

A Structured Bayesian Approach to ARMA
Time Series Analysis. Part IV: Program
Descriptions

John Monohan

Inst. of Statistics Mimeo Series #1627

I. Introduction

The program BAYFAC and its subsidiary analysis programs GOODO, GRATO, PLFIR and PL1FR are designed to do a fully Bayesian analysis of autoregressive-moving average (ARMA) time series models, as described by the author that appeared in the *Journal of Econometrics*, 1983. The purpose of this note is to describe these programs. Working knowledge of that paper will be assumed.

Speaking generally, the task is to take as input the data and the parameters of the prior and produce statistics that describe important aspects of the posterior distribution, viz. probabilities on M , mean and variances of the ARMA parameters (ϕ, θ) , of the series mean μ and disturbance precision R , and of the forecasts.

The primary computational effort is performed by BAYFAC. It receives as input the data and the parameters of the prior and produces a wealth of only partially processed information about the posterior that, for purposes here, will be written on a file called ZENBU. The other programs read this file and produce tables and figures. The first, GOODO, produces summary tables like Tables 6 and 7, directly from ZENBU. The next GRATO, receives as input the prior probabilities on the order of the model, $p(m)$ and produces tables like Table 8, also, after reading ZENBU. Two others, PLFIR and PL1FR, are designed to read ZENBU and produce files of abscissas and ordinates of the posterior distribution of the forecast(s), which can be executed by SAS/GRAPH to produce a graph similar to Figure 1 in the case of PLFIR, but in color. In the case of PL2FR, values of $p(m)$ are also required as input, and a single curve of the forecast posterior density is produced.

II. Standard Options

A) Number of integration points in BAYFAC

As currently designed, the numerical integration in BAYFAC is performed as follows:

for (1,0) and (0,1) 64 point midpoint rule

for (1,1) 32 x 32 product midpoint rule

for (2,0) and (0,2) 40^2 point triangular midpoint rule.

Thus 1024 and 1600 points are used for the last two types of regions. To change this, as described in the Program Documentation, in the first input line on line 1, the fields 5 through 8 must be filled with four 9's, that is NEWNP = 9999. This tells BAYFAC to read new values of NP1, NP11 and NP2, whose default values are 64, 32 and 40, respectively, as described above. Since this has a great bearing on the computational cost, this should not be changed unless necessary. For a very long series (NOBS is large) the likelihood would be expected to be more peaked and more points are likely to be needed to achieve the same level of accuracy.

B) Data format in BAYFAC

If the data a_t , $t = 1, \text{NOBS}$ are stored in a format (1X, 10F5.1), then BAYFAC is ready to read it. Since this is rare, the format can be adjusted to suit. As described in the Program Documentation and in the Sample Job Setups, in the first input line on unit 1, the fields 9 through 12 must be filled with four 9's, that is NEWFMT = 9999. This tells BAYFAC to read a new format for the data that must fit in 20 characters, including parenthesis, for example:

(1X, 4F5.2) .

If this is not adequate, then the driver must be changed accordingly and compiled and linked to the many subroutines.

C) Computing more tables in GRATO

As described in the Program Documentation and the Sample Job Setups, GRATO is designed to produce summary tables describing the posterior for each set of posterior probabilities, $p(m)$. Along with $p(m)$, a flag I is read in fields 2 and 3 which, if it indicates a positive integer, will cause a table to be produced from the accompanying values $p(m)$. To stop the processing, set I to 0 or negative.

III. Changing the Code

A) Unit numbers

These programs adhere to the IBM convention of reading on unit 1 and writing on unit 3 and most everything else is free. Here, either 2 or 9 is used as an additional output unit and 8 is used as an additional input unit. For different computer systems, these units should be adjusted to conform the conventions.

B) Priors on the ARMA parameters in BAYFAC

As it is currently written, BAYFAC uses a uniform prior distribution $\pi(\psi|M)$ for the ARMA parameters, as reflected in FUNCTION subprograms PIP1 and PIP2. To change this, FUNCTION subprograms must be written that are similar in type and number of arguments to PIP1 and PIP2 in lines 8940 through 9030 in the BAYFAC code. Then for the different models, the last argument in the CALL statements (300, 340, 380, 420, 460) must be changed to match the new $\pi(\psi|M)$'s. Also, their names must also appear in the EXTERNAL and the type declaration REAL*8 statements (20 and 30), in addition to or in place of PIP1 and PIP2. A final warning is that the new $\pi(\psi|M)$ must be normalized, that is, its integral over the relevant region (interval, square, triangle) must be unity.

C) Other forecasts

In GOODO, PLFIR and PL1FR, only forecasts for the first period ahead are made. This may be changed in the various programs by merely changing a few variables from 1 to something else and changing some appropriate FORMAT statement. In GOODO, this means changing three 1's in line 1330 and another in line 1410. In PLFIR, only the 1 in line 640 and FORMAT statement 70 (line 270) appear to need adjusting. In PL1FR, when a single curve is to be plotted, the similar changes are a 1 in line 780 and again a FORMAT 70 (but line 190).

IV. Caveats

The programs described herein are neither guaranteed nor warranted. The author pledges to seek and try to exterminate any bugs that may dare to raise its ugly head in the execution of the programs. Also, since the author seeks no profit in publishing these programs, the only copyright protection that is sought is that established in Common Law.

Many of the common implements of self-destruction for these programs have been disabled; however, for various reasons, some should be pointed out.

- 1) In the subroutines DO??D, there is a machine-dependent constant "-32" in statements labelled "4" and previous to each. These are designed to avoid underflow.
- 2) In GOODO and GRATO, when percentile points of the forecast distribution are computed, a natural cubic spline interpolates a set of points. If the interpolant is not nonnegative, some points are clipped from the two ends and a new spline is fit. If half of the points have been clipped and the fitted spline still dips to be negative somewhere, the problem is considered

hopeless and no percentile points found. This possibility is annoying and the user may want to disable this error trap. He must be warned that if the spline is somewhere negative then its cumulative is not monotone and some percentile points may not be unique.

- 3) Along the lines of (2) above, the rooting - finding algorithm for finding percentile points has an iteration limit (MIT) of 15, which can be adjusted.
- 4) One missing error trap that the author now sees is the lack of a check whether the data have been read in correctly.

IV. Reference

Monahan, John F., 1983, Fully Bayesian analysis of ARMA time series models, *Journal of Econometrics* 21, 307-331.

Program Documentation

Name: BAYFAC

Purpose: Computes Bayes factors, posterior means and variances of parameters and forecasts, and forecast posterior distributions.

Input:

Unit 1: NOBS, prior parameters, series name

Unit 8: Data

Output:

Unit 3: Bayes factors and normalization constant

Unit 9: Intermediate results on the posterior distributions to be processed by other programs.

Specifications:

- 1) $5 < \text{NOBS} < 500$
- 2) The number of points for integration are $\text{NP1} = 64$ for (1,0) and (0,1), $\text{NP11} = 32$ for (1,1) (32 x 32) and $\text{NP2} = 40$ for (2,0) and (0,2), (40 x 40 triangular). To: change this, make $\text{NEWNP} = 9999$ and add CARD1A
- 3) Preset format for data is (1X,10FS.1). To change make $\text{NEWFMT} = 9999$ and add CARD1B
- 4) To change prior on (ϕ, θ) , see Section IIIB.

Format for Input:

Unit 1:	Format
CARD1 NOBS,NEWNP,NEWFMT	(3I4)
CARD1A NP1,NP11,NP2	(3I4)
CARD1B format for reading data	(5A4)
CARD2 $\gamma, \tau, \alpha, \beta$	(1X,4F15.9)
CARD3 label for data	(3A4)
Unit 8:	
Observations 1 through NOBS	(1X,10F5.1)

Program Documentation

Name: GOOD0

Purpose: To produce tables of statistics from the posterior distributions for each of the six models. The table will be similar to Tables 6 and 7 in the *Journal of Econometrics* paper.

Input:

Unit 1: The entire file created by BAYFAC

Output:

Unit 2: Table of posterior information

Specifications: None

Format for Input: Not relevant

Program Documentation

Name: GRATO

Purpose: Produces tables (similar to Table 8 in the *Journal of Econometrics* paper) of posterior probabilities and posterior means, variances and percentiles for forecasts. One table for each set of prior probabilities on the order of the model.

Input:

Unit 1: The entire file created by BAYFAC

Unit 8: Stop code, prior probabilities

Output:

Unit 2: Table of posterior information

Specifications:

- 1) The variable I serves as the stopcode which stops the program. If $I > 0$ then the program will try to produce another table using the accompanying prior probabilities. If $I \leq 0$, then the program stops.

Format for Input

Unit 1:

I, p(m) for $m = 1, \dots, 6$

Format

(1X, I2, 6F8.6)

Program Documentation

Name: PLFIR

Purpose: To produce a file with commands and data. When processed by SAS/GRAPH and sent to a Tektronix 4662A flat bed plotter, a plot of forecast densities for each of the six models can be produced. The plot will be a color version of Fig. 1 in the *Journal of Econometrics* paper.

Input

Unit 1: The entire file created by BAYFAC

Output

Unit 2: A file of SAS/GRAPH commands for producing a color plot.

Specifications: None

Format for Input: Not relevant.

Program Documentation

Name: PL1FR

Purpose: To produce a file of commands and data. When processed by SAS/GRAPH and sent to a Tektronix 4662A flat bed plotter, a plot of the density of the first period forecast can be produced.

Input:

Unit 1: The entire file produced by BAYFAC

Unit 8: Prior probabilities on the order of the model

Output:

Unit 2: A file of SAS/Graph commands for producing a plot

Specifications: None

Format for Input:

Unit 8:

$p(m)$ for $m = 1, \dots, 6$

Format

(1X,6F8.6)

Sample Job Setups

```
00010 //BAYFAC JOB NCS.ES.B4912,ITSU,T=3
00020 // EXEC FTHCLG
00030 //C.SYSIN DD DSN=NCS.ES.B4912.ITSU.BAYFAC.FORT,DISP=SHR
00040 //G.FT03F001 DD SYSOUT=A
00050 //G.FT01F001 DD *
00060 100 19999
00070 (1X,4F5.2)
00080 4.000000000 1.000000000 4.000000000 4.000000000
00090 FULLER 641
00100 //G.FT08F001 DD DSN=NCS.ES.B4912.ITSU.FUL641.DATA,DISP=SHR
00110 //G.FT09F001 DD DSN=NCS.ES.B4912.ITSU.ZENBU.DATA,DISP=OLD
00120 //
```

```
00010 //GOODO JOB NCS.ES.B4912,ITSU
00020 // EXEC FTHCLG
00030 //C.SYSIN DD DSN=NCS.ES.B4912.ITSU.GOODO.FORT,DISP=SHR
00040 //G.FT03F001 DD SYSOUT=A
00050 //G.FT02F001 DD SYSOUT=A
00060 //G.FT01F001 DD DSN=NCS.ES.B4912.ITSU.ZENBU.DATA,DISP=SHR
00070 //
```

```
00010 //GRATO JOB NCS.ES.B4912,ITSU
00020 // EXEC FTHCLG
00030 //C.SYSIN DD DSN=NCS.ES.B4912.ITSU.GRATO.FORT,DISP=SHR
00040 //G.FT03F001 DD SYSOUT=A
00050 //G.FT02F001 DD SYSOUT=A
00060 //G.FT01F001 DD DSN=NCS.ES.B4912.ITSU.ZENBU.DATA,DISP=SHR
00070 //G.FT08F001 DD *
00080 1 .300000 .200000 .200000 .100000 .100000 .100000
00090 0 .300000 .200000 .200000 .100000 .100000 .100000
00100 //
```

Sample Job Setups

```
00010 //PLFIR JOB NCS.ES.B4912,ITSU
00020 // EXEC FTHCLG
00030 //C.SYSIN DD DSN=NCS.ES.B4912.ITSU.PLFIR.FORT,DISP=SHR
00040 //G.FT03F001 DD SYSOUT=A
00050 //G.FT02F001 DD DSN=NCS.ES.B4912.ITSU.PLOT6.DATA,DISP=OLD
00060 //G.FT01F001 DD DSN=NCS.ES.B4912.ITSU.ZENBU.DATA,DISP=SHR
00070 //
```

```
00010 //PL1FR JOB NCS.ES.B4912,ITSU
00020 // EXEC FTHCLG
00030 //C.SYSIN DD DSN=NCS.ES.B4912.ITSU.PL1FR.FORT,DISP=SHR
00040 //G.FT03F001 DD SYSOUT=A
00050 //G.FT02F001 DD DSN=NCS.ES.B4912.ITSU.PLOTP.DATA,DISP=OLD
00060 //G.FT01F001 DD DSN=NCS.ES.B4912.ITSU.ZENBU.DATA,DISP=SHR
00070 //G.FT08F001 DD *
00080 .300000 .200000 .200000 .100000 .100000 .100000
00090 //
```