1.2. Forecasted variable

The user declares a variable from the input data set, which is further considered as a dependent variable.

2. Index model forecast

2.1. Preparing the data set in terms of forecasting

2.2. Index model's forecast

2.2.1. Forecast for different lag distributions

Diffusion index models for all lag distributions defined in point 1 are estimated. In each model there is also an autoregressive part added (in accordance with formula (9)). The macro automatically examines the statistical significance of all the lags of the response variable up to order 10 inclusive (this value has been chosen arbitrarily[3]) and estimates only the autoregressive parameters, which are considered as statistically significant (the backward elimination method). Then, for each model, the forecast is calculated according to (9).

2.2.2. Putting the results into the outcome table

2.3. Autoregressive model's forecast

The autoregressive model with backward elimination is estimated with backward elimination. Similarly to models from point 2.2.1., only lags up to 10 are taken into account. For the final form of the model, the forecast is calculated according to (10).

2.4. Choice of the best model

The criterion for selecting the model with the best predictive properties is mean square error (MSE). The macro indicates the model for which the value of this criterion is the lowest.

3. DFA_Forecast macro

The command that runs the macro DFA_Forecast. The user calls the whole code saved in a macro by submitting a single line. The five parameters of the macro have to be declared: the name of the library where the input data set is saved (parameter LibName), the input set name (DataSetName), the name of the forecasted variable (PredVar; this variable must be included in the input data set), the parameter defining the maximum number of extracted factors ("NFactors") (see point 1), the parameter "FrcstHorizon", which defines the horizon of the forecast. It should be noted that the first three parameters are text type, the other two are numeric type.

Furthermore, note that (in case $h > 1$) only the last value prediction is consistent with the model (9). For example, selecting FrcstHorizon = 3 ($h = 3$), will set to forecast for the period ($t+1$), ($t+2$), ($t+3$), but only the forecast for period ($t + 3$) is consistent with the model (9).

All intermediary data sets as well as the result data sets are saved in the temporary system library Work. This ensures that the user's library containing the set of data is not filled up with unnecessary sets. However, it should be noted that the contents of the Work library are cleaned by the end of the SAS session. For this reason, the result data sets, if necessary, should be copied to another location.

The most important outcome data set of the macro is the Result data set. It indicates the model with best predictive properties. Values of mean square errors for all estimated models

[3] As mentioned earlier, it is expected that in many cases estimates of all of the autoregressive part's parameters are statistically insignificant.

are stored in the set F_lags (on the basis of this data set Table 4.1.1. and 4.1.2. have been prepared). Detailed information for each model can be found in sets F_lag[] and F_lag []_pred. For example, in the case of the F_lag7 model, parameter estimates and selected fit statistics of this model can be found in the F_lag7 set. The forecasted values with upper and lower confidence limits are in the F_lag7_pred set. In the case of the autoregressive model (see point 2.3), these sets are named as A_lag and A_lag_pred respectively.

The described program is available for download at the Internet address http://akson.sgh.waw.pl/~aj39683/DFA_Forecast.sas. The macro code has also been presented in Appendix 2.

The macro DFA_Forecast presented above is fully automated. Because of that, the user does not have to understand the programming aspects of the analysis. Deep knowledge of dynamic factor analysis is also not required. Despite this, to use the macro fully consciously, a good knowledge of DFM methods is suggested. In addition, to fully take advantage of the macro (such as changing the value of these formal criteria which have not been parameterized), at least a basic knowledge of SAS programming is recommended. It is, however, not necessary to efficiently obtain good quality forecasts using the diffusion index model.

In section 4.1. an example empirical analysis was presented. In this example, to achieve a fair comparison of the predictive ability of DFM models with different lag distributions and with the comparative model, it was necessary to estimate twenty models. Taking into account the fact that all compared models had the same number of factors, there is a much larger number of potential forecasting models which should be considered. Comparing all models by manually setting up their estimation would require very large amounts of time and work. Moreover, the amount of labor and time needed to conduct research increases several times when the analyst needs to get forecasts of more than one economic value. In the face of these facts it is reasonable to automate the researcher’s labor, at least in the technical part of the work. Although the final decisions about the shape of the researched relationships, model specification, and forecasting should belong to the researcher, some elements of the research process can be automated without loss of quality of the final results, thus greatly contributing to raising the effectiveness of the effort. The macro presented in this article automates most of the research activities in the process of forecasting based on dynamic factor analysis.

5. Summary

This study presents a SAS macro created to automate most of the research process while forecasting with dynamic factor analysis. As an example of the usage of the macro, the procurement price of wheat was predicted. The basis for this was 52 times series (each 130 observation long) representing the basic macroeconomic values in Poland for the period from January 2000 to October 2010.

The results of the study confirm that for one- and three- month forecast horizons DFM models perform better than autoregressive model. The study also indicates that a significant number of models with different specifications should be estimated to choose the model with best forecasting properties.

The macro automating most of the research activities of the process of forecasting based on dynamic factor analysis was presented. The program automatically produces forecasts for diffusion index models, depending on the lag distributions and autoregressive part. Then, the results are confronted with the comparative model. The program indicates which model has best forecasting properties measured by mean square error. On this basis, the researcher is free to compare the models and make the final choice of the model, as well as see the estimated parameters from the intermediary data sets. This saves the researcher’s time and labor– they can choose the best model without the tedium of estimating all the models manually. At the same time, conscious use of the macro avoids operating with a "black box" - an algorithm with unknown rules- while keeping full control over the research process. This allows for a more thorough examination of the shape of economic relations and the construction of forecasts based on them.

Further research is planned into expanding the program with statistical tests of significance between the forecasts of the models and the automation of model estimation with regard to the number of extracted factors. These actions could help to further improve the program.

References

1. Altissimo, F., A. Bassanetti, R. Cristadoro, M. Forni, M. Hallin, M. Lippi, L. Reichlin, (2001), *EuroCOIN: a real time coincident indicator of the euro area business cycle*, CEPR Working Paper 3108.
2. Angelini E., Henry J., Mestre, R. (2001), *Diffusion Indexes-based Inflation Forecasts for the Euro Area*, European Central Bank, Working Paper 061.
3. Bai J., Ng S. (2002), Determining the number of factors in approximate factor models, *Econometrica*, 70 (1), 191-221.
4. Baranowski P., Leszczyńska A., Szafrański G. (2010), Krótkookresowe prognozowanie inflacji z użyciem modeli czynnikowych, *Bank i Kredyt*, 41 (4), 23-44.
5. Breitung J., Eickmeier S. (2006), Dynamic Factor Models, *AStA Advances in Statistical Analysis*, 90 (1), 27-42.
6. Brzoza-Brzezina M., Kotłowski J. (2009), Bezwzględna stopa inflacji w gospodarce polskiej, *Gospodarka Narodowa*, 20 (9), 1–21.\\
7. Chamberlain, G., Rothschild, M. (1983), Arbitrage Factor Structure, and Mean-variance Analysis of Large Asset Markets, *Econometrica*, 51 (5), 1281–1304.
8. Costello A., Osborne J. (2005), Best Practices in Exploratory Factor Analysis: Four Recommendations for Getting the Most From Your Analysis, *Practical Assesment, Research & Evaluation*, 10 (7).
9. Cristadoro R., M. Forni, L. Reichlin, G. Veronese (2005), A Core Inflation Indicator for the Euro Area. *Journal of Money, Credit and Banking*, 37 (3), 539-60.
10. Del Negro M., Otrok C. (2007), 99 Luftballons: Monetary Policy and the House Price Boom Across U.S. States, *Journal of Monetary Economics*, 54 (7), 1962-1985.
11. Eickmeier, S. (2004), *Business cycle transmission from the US to Germany – a structural factor approach*, Bundesbank Discussion Paper nr 12/2004.
12. Engle, R., Watson M. (1981), A One-Factor Multivariate Time Series Model of Metropolitan Wage Rates, *Journal of the American Statistical Association*, 76, 774-81.
13. Forni M., Lippi M. (1997), *Aggregation and the Microfoundations of Dynamic Macroeconomics*, Oxford University Press, New York.
14. Gamst G., Guarino A., Meyers L. (2006), *Applied multivariate research. Design and interpretation*, Sage Publications Ltd., London.
15. Geweke J. (1977), The dynamic factor analysis of economic time series, ch. 19 [w:] Aigner, D.J., A.S. Goldberger (ed.), *Latent variables in socio-economic models*, North-Holland Pub. Co., Amsterdam.
16. Geweke J., Singleton K. (1981), Latent variable models for time series: A frequency domain approach with an application to the permanent income hypothesis, *Journal of Econometrics*, 17 (3), 287-304.
17. Kapetanios G., Labhard V., Price S. (2008), Forecast Combination and the Bank of England's Suite of Statistical Forecasting Models, *Economic Modelling*, 25 (4), 772-792.
18. Kotłowski J. (2008), *Forecasting inflation with dynamic factor analysis- the case of Poland*, Warsaw School of Economics, Warsaw.

19. Lackey N., Pett M., Sullivan J. (2003), *Making sense of factor analysis. The use of factor analysis for instrument development in health research*, Sage Publications Ltd., London.
20. Larose D., (2006), *Data mining methods and models*, John Wiley & Sons Inc, Hoboken, New Jersey.
21. Ostasiewicz W. (1998), *Statystyczne metody analizy danych*, Wydawnictwo Akademii Ekonomicznej im. Oskara Langego we Wrocławiu, Wrocław.
22. Ptak-Chmielewska (2009), Metoda głównych składowych i analiza czynnikowa [w:] Frątczak E., Gołata E., Klimanek T., Ptak-Chmielewska A., Pęczkowski M. (2009), *Wielowymiarowa analiza statystyczna. Teoria- przykłady zastosowań z systemem SAS*, Oficyna Wydawnicza Szkoły Głównej Handlowej, Warszawa.
23. Sargent, T., C. Sims (1977), Business cycle modelling without pretending to have too much a-priori economic theory, [w:] C. Sims (ed.), *New methods in business cycle research*, Federal Reserve Bank of Minneapolis, Minneapolis.
24. Statsoft (2010), *Składowe główne i analiza czynnikowa*, http://www.statsoft.pl/textbook/stathome_stat.html?http://www.statsoft.pl/textbook/stfacan.html%23index.
25. Stevens J. (2002), *Applied multivariate statistics for the social sciences*, Lawrence Erlbaum Associates, Inc., New Jersey.
26. Stock J., Watson M. (1991), A Probability Model of the Coincident Economic Indicators, [w:] Lahiri K., Moore G. (ed.), *Leading Economic Indicators: New Approaches and Forecasting Records*, ch. 4., Cambridge University Press, New York, 63-85.
27. Stock J., Watson M., 1998, *Diffusion Indexes*, National Bureau of Economic Research, Working Paper 6702.
28. Thurstone L. (1931), Multiple factor analysis, *Psychological Review*, 38, 406-427.
29. Tucker J., (1996), *Neural networks versus logistic regression in financial modeling: a methodological approach*, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.54.1759.
30. Vansteenkiste I. (2009), *How Important are Common Factors in Driving Non-Fuel Commodity Prices? A Dynamic Factor Analysis*, ECB Working Paper 1072.
31. Welfe A. (2003), *Ekonometria*. *Metody i ich zastosowanie*, Polskie Wydawnictwo Ekonomiczne, Warszawa.

Appendix 1

List of variables used in the study.

Variable	Name	Seasonally adjusted	Transformation performed
Prices			
Alcohol and tobacco prices	CENY_ALK_TYTON	yes	Δ ln
Procurement price of cattle	CENY_BYDLO	yes	Δ ln
Education	CENY_EDU	yes	Δ ln
Export transaction prices	CENY_EXPORT	yes	Δ ln
Import transaction prices	CENY_IMPORT	yes	Δ ln
Recreation and culture	CENY_KULTURA	yes	Δ ln
Telecommunications	CENY_LACZN	yes	Δ ln
Housing maintenance and energy prices	CENY_MIESZK_UZ	yes	Δ ln
Home furnishing and housekeeping	CENY_MIESZK_WYP	yes	Δ ln
Milk	CENY_MLEKO	yes	Δ ln
Average expected inflation rate in 12 months horizon	CENY_OCZEK_INFL	yes	Δ ln
Clothes and footwear	CENY_ODZIEZ	yes	Δ ln
Procurement price of wheat	CENY_PSZENICA	yes	Δ ln
Transport	CENY_TRANSPORT	yes	Δ ln
Procurement price of pigs	CENY_TRZODA	yes	Δ ln
Health	CENY_ZDROWIE	yes	Δ ln
Food	CENY_ZYWN	yes	Δ ln
Fractions of answers on inflation expectations survey			
Fraction (1) will grow faster than now	FRAKCJA1	no	Δ
Fraction (2) same growth as now	FRAKCJA2	no	Δ
Fraction (3) will grow slower	FRAKCJA3	no	Δ
Fraction (4) will be same as now	FRAKCJA4	no	Δ
Fraction (5) will be lower	FRAKCJA5	no	Δ
Fraction (6) hard to say	FRAKCJA6	no	Δ
Oil	OIL	no	Δ

Retirement			
Average monthly gross old-age pension	EMERYT_NROL	yes	Δ ln
Average monthly gross old-age pension in the agricultural social insurance system	EMERYT_ROL	yes	Δ ln
Stock exchange			
Dow Jones Industrial Average	DOWJONES_INDAVG	no	Δ ln
NASDAQ Index	NASDAQ	no	Δ ln
Crude Oil Future	OIL_FUTURE	no	Δ ln
S&P500 Index	SP500	no	Δ ln
WIG Index	WIG	no	Δ ln
WIG20 Index	WIG20	no	Δ ln
Foreign trade			
Balance of payments	BILANS_PLATN	no	Δ
Exports (in constant prices)	EKSPORT	yes	Δ ln
Imports (in constant prices)	IMPORT	yes	Δ ln
Exchange rates			
EUR/PLN	EURPLN	no	Δ ln
USD/PLN	USDPLN	no	Δ ln
Government			
Balance of the state budget	SALDO_BUDZETU	no	Δ
Money			
Foreign assets	AKTYWA_ZAGR	yes	Δ ln
Deposits of households	DEP_GD	yes	Δ ln
Deposits of enterprises	DEP_P	yes	Δ ln
Currency in circulation	GOTOWKA	yes	Δ ln
Loans to households	KRED_GD	yes	Δ ln
Loans to enterprises	KRED_P	yes	Δ ln
M1 aggregate	M1	yes	Δ ln
M3 aggregate	M3	yes	Δ ln
Real properties			
Number of completed dwellings	MIESZK	yes	Δ ln
Labor market			
Number of registered unemployed	BEZROBOT	yes	Δ ln

Interest rates			
Rediscount rate	STOPA_WEKSLE	no	Δ
Business climate indicators			
Overall situation in construction	OG_BUD	no	Δ
Overall situation in trade	OG_HAND	no	Δ
Overall situation in industry	OG_PRZET	no	Δ

Appendix 2

DFA_Forecast macro code in SAS 4GL. Point 0 of the description below does not belong directly in the macro code, it is an example of the transformation of input data.

```
/****** Emptying WORK library ******/

proc datasets lib=work memtype=data kill nolist;
quit;

/****** 0. Data transformation ******/
      /**** 0.1.  Variables with the transformations: seasonal adjustment,
logarithmizing, first order differentiation ****/
            /** 0.1.1.  Seasonal adjustment **/
proc x12 data=dfa.data date=TimeID seasons=12 noprint;
var   CENY_PSZENICA CENY_BYDLO      CENY_TRZODA CENY_MLEKO
CENY_ZYWN   CENY_ALK_TYTON    CENY_ODZIEZ CENY_MIESZK_UZ    CENY_MIESZK_WYP
      CENY_ZDROWIE      CENY_TRANSPORT    CENY_LACZN
CENY_KULTURA      CENY_EDU    CENY_EXPORT CENY_IMPORT EMERYT_NROL EMERYT_ROL
      EKSPORT     IMPORT      DEP_P DEP_GD      KRED_P
KRED_GD     GOTOWKA     AKTYWA_ZAGR M1    M3    MIESZK      BEZROBOT
      CENY_OCZEK_INFL
;
transform power=0;
arima model=((0,1,1)(0,1,1));
estimate;
x11;
output out=sa d11;
run;


proc datasets lib=work nolist;
      modify sa;
            attrib _all_ label =' ' ;
quit;


            /** 0.1.2.  Logarithmizing, first order differentiation **/
proc expand data=sa out=sa (drop = TimeID);
convert _ALL_
/     TRANSFORMIN =(LOG DIF 1);
run;

                  /* --> output: sa */

      /**** 0.2.  Variables with the transformations: logarithmizing, first
order differentiation ****/
proc expand data=dfa.data (drop =
TimeID      CENY_PSZENICA CENY_BYDLO      CENY_TRZODA CENY_MLEKO
CENY_ZYWN   CENY_ALK_TYTON    CENY_ODZIEZ CENY_MIESZK_UZ    CENY_MIESZK_WYP
      CENY_ZDROWIE      CENY_TRANSPORT    CENY_LACZN
CENY_KULTURA      CENY_EDU    CENY_EXPORT CENY_IMPORT EMERYT_NROL EMERYT_ROL
      EKSPORT     IMPORT      DEP_P DEP_GD      KRED_P
KRED_GD     GOTOWKA     AKTYWA_ZAGR M1    M3    MIESZK      BEZROBOT
      CENY_OCZEK_INFL )


out=sa2;
```

```
convert USDPLN    EURPLN      WIG   WIG20 NASDAQ      SP500 OIL_FUTURE
      DOWJONES_INDAVG
/     TRANSFORMIN =(LOG DIF 1);

      /**** 0.3.  Variables with the transformations: first order
differentiation ****/
convert BILANS_PLATN    FRAKCJA1    FRAKCJA2    FRAKCJA3    FRAKCJA4
      FRAKCJA5    FRAKCJA6    SALDO_BUDZETU
OG_PRZET    OG_BUD      OG_HAND     OIL   STOPA_WEKSLE
/     TRANSFORMIN =(    DIF 1);
run;

                  /* --> output: sa2 */

      /**** 0.4.  Merging the results ****/
proc SQL;
      create table dfa.data_tf as
      select * from sa AS sa
      inner join  sa2 AS sa2
      on sa.time=sa2.time;
Quit;


/*************************************************************************
************************************
*************************************************************************
************************************
*************************************************************************
************************************/


/*options nosymbolgen nomprint nomlogic;*/

%MACRO DFA_Forecast (LibName, DataSetName, PredVar, NFactors,
FrcstHorizon);

Proc datasets library=work kill nolist;
quit;

/****** 1.  Dynamic factor analysis ******/

data _null_;
  set &LibName..&DataSetName nobs=nobs;
  nlags=FLOOR(nobs*0.15);
  call symput('nobs',compress(put (nobs,11.)));
  call symput('nlags',compress(put (nlags,11.)));
run;

proc factor data=&LibName..&DataSetName out=dfa noprint
      method=prin
      vardef=df
      singular=1E-08
      mineigen=1
      proportion=0.6
      nfactors=&NFactors
      priors=one
      rotate=none;
var _all_;
run;
```

```
      /**** 1.1.  Transforming the results ****/
            /** 1.1.1.  Factors **/
data dfa_factors;
set dfa (keep= factor:);
run;

%let mlist=' ';
%MACRO Mmlist;
data _null_;
set dfa_factors  (firstobs=&nobs);
length list $ 200;
%do i=1 %to &NFactors;
      if Factor&i=. then list=trim(list)||''||trim("Factor&i");
%end;
call symput ('mlist', list);
put list;
run;
%MEND Mmlist;
%Mmlist;
%put &mlist;

data dfa_factors;
set dfa_factors (drop = &mlist);
run;

proc sql;
      create table nvar as
      select nvar from sashelp.vtable
      where libname='WORK' and memname='DFA_FACTORS' ;
quit;
data _null_;
  set nvar;
  call symput('NF',compress(put (nvar,11.)));
  stop;
run;
%put  &NF;


%MACRO FactorLags;
data dfa_factors;
set dfa_factors;
array DataLags (*) _ALL_ ;
%do j=0 %to &nlags;                /*lags loop*/

            %do i=1 %to &NF;       /*variables loop*/
                  Factor&i._lag&j= lag&j(DataLags(&i));
            %end;
%end;
drop Factor1-Factor&NF;
run;
%MEND;

%FactorLags;


            /** 1.1.2.  Forecasted variable **/

data dfa_PredVar;
set dfa (keep = &PredVar );
run;
```

```
data dfa_all;
merge dfa_PredVar dfa_factors;
run;

/****** 2.  Index model forecast ******/
      /**** 2.1. Preparing data set in terms of forecasting ****/

data dfa_indexForecast;
set dfa_factors (obs=&FrcstHorizon);
call missing (of _all_);
run;

proc sql;
insert into dfa_indexForecast
select * from dfa_factors;
quit;

data dfa_indexForecast;
merge dfa_PredVar dfa_indexForecast  ;
run;

      /**** 2.2.  Index model's forecast ****/
            /** 2.2.1.  Forecast for different lag distributions **/
%MACRO AutoRegF (reg_nlags);
proc autoreg data=dfa_indexForecast outest=F_Lag&reg_nlags noprint;
                  F_lag&reg_nlags: model &PredVar= factor1_lag0--
factor&NF._lag&Reg_nlags
                  /nlag=(1 2 3 4 5 6 7 8 9 10)
                  backstep;
                  output out=F_Lag&reg_nlags._pred Predicted=yhat lcl=lcl
ucl=ucl;
                  run;
                  quit;
%MEND;

%MACRO AutoRegF_loop;
data _null_;
%do j=0 %to &nlags;
      %AutoRegF (&j);
%end;
run;
%MEND;
%AutoRegF_loop;

            /** 2.2.2.  Putting the results into the outcome table **/
data F_lags;
set F_lag0 (keep = _MODEL_ _MSE_ );
run;

%MACRO SQL_insert (reg_nlags);
      proc sql;
      insert into F_lags
      select _MODEL_, _MSE_  from F_lag&reg_nlags;
      quit;
%MEND;

%MACRO SQL_insert_2;
data _null_;
%do j=1 %to &nlags;
      %SQL_insert(&j);
```

```
%end;
run;
%MEND;
%SQL_insert_2;

proc sql;
create table F_lags as
select * from F_lags
where _MSE_ <>0  ;
quit;

      /**** 2.3.  Autoregressive model's forecast ****/
proc autoreg data=dfa_indexForecast outest=A_Lag noprint;
                  A_lag: model &PredVar=
                  /nlag=(1 2 3 4 5 6 7 8 9 10)
                  backstep;
                  output out=A_Lag_pred Predicted=yhat lcl=lcl ucl=ucl;
                  run;
                  quit;


proc sql;
      create table A_lags as
      select _MODEL_, _MSE_  from A_lag;
quit;


      /**** 2.4.  Choice of the best model ****/
proc sql;
      insert into F_lags
      select * from A_lags;

      create table F_lags as
      select * from F_lags
      where _MSE_ <>0 ;

      create table Result as
      select * from F_lags
      having _MSE_=min(_MSE_);
quit;

%MEND DFA_FORECAST;

/****** 3.  DFA_Forecast macro ******/

%DFA_Forecast(LibName= dfa, DataSetName= data_tf, PredVar=
CENY_pszenica_d11, NFactors= 12, FrcstHorizon= 1);

/*%MACRO DFA_Forecast (LibName, DataSetName,
PredVar,NFactors,FrcstHorizon);*/
```