An Introduction to Graphs in Dynare

Johannes Pfeifer*

University of Mannheim

July 3, 2014

First version: March 2013

Abstract

This paper provides an introduction to the graphs displayed by Dynare in its `stoch_simul`, `rplot`, `estimation`, `shock_decomposition`, and `identification`-commands. Note that it is work in progress and may not consider the most recent changes yet.

Keywords: Dynare 4, graphs, stoch_simul, estimation, shock_decomposition, identification.

*For corrections or suggestions, please email me at pfeifer@uni-mannheim.de

List of figures (as named in Dynare)

List of Abbreviations

BVAR	Bayesian Vector Autoregression.
DSGE	Dynamics Stochastic General Equilbrium.
HPDI	Highest Posterior Density Interval.
IRF	Impulse Response Function.
LRE	Linear Rational Expectation.
MC	Monte Carlo.
MCMC	Monte Carlo Markov Chain.
ML	Maximum Likelihood.
MS	Markov Switching.
SV	Singular Value.
SVD	Singular Value Decomposition.
VAR	Vector Autoregression.

List of Symbols

γ	vector of linear rational expectations model coefficients.
m	moment vector.
θ	generic parameter vector.
τ	vector of model solution coefficients.

1 Graphs Produced by `stoch_simul`

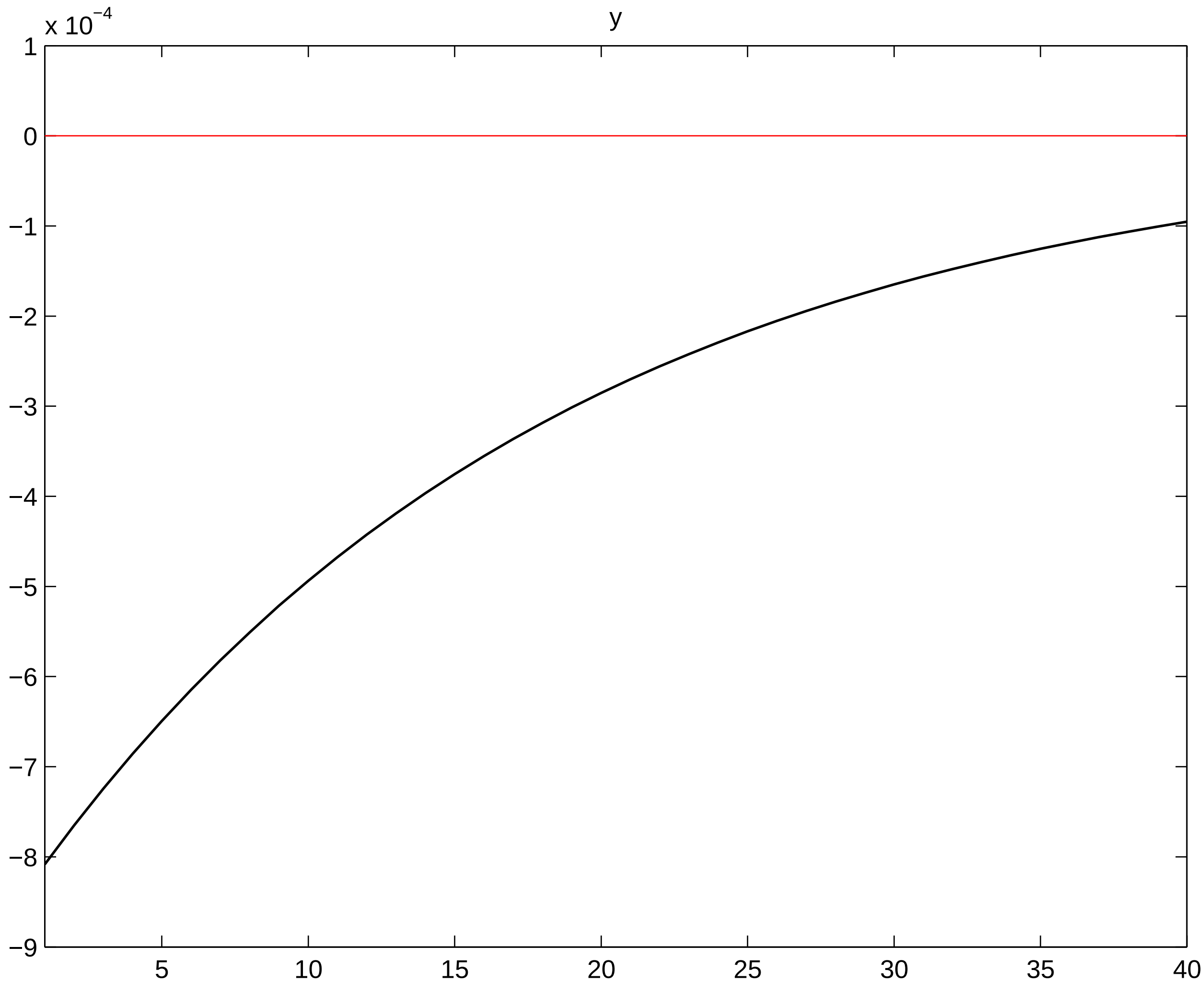

Figure 1: Orthogonalized shock to e__a - Impulse response functions (Impulse Response Function (IRF)) generated by `stoch_simul`. The underlying shock can be seen from either the figure title in Matlab or the file name. The x-axis displays the time horizon, while the y-axis displays deviations from the deterministic steady state, either in absolute deviations for linearized models or in percentage deviations for loglinearized models. The model used for the present graph is only linearized. Thus, the impact response for y means that y decreases by about 0.08 units. Predetermined variables like the capital stock in the present example are displayed in Dynare's end of period notation. Thus, k reacts on impact, because it is the capital stock at the end of the period, which is chosen by today's investment, but only becomes productive tomorrow.

If there are correlated shocks, Dynare uses a Cholesky decomposition to orthogonalize the shocks. The ordering in the Cholesky decomposition depends on the declaration order of the exogenous variables in the `var_exo`-statement. For more information, see http://www.dynare.org/phpBB3/viewtopic.php?f=1&t=2574. In case you want to simulate two simultaneous shocks, see http://www.dynare.org/phpBB3/viewtopic.php?f=1&t=2515

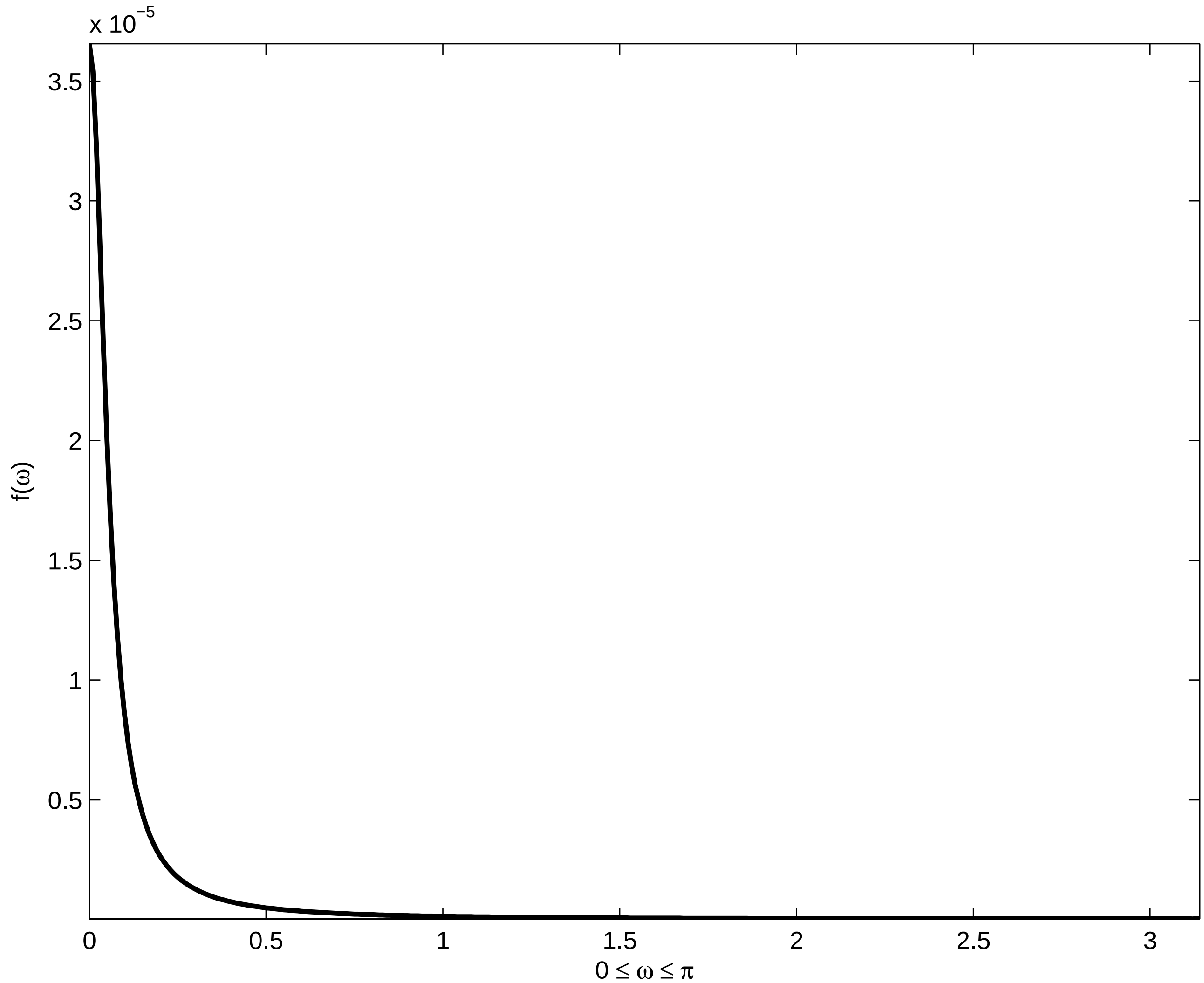

Figure 2: Spectral density of y – Spectral density plot generated by `stoch_simul` if `options_.SpectralDensity.trigger = 1;` has been set. The graphs are saved to the `graphs`-subfolder. Graphs display the univariate spectral density of the respective variable under consideration. The x-axis depicts the frequency, while the y-axis shows the power of this time-series at the respective frequency.

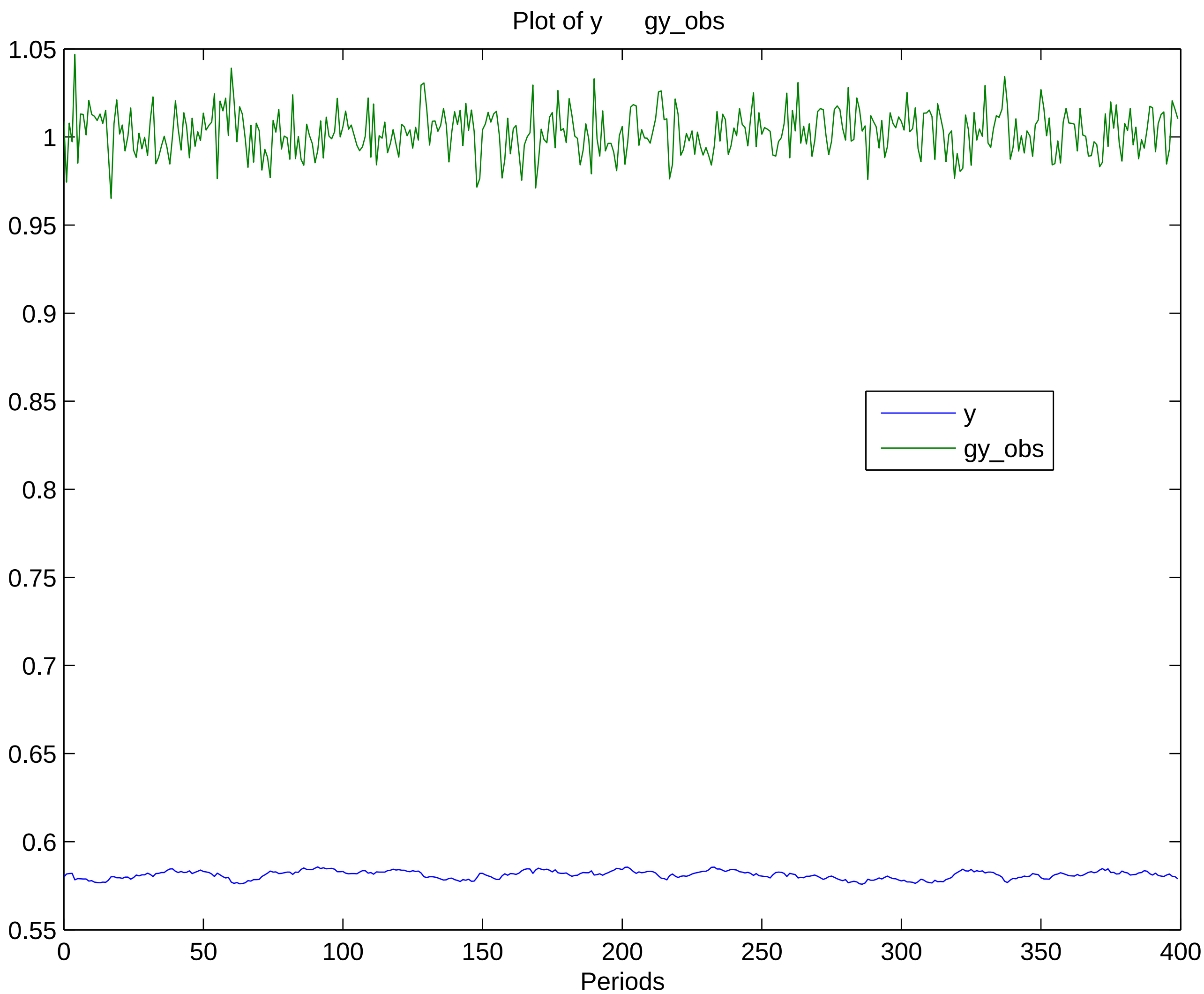

Figure 3: Simulated Trajectory – Plot of the simulated trajectory of endogenous variables that can be generated by using the `rplot`-command after `stoch_simul` or `simul`. Graphs display the simulated series over the simulation horizon. When used with the default option `options_.rplottype=0`, the command prints all simulated series in one figure. The graphs are saved in the graphs-subfolder.

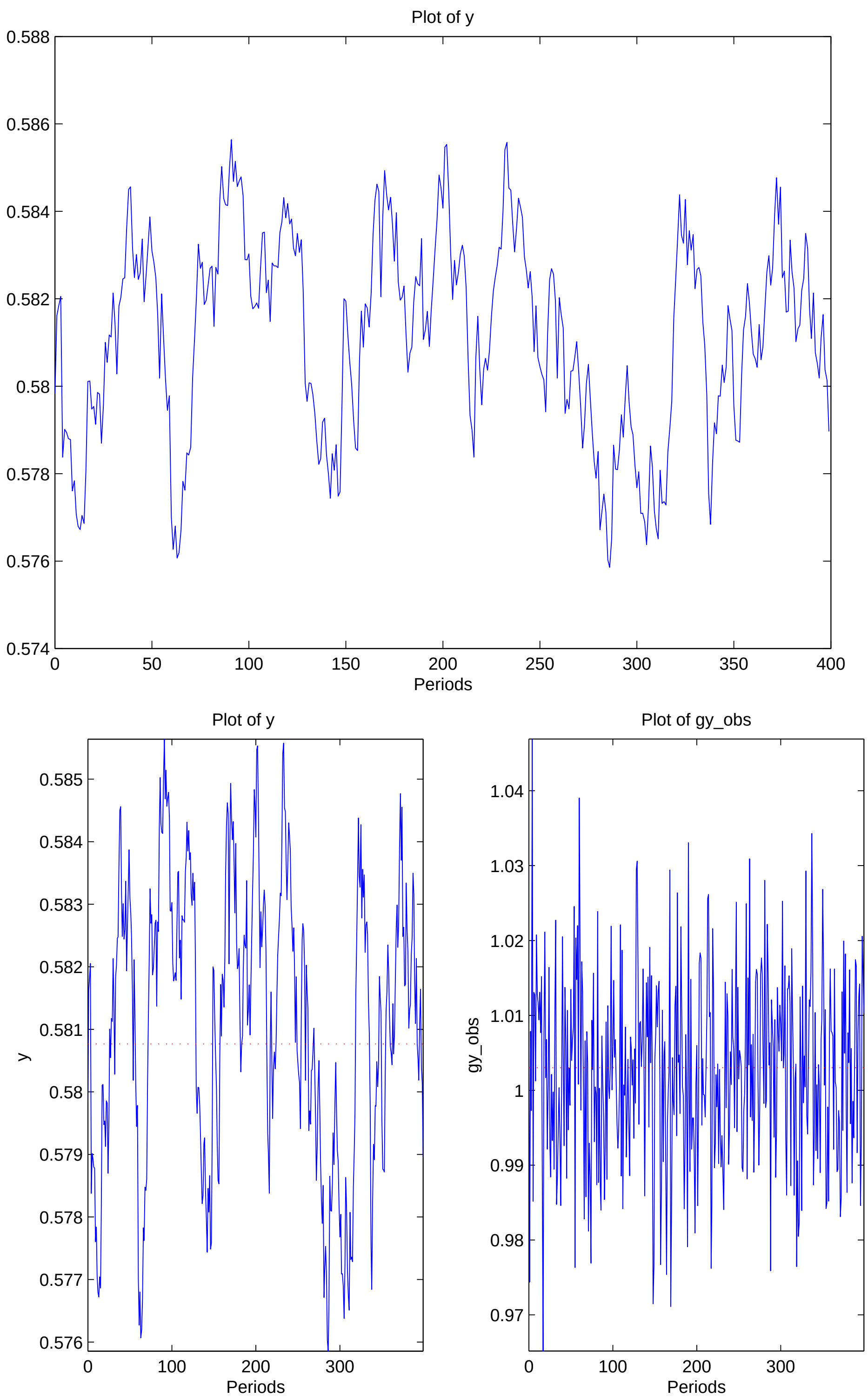

Figure 4: Simulated Trajectory – Plot of the simulated trajectory that can be generated by using the `rplot`-command after `stoch_simul` or `simul`. Graphs display the simulated series over the simulation horizon. When used with the option `options_.rplottype=1` (top figure), the command prints one figure for each variable to plot. With option `options_.rplottype=2` (bottom figure), the command prints all series into one figure, but with different subplots for each series. The graphs are saved in the graphs-subfolder.

2 Graphs Produced by `estimation`

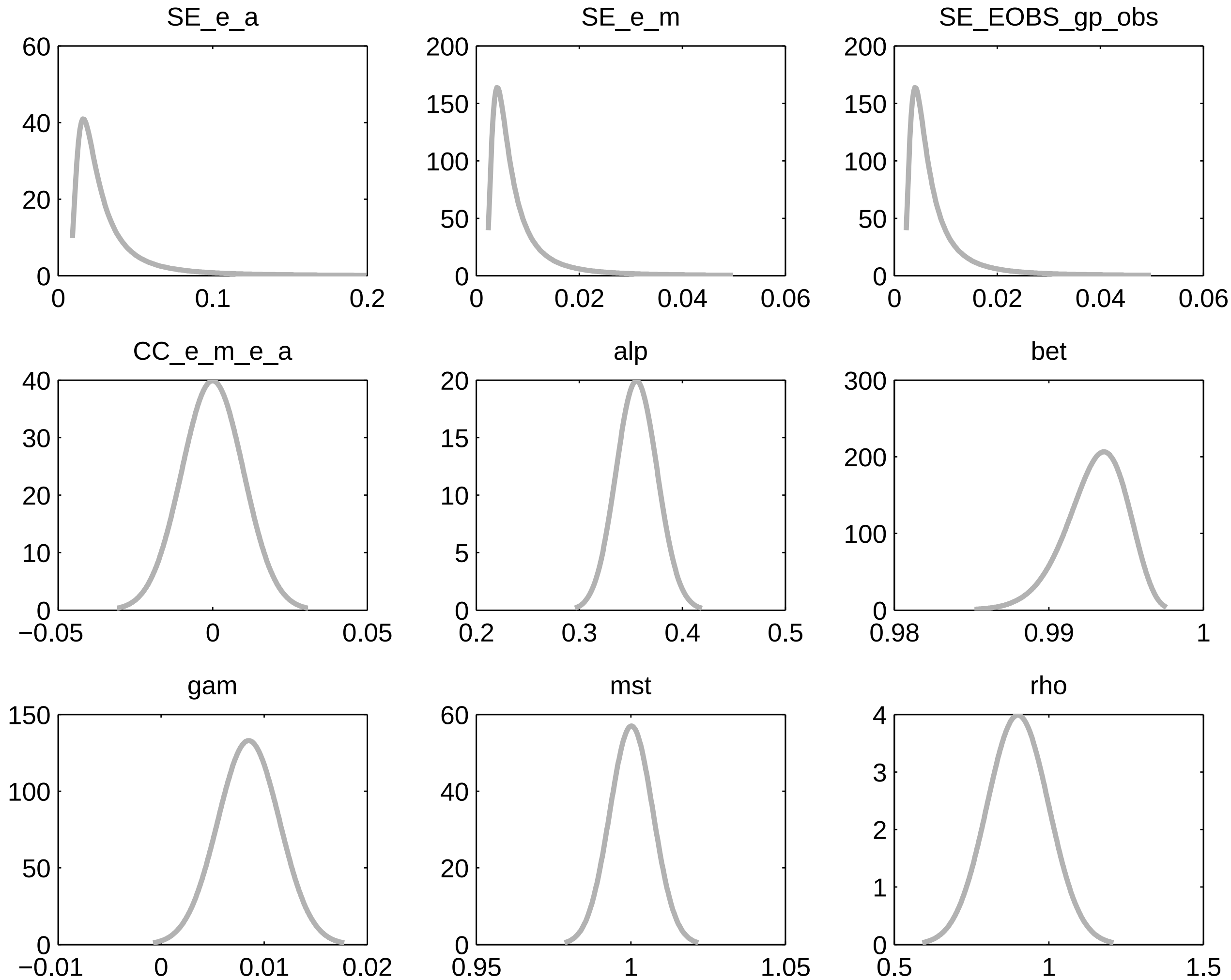

Figure 5: Priors – Prior plot generated by the `plot_priors`-option of the `estimation`-command. The x-axis displays part of the support of the prior distribution, while the y-axis displays the corresponding density. Standard deviations of shocks are designated by `SE_`, followed by the name of the shock, while observational errors are indicated by `SE_EOBS_`, followed by the name of the observable variables. Correlation between shocks and measurement errors are indicated by `CC_` and `CC_EOBS_` followed again by the names of the shock or observables.

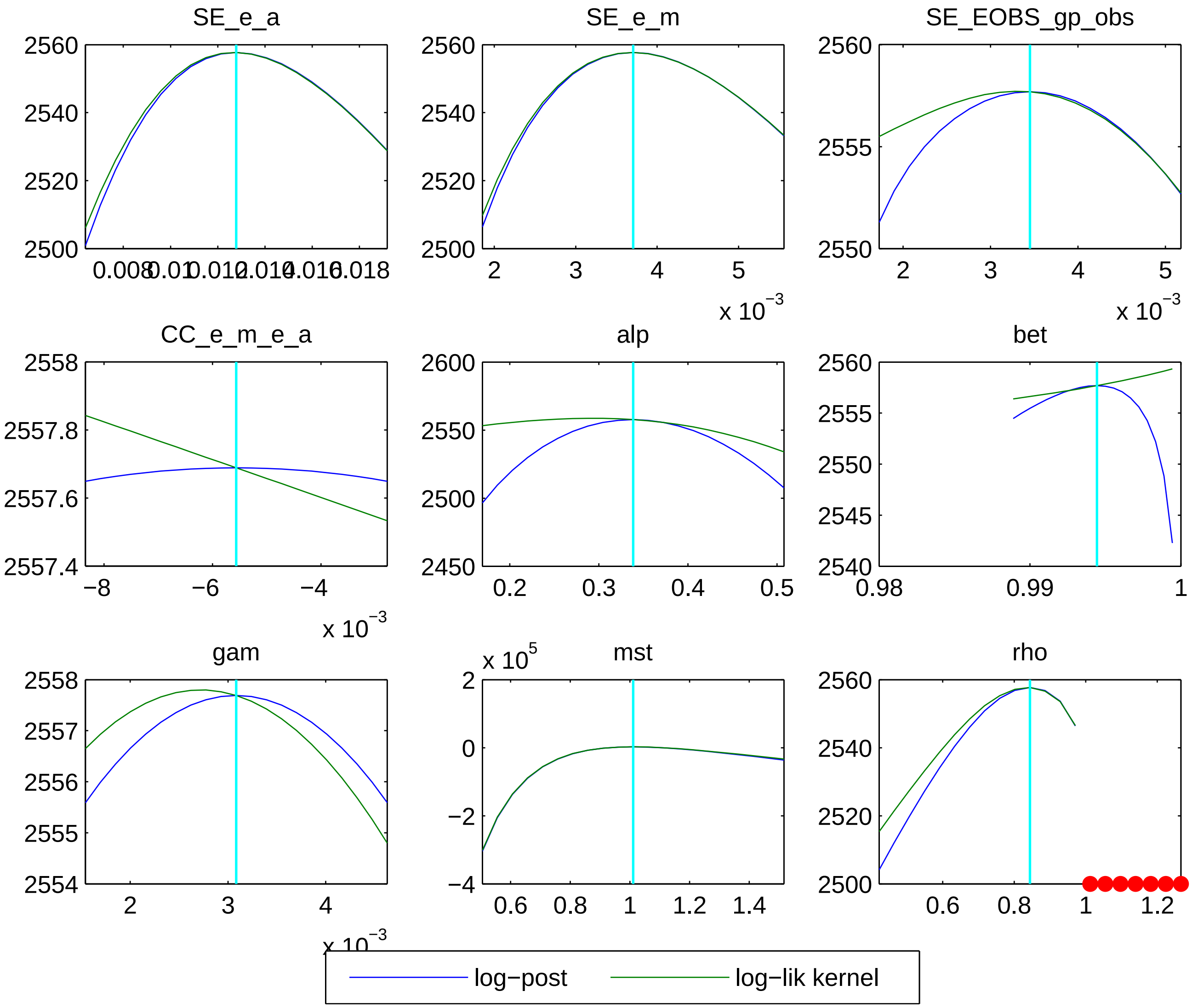

Figure 6: Mode check plots – Mode check plot generated by the `mode_check`-option of the `estimation`-command. This figure allows for checking whether the mode-computation found the (local) mode. The x-axis of each panel displays an interval of parameter values centered around the estimated mode (horizontal magenta line), while the y-axis displays the corresponding value of the log-likelihood kernel shifted up or down by the prior value at the posterior mode (green line) and of the posterior likelihood function (blue line). Differences in the shape between the likelihood kernel and the posterior likelihood indicate the role of the prior in influencing the curvature of the likelihood function. Ideally, the estimated mode should be at the maximum of the posterior likelihood. Big red dots indicate parameter values for which the model could not be solved due to e.g. violations of the Blanchard-Kahn conditions (indeterminacy or no bounded solution). In the current plot, such points are obtained when the autocorrelation coefficient *rho* is bigger than 1. The labeling of standard deviations and correlations is the same as in Figure 5. In case of Maximum Likelihood estimation, the log-likelihood kernel and the posterior are identical.

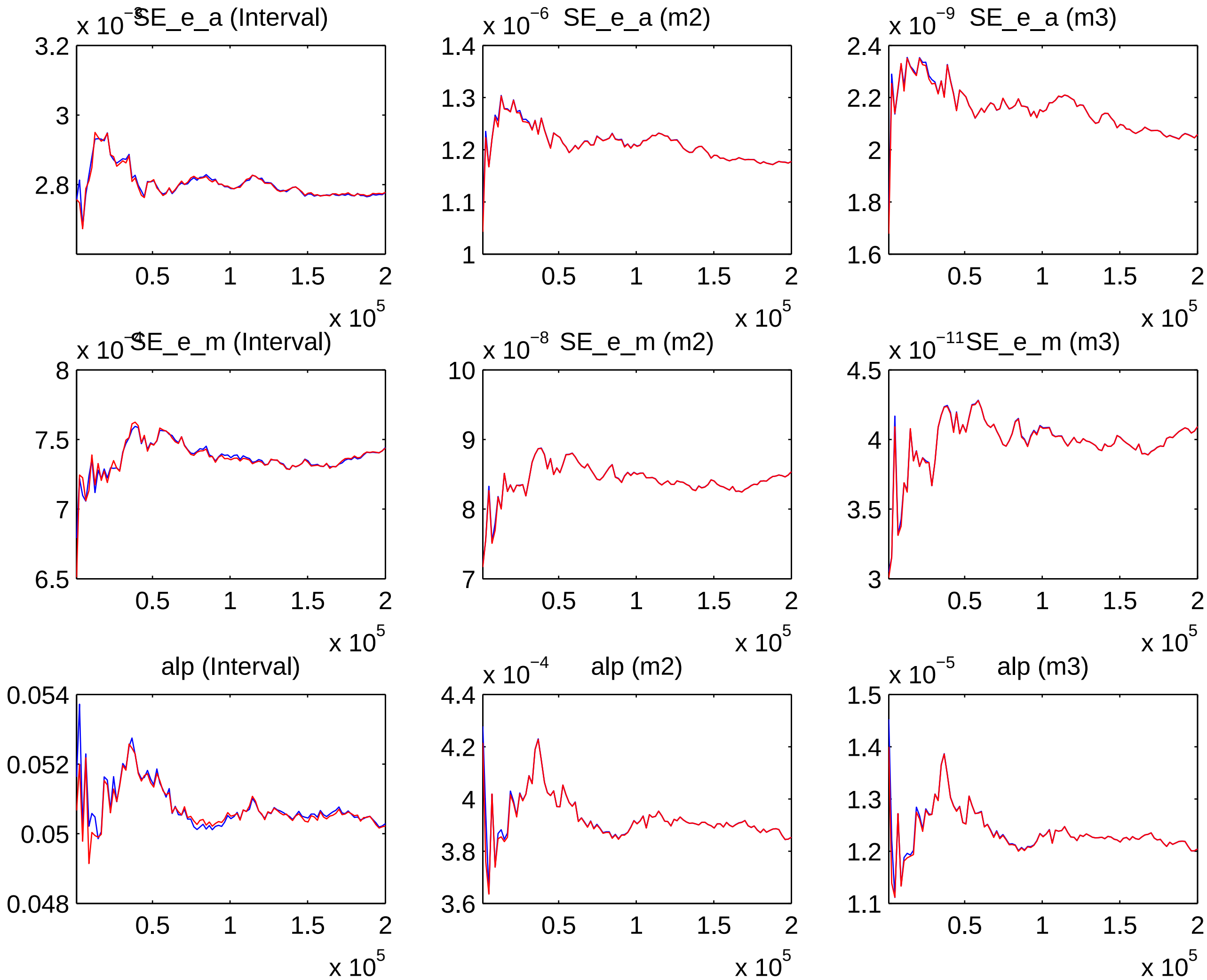

Figure 7: Monte Carlo Markov Chain (MCMC) univariate diagnostics (Brooks and Gelman, 1998) – Univariate convergence diagnostics generated by the `estimation`-command if `mh_nblocks` is larger than 1 and `mh_replic` larger than 2000. It is stored in the Output-subfolder. The first column with the appended (Interval) shows the Brooks and Gelman (1998, Section 3) convergence diagnostics for the 80% interval. The blue line shows the 80% interval/quantile range based on the pooled draws from all sequences, while the red line shows the mean interval range based on the draws of the individual sequences. The second and third column with the appended (m2) and (m3) show an estimate of the same statistics for the second and third central moments, i.e. the squared and cubed absolute deviations from the pooled and the within-sample mean, respectively. If the chains have converged, the two lines should stabilize horizontally and should be close to each other. The depicted graphs are based on an increasing number of parameter draws. The step size is `ceil((NumberOfDraws-1000)/100)`. The first data point is always computed on a window from draw 500 to draw 1000. The subsequent window *ii* ranges from from draw `(1000+ii*stepsize)/2` to draw `(1000+ii*stepsize)/2`. The labeling of standard deviations and correlations is the same as in Figure 5.

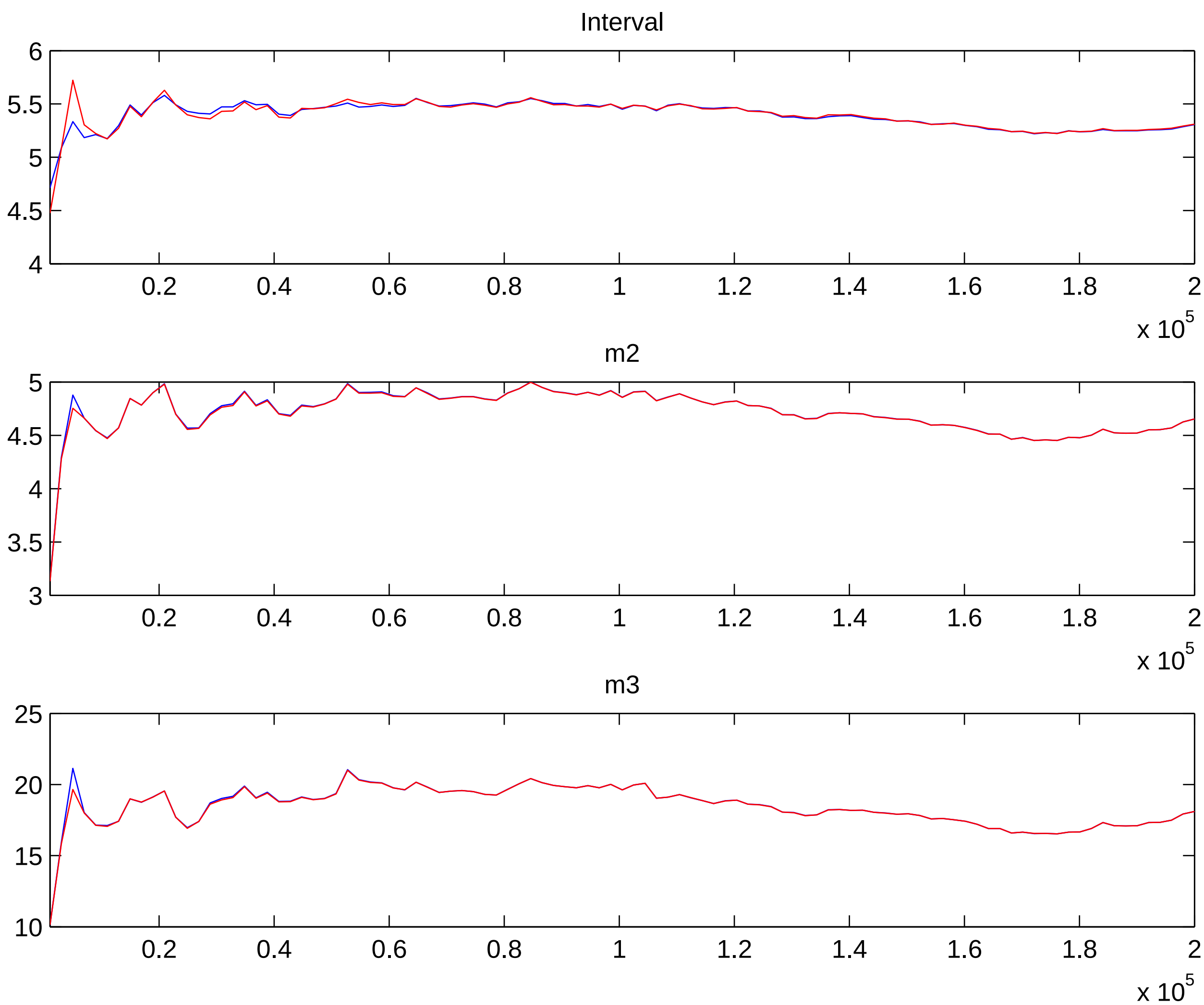

Figure 8: Multivariate convergence diagnostic – Multivariate convergence diagnostics generated by the `estimation`-command if `mh_nblocks` is larger than 1 and `mh_replic` larger than 1000. It is stored in the Output-subfolder. This diagnostics is the same as the univariate one depicted in Figure 7, except for the statistics now being based on the range of the posterior likelihood function instead of the individual parameters. Thus, the posterior kernel is used to aggregate the parameters. Again, convergence is indicated by the two lines stabilizing and being close to each other. The window size is the same as in the univariate case (see Figure 7 for a description).

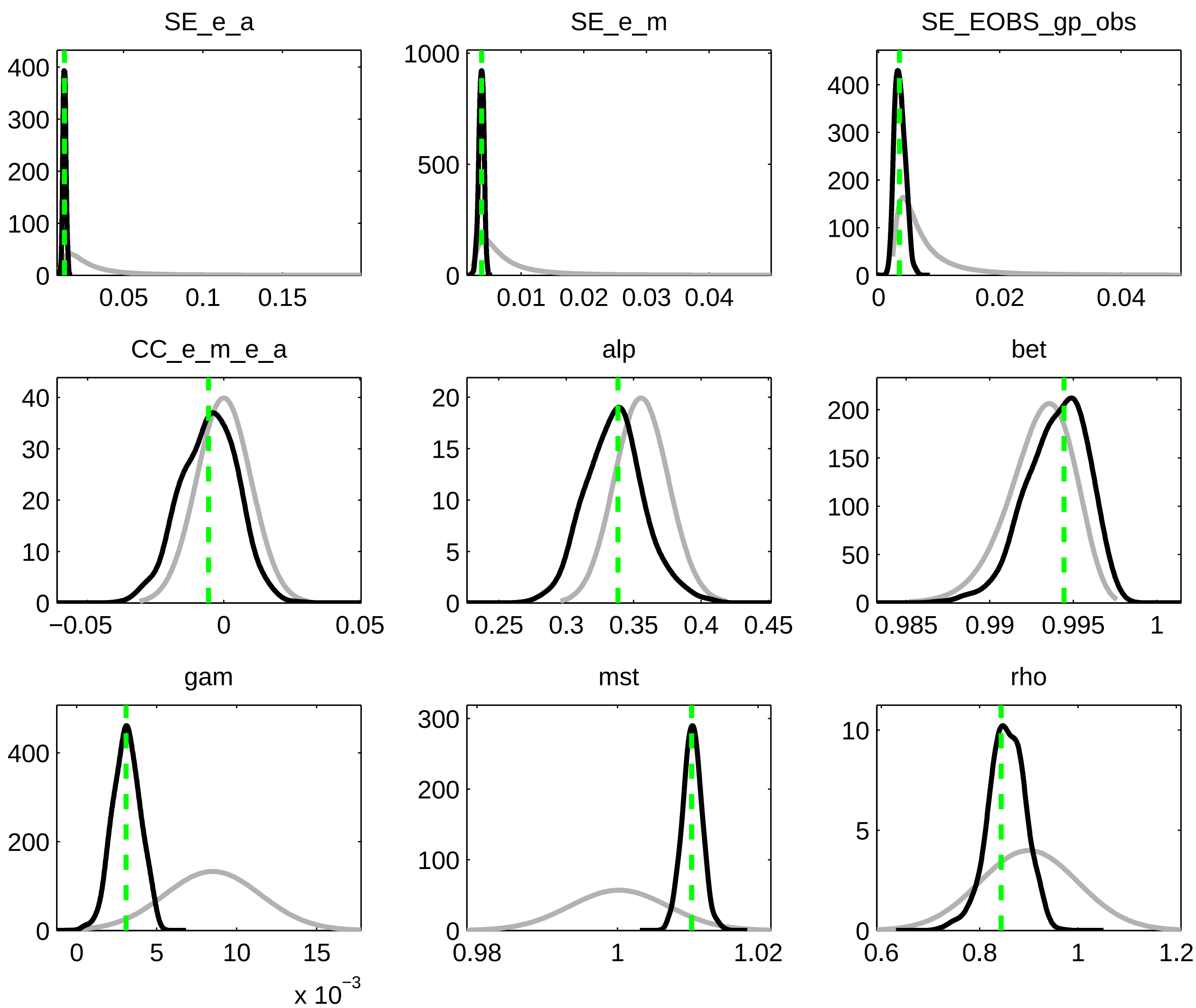

Figure 9: Priors and posteriors – Prior-posterior plot generated by the `estimation`-command if `mh_replic` > 0. The x-axis displays part of the support of the prior distribution, while the y-axis displays the corresponding density. The grey line shows the prior density also shown in Figure 5, while the black line shows the density of the posterior distribution. The green horizontal line indicates the posterior mode. If the posterior looks like the prior, either your prior was a very accurate reflection of the information in the data or, more usually, the parameter under consideration is only weakly identified and the data does not provide much information to update the prior (see e.g. Canova, 2007). The strength of identification can be checked using the `identification`-command. The labeling of standard deviations and correlations is the same as in Figure 5.

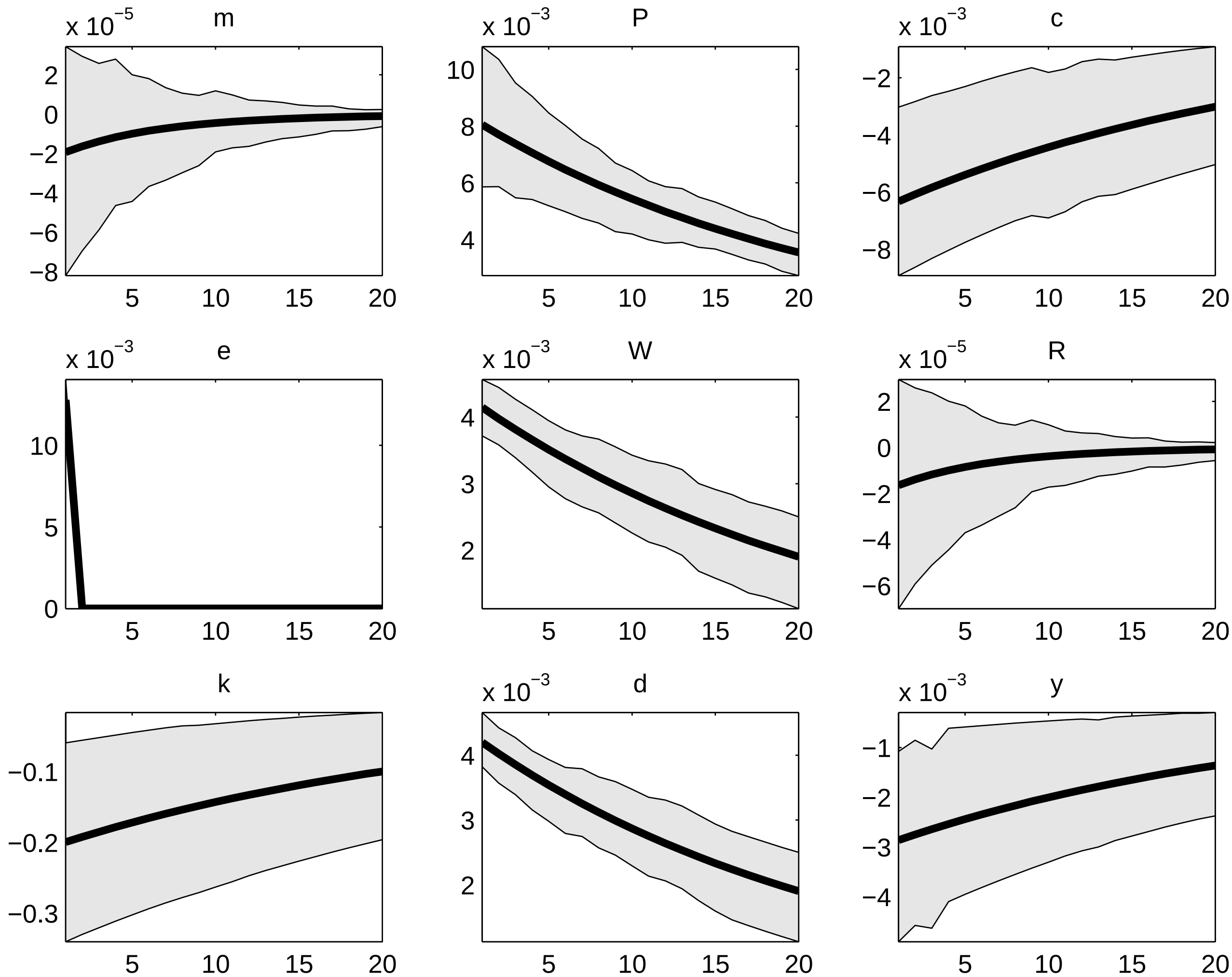

Figure 10: Orthogonalized shock to e_a – Bayesian IRF plot generated by the `bayesian_irf`-option of the `estimation`-command. It is stored in the Output-subfolder. Generally, these IRFs are similar to the ones displayed in Figure 1 and use the same orthogonalization scheme. The main difference is that the `stoch_simul`-IRFs are computed at the calibrated parameter combination, while the Bayesian IRFs are the mean impulse responses (not to be confused with the IRFs at the mean). The gray shaded areas provide highest posterior density intervals (Highest Posterior Density Interval (HPDI)). If you want to compute classical IRFs after estimation, use `stoch_simul` after `estimation` as the latter will set the parameters to the posterior mode/mean, depending on whether you use maximum likelihood or Bayesian estimation. More information can be found in Adjemian, Bastani, Juillard, Karamé, Mihoubi, Perendia, Pfeifer, Ratto, and Villemot (2011)

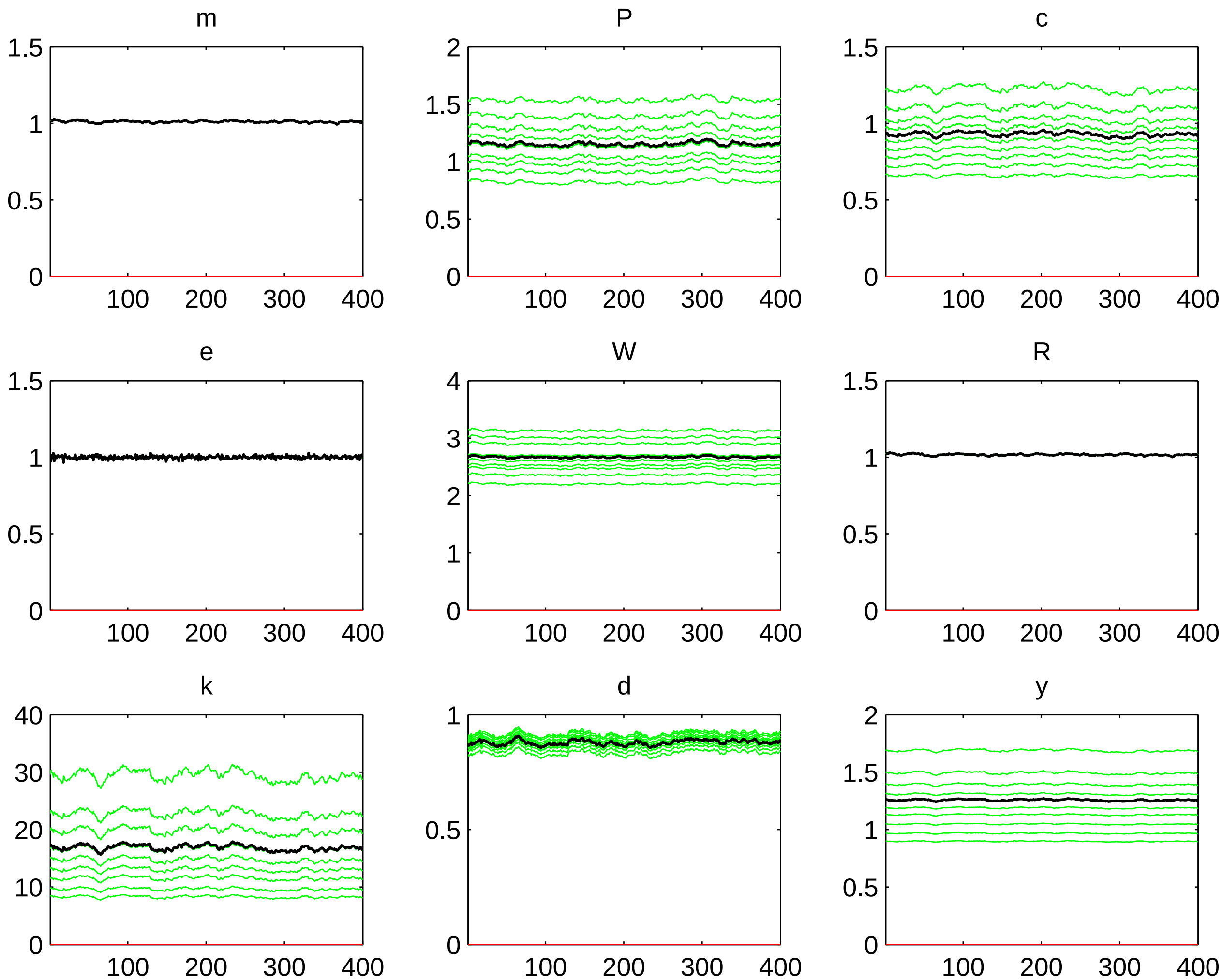

Figure 11: Smoothed variables – Smoothed variables plot generated by the `filtered_vars`-option of the `estimation`-command. It is stored in the Output-subfolder. The black line depicts the mean estimate of the smoothed variables ("best guess for the endogenous variables given all observations"), derived from the Kalman smoother. The green lines represent the deciles of the smoother distribution.

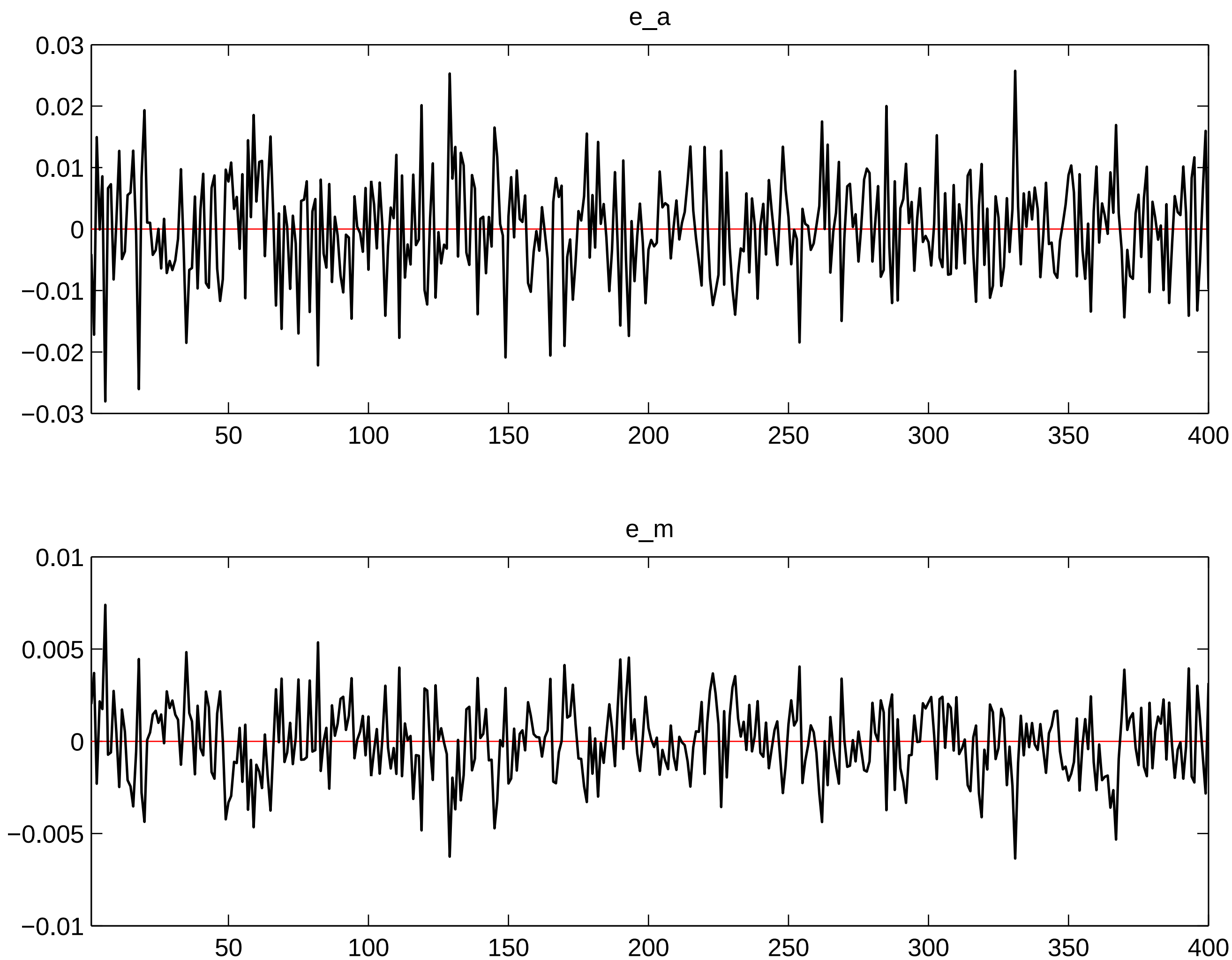

Figure 12: Smoothed Shocks – Smoothed Shocks plot generated by the `estimation`-command when either maximum likelihood estimation (Maximum Likelihood (ML)) is used or Bayesian estimation *without* the `smoother`-option. It is stored in the main folder. The black line depicts the estimate of the smoothed structural shocks ("best guess for the structural shocks given all observations"), derived from the Kalman smoother at the posterior mode (ML) or posterior mean (Bayesian estimation).

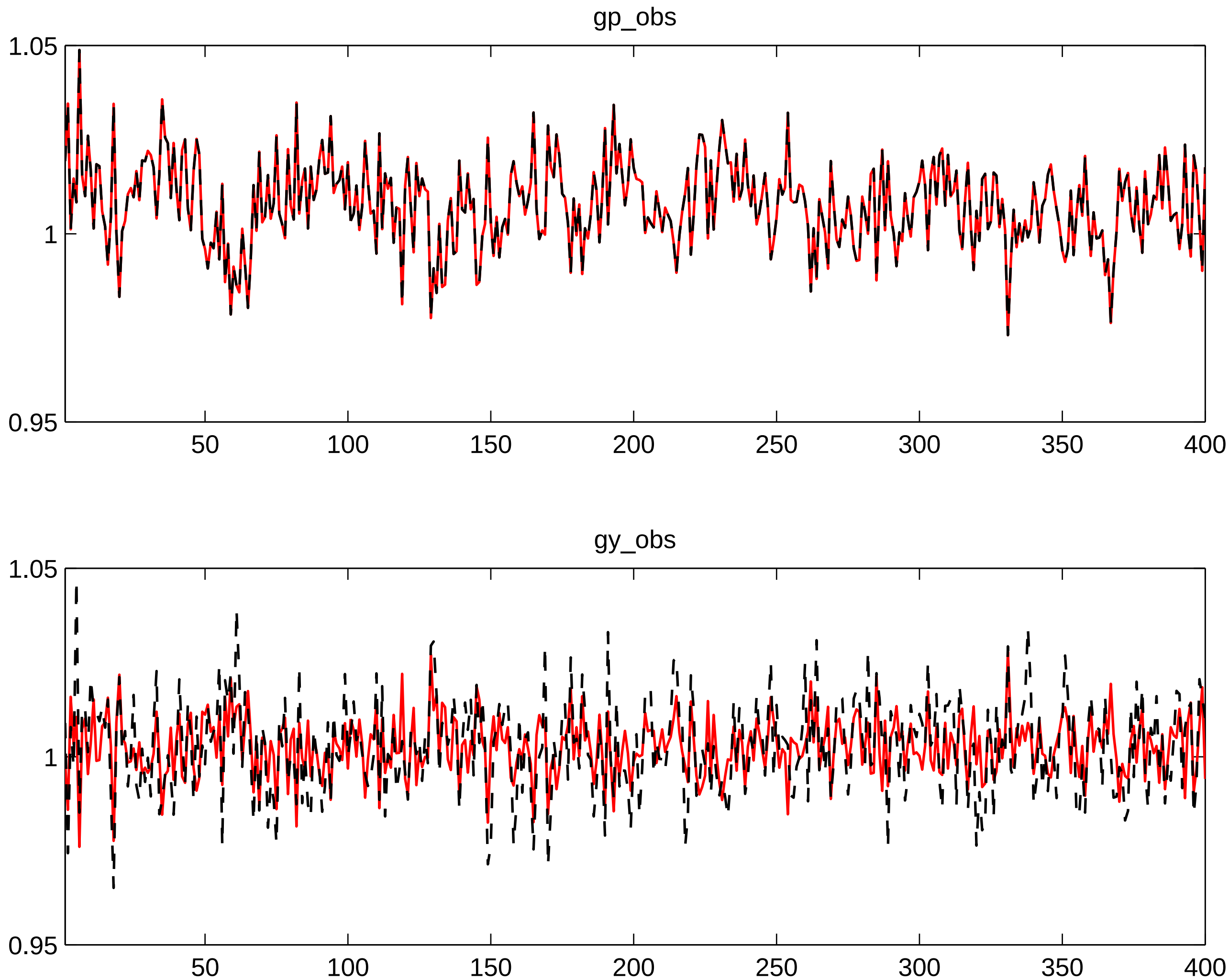

Figure 13: Historical and smoothed variables – Historical and smoothed variables plot generated by the `estimation`-command when either maximum likelihood estimation (ML) is used or Bayesian estimation *without* the `smoother`-option. It is stored in the main folder. The dotted black line depicts the actually observed data, while the red line depicts the estimate of the smoothed variable ("best guess for the observed variable given all observations"), derived from the Kalman smoother at the posterior mode (ML) or posterior mean (Bayesian estimation). In case of no measurement error, both series are identical as is the case in the upper panel.

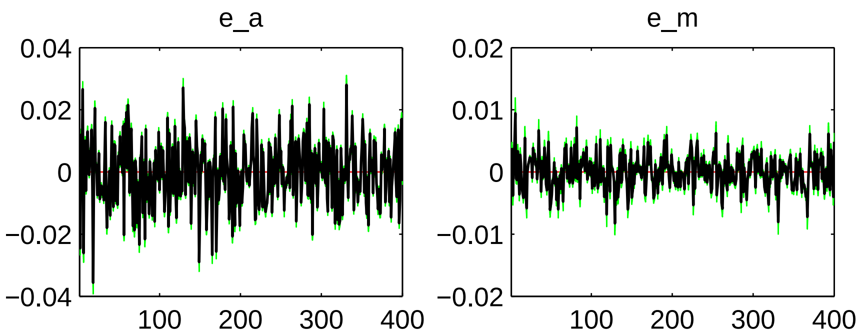

Figure 14: Smoothed shocks – Smoothed shocks plot generated by the `filtered_vars`-option of the `estimation`-command. It is stored in the Output-subfolder. The black line depicts the mean estimate of the smoothed structural shocks ("best guess for the structural shocks given all observations"), derived from the Kalman smoother. The green lines represent the deciles of the smoother distribution.

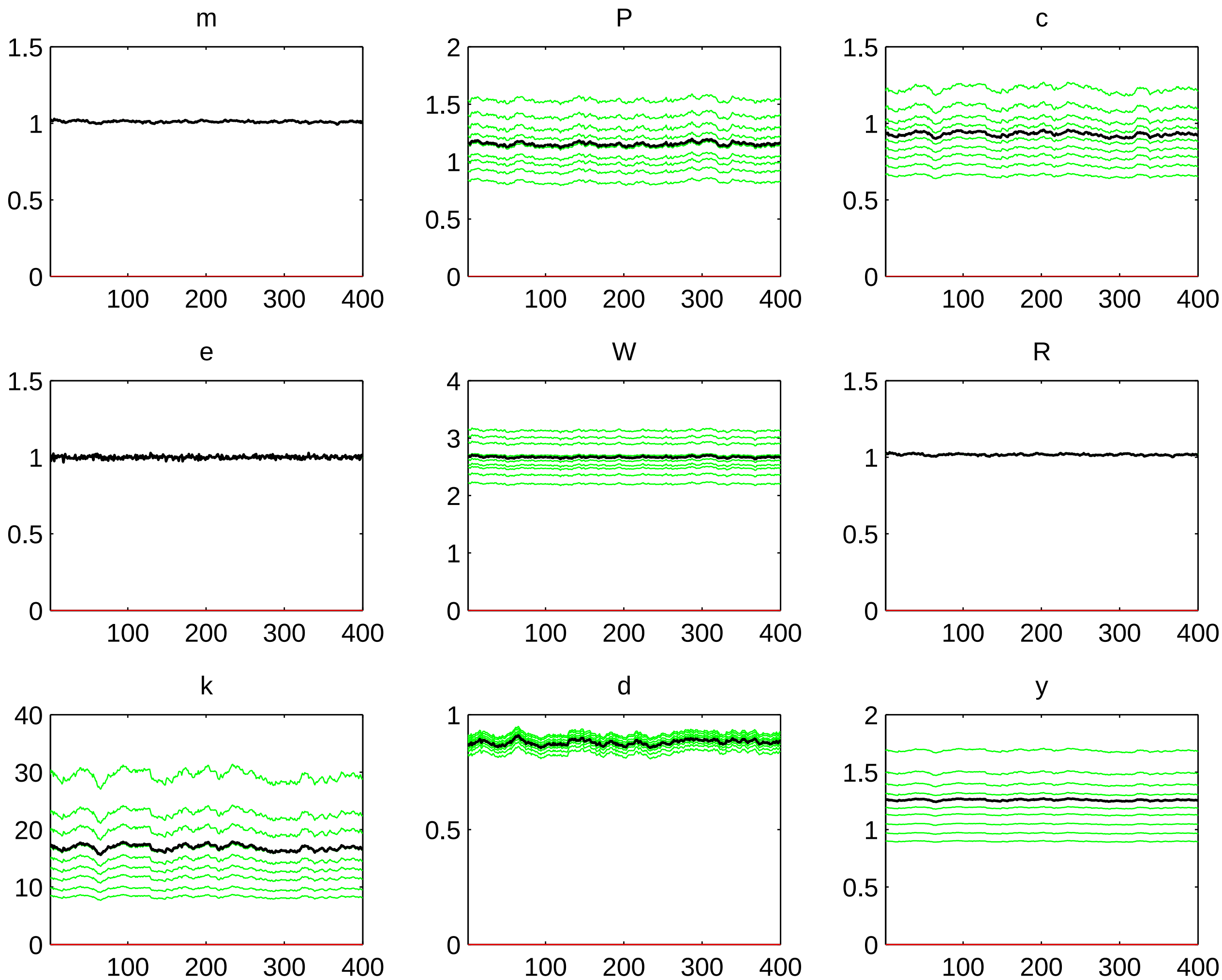

Figure 15: Updated Variables – Updated variables plot generated by the `filtered_vars`-option of the `estimation`-command. It is stored in the Output-subfolder. The black line depicts the mean estimate of the filtered endogenous variables ("best guess for the endogenous variables at time t given information up to the current observations t"), derived from the Kalman filter. The green lines represent the deciles of the smoother distribution.

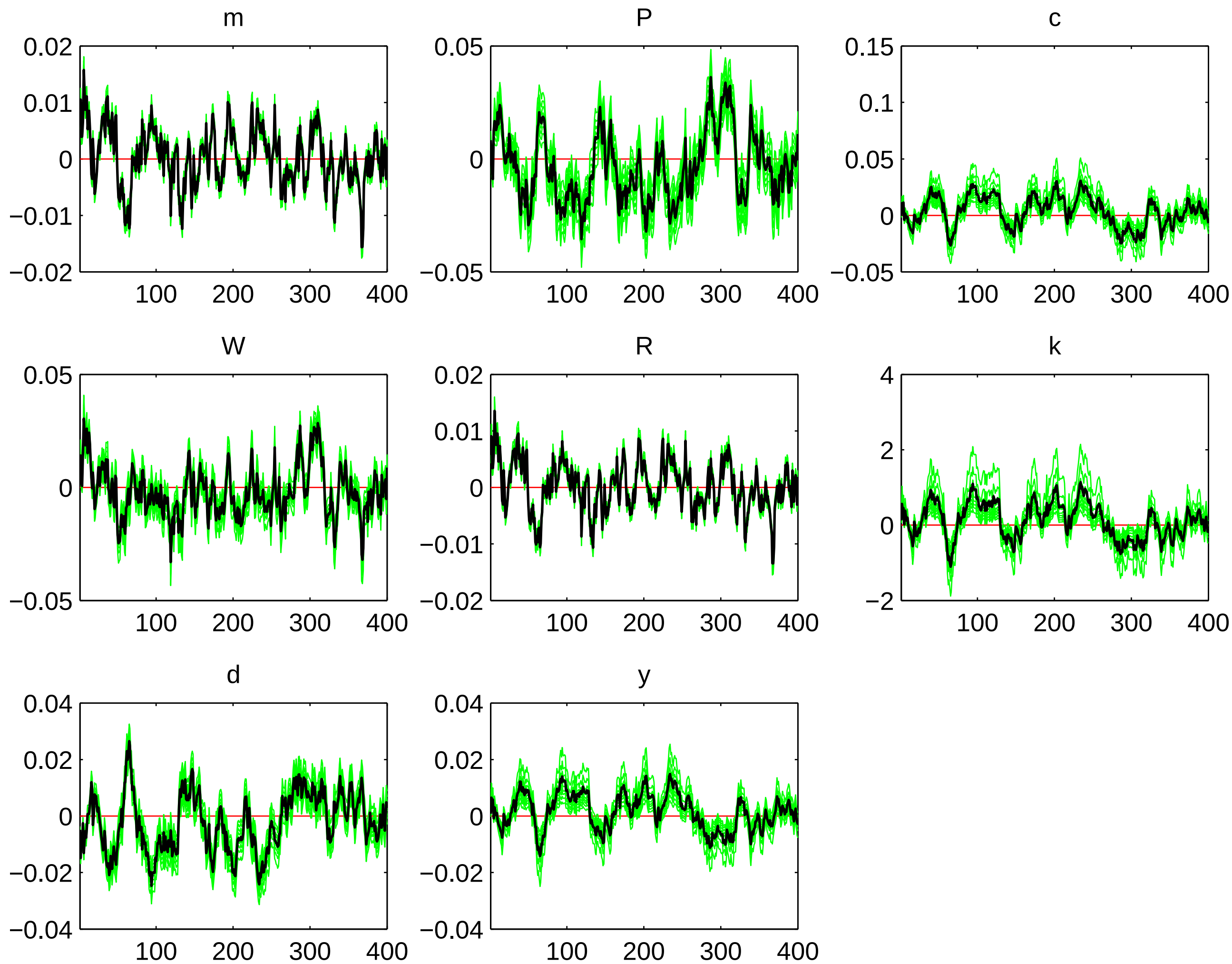

Figure 16: One step ahead forecast (filtered variables) – One step ahead forecast plot generated by the `filtered_vars`-option of the `estimation`-command. It is stored in the Output-subfolder. The black line depicts the mean estimate of the one step ahead forecast of the endogenous variables ("best guess for the endogenous variables at time $t + 1$ given information up to the current observations t"), derived from the Kalman filter. The green lines represent the deciles of the one step ahead forecast distribution.

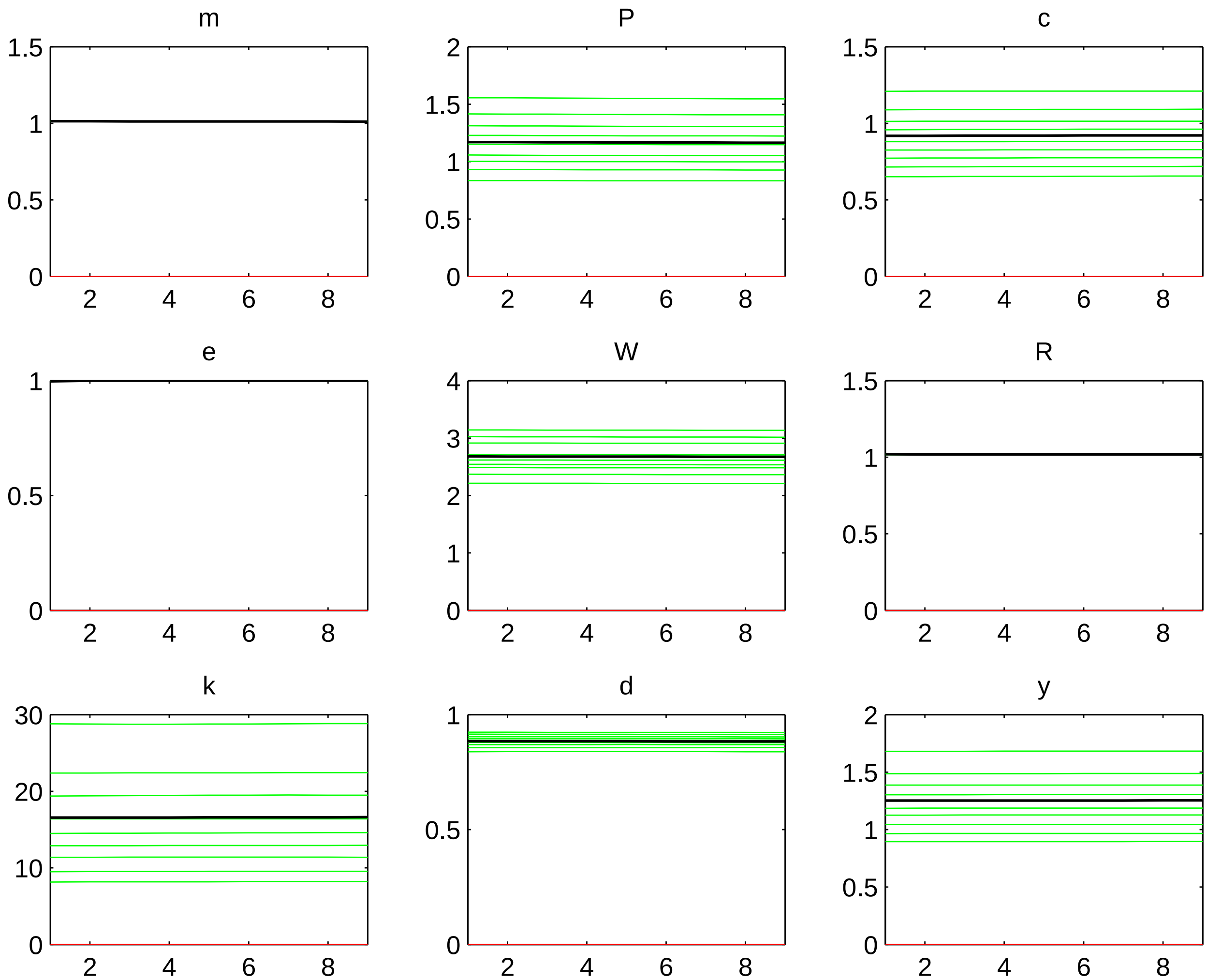

Figure 17: Forecasted variables (mean) – Mean forecast plot generated by the `forecast`-option of the `estimation`-command. It is stored in the Output-subfolder. The black line depicts the mean forecasts for the endogenous variables, starting at the last observation of the sample and going as many steps into the future as specified in the `forecast`-option. The green lines again depict the mean forecast deciles. The mean forecasts only take the parameter uncertainty into account, but omit the uncertainty about future shocks. Future shocks are averaged out/set to 0.

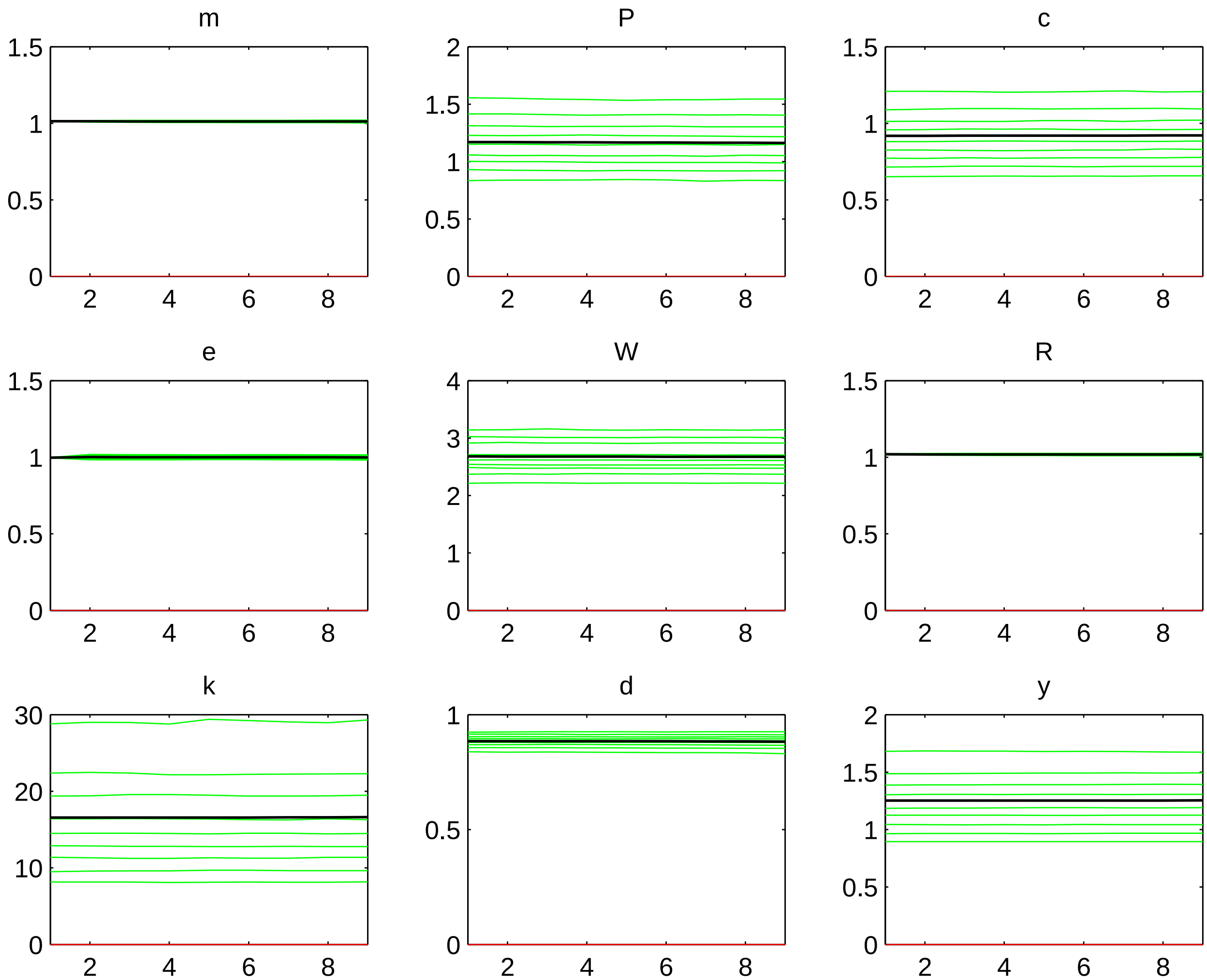

Figure 18: Forecasted variables (point) – Point forecast plot generated by the `forecast`-option of the `estimation`-command. It is stored in the Output-subfolder. The black line depicts the point forecasts for the endogenous variables, starting at the last observation of the sample and going as many steps into the future as specified in the `forecast`-option. The green lines again depict the point forecast deciles. In contrast to the mean forecasts, the points forecast not only take the parameter uncertainty into account, but also take into consideration the uncertainty about future shocks.

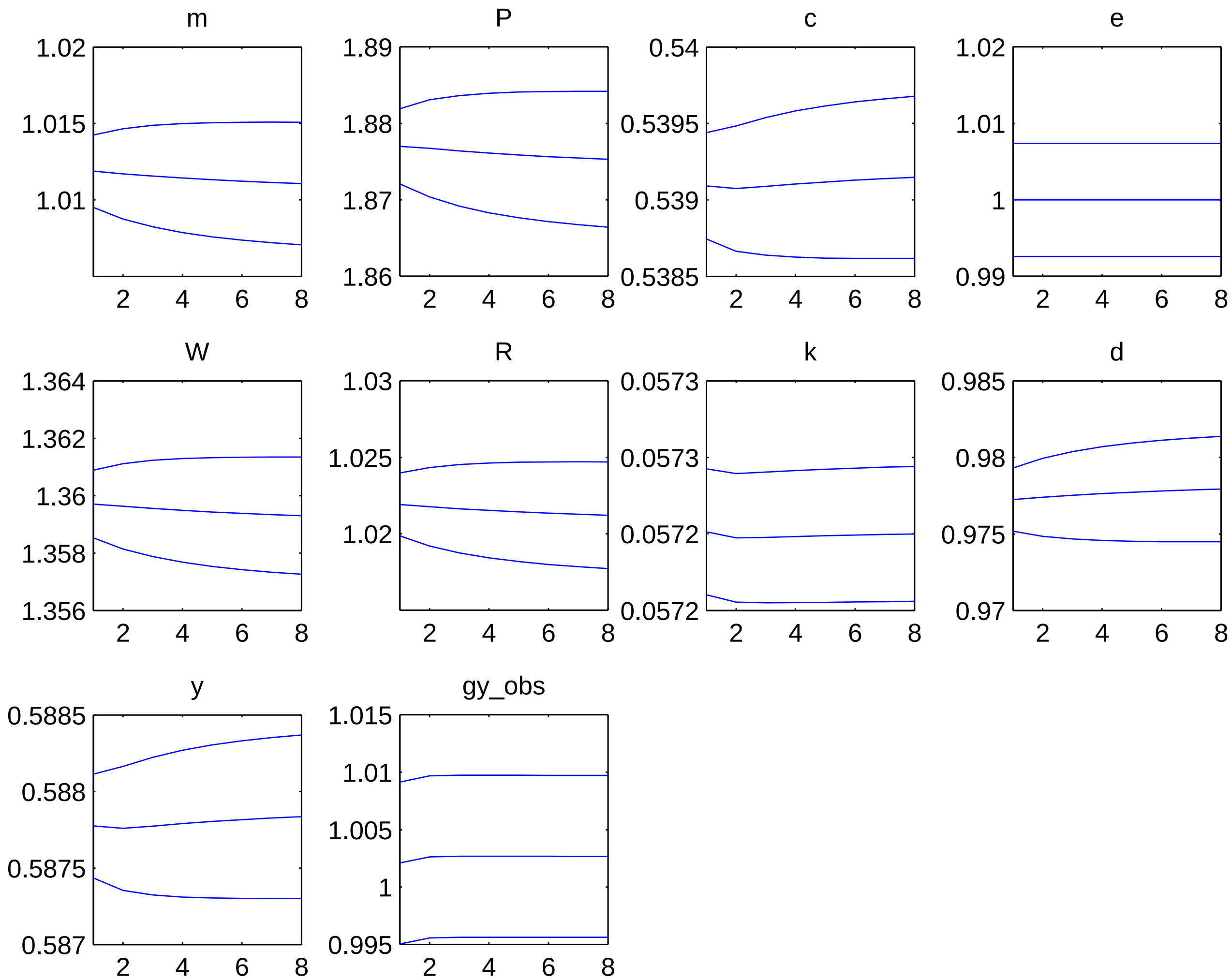

Figure 19: Forecasts – Forecast plot generated by the `forecast`-option of the `estimation`-command when used for ML-estimation. It is stored in the `graphs`-subfolder. The middle lines depict the point forecasts for the endogenous variables, starting at the last observation of the sample and going as many steps into the future as specified in the `forecast`-option. This point forecast is obtained at the posterior mode (ML). The blue lines represent classical confidence intervals with coverage specified by `conf_sig`. They are derived from the forecast error variance of the state space representation. The default coverage is 60%.

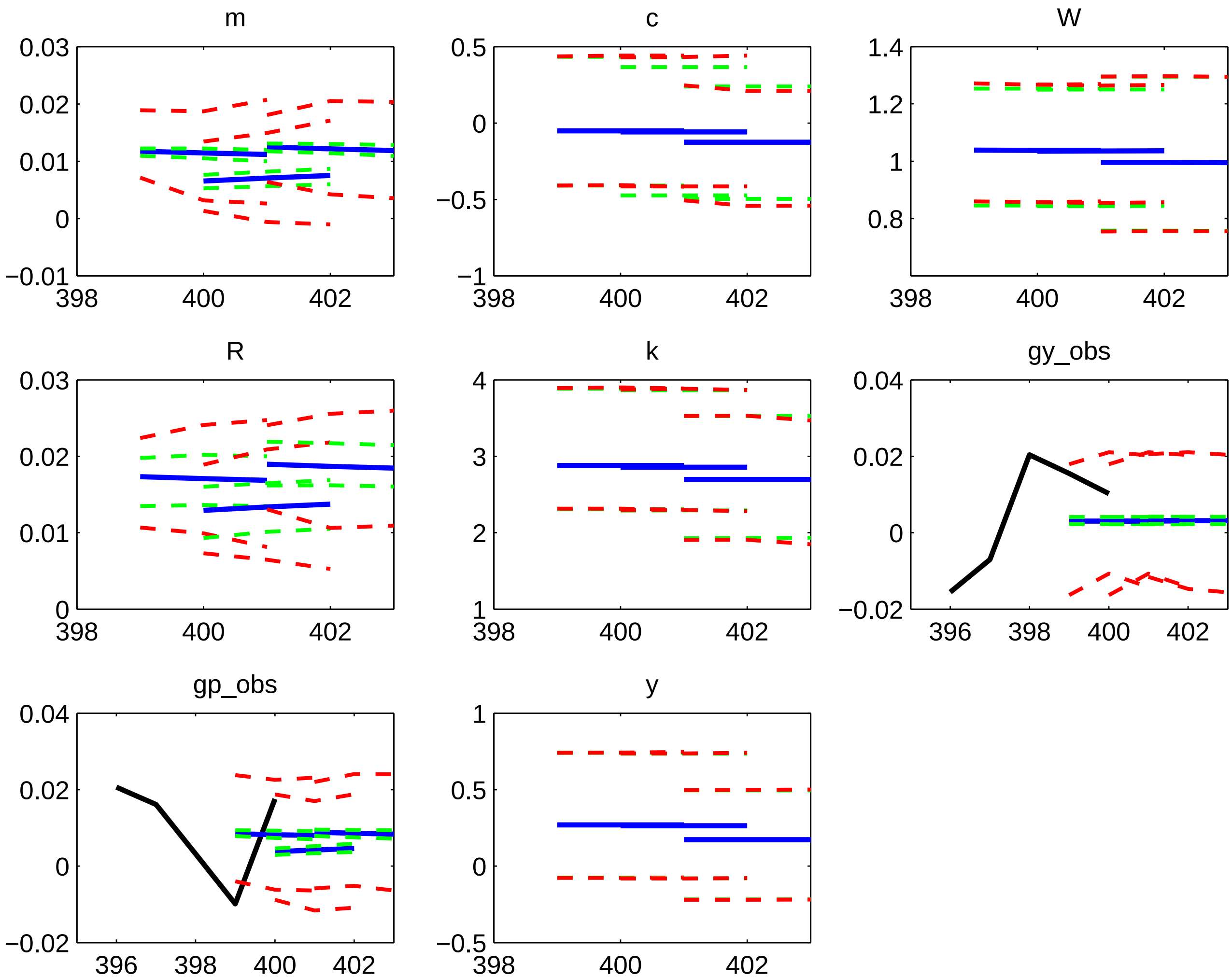

Figure 20: Out of sample forecasts – Recursive forecast plot generated by the `forecast`-option of the `estimation`-command when used with the `[nobs1:nobs2]`-option. It is stored in the `graphs`-subfolder. The blue solid lines depict the k-step ahead mean forecasts starting at the specified observations while the black solid lines depict the actual data values for the observables. The green dashed lines represent the 90% HPDI taking into account only parameter uncertainty, while the red dashed line represent the 90% HPDI taking into account both parameter and future shock uncertainty.

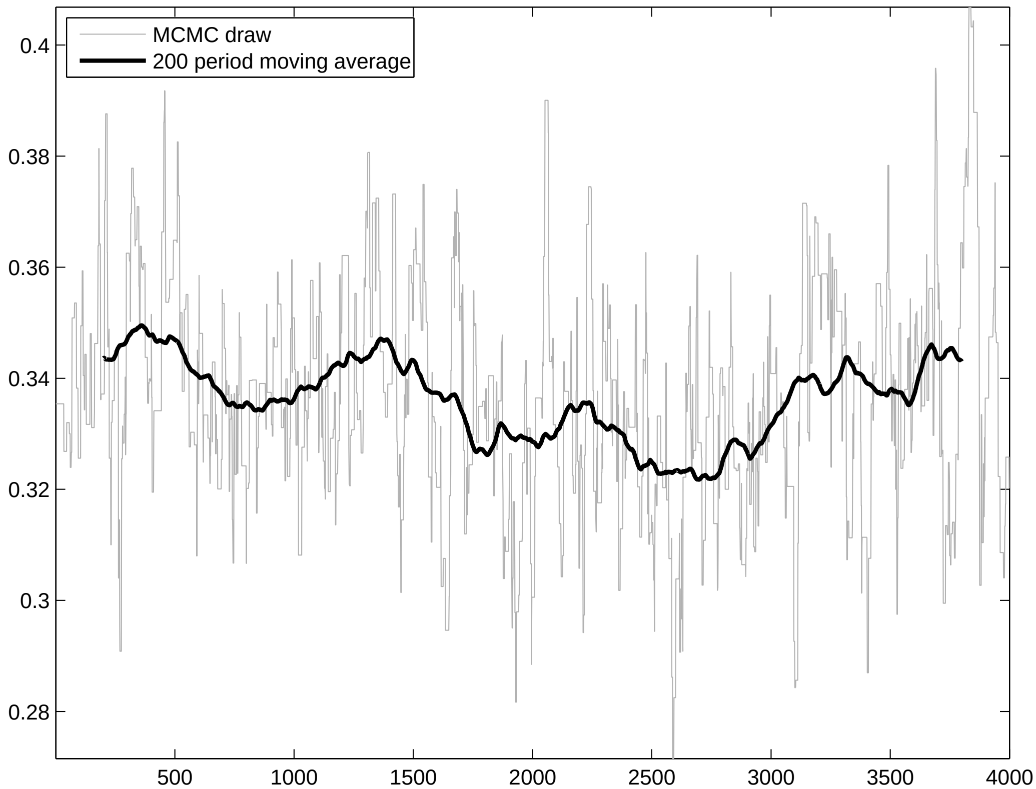

Figure 21: Trace plot for parameter alp (block number 2) – Trace plot generated by the `trace_plot`-command after `estimation`. It is stored in the `graphs`-subfolder. The gray line shows the MC iterations encountered during the MC chain indicated in the figure title for the indicated parameter. The black line depicts the N-period moving average (default: N=200). The current plot, showing the deep parameter *alp*, was created using the command `trace_plot(options_,M_,estim_params_,'DeepParameter',2,'alp')`. The first three arguments are mandatory and always the same. The fourth input argument designates the type of parameter to be plotted. It can take the values

- 'DeepParameter' for a deep structural parameter defined in the `parameters`-command
- 'StructuralShock' for the standard deviation or correlation of structural shocks (defined in Dynare's `estimated_params`-block using the `stderr` or `corr`-commands on exogenous variables)
- 'MeasurementError' for the standard deviation or correlation of measurement errors (defined in Dynare's `estimated_params`-block using the `stderr` or `corr`-commands on endogenous variables)

The fifth input argument takes an integer specifying the number of the Markov Chain for which you want to plot the draws. The sixth input argument provides the name of the parameter to be plotted/the name of the exogenous or endogenous variable for which to plot the standard deviation of the structural shock or measurement error. In case of a correlation that should be plotted, you have to specify the name of the second affected endogenous or exogenous variable as the seventh input argument. For example, to plot the correlation between the shocks e_m and e_a, use `trace_plot(options_,M_,estim_params_,'StructuralShock',2,'e_m','e_a')`

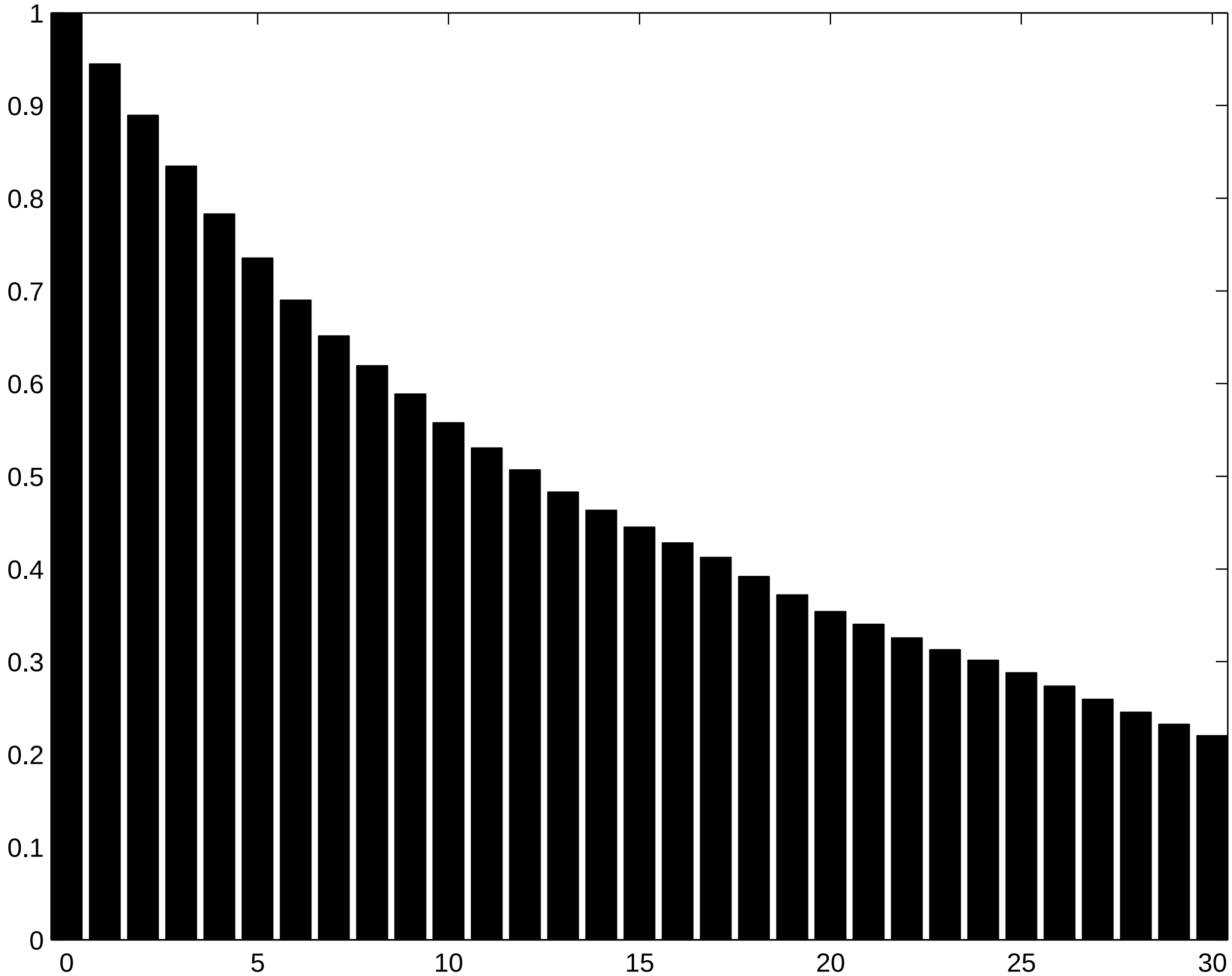

Figure 22: Autocorrelogram for correlation between structural shocks e_m and e_a (block number 2) – Autocorrelogram plot generated by the `mh_autocorrelation_function`-command after `estimation`. It is stored in the `graphs`-subfolder. The black bars indicate the n-th order autocorrelation of the parameter in the specified Markov Chain shown in the figure title. The horizontal axis shows the lag order n, while the vertical axis depicts the corresponding autocorrelation at lag n. At 0, the correlation is always 1. Ideally, we would like the draws from the posterior to be iid. The higher the autocorrelation the more inefficient the MCMC is due to the correlation making one additional draw from the MCMC less informative compared to the already present draws. If the autocorrelation is very high, you will need a lot of draws in your MCMC. The syntax for plotting is the same as for the `trace_plot`-command in Figure 21. The present graph has been created using `mh_autocorrelation_function(options_,M_,estim_params_,'StructuralShock',2,'e_m','e_a')`

3 Graphs Produced by `dsge_var`

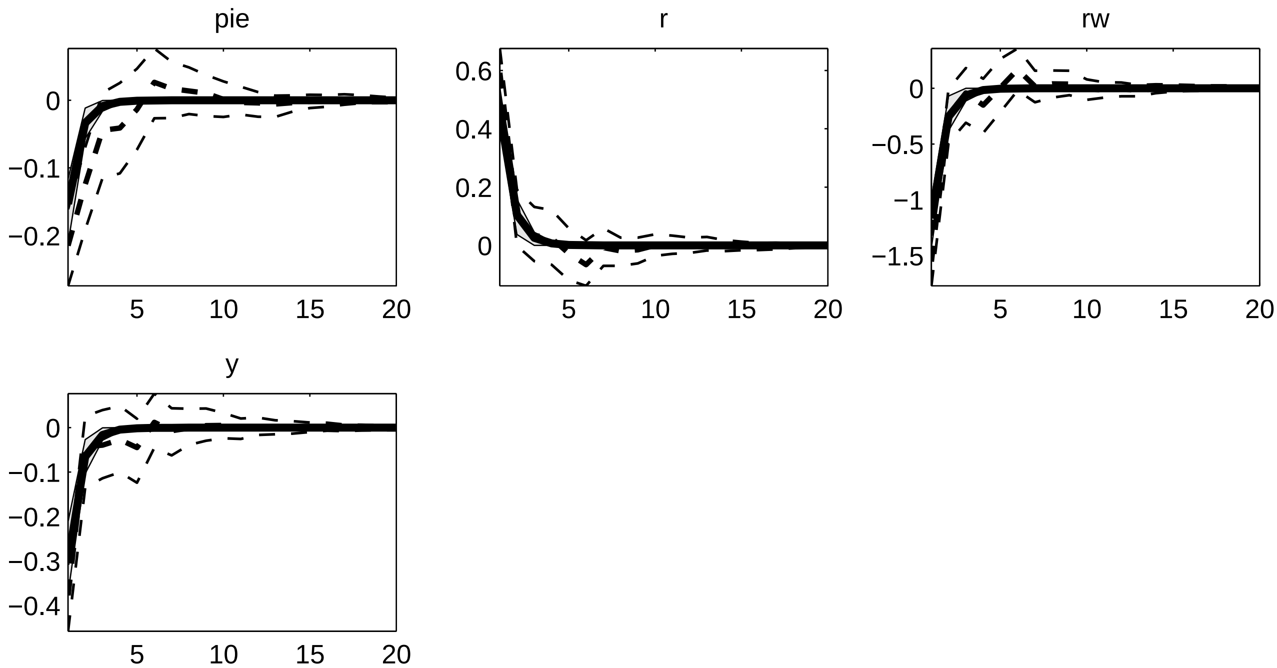

Figure 23: Orthogonalized shock to e_ms – Bayesian IRF plot generated by the `bayesian_irf`-option of the `estimation`-command if the `dsge_var`-option is specified. It is stored in the Output-subfolder. The thick black dashed line in the middle is the median IRF of the Dynamics Stochastic General Equilbrium (DSGE) Vector Autoregression (VAR). The thin black dashed lines are the first and ninth posterior deciles of the DSGE-VAR's IRFs. The thick solid black line is the median posterior IRF of the DSGE model, while the shaded area covers the space between the first and ninth posterior deciles of the DSGE's IRFs (i.e. basically the content of Figure 10).

4 Graphs Produced by `forecast`

The `forecast`-command makes use of the same functions as the forecasts after maximum likelihood estimation. See Figure 19 for a description.

5 Graphs Produced by `shock_decomposition`

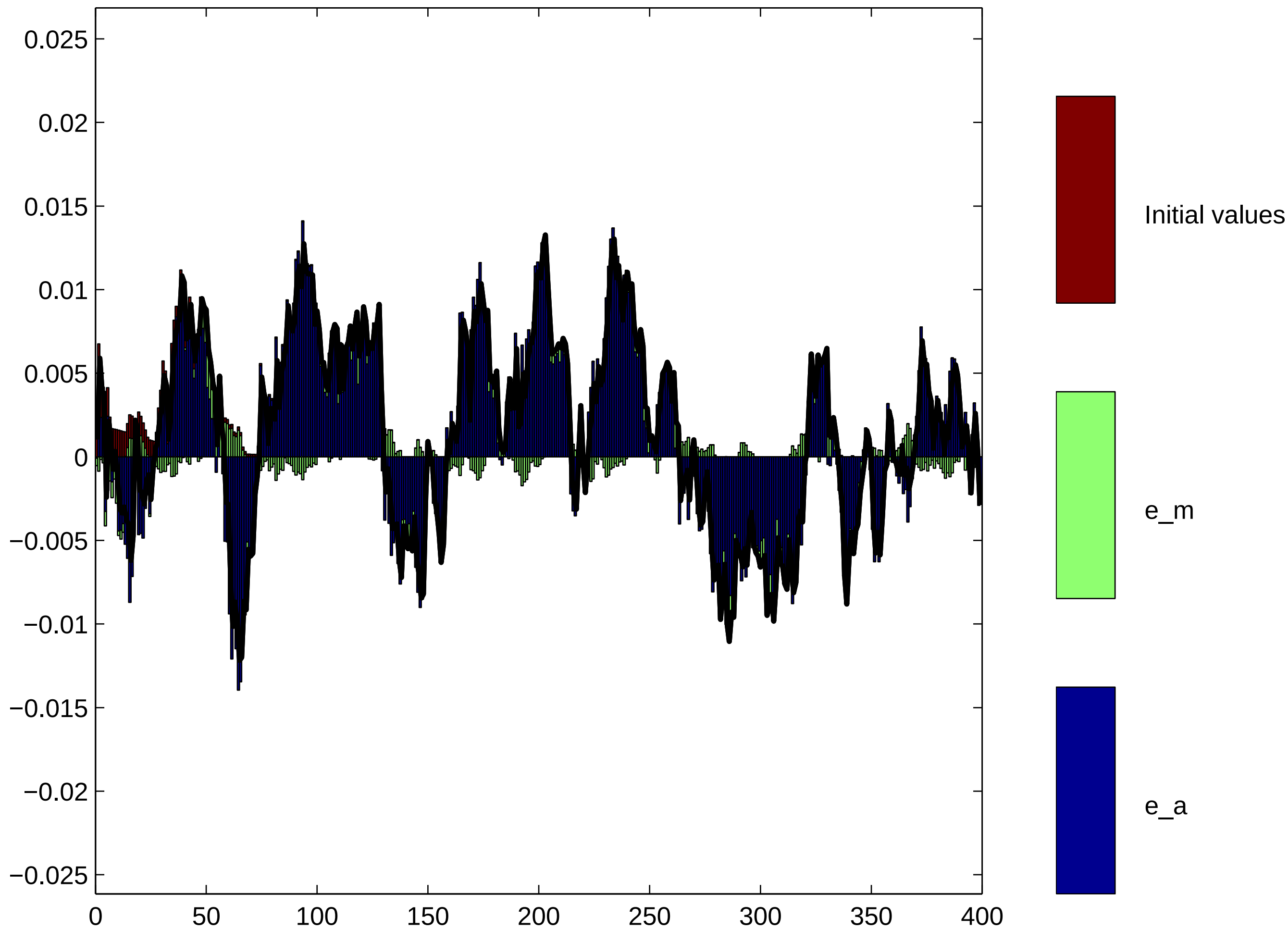

Figure 24: Shock decomposition – Shock decomposition plot generated by the `shock_decomposition`-command. It is stored in the main folder. The black line depicts the deviation of the smoothed value of the corresponding endogenous variable from its steady state at the specified `parameter_set`. Note that this differs from Figure 11, where the steady state was added. By default, the `parameter_set` is the `posterior_mean` if Bayesian estimation has been used and the `posterior_mode` otherwise. The colored bars correspond to the contribution of the the respective smoothed shocks to the deviation of the smoothed endogenous variable from its steady state, i.e. our "best guess" of which shocks lead to our "best guess" for our unobserved variables. "Initial values" in the graphs refers to the part of the deviations from steady state not explained by the smoothed shocks, but rather by the unknown initial value of the state variables. This influence of the starting values usually dies out relatively quickly.

6 Graphs Produced by `identification`

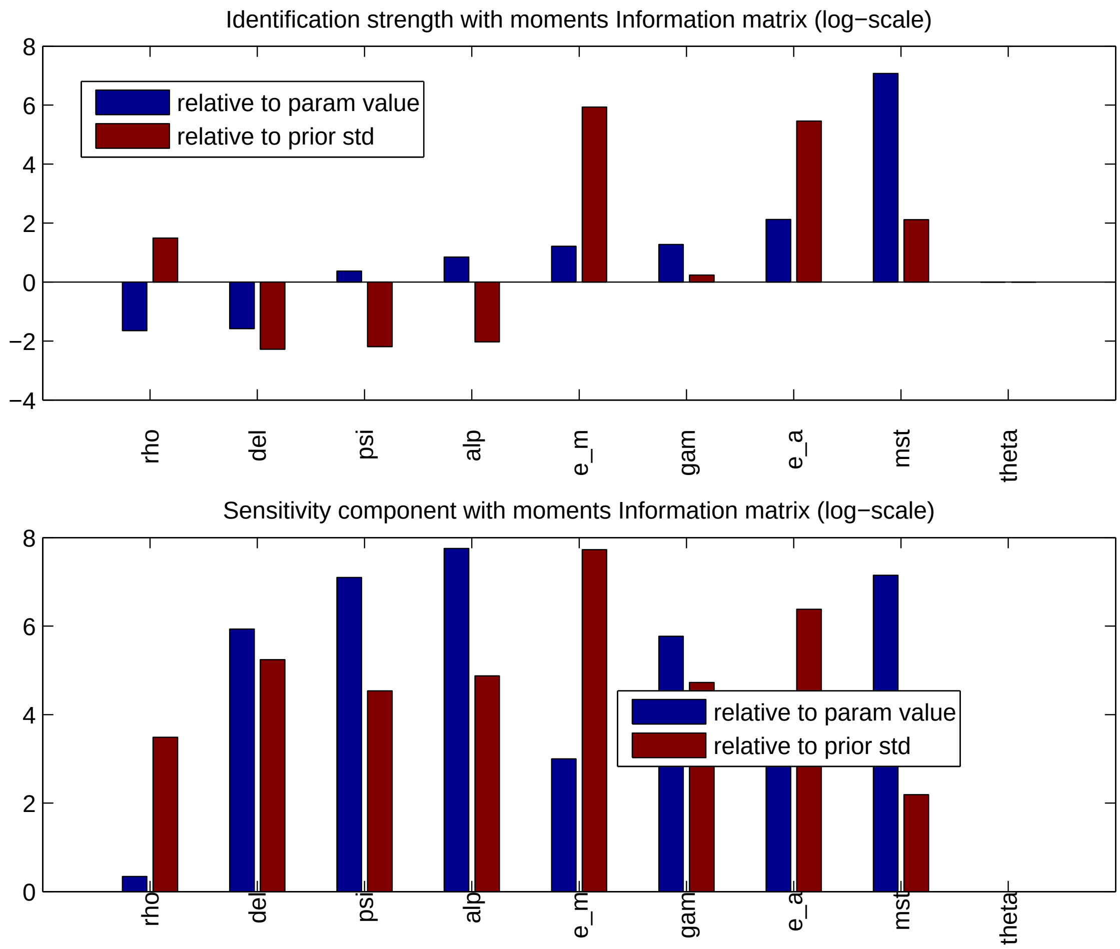

Figure 25: prior_mean - Identification using info from observables – Identification strength-plot generated by the `identification`-command. It is stored in the identification subfolder. **Upper Panel:** the bar charts depict the identification strength of the parameters based on the Fischer information matrix normalized by either the parameter at the prior mean (blue bars) or by the standard deviation at the prior mean (red bars) (see Ratto and Iskrev, 2011, p. 15). The weighting with the prior standard deviation is only available if priors have been specified. For ML estimation, no blue bars will be displayed. The weighting with the prior standard deviation uses the prior uncertainty for weighting and is particularly useful for cases where the prior mean is 0. In this case the weighting with the prior mean would falsely signal an identification strength of 0. The Fisher information matrix is either computed analytically (Iskrev, 2011) or based on simulations. Which version has been used is indicated in the figure title. Intuitively, the bars represent the normalized curvature of the log likelihood function at the prior mean in the direction of the parameter. If the strength is 0 as is the case for theta in the rightmost diagram, the parameter is not identified as the likelihood function is flat in this direction. In contrast, the larger the absolute value of the bars, the stronger is the identification. Note that the graphs generally use a log-scale except for parameters that are unidentified, which are shown with a bar length of exactly 0, i.e. no bar at all.

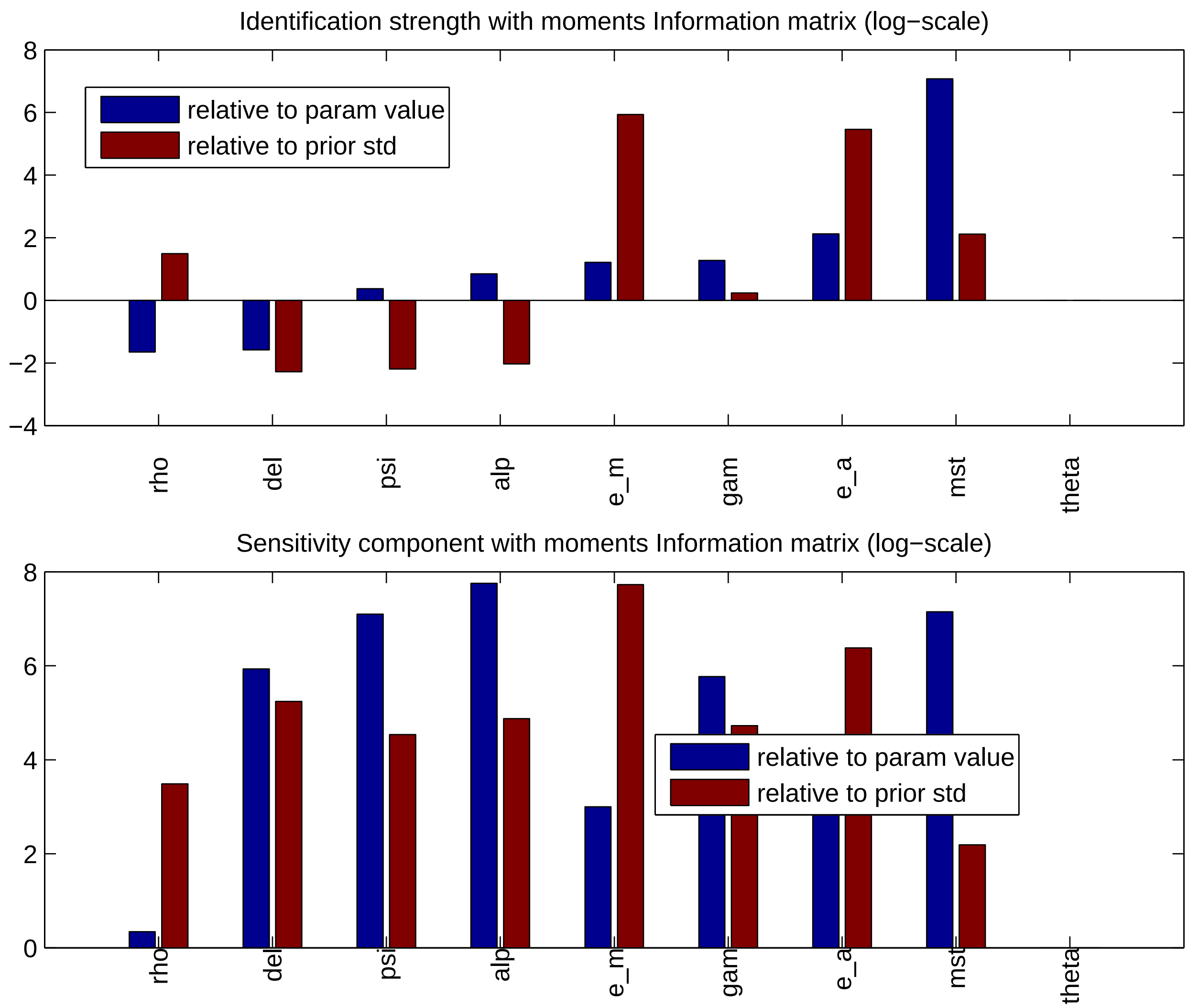

Figure 26: prior_mean - Identification using info from observables – Identification strength-plot generated by the `identification`-command. It is stored in the identification subfolder. **Lower Panel:** This panel further decomposes the effect shown in the upper panel. A weak identification can be due to either other parameters linearly compensating/replacing the effect of a parameter (i.e. parameters having exactly the same effect on the likelihood) or the fact that the likelihood does not change at all with the respective parameter. This latter effect is called sensitivity (computed according to formula (12) in Ratto and Iskrev (2011)) and is plotted in the bottom panel. Again, weighting can take place either with the prior mean (blue bars) or the prior standard deviation (red bars). The weighting with the prior standard deviation is only available if priors have been specified. For ML estimation, no blue bars will be displayed. The bottom panel shows that the sensitivity of theta is 0. Thus, theta is not identified at the local mean, because it does not affect the likelihood and hence the model moments.

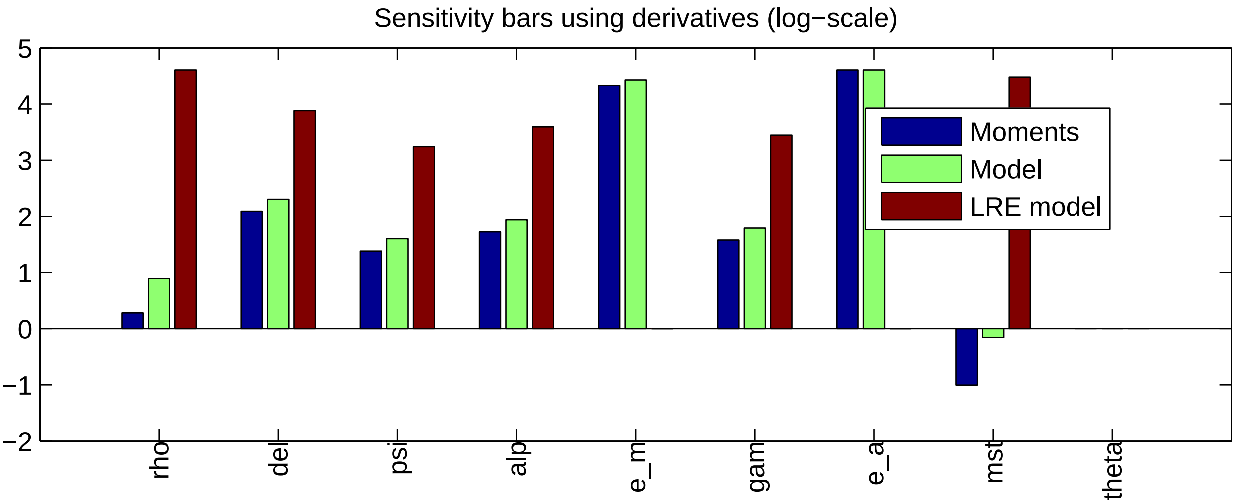

Figure 27: prior_mean - Sensitivity plot – Sensitivity-plot generated by the identification-command. It is stored in the identification subfolder. It shows an aggregate measure of how changes in the elements of the parameter vector θ impact on the model moments. The "impact" is measured locally using the Jacobian, i.e. the local derivatives. The problem is that derivatives are not scale-invariant and thus not easily comparable. Hence, a normalization of the derivative of the jth moment of the moment vector, m_j, with respect to the parameter entry i, $\partial m_j/\partial\theta_i$ is performed. This is done by multiplying with the ratio of standard deviations $std(\theta_i)/std(m_j)$ (square root of formula (16) in Ratto and Iskrev (2011)). This normalization of the change in the parameter i, $\partial\theta_i$, by its variance, $std(m_j)$, accounts for different parameter uncertainty by ascribing ceteris paribus more importance to more variable parameters, because they will be responsible for higher changes in the moments (thereby effectively normalizing across parameters i). At the same time, the normalization of the change in the moment ∂m_j with its standard deviation, $std(m_j)$, allows for comparing the impact of parameter i on differently volatile moments (effectively normalizing across moments j). Thus, when taking the norm (i.e. length) of the columns of the standardized Jacobian, one ends up with a single aggregate sensitivity measure over all moments j for each parameter i. Dynare plots three different measures of sensitivity. The bars shown in the figure depict the norm of the columns of three different standardized Jacobian matrices for the respective parameter shown on the x-axis. The respective Jacobian matrices refer to i) the moments matrix $(\partial m_T/\partial\theta')$, indicting how well a parameter can be identified due the strength of its impact on the observed moments, ii) the model solution matrices $(\partial\tau/\partial\theta')$, indicating how well a parameter could in principle be identified if all state variables were observed, and iii) the Linear Rational Expectations (Linear Rational Expectation (LRE)) model $(\partial\gamma/\partial\theta')$, indicating trivial cases of non-identifiability due to e.g. some parameters always showing up as a product only in the model equations. If the moment matrix indicates non-identifiability and the model solution matrix indicates identifiability, it means that your observables in the var_obs-command are not sufficient.

As the sensitivity is analyzed at the prior mean, this is a point estimate which uses analytic derivatives to compute the variance of the moments m.

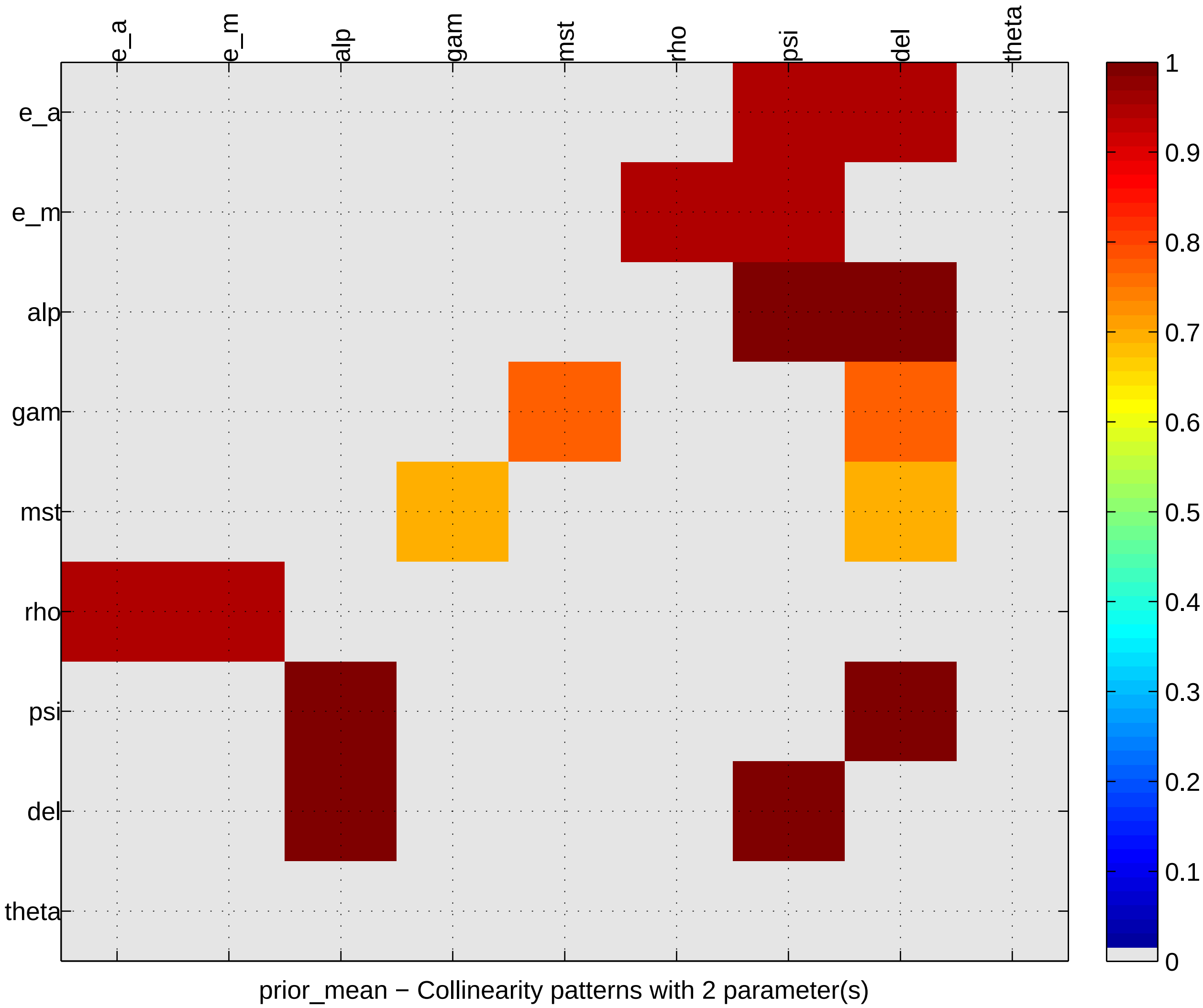

Figure 28: prior_mean - Collinearity pattern with 2 parameter(s) – Collinearity pattern-plot generated by the `identification`-command. It is stored in the identification subfolder. The plot shows which linear combination of parameters shown in the columns best replicates/replaces the effect of the parameter depicted in the row on the moments of the observables. Higher values imply the relative redundancy and thus weak or un-identifiability of the parameter under consideration. This analysis is conducted via brute force. For each single parameter a set of regressions is run of the column of the Jacobian corresponding to the parameter in the row on all possible combinations of x other Jacobian columns ($x \in \{1, \ldots, \texttt{max\_dim\_cova\_group}\}$). The aim is finding the column (and thus parameter) combination with the highest R^2. The resulting collinearity pattern between the parameter in the row and the set of parameters in the columns is then shown in the figure. Dynare generates plots for each set size of columns and thus parameters starting at 1 up to `max_dim_cova_group`. The depicted plot shows an example for a set size of two. The darker red the squares are, the more critical is the collinearity between parameters. For example, the first row signifies that there is a strong correlation between the effect of e_a on the model moments and the effect of alp and psi.

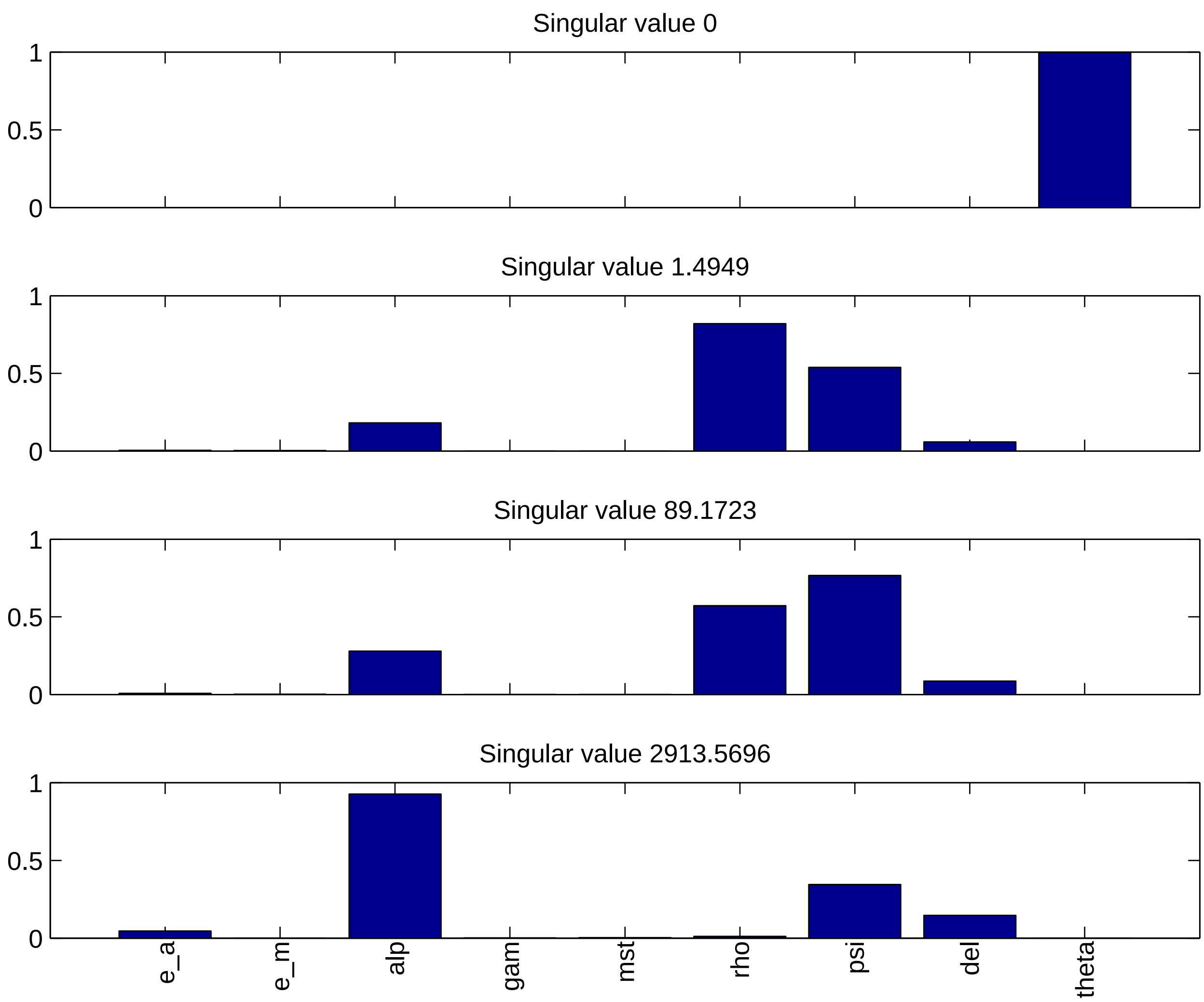

Figure 29: prior_mean - Identification patterns (Information matrix): SMALLEST SV – Identification pattern-plot generated by the `identification`-command. It is stored in the identification subfolder. Following Andrle (2010), the parameter groups with the strongest and weakest identification can be identified from the singular value decomposition (Singular Value Decomposition (SVD)) of the Fischer information matrix. This graph shows the smallest singular values (Singular Value (SV)) and the associated eigenvectors of parameters. The parameter combinations associated with the smallest singular values are closest to being perfectly collinear and thus redundant. A singular value of 0, as is the case for theta, implies that the parameter is completely unidentified as it is responsible for the information matrix being rank deficient due to the parameter having no effect at all on the likelihood.

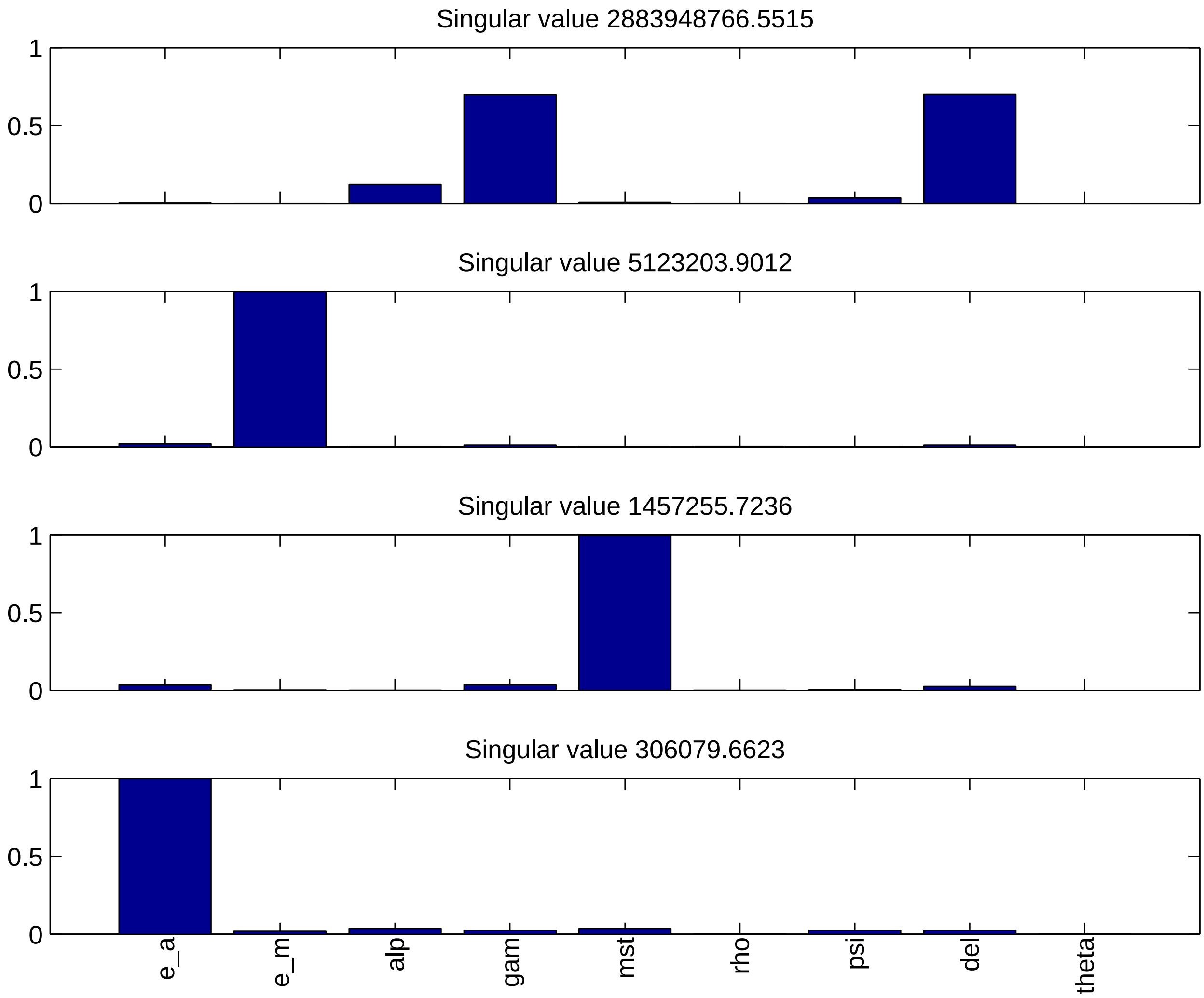

Figure 30: prior_mean - Identification patterns (Information matrix): HIGHEST SV –Identification pattern-plot generated by the `identification`-command. It is stored in the identification subfolder. This graph shows the largest singular values and the associated eigenvectors of parameters. It has the same interpretation as the graph for the smallest singular values, Figure 29, except for now depicting the parameters being most uncorrelated and thus best identified.

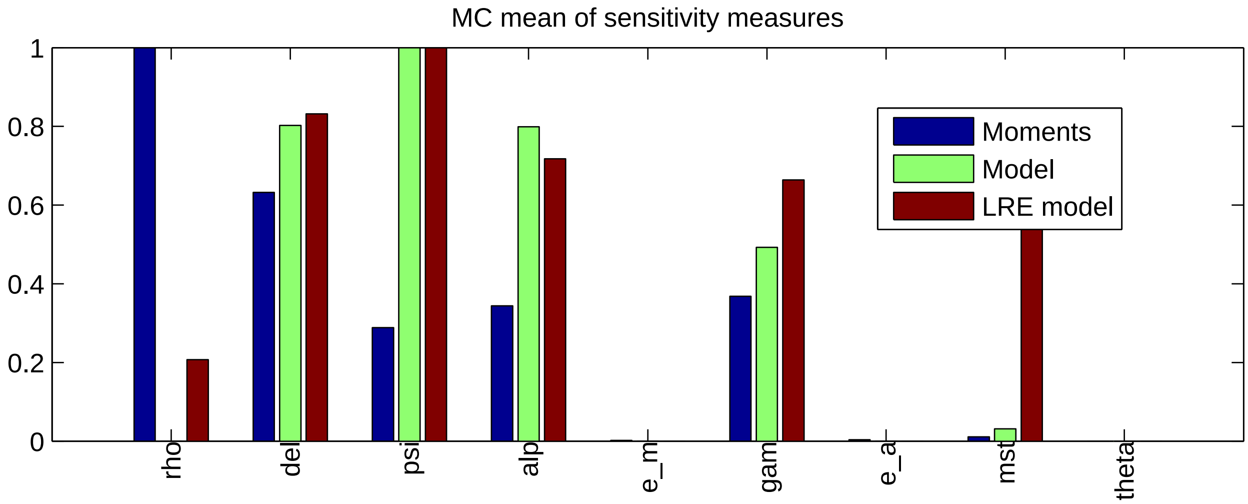

Figure 31: MC sensitivities – Collinearity pattern-plot generated by the `identification`-command. It is stored in the identification subfolder. It shows an aggregate measure of how changes in the elements of the parameter vector θ impact on the model moments. The interpretation of the graphs is the same as for the prior mean sensitivities depicted in Figure 27, except for the measure now being not a point estimate but being averaged over the Monte Carlo (Monte Carlo (MC)) sample of the parameters θ, thus providing a description of sensitivity over the whole parameter space.

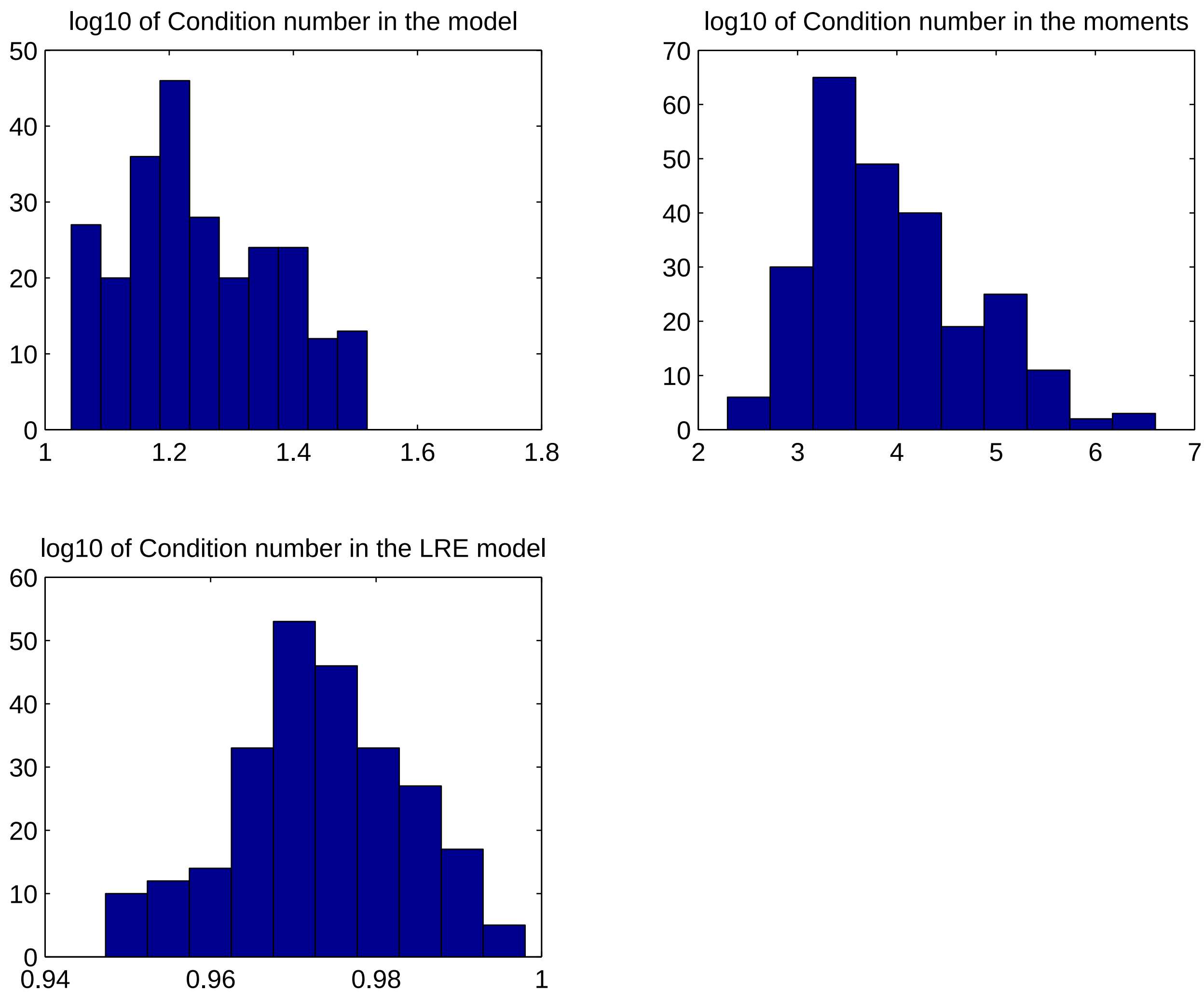

Figure 32: MC condition number – Condition number-plot generated by the `identification`-command. Requires the `advanced` and `prior_mc`-options. It is stored in the identification subfolder. It depicts the condition numbers of the Jacobians of the model moments ($\partial m_T/\partial\theta'$, first plot), the model solution matrices ($\partial\tau/\partial\theta'$, second plot), and the LRE model ($\partial\gamma/\partial\theta'$, third plot) encountered during the MC simulations. A large condition number signals near-singularity of the respective Jacobian and thus weak identifiability. Hence, the plot provides an overview about the occurrence of identification problems over the whole parameter range as encountered during MC simulations.

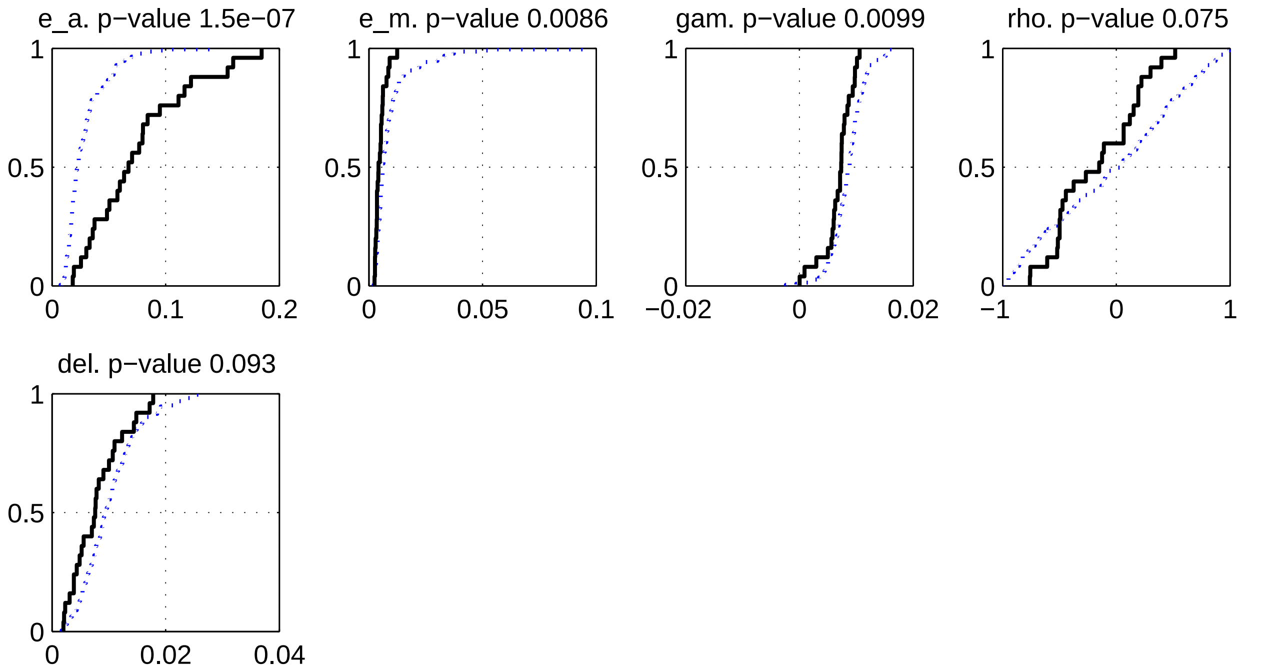

Figure 33: MC condition number – Condition number-plot generated by the `identification`-command. Requires the `advanced` and `prior_mc`-options. It is stored in the identification subfolder. This plot uses the condition number of the Jacobian to trace out which parameters are only weakly identified and thus responsible for the Jacobian being close to singular. To do so, the parameters encountered during MC are split into two samples, depending on whether they occurred in a draw with a condition number above the median or not. Testing whether the parameter distributions in both samples significantly differ, provides an indication whether the parameter under consideration is the source of the Jacobian being close to singular.
The dotted line represents the cumulative distribution of parameter values associated with condition numbers in the bottom half, while the solid line depicts the cumulative distribution associated with condition numbers in the upper half. The p-value above each subplot indicates the probability of both distributions being equal according to a Smirnov test (see Ratto (2008) for details about this procedure and its foundation in Monte Carlo Filtering). Generally, only parameters with p-value below 0.1 are plotted in Dynare.
Dynare generates three plots corresponding to the condition numbers of the Jacobians of the model moments ($\partial m_T/\partial\theta'$, first plot), the model solution matrices ($\partial\tau/\partial\theta'$, second plot), and the LRE model ($\partial\gamma/\partial\theta'$, third plot) encountered during the MC simulations.

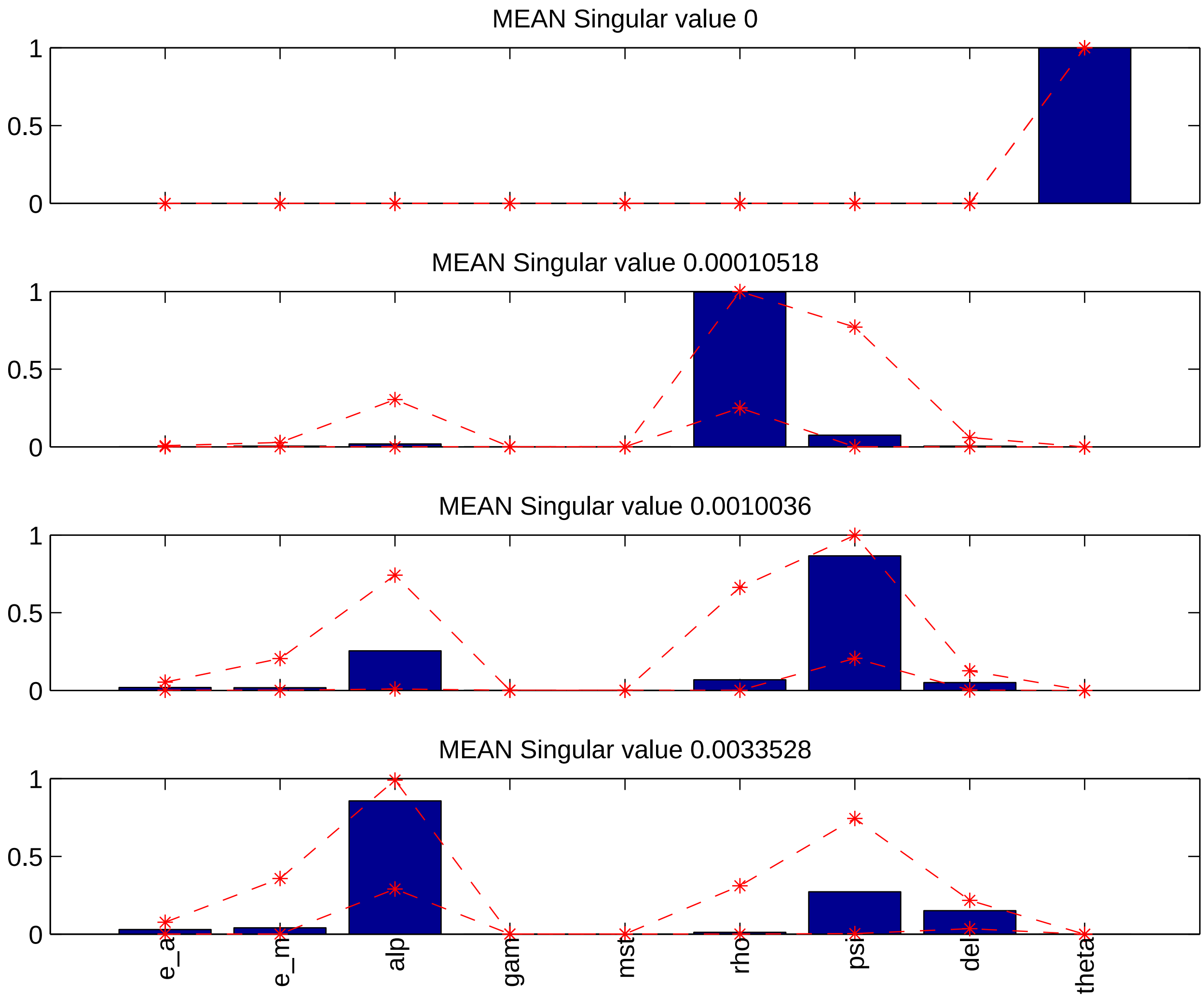

Figure 34: MC Sample - MC identification patterns (moments): SMALLEST SV – Identification pattern-plot generated by the `identification`-command. Requires the `advanced` and `prior_mc`-options. It is stored in the identification subfolder. It shows the distribution of the parameter vectors associated with the smallest singular value of the Jacobian (see Figure 29 for a description for the case of the information matrix) during the MC runs. Note that in contrast to Figure 29 the SVD applies to the Jacobian of the moments m with respect to the parameters θ, not the Fischer information matrix. This also explains the large differences in the scale of the singular values.

The blue bars depict the mean value, while the red starred lines represent the 90% quantiles. If the quantiles are narrow or one quantile is close to the mean, it suggests that the identification pattern is similar over the whole parameter space. This can be clearly seen for theta, which is unidentified over the whole sample. Similarly, for the second smallest singular value, the mean of *rho* is identical to the upper 90% bound, while the other means are close to the lower bound. This suggests that *psi* is weakly identified over the whole parameter space and that the weak identification is not due to collinearity with other parameters. This evidence is consistent with second panel of Figure 29.

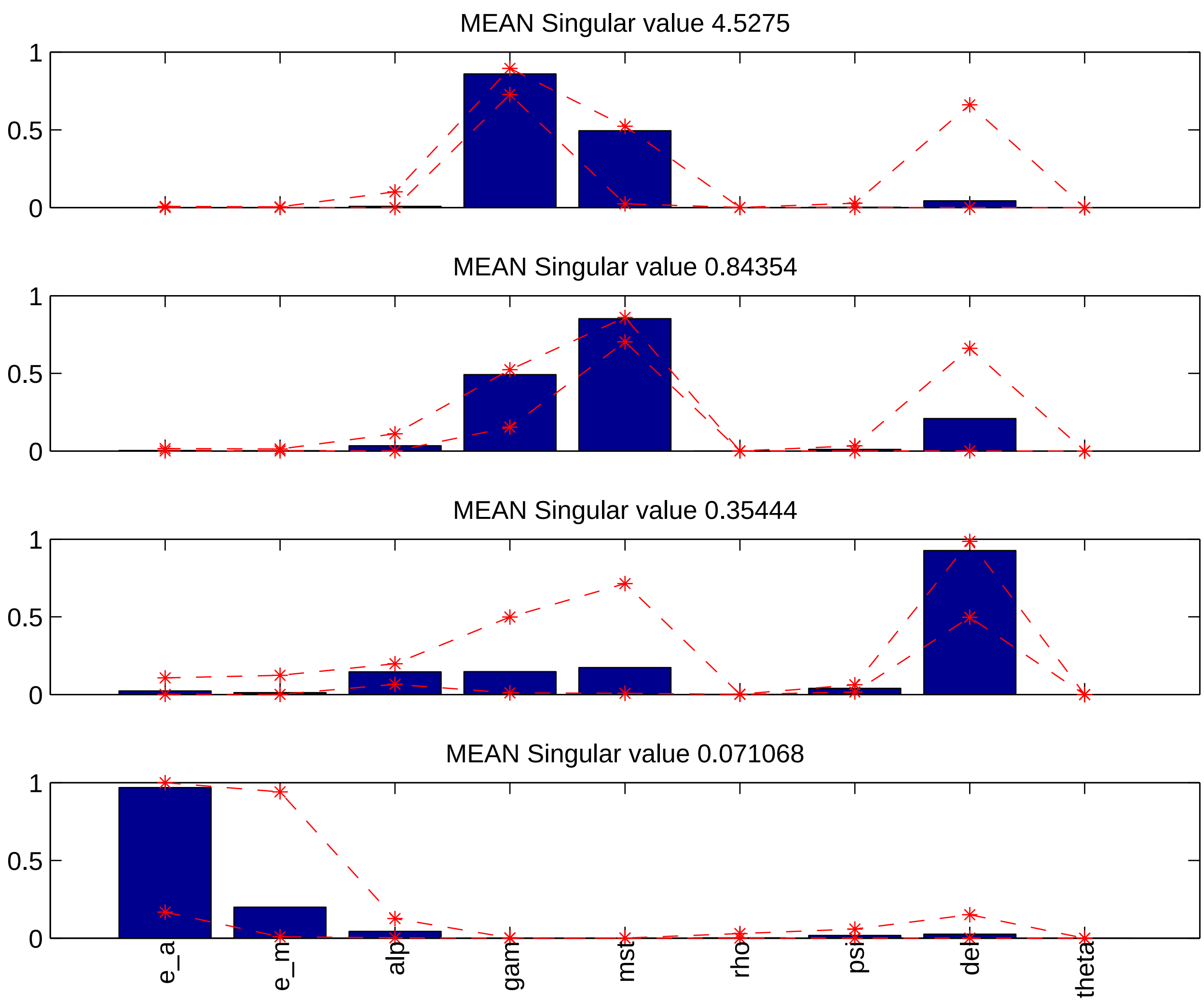

Figure 35: MC Sample - MC identification patterns (moments): HIGHEST SV – Identification pattern-plot generated by the `identification`-command. Requires the `advanced` and `prior_mc`-options. It is stored in the identification subfolder. It shows the distribution of the parameter vectors associated with the largest singular value of the Jacobian of the moments (see Figures 30 and 34) during the MC runs. This figure is similar to Figure 34, except for showing the parameter combinations with the strongest identification.

7 Graphs Produced by `dynare_sensitivity`

8 Graphs Produced by `bvar_forecast`

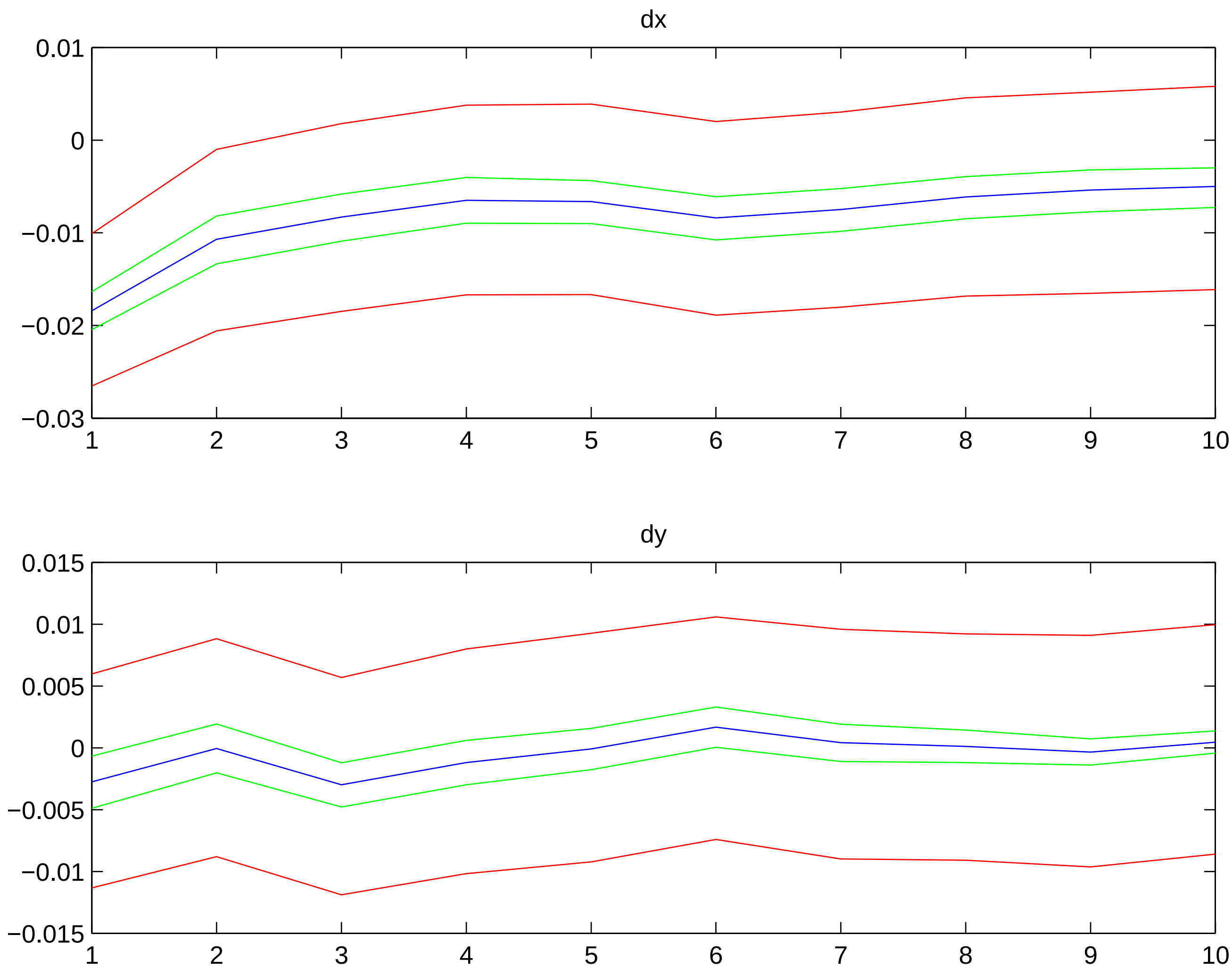

Figure 36: BVAR forecasts (nlags=8) – BVAR forecasts plot generated by the `bvar_forecast`-command. It is stored in the `graphs`-subfolder. The blue line in the middle is the median forecast of the Bayesian Vector Autoregression (BVAR) in the absence of shocks, i.e. given only parameter uncertainty. The area between the green lines covers the `options_.conf_sig` percent HPDI interval (default: 60%) of the forecasts given only parameter uncertainty. The area between the red lines covers the `options_.conf_sig` percent HPDI interval (default: 60%) of the forecasts given both parameter uncertainty and uncertainty about future shock realization.

9 Graphs Produced by `forecast`

The graphs generated by the `forecast`-command are the same as for the `forecast`-option of the estimation command with Maximum Likelihood-estimation shown in Figure 19.

10 Graphs Produced by `conditional_forecast`

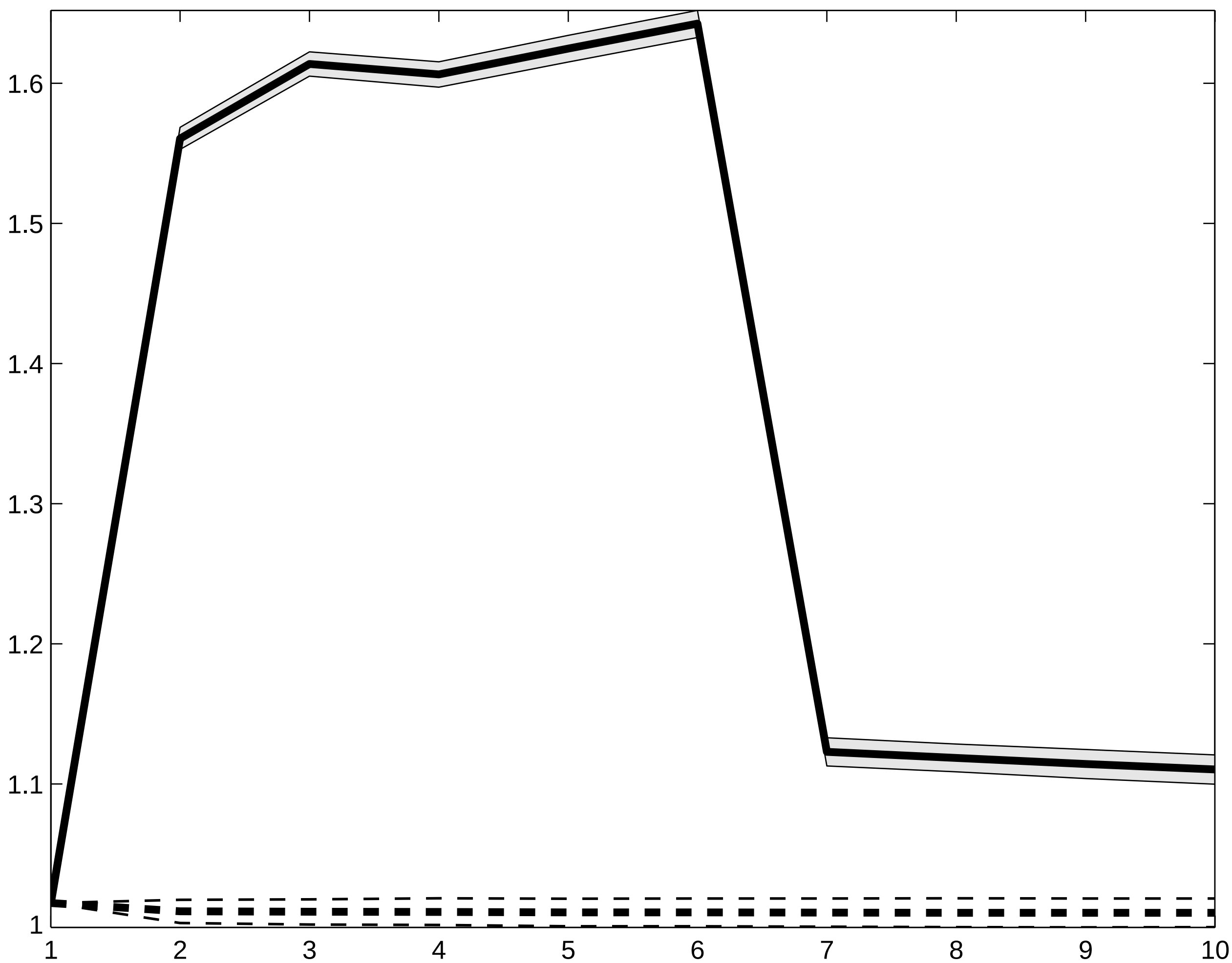

Figure 37: Conditional forecast (Posterior Mode): gp_obs – Conditional forecast plot generated by the `conditional_forecast`-command. It is stored in the `graphs`-subfolder. The solid black line is the mean conditional forecast. The grey area around the black solid line covers the `options_.conf_sig` percent confidence interval (default: 80%), computed as the corresponding percentiles of the forecasts. The dashed black line is the mean unconditional forecast. The area between the thin black dashed lines covers the `options_.conf_sig` percent confidence interval (default: 80%) of the unconditional forecasts, computed as the corresponding percentiles of the forecasts.

11 Graphs Produced by the Markov Switching VAR Codes

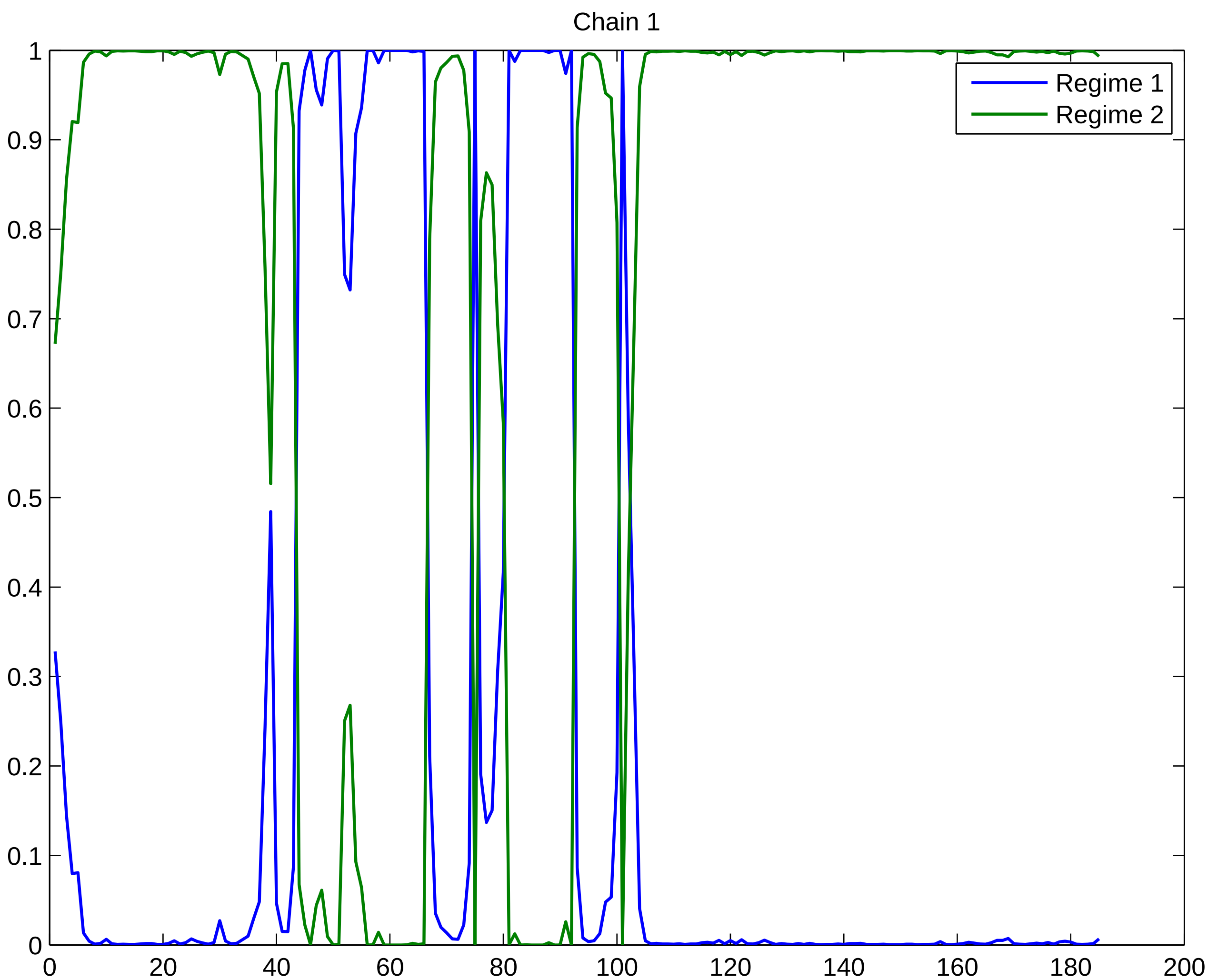

Figure 38: MS-Probabilities, Chain 1 – Markov Switching (MS) probabilities plot generated by the `ms_compute_probabilities`-command. It is stored in the `Output/Probabilities`-subfolder. The x-axis shows the number of time periods and the y-axis the probabilities of the corresponding regimes. The title shows the number of the corresponding Markov Chain (with the total number set by the `chain`-option) for which the probabilities are displayed. By default, the probabilities are smoothed probabilities, i.e. using information up to time T. Using the `filtered_probabilities`-option of the `ms_compute_probabilities`, filtered probabilities using only information up to time t can be requested.

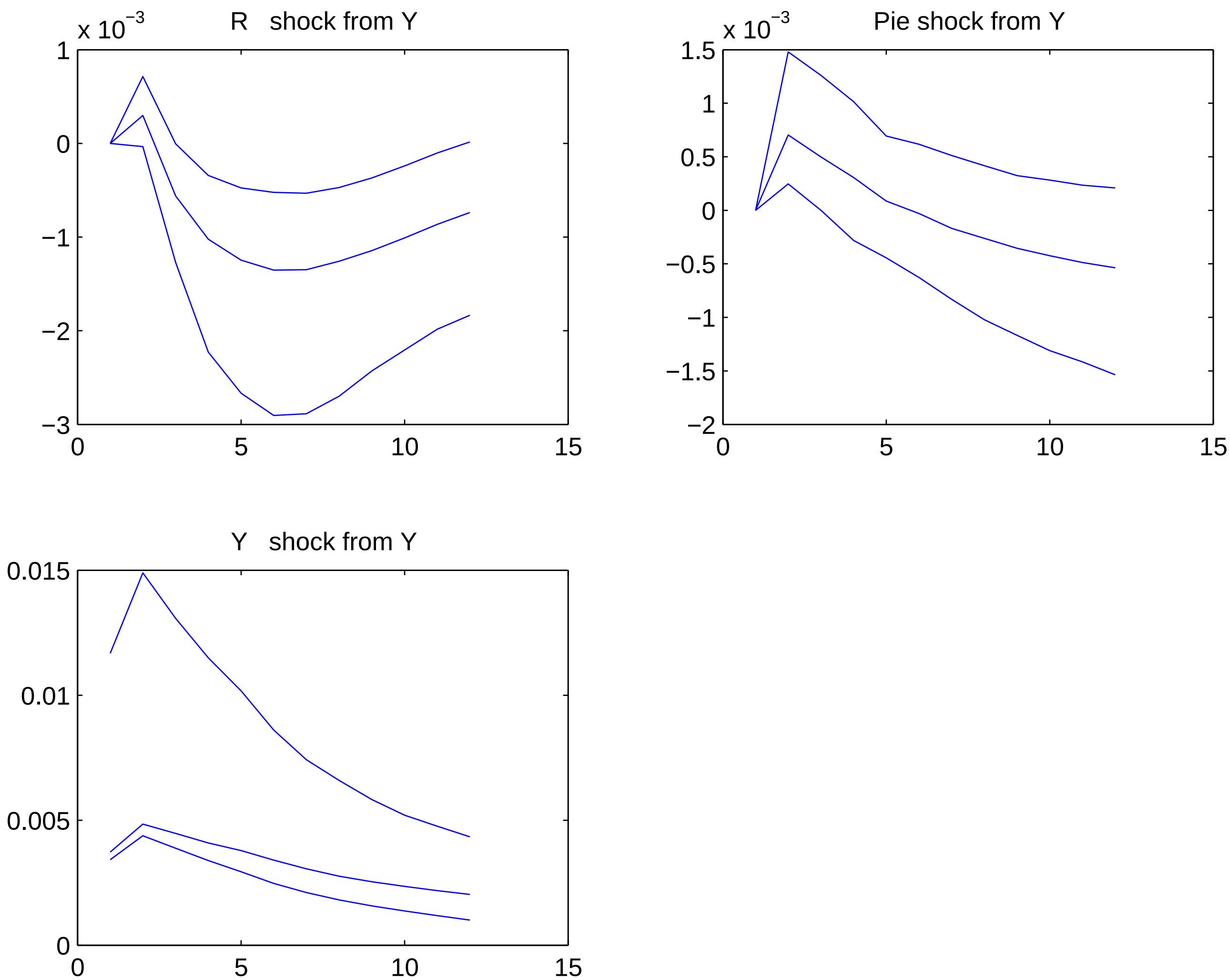

Figure 39: Impulse Response Ergodic – Impulse response function plot generated by the `ms_irf`-command. It is stored in the `Output/IRF`-subfolder. The x-axis shows the number of time periods and the y-axis the deviation from the unconditional mean. The middle line shows the median IRF, while the upper and lower blue line depict the percentiles of the IRF distribution. They can be set using the `error_band_percentiles`-command and are 16% and 84% percentiles by default.

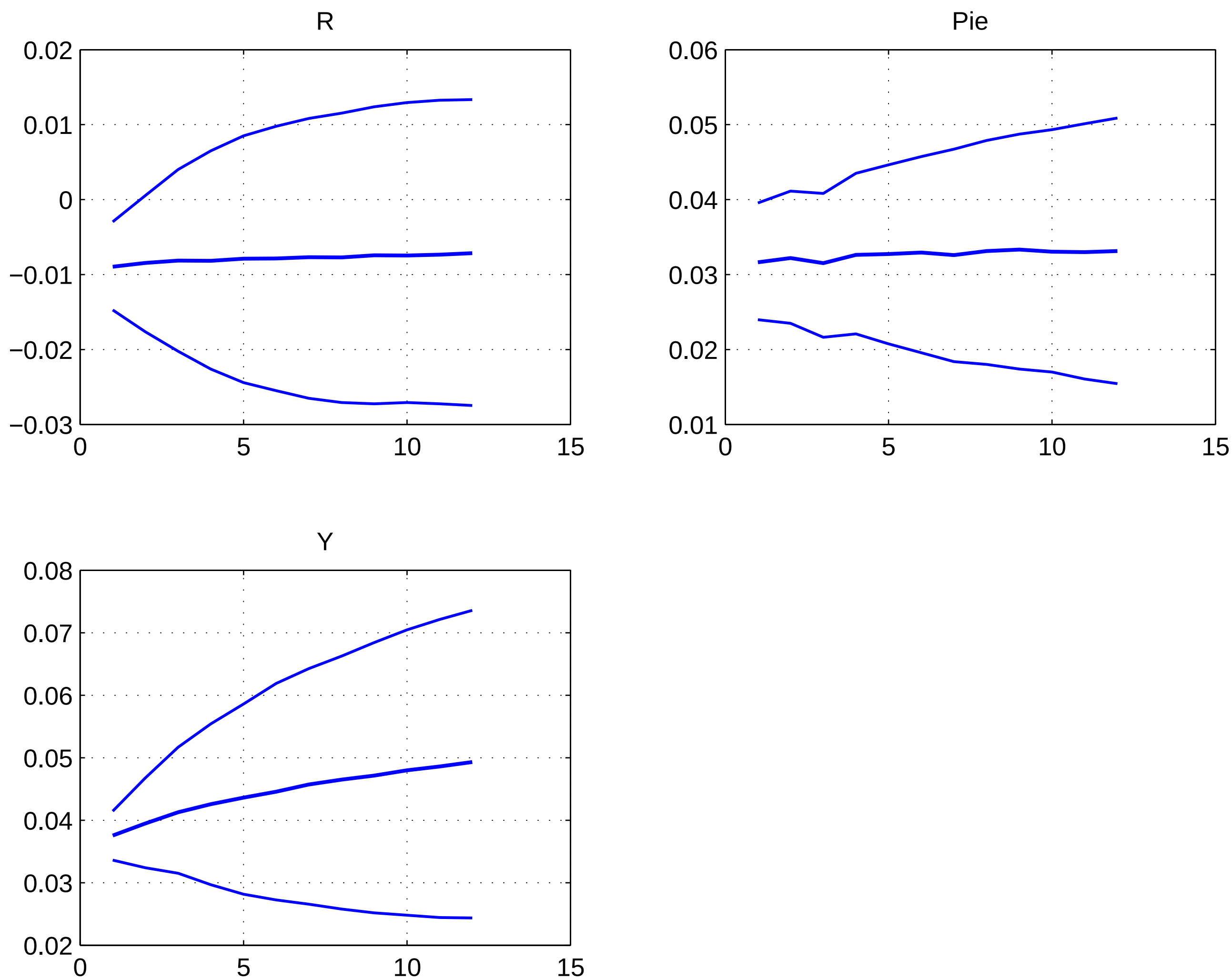

Figure 40: Forecast – Forecast plot generated by the `ms_forecast`-command. It is stored in the `Output/Forecast`-subfolder. The x-axis shows the number of time periods and the y-axis the forecasted deviation from the unconditional mean. The thick blue line is the mean of median forecast, with the two thin blue lines indicating the upper and lower percentiles of the forecast distribution. The percentiles are set in `options_.ms.percentiles` and are 16% and 84% by default. The panel title shows the name of the respective variable.

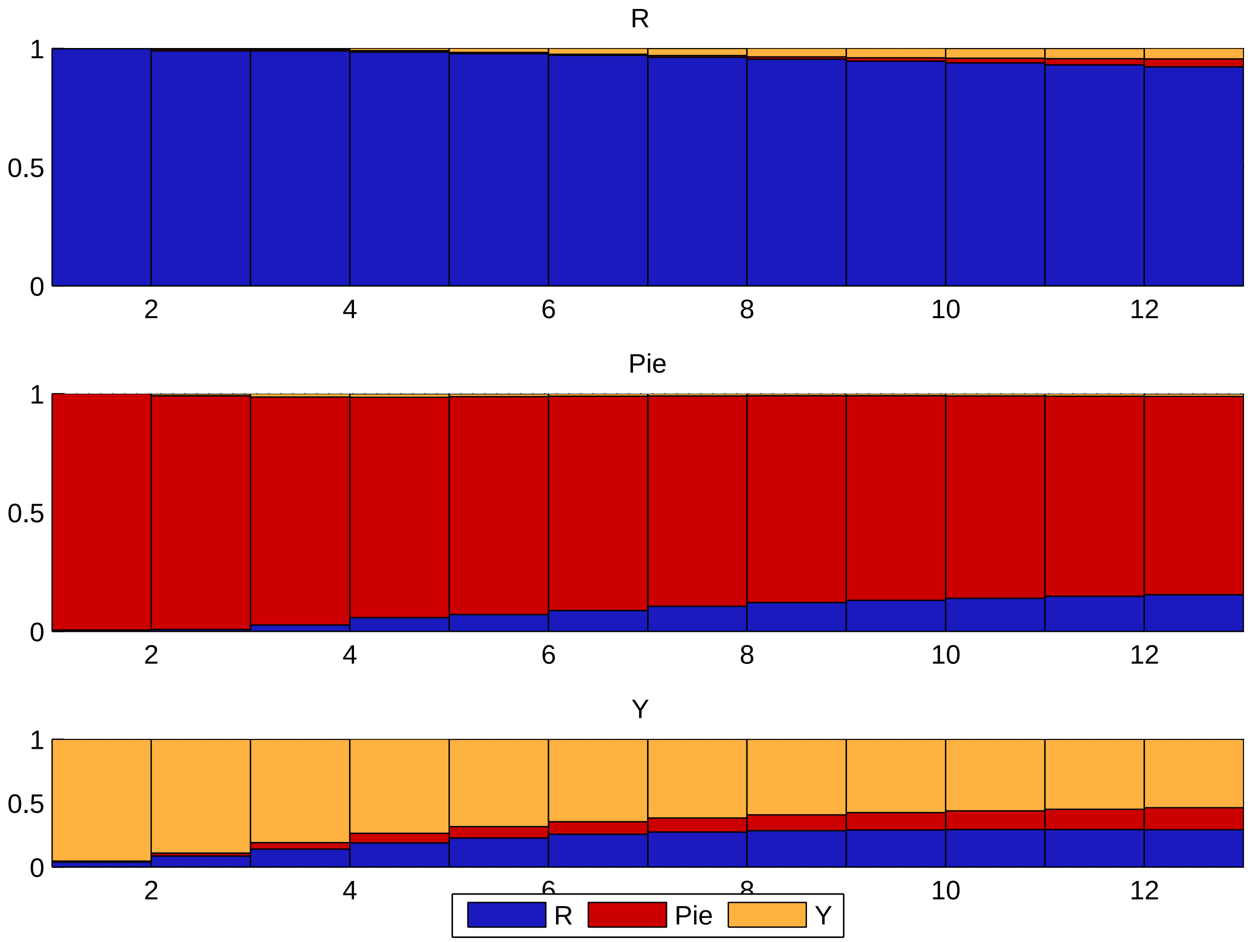

Figure 41: Variance Decomposition Ergodic – Forecast Error Variance Decomposition plot generated by the `ms_variance_decomposition`-command. It is stored in the `Output/Variance_Decomposition`-subfolder. The x-axis shows the forecast horizon and the stacked colored columns the contribution of the corresponding shock to the overall forecast error variance at the respective horizon. The title shows the name of the variable for which the forecast error variance decomposition is performed.

References

Adjemian, Stéphane, Houtan Bastani, Michel Juillard, Frédéric Karamé, Ferhat Mihoubi, George Perendia, Johannes Pfeifer, Marco Ratto, and Sébastien Villemot (2011). "Dynare: Reference Manual, Version 4". Dynare Working Papers, 1. CEPREMAP.

Andrle, Michal (2010). "A note on identification patterns in DSGE models". Working Paper Series 1235. European Central Bank.

Brooks, Stephen P and Andrew Gelman (1998). "General methods for monitoring convergence of iterative simulations". *Journal of Computational and Graphical Statistics 7* (4), 434–455.

Canova, Fabio (2007). "Bayesian Analysis of DSGE Models by S. An and F. Schorfheide". *Econometric Reviews 26* (2-4), 187–192. DOI: 10.1080/07474930701220162. eprint: http://www.tandfonline.com/doi/pdf/10.1080/07474930701220162.

Iskrev, Nikolay (2011). "Evaluating the strength of identification in DSGE models. An a priori approach". Tech. rep. Banco de Portugal.

Ratto, Marco (2008). "Analysing DSGE Models with Global Sensitivity Analysis". *Computational Economics 31* (2), 115–139.

Ratto, Marco and Nikolay Iskrev (2011). "Algorithms for identification analysis under the Dynare environment: final version of the software". Tech. rep. Joint Research Centre, European Commission.