JRC Scientific and Technical Reports

European Community project MONFISPOL (grant agreement SSH-CT-2009-225149)

Deliverable 3.1.2: Algorithms for identification analysis under the DYNARE environment: final version of software.

Marco Ratto, Nikolai Iskrev

EUR 25032 EN - 2011

JRC
EUROPEAN COMMISSION

ipSc
Institute for the Protection and Security of the Citizen

The mission of the JRC-IPSC is to provide research results and to support EU policy-makers in their effort towards global security and towards protection of European citizens from accidents, deliberate attacks, fraud and illegal actions against EU policies.

European Commission
Joint Research Centre
Institute for the Protection and Security of the Citizen

Contact information
Address: TP361, via Fermi, Ispra (VA), 21027 Italy
E-mail: marco.ratto@jrc.ec.europa.eu
Tel.: +39 0332 785639
Fax: +39 0332 785733

http://ipsc.jrc.ec.europa.eu/
http://www.jrc.ec.europa.eu/

Legal Notice
Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use which might be made of this publication.

JRC 67624

EUR 25032 EN
ISBN 978-92-79-22035-7
ISSN 1831-9424
doi:10.2788/97652

Luxembourg: Publications Office of the European Union

Printed in Italy

European Community project MONFISPOL (grant agreement SSH-CT-2009-225149)

Deliverable 3.1.2

Algorithms for identification analysis under the DYNARE environment: final version of software.

Marco Ratto, Joint Research Centre, European Commission
with the contribution of Nikolai Iskrev, Bank of Portugal

August 31, 2011

Abstract

In this report we document in detail the Identification package developed for the DYNARE environment. The package implements methodologies and collects developed algorithms to assess identification of DSGE models in the entire prior space of model deep parameters, by combining 'classical' local identification methodologies and global tools for model analysis, like global sensitivity analysis.

1 Summary

In developing the identification software, we took into consideration the most recent developments in the computational tools for analyzing identification in DSGE models. A growing interest is being addressed to identification issues in economic modeling (Canova and Sala, 2009; Komunjer and Ng, 2009; Iskrev, 2010b). The identification toolbox includes the new efficient method for derivatives computation presented in Ratto and Iskrev (2010a,b) and the identification tests proposed by Iskrev (2010a,b).

1.1 Main features of the software

The new DYNARE keyword `identification` triggers the routines developed at JRC. This option has two modes of operation.

Point estimate : this is the default option.

- when there is a prior definition for a subset of model parameters that are going to be estimated, the program performs the local identification checks for the parameters declared in the prior definition at the prior mean (prior mode, posterior mean and posterior mode are also alternative options);
- when there is no prior definition for model parameters, the program computes the local identification checks for *all* the model parameter values declared in the DYNARE model file. The parameter values used for identification computations are those defined in the model declaration.

Monte Carlo exploration : when information about prior distribution is provided, a full Monte Carlo analysis is also possible. In this case, for a number of parameter sets sampled from prior distributions, the local identification analysis is performed in turn. This provides a 'global' prior exploration of local identification properties of DSGE models. This Monte Carlo mode can also be linked to the the global sensitivity analysis toolbox, also available in the official DYNARE as of version 4.3 (currently downloadable as the 'unstable' DYNARE).

A library of test routines is also provided in the official DYNARE test folder. Such tests implement some of the examples described in the present document.

Kim (2003) : the DYNARE routines for this example are placed in the folder `dynare_root/tests/identification/kim`;

An and Schorfheide (2007) : the DYNARE routines for this example are placed in `dynare_root/tests/identification/as2007`;

2 DSGE Models

We recall here the notation adopted in Iskrev (2010b) for linearized DSGE models and the restrictions they imply on the first and second order moments of the observed variables. Readers can refer to the latter paper for a more detailed discussion of the identification criteria.

2.1 Structural model and reduced form

A DSGE model can be expressed as a system $\boldsymbol{g}$ of m non-linear equations:

$$E_t\Big(\boldsymbol{g}(\hat{\boldsymbol{z}}_t, \hat{\boldsymbol{z}}_{t+1}, \hat{\boldsymbol{z}}_{t-1}, \boldsymbol{u}_t|\boldsymbol{\theta})\Big) = 0 \tag{1}$$

where $\hat{\boldsymbol{z}}_t$ is a $m-$dimensional vector of endogenous variables, $\boldsymbol{u}_t$ an n-dimensional random vector of structural shocks with $\mathrm{E}\,\boldsymbol{u}_t = \boldsymbol{0}$, $\mathrm{E}\,\boldsymbol{u}_t\boldsymbol{u}_t' = \boldsymbol{I}_n$ and $\boldsymbol{\theta}$ a $k-$dimensional vector of deep parameters. Here, $\boldsymbol{\theta}$ is a point in $\boldsymbol{\Theta} \subset \mathbb{R}^k$ and the parameter space $\boldsymbol{\Theta}$ is defined as the set of all theoretically admissible values of $\boldsymbol{\theta}$.

Most studies involving either simulation or estimation of DSGE models use linear approximations of the original models around the steady-state $\hat{\boldsymbol{z}}^*$, namely:

$$\boldsymbol{\Gamma}_0(\boldsymbol{\theta})\boldsymbol{z}_t = \boldsymbol{\Gamma}_1(\boldsymbol{\theta})\,\mathrm{E}_t\,\boldsymbol{z}_{t+1} + \boldsymbol{\Gamma}_2(\boldsymbol{\theta})\boldsymbol{z}_{t-1} + \boldsymbol{\Gamma}_3(\boldsymbol{\theta})\boldsymbol{u}_t \tag{2}$$

where $\boldsymbol{z}_t = \hat{\boldsymbol{z}}_t - \hat{\boldsymbol{z}}^*$. The elements of the matrices $\boldsymbol{\Gamma}_0$, $\boldsymbol{\Gamma}_1$, $\boldsymbol{\Gamma}_2$ and $\boldsymbol{\Gamma}_3$ are functions of $\boldsymbol{\theta}$.

Assuming that a unique solution exists, it can be written as:

$$\boldsymbol{z}_t = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{z}_{t-1} + \boldsymbol{B}(\boldsymbol{\theta})\boldsymbol{u}_t \tag{3}$$

where the $m \times m$ matrix $\boldsymbol{A}$ and the $m \times n$ matrix $\boldsymbol{B}$ are functions of $\boldsymbol{\theta}$.

For a given value of $\boldsymbol{\theta}$, the matrices $\boldsymbol{A}$, $\boldsymbol{\Omega} := \boldsymbol{B}\boldsymbol{B}'$, and $\hat{\boldsymbol{z}}^*$ completely characterize the equilibrium dynamics and steady state properties of all endogenous variables in the linearized model. Typically, some elements of these matrices are constant, i.e. independent of $\boldsymbol{\theta}$ and it is useful to separate the solution parameters that depend on $\boldsymbol{\theta}$ from those that do not. Iskrev (2010b) uses $\boldsymbol{\tau}$ to denote the vector collecting the non-constant elements of $\hat{\boldsymbol{z}}^*$, $\boldsymbol{A}$, and $\boldsymbol{\Omega}$, i.e. $\boldsymbol{\tau} := [\boldsymbol{\tau}_z', \ \boldsymbol{\tau}_A', \ \boldsymbol{\tau}_\Omega']'$, where $\boldsymbol{\tau}_z$, $\boldsymbol{\tau}_A$, and $\boldsymbol{\tau}_\Omega$ denote the elements of $\hat{\boldsymbol{z}}^*$, $\mathrm{vec}(\boldsymbol{A})$ and $\mathrm{vech}(\boldsymbol{\Omega})$ that depend on $\boldsymbol{\theta}$.

In some cases, 'trivial' singularities in the model dependency w.r.t. $\boldsymbol{\theta}$ can directly tracked in the structural form (2), so here we will also use $\boldsymbol{\gamma}$ to denote the vector collecting the non-constant elements of $\boldsymbol{\Gamma}_0, \ \boldsymbol{\Gamma}_1, \ \boldsymbol{\Gamma}_2$ and $\boldsymbol{\Gamma}_3$, i.e. $\boldsymbol{\gamma} := [\boldsymbol{\gamma}_{\boldsymbol{\Gamma}_0}', \ldots, \ \boldsymbol{\gamma}_{\boldsymbol{\Gamma}_3}']'$, where $\boldsymbol{\gamma}_{\boldsymbol{\Gamma}_i}$ denote the elements of $\boldsymbol{\Gamma}_i$ that depend on $\boldsymbol{\theta}$.

In most applications the model in (3) cannot be taken to the data directly since some of the variables in $\boldsymbol{z}_t$ are not observed. Instead, the solution of the DSGE model is expressed in a state space form, with transition equation given by (3), and a measurement equation

$$\boldsymbol{x}_t = \boldsymbol{C}\boldsymbol{z}_t + \boldsymbol{D}\boldsymbol{u}_t + \boldsymbol{\nu}_t \tag{4}$$

where $\boldsymbol{x}_t$ is a l-dimensional vector of observed variables and $\boldsymbol{\nu}_t$ is a l-dimensional random vector with $\mathrm{E}\,\boldsymbol{\nu}_t = \boldsymbol{0}$, $\ \mathrm{E}\,\boldsymbol{\nu}_t\boldsymbol{\nu}_t' = \boldsymbol{Q}$, where $\boldsymbol{Q}$ is $l \times l$ symmetric semi-positive definite matrix [1].

In the absence of a structural model it would, in general, be impossible to fully recover the properties of $\boldsymbol{z}_t$ from observing only $\boldsymbol{x}_t$. Having the model in (2) makes this possible by imposing restrictions, through (3) and (4), on the joint probability distribution of the observables. The model-implied restrictions on the first and second order moments of the $\boldsymbol{x}_t$ are summarized

[1] In the DYNARE framework, the state-space and measurement equations are always formulated such that $\boldsymbol{D} = 0$

next (see Iskrev, 2010b, for more details).

2.2 Theoretical first and second moments

From (3)-(4) it follows that the unconditional first and second moments of $\boldsymbol{x}_t$ are given by

$$\begin{aligned} \mathrm{E}\,\boldsymbol{x}_t &:= \boldsymbol{\mu}_{\boldsymbol{x}} = \boldsymbol{s} \quad (5) \\ \operatorname{cov}(\boldsymbol{x}_{t+i}, \boldsymbol{x}_t') &:= \Sigma_{\boldsymbol{x}}(i) = \begin{cases} \boldsymbol{C}\Sigma_{\boldsymbol{z}}(0)\boldsymbol{C}' & \text{if } i = 0 \\ \boldsymbol{C}\boldsymbol{A}^i\Sigma_{\boldsymbol{z}}(0)\boldsymbol{C}' & \text{if } i > 0 \end{cases} \quad (6) \end{aligned}$$

where $\Sigma_{\boldsymbol{z}}(0) := \mathrm{E}\,\boldsymbol{z}_t\boldsymbol{z}_t'$ solves the matrix equation

$$\Sigma_{\boldsymbol{z}}(0) = \boldsymbol{A}\Sigma_{\boldsymbol{z}}(0)\boldsymbol{A}' + \boldsymbol{\Omega} \quad (7)$$

Given the observed data $\boldsymbol{X}_T := [\boldsymbol{x}_1', \ldots, \boldsymbol{x}_T']'$, and the corresponding covariance matrix $\boldsymbol{\Sigma}_T$

$$\begin{aligned} \boldsymbol{\Sigma}_T &:= \mathrm{E}\,\boldsymbol{X}_T\boldsymbol{X}_T' \\ &= \begin{pmatrix} \Sigma_{\boldsymbol{x}}(0), & \Sigma_{\boldsymbol{x}}(1)', & \ldots, & \Sigma_{\boldsymbol{x}}(T-1)' \\ \Sigma_{\boldsymbol{x}}(1), & \Sigma_{\boldsymbol{x}}(0), & \ldots, & \Sigma_{\boldsymbol{x}}(T-2)' \\ \cdots & \cdots & \cdots & \cdots \\ \Sigma_{\boldsymbol{x}}(T-1), & \Sigma_{\boldsymbol{x}}(T-2), & \ldots, & \Sigma_{\boldsymbol{x}}(0) \end{pmatrix} \quad (8) \end{aligned}$$

$\boldsymbol{\sigma}_T$ is the vector collecting the unique elements of $\boldsymbol{\Sigma}_T$, i.e.

$$\boldsymbol{\sigma}_T := [\mathrm{vech}(\Sigma_{\boldsymbol{x}}(0))', \mathrm{vec}(\Sigma_{\boldsymbol{x}}(1))', ..., \mathrm{vec}(\Sigma_{\boldsymbol{x}}(T-1))']'$$

As in Iskrev (2010b) we can then define $\boldsymbol{m}_T := [\boldsymbol{\mu}', \boldsymbol{\sigma}_T']'$, a $(T-1)l^2 + l(l+3)/2$-dimensional vector collecting the parameters that determine the first two moments of the data. Assuming that the linearized DSGE model is determined everywhere in $\boldsymbol{\Theta}$, i.e. $\boldsymbol{\tau}$ is unique for each admissible value of $\boldsymbol{\theta}$, it follows that $\boldsymbol{m}_T$ is a function of $\boldsymbol{\theta}$. If either $\boldsymbol{u}_t$ is Gaussian, or there are no distributional assumptions about the structural shocks, the model-implied restrictions on $\boldsymbol{m}_T$ contain all information that can be used for the estimation of $\boldsymbol{\theta}$. The identifiability of $\boldsymbol{\theta}$ depends on whether that information is sufficient or not. This is the subject of the next section, where the main results and identification criteria of Iskrev are summarized.

3 Identification

This section recalls the role of the Jacobian matrix of the mapping from $\boldsymbol{\theta}$ to $\boldsymbol{m}_T$ for identification, as discussed in Iskrev (2010b), and summarizes the main computational results concerning its analytic derivation.

3.1 The rank condition

In most applications the distribution of $\boldsymbol{X}$ is unknown or assumed to be Gaussian. Thus, the estimation of $\boldsymbol{\theta}$ is usually based on the first two moments of the data. If the data is not normally distributed, higher-order moments may provide additional information about $\boldsymbol{\theta}$, not contained in the first two moments. Therefore, identification based on the mean and the variance of $\boldsymbol{X}$ is only sufficient but not necessary for identification with the complete distribution. In general, there are no known global conditions for unique solutions of systems of non-linear equations, and it is therefore difficult to establish the global identifiability of $\boldsymbol{\theta}$. Local identification, on the other hand, can be verified with the help of the following condition

Theorem 1. *Suppose that $\boldsymbol{m}_T$ is a continuously differentiable function of $\boldsymbol{\theta}$. Then $\boldsymbol{\theta}_0$ is locally identifiable if the Jacobian matrix $J(q) := \dfrac{\partial \boldsymbol{m}_q}{\partial \boldsymbol{\theta}'}$ has a full column rank at $\boldsymbol{\theta}_0$ for $q \leq T$. This condition is both necessary and sufficient when $q = T$ if $\boldsymbol{u}_t$ is normally distributed.*

Note that, $J(T)$ having full rank is necessary for identification from the first and second order moments. Therefore, when the rank of $J(T)$ is less than k, $\boldsymbol{\theta}_0$ is said to be unidentifiable from a model that utilizes only the mean and the variance of $\boldsymbol{X}_T$. A necessary condition for identification in that sense is that the number of deep parameters does not exceed the dimension of $\boldsymbol{m}_T$, i.e. $k \leq (T-1)l^2 + l(l+3)/2$.

Another necessary condition discussed in Iskrev (2010b) reads:

Corollary 1. *The point $\boldsymbol{\theta}_0$ is locally identifiable only if the rank of $J_2 = \frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\theta}'}$ at $\boldsymbol{\theta}_0$ is equal to k.*

The condition is necessary because the distribution of $\boldsymbol{X}_T$ depends on $\boldsymbol{\theta}$ only through $\boldsymbol{\tau}$, irrespectively of the distribution of $\boldsymbol{u}_t$. It is not sufficient since, unless all state variables are observed, $\boldsymbol{\tau}$ may be unidentifiable.

The local identifiability of a point $\boldsymbol{\theta}_0$ can be established by verifying that the Jacobian matrix $J(T)$ has full column rank when evaluated at $\boldsymbol{\theta}_0$. With the help of the Jacobian matrix we can detect problems that are a common cause for identification failures in these models. If, for instance, a deep parameter θ_j does not affect the solution of the model, $\frac{\partial \boldsymbol{m}_T}{\partial \theta_j}$ - the column of $J(T)$ corresponding to θ_j, will be a vector of zeros for any T, and the rank condition for identification will fail. Another type of identification failure occurs when two or more parameters enter in the solution in a manner which makes them indistinguishable, e.g. as a product or a ratio. As a result it will be impossible to identify the parameters separately, and some of the columns of the Jacobian matrix will be linearly dependent. For small scale models, like Cochrane (2007) or Kim (2003), these problems are discovered by solving the models explicitly in terms of the deep parameters. That approach, however, is not feasible for larger models, which can only be solved numerically. However, the Jacobian matrix in Theorem 1 is straightforward to compute analytically for linearized models of any size or complexity, as shown in (Iskrev, 2010b) and in Ratto and Iskrev (2010a,b). We summarize these results next.

3.2 Computing the Jacobian matrix

The simplest method for computing the Jacobian matrix of the mapping from $\boldsymbol{\theta}$ to $\boldsymbol{m}_T$ is by numerical differentiation. The problem with this approach is that numerical derivatives tend to be inaccurate for highly non-linear functions. In the present context this may lead to wrong conclusions concerning the rank of the Jacobian matrix and the identifiability of the parameters in the model. For this reason, Iskrev (2010b) applied analytical derivatives, employing implicit derivation. As shown in Iskrev (2010b), it helps to consider the mapping from $\boldsymbol{\theta}$ to $\boldsymbol{m}_T$ as comprising two steps: (1) a transformation from $\boldsymbol{\theta}$ to $\boldsymbol{\tau}$; (2) a transformation from $\boldsymbol{\tau}$ to $\boldsymbol{m}_T$. Thus, the Jacobian matrix can be expressed as

$$J(T) = \frac{\partial \boldsymbol{m}_T}{\partial \boldsymbol{\tau}'} \frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\theta}'} \tag{9}$$

The derivation of the first term on the right-hand side is straightforward since the function mapping $\boldsymbol{\tau}$ into $\boldsymbol{m}_T$ is available explicitly (see the definition of $\boldsymbol{\tau}$ and equations (5)-(7)); thus the Jacobian matrix $J_1(T) := \frac{\partial \boldsymbol{m}_T}{\partial \boldsymbol{\tau}'}$ may be obtained by direct differentiation.

The elements of the second term $J_2(T) := \frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\theta}'}$, the Jacobian of the transformation from $\boldsymbol{\theta}$ to $\boldsymbol{\tau}$, can be divided into three groups corresponding to the three blocks of $\boldsymbol{\tau}$: $\boldsymbol{\tau}_z$, $\boldsymbol{\tau}_A$ and $\boldsymbol{\tau}_\Omega$. In order to properly compute the derivatives of $\boldsymbol{\tau}_A$ and $\boldsymbol{\tau}_\Omega$, the structural form (2) has to be re-written explicitly accounting for the dependency to $\hat{\boldsymbol{z}}^*$:

$$\boldsymbol{\Gamma}_0(\boldsymbol{\theta}, \hat{\boldsymbol{z}}^*)\boldsymbol{z}_t = \boldsymbol{\Gamma}_1(\boldsymbol{\theta}, \hat{\boldsymbol{z}}^*)\,\mathrm{E}_t\,\boldsymbol{z}_{t+1} + \boldsymbol{\Gamma}_2(\boldsymbol{\theta}, \hat{\boldsymbol{z}}^*)\boldsymbol{z}_{t-1} + \boldsymbol{\Gamma}_3(\boldsymbol{\theta}, \hat{\boldsymbol{z}}^*)\boldsymbol{u}_t \tag{10}$$

Taking advantage of the DYNARE symbolic pre-processor and after some implicit derivation steps, discussed in Ratto and Iskrev (2010a,b), all the derivatives information of the structural form (10) can be obtained.

The derivatives of $\boldsymbol{\tau}_A$ and $\boldsymbol{\tau}_\Omega$ can be obtained from the derivatives of $\mathrm{vec}(\boldsymbol{A})$ and $\mathrm{vech}(\boldsymbol{\Omega})$, by removing the zeros corresponding to the constant elements of $\boldsymbol{A}$ and $\boldsymbol{\Omega}$. In Iskrev (2010b) the derivative of $\mathrm{vec}(\boldsymbol{A})$ is computed using the implicit function theorem. Such a derivation requires the use of Kronecker products, implying a dramatic growth in memory allocation requirements and in computational time as the size of the model increases. The typical size of matrices to be handled in Iskrev (2010b) is of $m^2 \times m^2$, which grows very rapidly with m. Here we apply the alternative method discussed in Ratto and Iskrev (2010a,b), which allows to reduce both memory requirements and the computational time, where it is shown that the derivation problem can be cast in the form of a generalized Sylvester equation and can be solved using available algebraic solvers. In practice the single big algebraic problem of dimension $m^2 \times m^2$ of Iskrev (2010b) is replaced by a set of k problems of dimension $m \times m$. This allows a significant gain in computational time of the Sylvester equation solution with respect to the approach in Iskrev (2010b), making the evaluation of analytic derivatives affordable also for DSGE models of medium/large scale, enabling to perform detailed identification analysis for such kind of models.

4 Analyzing local identification of DSGE models: DYNARE implementation

4.1 Identification analysis procedure

The local identifiability of the parameter set $\boldsymbol{\theta}$ is established using the necessary and sufficient conditions discussed by Iskrev (2010b):

- Finding that matrix J_2 is rank deficient at $\boldsymbol{\theta}$ implies that this particular point in $\boldsymbol{\Theta}$ is unidentifiable in the model.

- Finding that J_2 has full rank but $J(T)$ does not, means that $\boldsymbol{\theta}$ cannot be identified given the set of observed variables and the number of observations.

- On the other hand, if $\boldsymbol{\theta}$ is identified at all, it would typically suffice to check the rank condition for a small number of moments, since $J(q)$ is likely to have full rank for q much smaller than T. According to Theorem 1 this is sufficient for identification; moreover, the smaller matrix may be much easier to evaluate than the Jacobian matrix for all available moments. A good candidate to try first is the smallest q for which the order condition is satisfied, and then increase the number of moments if the rank condition fails;

- the present DYNARE implementation also analyzes the derivatives of the LRE form of the model ($J_{\boldsymbol{\Gamma}} = \frac{\partial \gamma}{\partial \boldsymbol{\theta}'}$), to check for 'trivial' non-identification problem, like two parameters always entering as a product in Γ_i matrices;

- whenever some of the matrices J_2, $J(T)$ or $J_{\boldsymbol{\Gamma}}$ is rank deficient, the code tries to diagnose the subset of parameters responsible for the rank deficiency: this is done by doing a number of tests

  1. if there are columns of zeros in the $J(\cdot)$ matrix, the associated parameter is printed on the MATLAB command window;
  2. compute pairwise- and multi-correlation coefficients for each column of the $J(\cdot)$ matrix: if there are parameters with correlation coefficients equal to unity, these are printed on the MATLAB command window;
  3. take the Singular Values Decomposition (SVD) of $J(\cdot)$ and track the eigenvectors associated to the zero singular values.

4.2 Identification strength

The previous conditions are related to whether of not columns of $J(T)$ or J_2 are linearly dependent. Another typical avenue in DSGE models is weak identification: in this case it is interesting to rank model parameters in terms of strength of identification. A measure of identification strength is introduced, following the work of Iskrev (2010a) and Andrle (2010). This can be based on either using the asymptotic information matrix or using simulated moments uncertainty mapped onto the deep parameters.

Asymptotic Information Matrix. Given a sample size T, the Fischer information matrix $\mathcal{I}_T(\boldsymbol{\theta})$ is computed as discussed in Iskrev (2010a).

Simulated moments. The uncertainty of simulated moments is evaluated, by performing stochastic simulations for T periods and computing sample moments of observed variables; this is repeated for N_r replicas, giving a sample of dimension N_r of simulated moments; from this the covariance matrix $\Sigma(\boldsymbol{m}_T)$ of (first and second) simulated moments is obtained. A 'moment information matrix' can be defined as $\mathcal{I}_T(\boldsymbol{\theta}|\boldsymbol{m}_T) = J_2' \cdot \Sigma(\boldsymbol{m}_T) \cdot J_2$;

The procedure implemented in DYNARE takes the following steps:

1. given the information matrix (either the true one or the one based on simulated moments), the strength of identification for parameter θ_i is defined as

$$s_i = \sqrt{\theta_i^2/(\mathcal{I}_T(\boldsymbol{\theta})^{-1})_{(i,i)}} \tag{11}$$

   which is a sort of a priori 't-test' for θ_i;

2. as discussed in Iskrev (2010a), this measure is made of two components: the ‘sensitivity’ and the ‘correlation’, i.e. weak identification may be due to the fact that moments do not change with θ_i or or that other parameters can compensate linearly the effect of θ_i; the sensitivity component is defined as

$$\Delta_i = \sqrt{\theta_i^2 \cdot \mathcal{I}_T(\boldsymbol{\theta})_{(i,i)}} \tag{12}$$

3. as an alternative option to ‘normalize’ the identification strength measures, in place of using θ_i, the toolbox also uses the value for its prior standard deviation $\sigma(\theta_i)$, i.e.:

$$\begin{aligned} s_i^{prior} &= \sigma(\theta_i)/\sqrt{(\mathcal{I}_T(\boldsymbol{\theta})^{-1})_{(i,i)}} \\ \Delta_i^{prior} &= \sigma(\theta_i) \cdot \sqrt{\mathcal{I}_T(\boldsymbol{\theta})_{(i,i)}} \end{aligned}$$

   this normalization for the identification strength weights the information derived from the likelihood using the prior uncertainty; this is specially useful to distinguish those cases where (11) is singular simply because $\theta_i \approx 0$ (e.g. for parameters whose prior distribution is centered in zero) and therefore θ_i should NOT be flagged as non- (or weakly) identified.

The default of the identification toolbox is to show, after the check of rank conditions, the plots of the strength of identification and of the sensitivity component for all estimated parameters.

4.3 Analyzing identification patterns

Identification patterns are essentially tracked in two ways:

1. As suggested by Andrle (2010), the identification patterns are shown by taking the singular value decomposition of $\mathcal{I}_T(\boldsymbol{\theta})$ or of the $J(q)$ matrix and displaying the eigenvectors corresponding to the smallest (or highest) singular values;

2. As discussed in Iskrev (2010b), it is also interesting to check which group of one, two or more parameters is most capable to mimic (replace) the effect of each parameter; in other words, a brute force search is done for each column of $J(q)_{(j)}$ to detect the group of columns $J(q)_{(I \not\ni j)}$, indexed by I, that has the highest explanatory power for $J(q)_{(j)}$ by a linear regression.

In this analysis, scaling issues in the Jacobian can matter in interpreting results. In medium-large scale DSGE models there can be as many as thousands entries in $J(q)$ and J_2 matrices (as well as in corresponding $\boldsymbol{m}_q$ and $\boldsymbol{\tau}$ matrices). Each row of $J(q)$ and J_2 correspond to a specific moment or $\boldsymbol{\tau}$ element and there can be differences by orders of magnitude between the values in different rows. In this case, the analysis would be dominated by the few rows with large elements, while it would be unaffected by all remaining elements. This can imply loss of ‘resolution’. Iskrev (2010b) used the elasticities, so that the (j,i) element of the Jacobian is $\frac{\partial m_j}{\partial \theta_i}\frac{\theta_i}{m_j}$. This give the percentage change in the moment for 1% change in the parameter value. Here we re-scale each row of $J(q)$ and J_2 by its largest element in absolute value. In other words, assuming J_2 made of the two rows:

$$\begin{pmatrix} 0.1 & -0.5 & 2.5 \\ -900 & 500 & 200 \end{pmatrix}$$

multi-collinearity analysis will be performed on the scaled matrix:

$$\begin{pmatrix} 0.04 & -0.2 & 1 \\ -1 & 0.5556 & 0.2222 \end{pmatrix}$$

The effect of this scaling is that the order of magnitude of derivatives of any moment (or any $\boldsymbol{\tau}$ element) is the same. In other words, this grossly corresponds to an assumption that the model is equally informative about moments, thus implying equal weights across different rows of the Jacobian matrix.

4.4 The optional Monte Carlo implementation

This optional implementation is based on Monte Carlo exploration of the space $\boldsymbol{\Theta}$ of model parameters. In particular, a sample from $\boldsymbol{\Theta}$ is made of many randomly drawn points from $\boldsymbol{\Theta}'$, where $\boldsymbol{\Theta} \in \boldsymbol{\Theta}'$ discarding values of $\boldsymbol{\theta}$ that do not imply a unique solution. The set $\boldsymbol{\Theta}'$ contains all values of $\boldsymbol{\theta}$ that are theoretically plausible, and may be constructed by specifying a lower and an upper bound for each element of $\boldsymbol{\theta}$ or by directly specifying prior distributions. After specifying a distribution for $\boldsymbol{\theta}$ with support on $\boldsymbol{\Theta}'$, one can obtain points from $\boldsymbol{\Theta}$ by drawing from $\boldsymbol{\Theta}'$ and removing draws for which the model is either indetermined or does not have a solution. Conditions for existence and uniqueness are automatically checked by DYNARE. The checks for local identification are then performed for each parameter set in turn, obtaining a Monte Carlo sample of identification features for the model under analysis.

4.4.1 Sensitivity measures

The identification analysis is linked to the idea of sensitivity analysis, where the analyst would like to track how model 'outputs' (in this case $\boldsymbol{\tau}$ or $\boldsymbol{m}_T$ or the likelihood itself) are affected by changes in $\boldsymbol{\theta}$. In a context where plausible parameter ranges are wide and the associated model features can significantly change in such a wide domain, global sensitivity analysis (GSA) provides useful diagnostic tools to measures that importance of the various parameters. The GSA, the reference measures are the variance-based sensitivity indices (Ratto, 2008).

Let Y be a generic 'output' of the model: in this case $\boldsymbol{\tau}$ or $\boldsymbol{m}_T$ elements. Given the model structure, the outcome of Y depends on the values of $\boldsymbol{\theta}$. This dependence can be expressed as a non linear relationship $Y = f(\theta_1, \dots, \theta_k)$, whose analytic form is unknown to the analyst. Such a relationship can be represented by mean of ANOVA (or ANOVA-HDMR) decompositions, where ANOVA stands for 'analysis of variance', the ensemble the statistical tools for in which the observed variance in a particular variable Y is partitioned into components attributable to different sources of variation and HDMR stands for High-Dimensional Model representation (Sobol', 1990; Gu, 2002). In ANOVA, the function f is decomposed by means of a *finite* decomposition into terms of increasing dimensionality:

$$f(\theta_1, \theta_2, \ldots, \theta_k) = f_0 + \sum_i f_i + \sum_i \sum_{j>i} f_{ij} + \ldots + f_{12\ldots k} \tag{13}$$

where each term is a function only of the parameters in its indexes, i.e. $f_i = f(\theta_i)$, $f_{ij} = f(\theta_i, \theta_j)$ and so on. The various terms are expressed as:

$$\begin{aligned} f_0 &= E(Y) \\ f_i(\theta_i) &= E(Y|\theta_i) - f_0 \\ f_{ij} &= E(Y|\theta_i, \theta_j) - f(\theta_i) - f(\theta_j) - f_0 \end{aligned} \tag{14}$$

If $\boldsymbol{\theta}$ values are sampled independently, all the terms of the decomposition are orthogonal and the decomposition (13) is unique. The $f_i's$ are called the main effects, $f_{ij}'s$ are the second order interaction effects, and so on. Each term of the decomposition tells the analyst how much Y moves around its mean level f_0 as a function of each of the input factors or group of them. An intuitive way to derive a scalar measure of the importance of θ_i on the variation of Y, is to take the variance $V(f_i) = V_i$ and compare it to the total variance $V(Y)$. Normalizing the partial variances with the unconditional variance $V = V(Y)$, the sensitivity indices are obtained: $S_i = V_i/V$ (main effects) , $S_{ij} = V_{ij}/V$ (second order pure interaction effects), etc. Variance based sensitivity indices provide the percentage of output model variance which is explained by each input.

Various approaches for estimating variance based sensitivity indices are available. These may require either the use of specific sampling strategies to directly estimate sensitivity indices (Sobol', 1990; Saltelli et al., 2010) or by estimating some 'metamodel' using a Monte Carlo sample of the mapping f and deriving sensitivity indices from the metamodel (Oakley and O'Hagan, 2004; Storlie and Helton, 2007; Ratto and Pagano, 2010). In the present context, running such kind of sophisticated estimators may imply either too many runs of the model solutions than affordable (specially for medium-large scale models) or to repeat computer intensive metamodelling estimations for the large number of 'outputs' under consideration (in particular the $\boldsymbol{m}_T$ elements).

For the purpose of the identification toolbox it is sufficient to take an approximation of the full variance based approach which allows to synthesize effectively the information coming from the Jacobian $J(q)$. The Jacobian (i.e. local derivatives) provide per se 'local' sensitivity measures and in fact this has been historically the first approach to sensitivity analysis. It is also well known that derivative based sensitivity poses some problems of interpretation, due to scaling issues and to the fact that derivatives may not be directly comparable to each others and also due to the fact they only reflect the *local* behavior of the mapping f possibly leading to erroneous results. So, first, the derivatives are usually normalized if they ought to be used for sensitivity purposes: Iskrev (2010b) uses elasticities, implying that the new sensitivity measure gives the percentage change in the moment for 1% change in the parameter value. This somehow improves the interpretability of the measure, but still carries in some issues relative to the specific point where the derivatives are taken: for example what if derivatives are taken at a location where $\theta_j = 0$ or $Y = 0$? Moreover, no information about the degree of uncertainty about each parameters comes in in such a sensitivity measure. Therefore, in order to link derivatives to the reference variance-based measure, it is most advisable to proceed as follows.

Derivatives provide a local approximation of the mapping f around the specific sample point $\boldsymbol{\theta}_{(j)}$ where derivatives are taken:

$$f(\boldsymbol{\theta}) \approx Y(\boldsymbol{\theta}_{(j)}) + J(\boldsymbol{\theta}_{(j)}) \cdot (\boldsymbol{\theta} - \boldsymbol{\theta}_{(j)}) \tag{15}$$

and the variance based sensitivity indices for this linearized mapping simply read (assuming independent θ's):

$$S_i \approx \left(\frac{\partial Y}{\partial \theta_i} \right)^2_{(j)} \cdot V(\theta_i)/V(Y) \tag{16}$$

where $V(\theta_i)$ is the prior variance of each parameter and $V(Y)$ is the variance of the model 'output' (i.e. $\boldsymbol{m}_T$ elements). This approach to scaling derivatives links the local sensitivity to two key global elements: the 'uncertainty' about the parameters and the 'uncertainty' about the outputs. This measure provides the portion of the linearized mapping f that is explained by each parameter. This implies that, if Y has the same derivative w.r.t. two parameters, the most 'important' parameter will be the one with the highest uncertainty (i.e. variance), since it will be responsible for higher changes in the output value. With the Monte Carlo exploration of the parameter space, one will have the availability of these sensitivity measures for each element of the sample: taking the average of these measures provides a synthetic measure of sensitivity measure that averages the local effect over the entire prior space.

To summarize, the program performs the following steps to compute sensitivity measures of theoretical moments with respect to model parameters:

1. given the MC sample of $\boldsymbol{\theta}$, the sample for $\boldsymbol{m}_T$ is stored, allowing to compute the sample variance for each of their elements $V(m_j)$;

2. all computed derivatives are normalized, so that the (j,i) element of the Jacobian is $\frac{\partial m_j}{\partial \theta_i}\frac{std(\theta_i)}{std(m_j)}$;

3. for each MC sample point we take the norm of the normalized Jacobian, obtaining one single aggregate sensitivity measure for each parameter and each sample point;

4. this aggregate measure is finally averaged over the MC sample, providing the final sensitivity measure reported by the identification toolbox.

In the case of the point estimate (the default case), no direct information about the variances $V(m_j)$ is available. Hence, given the covariance $\Sigma_{\boldsymbol{\theta}}$ of the parameters, the variance of the moments is obtained as $J(q) \cdot \Sigma_{\boldsymbol{\theta}} \cdot J(q)'$.

5 DYNARE syntax

A new syntax is made available for DYNARE users. The simple keyword `identification(<options>=<values>);` triggers the point identification analysis, performed at the prior mean. Prior definitions and the list of observed values are needed, using the standard DYNARE syntax for setting-up an estimation.

Current options are as follows (all options will be discussed in detail in the Examples section 6):

- parameter_set = prior_mode | prior_mean | posterior_mode | posterior_mean | posterior_median. Specify the parameter set to use. Default: prior_mean.
- prior_mc = INTEGER sets the number of Monte Carlo draws (default $= 1$); prior_mc=1 triggers the default point identification analysis; prior_mc>1 triggers the Monte Carlo mode;
- prior_range = INTEGER triggers uniform sample within the range implied by the prior specifications (when prior_mc>1). Default: 0
- load_ident_files = 0, triggers a new analysis, while load_ident_files = 1, loads and displays a previously performed analysis (default $= 0$);
- ar = <integer> (default $= 3$), triggers the value for q in computing $J(q)$;
- useautocorr: this option triggers $J(q)$ in the form of auto-covariances and cross-covariances (useautocorr = 0), or in the form of auto-correlations and cross-correlations (useautocorr = 1). The latter form normalizes all $\boldsymbol{m}_q$ entries in $[-1, 1]$ (default $= 0$).
- advanced = INTEGER triggers standard or advanced identification analysis (default $= 0$). In standard identification analysis the rank condition is checked and, if it fails, indicates the parameter(s) that are not identified; the strength of identification plus the sensitivity component is also provided. The advanced identification analysis shows

a more detailed analysis, comprised of an analysis for the linearized rational expectation model as well as the associated reduced form solution. Moreover, a detailed information about identification patterns is also provided, as discussed in Section 4.3. In the collinearity analysis by brute force search of the groups of parameters best reproducing the behavior of each single parameter (Iskrev, 2010b), the maximum dimension of the group searched is triggered by `max_dim_cova_group`. In the Monte Carlo mode, the advanced options triggers a number of additional diagnostics: (i) analysis of the condition number of the Jacobians $J(q)$, J_2 and J_Γ and detection of the parameters that mostly drive large condition numbers (i.e. weaker identification); (ii) analysis of the identification patters across the Monte Carlo sample; (iii) detailed point-estimate (identification strength and collinearity analysis) of the parameters set having the smallest/largest condition number; (iv) when some singularity (rank condition failure) is detected for some elements of the Monte Carlo sample, detailed point-estimates are performed for such critical points.

- `max_dim_cova_group = INTEGER` In the brute force search (performed when `advanced=1`) this option sets the maximum dimension of groups of parameters that best reproduce the behavior of each single model parameter. Default: 2

- `periods = INTEGER` triggers the length of the stochastic simulation to compute the analytic Hessian. Default: 300

- `periods = INTEGER` When the analytic Hessian is not available (i.e.

with missing values or diffuse Kalman filter or univariate Kalman filter), this triggers the length of stochastic simulation to compute Simulated Moments Uncertainty. Default: 300

- `replic = INTEGER` When the analytic Hessian is not available, this triggers the number of replicas to compute Simulated Moments Uncertainty. Default: 100.
- `gsa_sample_file = INTEGER` If equal to 0, do not use sample file. If equal to 1, triggers GSA prior sample. If equal to 2, triggers GSA Monte-Carlo sample (i.e. loads a sample corresponding to `pprior=0` and `ppost=0` in the `dynare_sensitivity` options). Default: 0
- `gsa_sample_file = FILENAME` Uses the provided path to a specific user defined sample file. Default: 0

5.1 Notes

The analytic derivation procedure takes advantage of the symbolic preprocessor of DYNARE. The latter interprets and implements the model definitions as expressed in the DYNARE file. However, it will not reflect all parameter definitions which may be hidden in the `<>_steadystate.m` file. The latter, in some cases, may include calibration constraints for model parameters. Those conditions are totally transparent to the preprocessor, and the identification analysis, which is based on analytical derivatives, will not reflect entirely the true mapping between structural parameters and the model solution. In order to obtain a proper operation of the analytic derivation engine, the `#`

syntax should be used in the `model` block of the DYNARE file, providing the calibration and steady state constraints in a way that can be interpreted by the pre-processor. Of course, this problem does not affect the DYNARE estimation engine, which uses numerical derivatives and therefore reflects the entire set of user definitions, including the steady-state file.

6 Examples

6.1 Kim (2003)

In this paper, Kim demonstrated a functional equivalence between two types of adjustment cost specifications, coexisting in macroeconomic models with investment: intertemporal adjustment costs which involve a nonlinear substitution between capital and investment in capital accumulation, and multisectoral costs which are captured by a nonlinear transformation between consumption and investment. We reproduce results of Kim (2003), worked out analytically, applying the DYNARE procedure on the non-linear form of the model. The representative agent maximizes

$$\sum_{t=0}^{\infty} \beta^t \log C_t \tag{17}$$

subject to a national income identity and a capital accumulation equation:

$$(1-s)\Big(\frac{C_t}{1-s}\Big)^{1+\theta} + s\Big(\frac{I_t}{s}\Big)^{1+\theta} = (A_t K_t^{\alpha})^{1+\theta} \tag{18}$$

$$K_{t+1} = \left[\delta\left(\frac{I_t}{\delta}\right)^{1-\phi} + (1-\delta)K_t^{1-\phi}\right]^{\frac{1}{1-\phi}} \tag{19}$$

where $s = \frac{\beta\delta\alpha}{\Delta}$, $\Delta = 1 - \beta + \beta\delta$, $\phi(\geq 0)$ is the inverse of the elasticity of substitution between I_t and K_t and $\theta(\geq 0)$ is the inverse of the elasticity of transformation between consumption and investment. Parameter ϕ represents the size of intertemporal adjustment costs while θ is called the multisectoral adjustment cost parameter. Kim shows that in the linearized form of the model, the two adjustment cost parameter only enter through an 'overall' adjustment cost parameter $\Phi = \frac{\phi+\theta}{1+\theta}$, thus implying that they cannot be identified separately.

Here we assume that the Kim model is not analytically worked out to highlight this problem of identification. Instead, the analyst feeds the non-linear model (constraints and Euler equation) to DYNARE (also note that the adjustment costs are defined in such a way that the steady state is not affected by them).

6.1.1 The default identification procedure

First we run the default identification procedure: point estimate at the prior mean, standard diagnostics (see Syntax 1). In the list of estimated parameters we have added α (`alph`) to illustrate the behavior of the program with other parameters that are identifiable. Observable variables are C_t and I_t.

```
estimated_params;
alph ,uniform_pdf,0.6,0.04,0.5,0.7;
phi ,uniform_pdf,0.5,0.2,0,10;
theta ,uniform_pdf,0.3,0.1,0,10;
end;

varobs c i;
identification;
```

Syntax 1. Standard identification syntax for the Kim (2003) example.

The printed output in the command window is shown in Box 1 while the identification strength is shown in Figure 1.

```
==== Identification analysis ====

Testing prior mean
Evaluting simulated moment uncertainty ... please wait
Doing 100 replicas of length 300 periods.
Simulated moment uncertainty ... done!

WARNING !!!
The rank of H (model) is deficient!


    [theta,phi] are PAIRWISE collinear (with tol = 1.e-10) !


WARNING !!!
The rank of J (moments) is deficient!


    [theta,phi] are PAIRWISE collinear (with tol = 1.e-10) !


==== Identification analysis completed ====
```

Box 1. Standard printed output for the Kim (2003) example.

Comments on DYNARE outputs.

The printed output first tells that the point estimate is done for the prior mean. Then, the code proceeds with the identification strength: since the model has two observables and only one shock (technology) the Kalman filter is rank deficient and the analytic asymptotic Hessian cannot be computed. Hence, the alternate procedure based on simulated moments uncertainty is applied. The rank condition fails for both for J_2 (called `H` in the DYNARE output) and $J(q)$, implying that sufficient and necessary conditions for local identification are not fulfilled by this model. Moreover, perfect collinearity among columns for θ and ϕ is detected both for J_2 and $J(q)$, thus allowing to highlight which combination of parameters is responsible for the lack of identification. The plot of the strength of identification (Figure 1) tells that α is identified while identification strength for θ and ϕ is null (in practice no bar is shown in the log-scale plot). It is interesting to look at the sensitivity component for the three parameters, which tells that all parameters have individually an effect on the model behavior, implying that it is indeed the correlation component that drives the rank-deficiency. The strength of identification plot is saved in the `identification` subfolder with the name `MODEL_ident_strength`. So, the diagnostic tests implemented in DYNARE perfectly reveal the identification problem demonstrated analytically by Kim. This result shows that the procedures implemented in DYNARE can help the analyst in detecting identification problems in all typical cases where such problems cannot easily worked out analytically.

It seems also interesting to show here the effect of the number of states fed to DYNARE on the results of the identification analysis. For simplicity of coding, Lagrange multipliers may be explicitly included in the model equations. In this case, one would have an additional equation for the La-

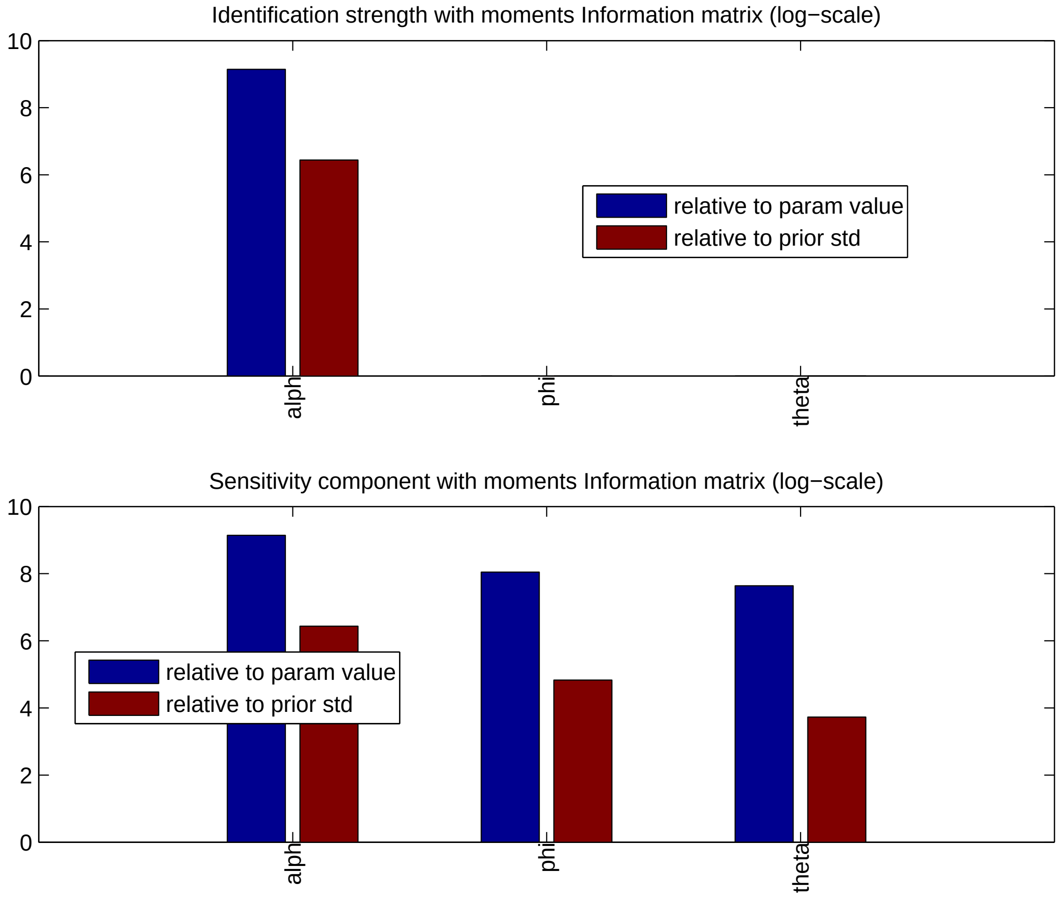


Figure 1: DYNARE identification strength of the Kim model.

grange multiplier $\lambda_t = \frac{(1-s)^\theta}{(1+\theta)C_t^{(1+\theta)}}$, with λ_t entering the Euler equation. The DYNARE output is shown in Box 2. Under this kind of implementation, and still assuming that only C_t and I_t can be observed, the test for $J(q)$ still the rank deficiency, thus confirming the identification problem. On the other hand, due to the specific effect of θ on λ_t, our identification tests would tell that θ and ϕ are separably identified in the model, provided that all states are observed. This exemplifies the nature of the *necessary* condition stated in Corollary 1.

```
==== Identification analysis ====

Testing prior mean
Evaluting simulated moment uncertainty ... please wait
Doing 100 replicas of length 300 periods.
Simulated moment uncertainty ... done!

All parameters are identified in the model (rank of H).


WARNING !!!
The rank of J (moments) is deficient!


    [theta,phi] are PAIRWISE collinear (with tol = 1.e-10) !


==== Identification analysis completed ====
```

Box 2. Standard printed output for the Kim (2003) example when λ_t is included in the variables of the model.

6.1.2 The advanced identification procedure

The advanced identification checks are triggered by the DYNARE commands in Syntax 2. Note that, to highlight further features of the identification toolbox, we also added a ‘dummy’ parameter in the estimate parameter list called `dumpy`, which does not enter the model.

```
estimated_params;
alph ,uniform_pdf,0.6,0.04,0.5,0.7;
phi ,uniform_pdf,0.5,0.2,0,10;
theta ,uniform_pdf,0.3,0.1,0,10;
dumpy ,uniform_pdf,0.5,0.2,0,10;
end;

varobs c i;
identification(advanced=1,max_dim_cova_group=3);
```

Syntax 2. Example of advanced identification syntax for the Kim (2003) example.

We can see in Box 3 that first the standard output is printed, showing that not only θ and ϕ are not separably identified, but also that `dumpy` is not identified (analytic derivatives produce a column of zeros for this parameter). Then, the program prompts the user to proceed for advanced outputs and diagnostics, shown in Box 4.

The advanced diagnostics prints the result of the brute force search for the groups of parameters whose columns of $J(q)$ best explain each column of $J(q)$: the name of the parameters and the cosine between the $J(q)$ column of each parameter and of the regressors is displayed. When there is no explanatory power (like for `dumpy`), nothing is displayed, and the cosine is null. Moreover, when there is no improvement in the explanatory power by increasing the dimension of the candidate group of parameters, no additional regressor is added (see printed results for 3 parameters) and the cosine does not improve with respect to the group of smaller dimensionality.

The advanced diagnostics also produces three kinds of additional plots: the sensitivity plot, the collinearity patterns, the identification patterns à la Andrle (2010).

```
==== Identification analysis ====

Testing prior mean
Evaluting simulated moment uncertainty ... please wait
Doing 100 replicas of length 300 periods.
Simulated moment uncertainty ... done!

WARNING !!!
The rank of H (model) is deficient!

    dumpy is not identified in the model!
    [dJ/d(dumpy)=0 for all tau elements in the model solution!]

    [theta,phi] are PAIRWISE collinear (with tol = 1.e-10) !


WARNING !!!
The rank of J (moments) is deficient!

    dumpy is not identified by J moments!
    [dJ/d(dumpy)=0 for all J moments!]

    [theta,phi] are PAIRWISE collinear (with tol = 1.e-10) !


Press ENTER to display advanced diagnostics
```

Box 3. Standard printed output for the Kim (2003) example when `dumpy` is included in the prior list for estimation.

```
Collinearity patterns with 1 parameter(s)
Parameter       [ Expl. params   ]       cosn
alph            [ theta          ]      0.800
phi             [ theta          ]      1.000
theta           [ phi            ]      1.000
dumpy           [ --             ]      0.000

Collinearity patterns with 2 parameter(s)
Parameter       [ Expl. params                   ]       cosn
alph            [ phi             theta          ]      0.975
phi             [ theta           --             ]      1.000
theta           [ phi             --             ]      1.000
dumpy           [ --              --             ]      0.000

Collinearity patterns with 3 parameter(s)
Parameter       [ Expl. params                                   ]       cosn
alph            [ phi             theta           --             ]      0.975
phi             [ theta           --              --             ]      1.000
theta           [ phi             --              --             ]      1.000
dumpy           [ --              --              --             ]      0.000

==== Identification analysis completed ====
```

Box 4. Advanced printed output for the Kim (2003) example.

Sensitivity plot. This plot shows the sensitivity measures computed as described in Section 4.4.1, i.e. the norm of the columns of the jacobian matrices for the moments ($J_(q)$), the model (J_2), the LRE model (J_Γ) normalized by the relative uncertainties in model parameters and output values. An example of this is in Figure 2. This plot is saved in the `identification` subfolder with the name `MODEL_sensitivity`.

Collinearity patterns. These plots synthesize the results of the brute force search à la Iskrev (2010b) already printed in Box 4. Specially for large models and large number of parameters, these plots allow a quick search for the critical collinearities (i.e. dark red spots). Examples of these are shown in Figure 3: the critical combination between θ and ϕ is clearly indicated. These plots are saved in the `identification` subfolder with the names `MODEL_ident_collinearity_*`.

Identification patterns. As suggested by Andrle (2010), identification patterns can be analyzed by taking the singular value decomposition of the Information Matrix or of the Jacobian. The plots produced by the identification toolbox show in bar form the eigenvectors relative to the smallest and largest singular values: the former indicate the parameters that are mostly affected by weak or lack of identification, while the latter indicate parameters that can be best identified. An example of this for the Kim model is shown in Figure 4. We can see in this figure that there are two null singular values, one associated to `dumpy`, the other associated to the couple [θ, ϕ]. Moreover, the positive singular values are associated with α and [θ, ϕ], indicating that only the

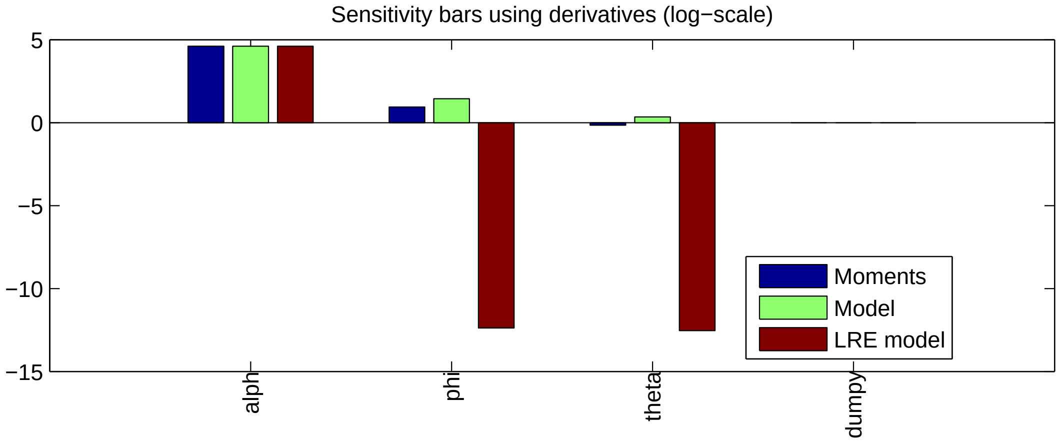


Figure 2: Sensitivity plot for the Kim model.

linear combination of the latter couple of parameters can be identified. These plots are saved in the `identification` subfolder with the names `MODEL_ident_pattern_*`.

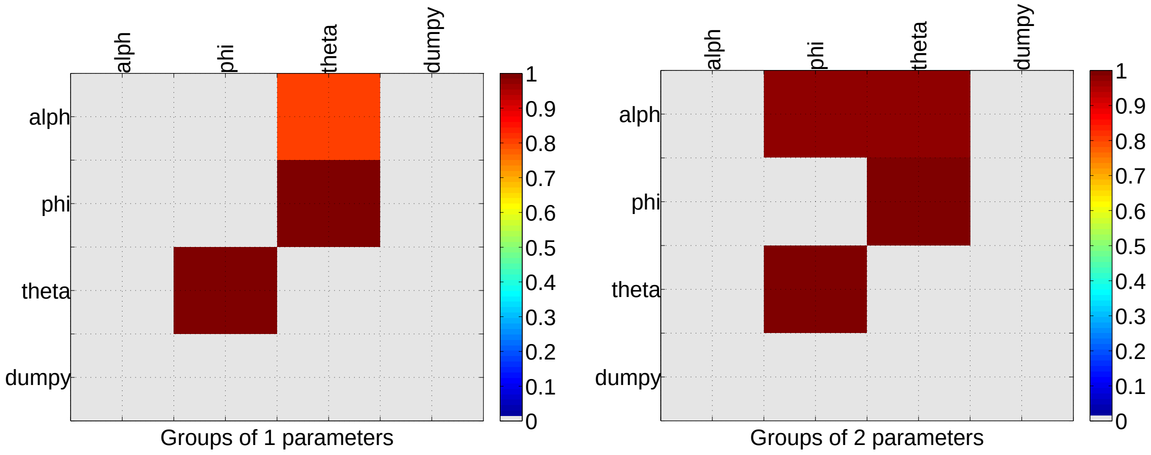


Figure 3: Plot of collinearity patterns for the Kim model.

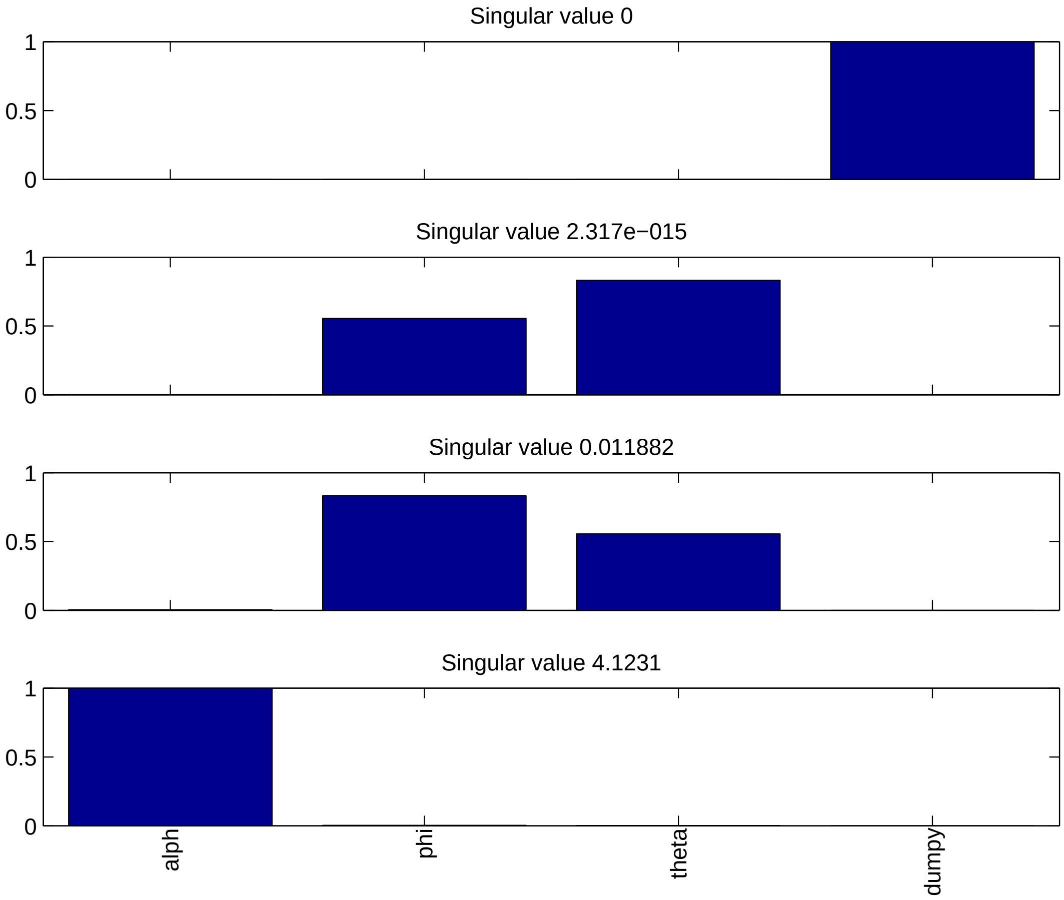


Figure 4: Plot of identification patterns for the Kim model.

6.2 An and Schorfheide (2007)

The model An and Schorfheide (2007), linearized in log-deviations from steady state, reads:

$$y_t = \mathbb{E}_t[y_{t+1}] + g_t - \mathbb{E}_t[g_{t+1}] - 1/\tau \cdot (R_t - \mathbb{E}_t[\pi_{t+1}] - \mathbb{E}_t[z_{t+1}]) \tag{20}$$

$$\pi_t = \beta \mathbb{E}_t[\pi_{t+1}] + \kappa(y_t - g_t) \tag{21}$$

$$R_t = \rho_R R_{t-1} + (1 - \rho_R)\psi_1 \pi_t + (1 - \rho_R)\psi_2(\Delta y_t - z_t) + \varepsilon_{R,t} \tag{22}$$

$$g_t = \rho_g g_{t-1} + \varepsilon_{g,t} \tag{23}$$

$$z_t = \rho_z z_{t-1} + \varepsilon_{z,t} \tag{24}$$

where y_t is GDP in efficiency units, π_t is inflation rate, R_t is interest rate, g_t is government consumption and z_t is change in technology. The model is completed with three observation equations for quarterly GDP growth rate (YGR_t), annualized quarterly inflation rates (INF_t) and annualized nominal interest rates (INT_t):

$$YGR_t = \gamma^Q + 100 * (y_t - y_{t-1} + z_t) \tag{25}$$

$$INFL_t = \pi^A + 400\pi_t \tag{26}$$

$$INT_t = \pi^A + r^A + 4\gamma^Q + 400R_t \tag{27}$$

where $\beta = \frac{1}{1+r^A/400}$.

For this example we will consider and discuss in detail the Monte Carlo option.

6.2.1 Standard Monte Carlo procedure

An example of DYNARE syntax for this is shown in Syntax 3: we perform a loop of 250 local identification analyzes at parameter values randomly chosen in the prior distributions.

```
estimated_params;
tau, gamma_pdf, 2, 0.5;
kap, beta_pdf, 0.2, 0.1;
psi1, gamma_pdf, 1.5, 0.25;
psi2, gamma_pdf, 0.5, 0.25;
rhoR, beta_pdf, 0.5, 0.2;
rhog, beta_pdf, 0.8, 0.1;
rhoz, beta_pdf, 0.66, 0.15;
rr_steady, gamma_pdf, 0.8, 0.5;
pi_steady, gamma_pdf, 0.8, 0.5;
gam_steady, gamma_pdf, 0.8, 0.5;
std_R, inv_gamma_pdf, 0.05, inf;
std_g, inv_gamma_pdf, 0.05, inf;
std_z, inv_gamma_pdf, 0.05, inf;
end;

varobs YGR INFL INT;

identification(prior_mc=250);
```

Syntax 3. Identification syntax for Monte Carlo identification analysis of the An and Schorfheide (2007) example.

The standard output printed on the command window by the program is reported in Box 5: the program execution first performs the point estimated at the prior mean, as in the default execution. Then, it loops over the 250 replicas. The result of the identification analysis indicates that all parameters are identified in the model and in the moments in the entire prior space. The graphical output is made of two plots: first the identification strength at the prior mean (Figure 5), second the Monte Carlo mean of the sensitivity measures as described in Section 4.4.1 (Figure 6). In this case, the number

```
==== Identification analysis ====

Testing prior mean

All parameters are identified in the model (rank of H).


All parameters are identified by J moments (rank of J)


Monte Carlo Testing

Testing MC sample

All parameters are identified in the model (rank of H).


All parameters are identified by J moments (rank of J)


==== Identification analysis completed ====
```

Box 5. Printed output for the An and Schorfheide (2007) example when the Monte Carlo option is triggered.

of shocks equals the number of observables, so the asymptotic Hessian of the likelihood function can be evaluated for the identification strength. In Figure 5 we can see that all parameters are identified: the model parameters on the x-axis are ranked in increasing order of strength of identification. In the plot for the sensitivity analysis component and of the Monte Carlo sensitivity, we can also see that all parameters have a non-negligible effect on the moments.

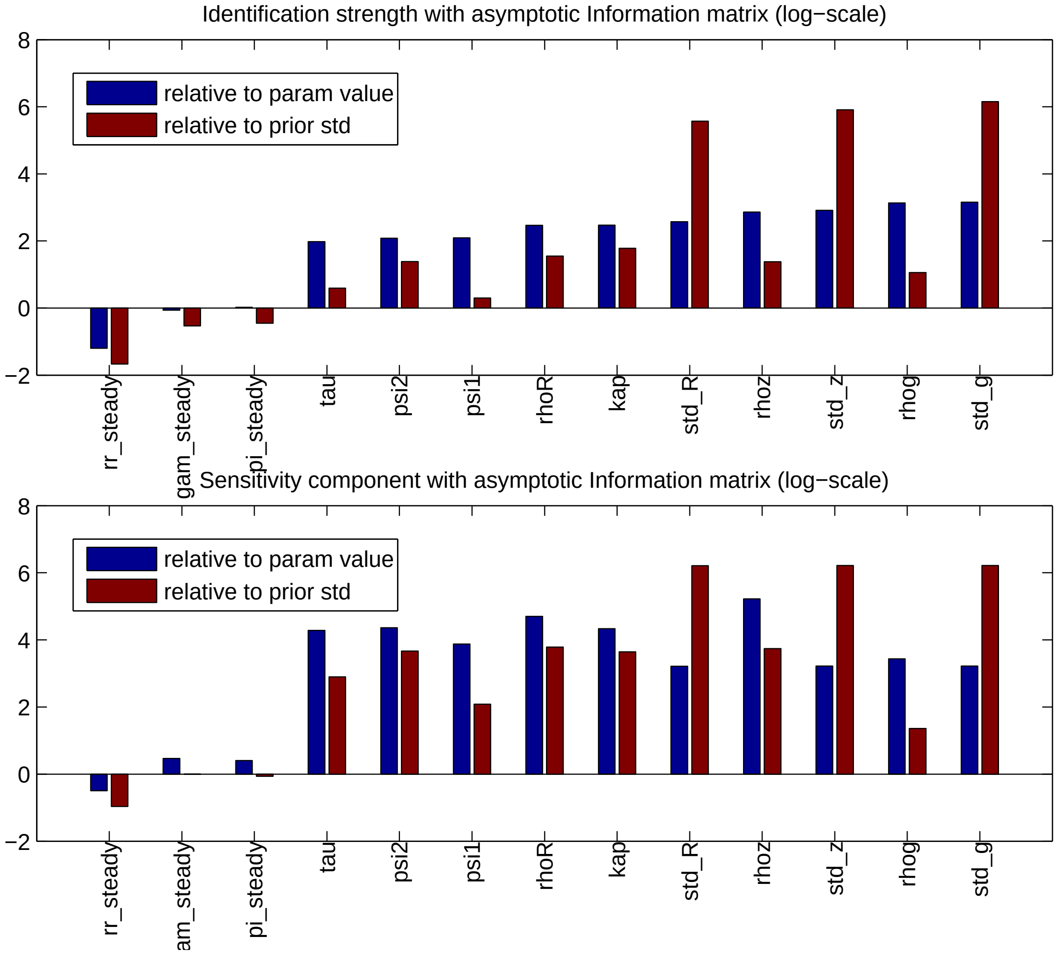


Figure 5: DYNARE identification strength at the prior mean for An and Schorfheide (2007).

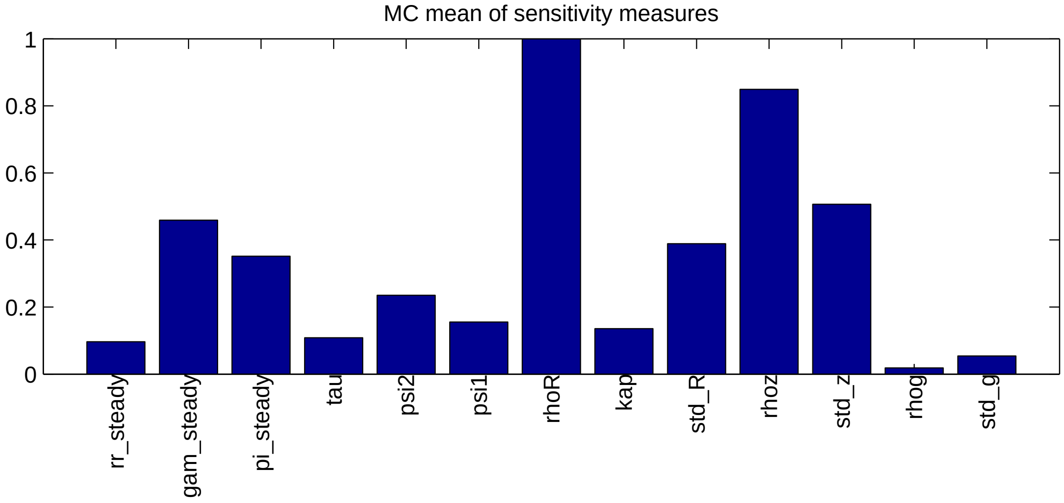


Figure 6: Monte Carlo sensitivity of the moments for An and Schorfheide (2007).

6.2.2 Advanced Monte Carlo procedure

An example of syntax for the advanced Monte Carlo identification procedure is shown in Syntax 4. For the point-estimate at the prior mean, with the advanced option, the brute force collinearity analysis à la Iskrev is displayed on the command window (Box 6) and in graphical form (Figure 7). The latter is specially informative: in particular we can note the correlation links that involve ψ_1, ψ_2 and the autoregressive parameter ρ_R. The identification patterns à la Andrle are also shown: when the number of estimated parameters is larger than four, the identification patterns are shown using two panels, one for the smallest singular values (Figure 8) and the second for the largest ones (Figure 9). The former spans the weakest identifiable patterns, the latter spans the strongest ones. Note that these identification patterns take the SVD of the Hessian. Later on we will see somewhat different patterns obtained taking the SVD of the Jacobian $J(q)$.

```
...

identification(advanced=1,max_dim_cova_group=3,prior_mc=250);

...
```

Syntax 4. Advanced Identification syntax for Monte Carlo identification analysis of the An and Schorfheide (2007) example.

```
Collinearity patterns with 1 parameter(s)
Parameter       [ Expl. params   ]       cosn
tau             [ psi1           ]      0.750
kap             [ tau            ]      0.744
psi1            [ tau            ]      0.750
psi2            [ rhoR           ]      0.691
rhoR            [ psi2           ]      0.691
rhog            [ std_g          ]      0.615
rhoz            [ psi2           ]      0.650
rr_steady       [ gam_steady     ]      0.707
pi_steady       [ rr_steady      ]      0.243
gam_steady      [ rr_steady      ]      0.707
std_R           [ psi1           ]      0.726
std_g           [ psi1           ]      0.626
std_z           [ rhoz           ]      0.513
Collinearity patterns with 2 parameter(s)
Parameter       [ Expl. params                   ]       cosn
tau             [ kap             psi1           ]      0.856
kap             [ tau             rhog           ]      0.776
psi1            [ rhoR            std_g          ]      0.958
psi2            [ rhoz            std_z          ]      0.911
rhoR            [ psi1            std_g          ]      0.932
rhog            [ psi1            rhoR           ]      0.732
rhoz            [ psi2            std_z          ]      0.932
rr_steady       [ pi_steady       gam_steady     ]      0.718
pi_steady       [ psi1            rr_steady      ]      0.243
gam_steady      [ psi1            rr_steady      ]      0.707
std_R           [ psi1            rhog           ]      0.836
std_g           [ psi1            rhoR           ]      0.929
std_z           [ psi2            rhoz           ]      0.885
Collinearity patterns with 3 parameter(s)
Parameter       [ Expl. params                                   ]       cosn
tau             [ kap             psi1            psi2           ]      0.907
kap             [ tau             rhog            std_g          ]      0.835
psi1            [ rhoR            rhog            std_g          ]      0.970
psi2            [ psi1            rhoz            std_z          ]      0.960
rhoR            [ psi1            rhoz            std_g          ]      0.959
rhog            [ psi1            psi2            rhoR           ]      0.821
rhoz            [ psi1            psi2            std_z          ]      0.962
rr_steady       [ psi1            pi_steady       gam_steady     ]      0.718
pi_steady       [ psi1            rr_steady       std_R          ]      0.243
gam_steady      [ psi1            rr_steady       std_R          ]      0.707
std_R           [ psi1            rhog            rhoz           ]      0.891
std_g           [ psi1            rhoR            rhoz           ]      0.945
std_z           [ psi1            psi2            rhoz           ]      0.936
```

Box 6. Printed output for the An and Schorfheide (2007) example when the Monte Carlo option is triggered.

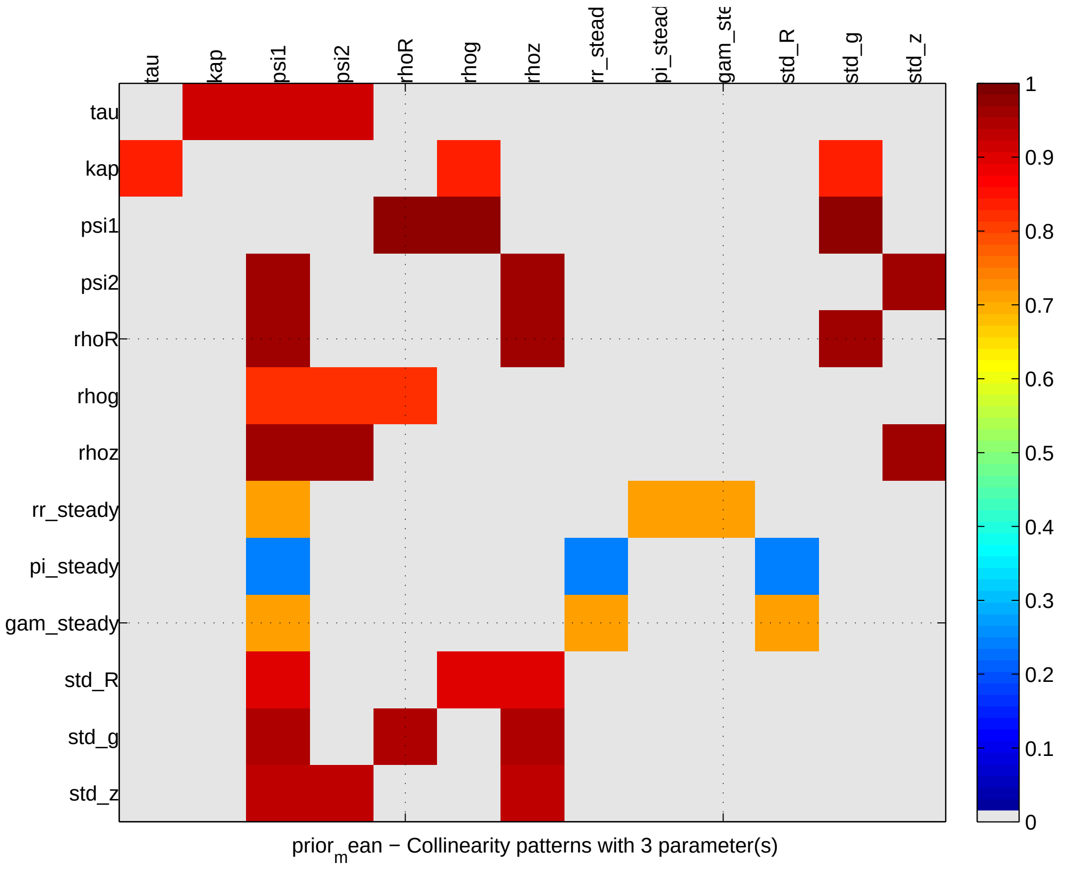


Figure 7: Brute force collinearity search of one parameter vs. groups of there other parameters for the An and Schorfheide (2007).

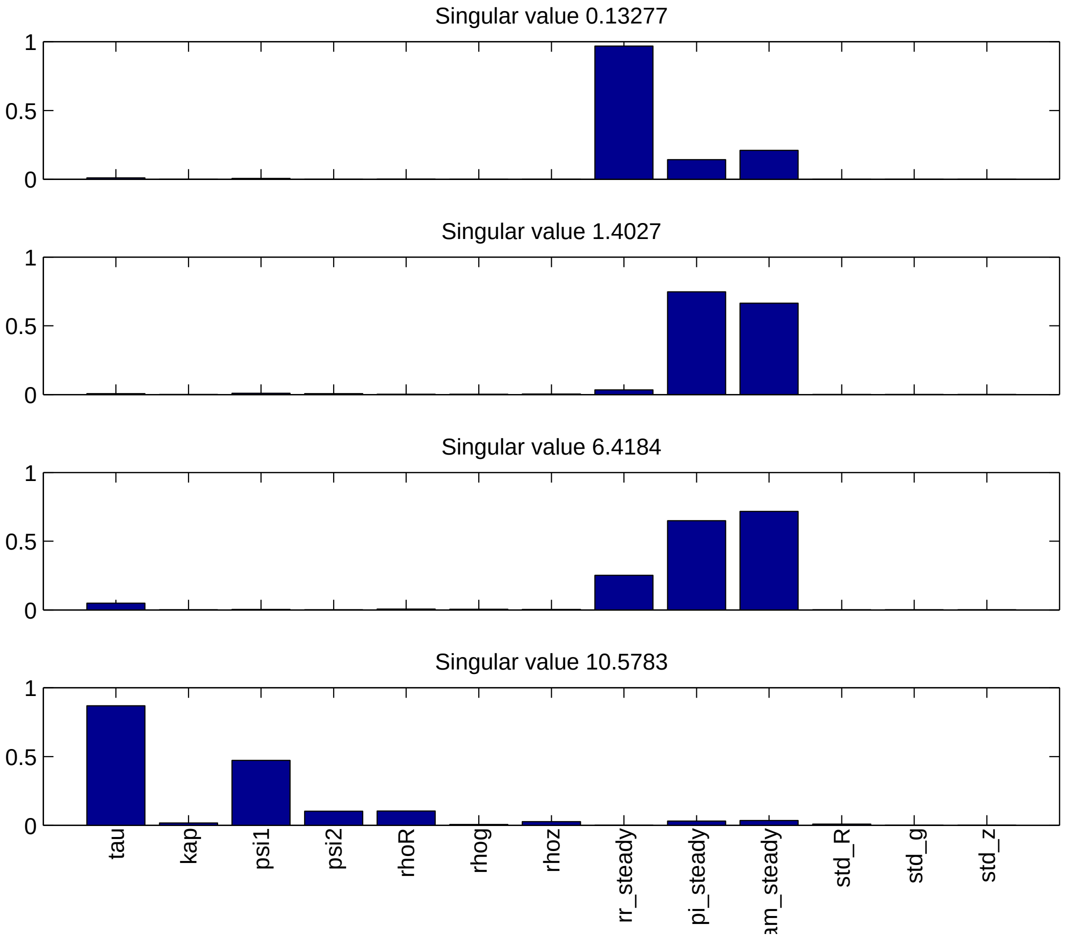


Figure 8: Identification patterns for the An and Schorfheide (2007) (smallest singular values).

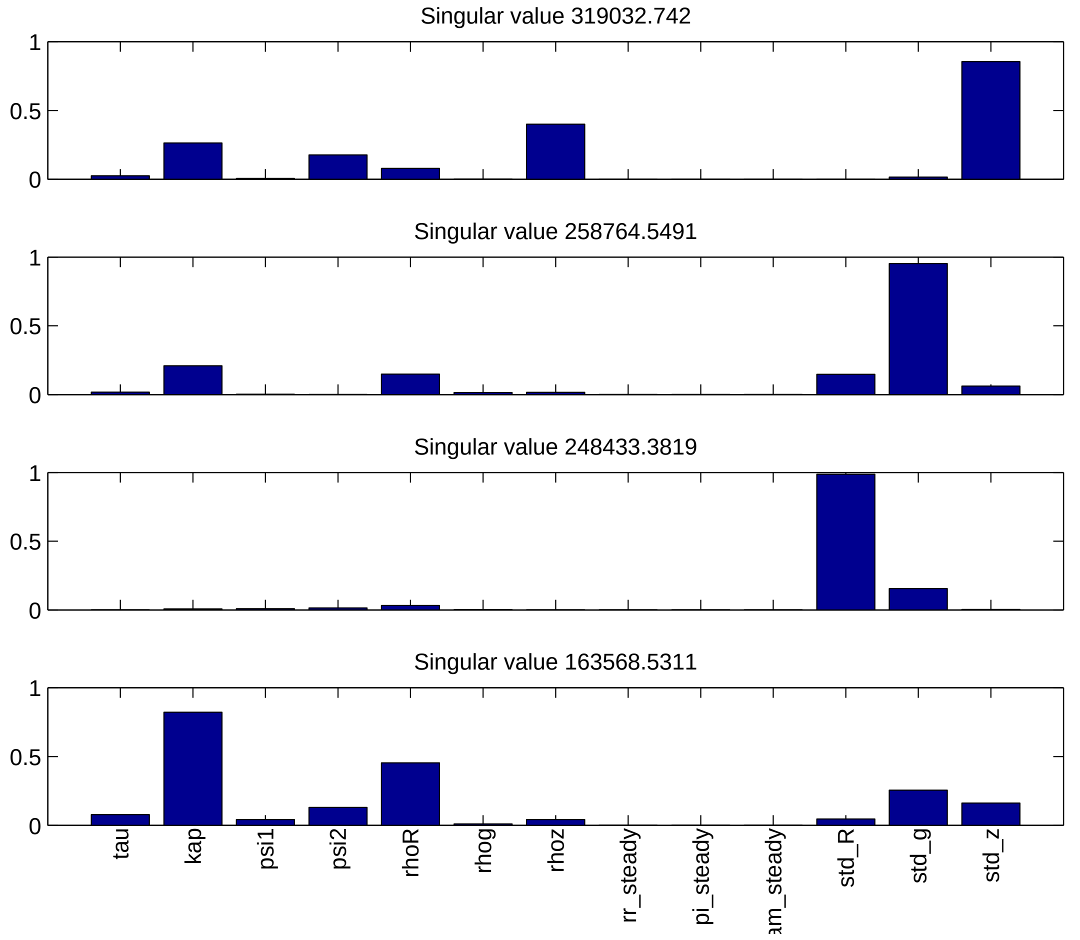


Figure 9: Identification patterns for the An and Schorfheide (2007) (largest singular values).

When the advanced Monte Carlo is performed, first the sensitivity plot is shown: in the advanced mode, sensitivities are shown not only for the moments (i.e. based on $J(q)$) but also for the model and LRE model (i.e. based on J_2 and J_Γ), see Figure 10.

The advanced Monte Carlo diagnostics proceeds by analyzing the condition number of the Jacobians.

- First the histogram of the condition numbers across the Monte Carlo sample is shown (Figure 11).

- Second the sensitivity of the condition number versus parameters is analyzed applying the Monte Carlo filtering technique (Ratto, 2008): the parameters that mostly drive the occurrence of the largest condition numbers are searched. An example for the condition number of $J(q)$ is shown in Figure 12, where one can see that large values for ρ_R and ρ_z are most responsible for the largest condition number, i.e. possibly driving to weaker identification.

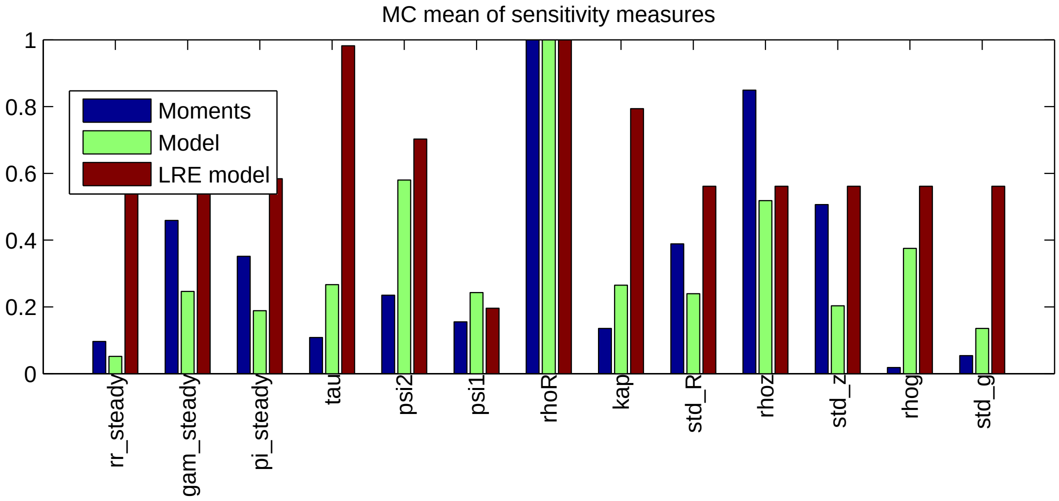


Figure 10: Advanced Monte Carlo sensitivity plot for the An and Schorfheide (2007).

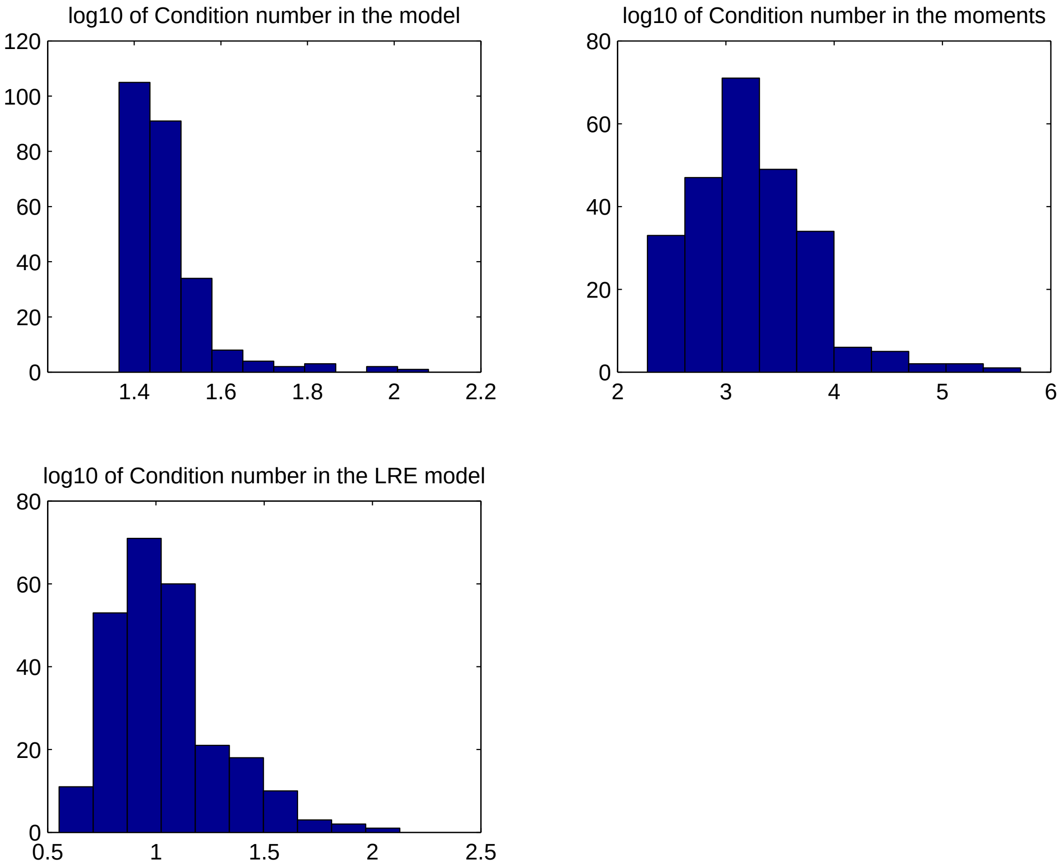


Figure 11: Histograms of condition numbers of Jacobians for the An and Schorfheide (2007) model.

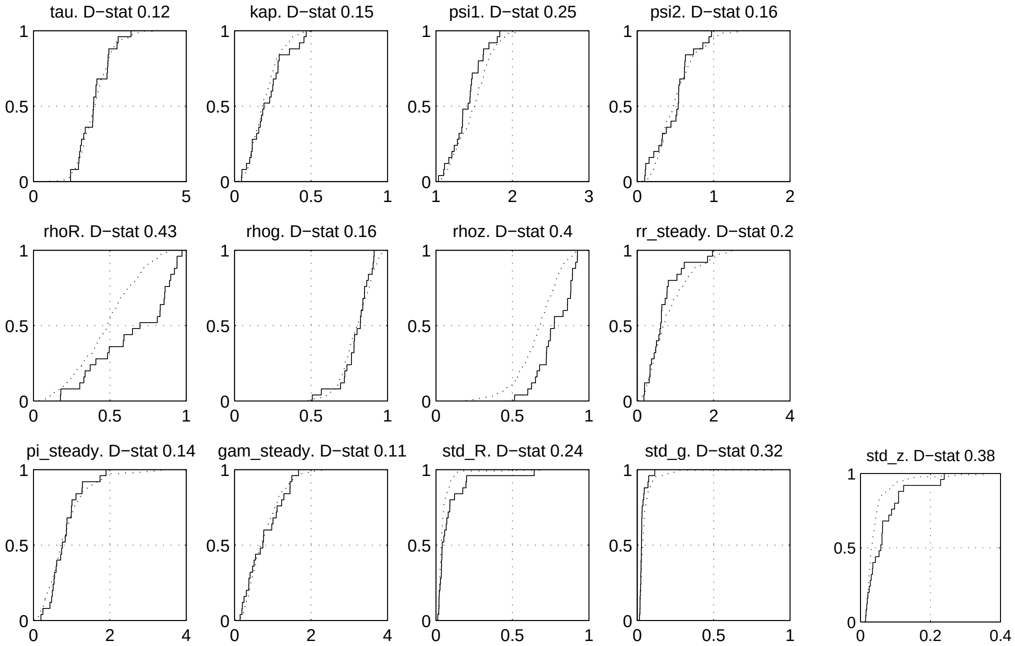
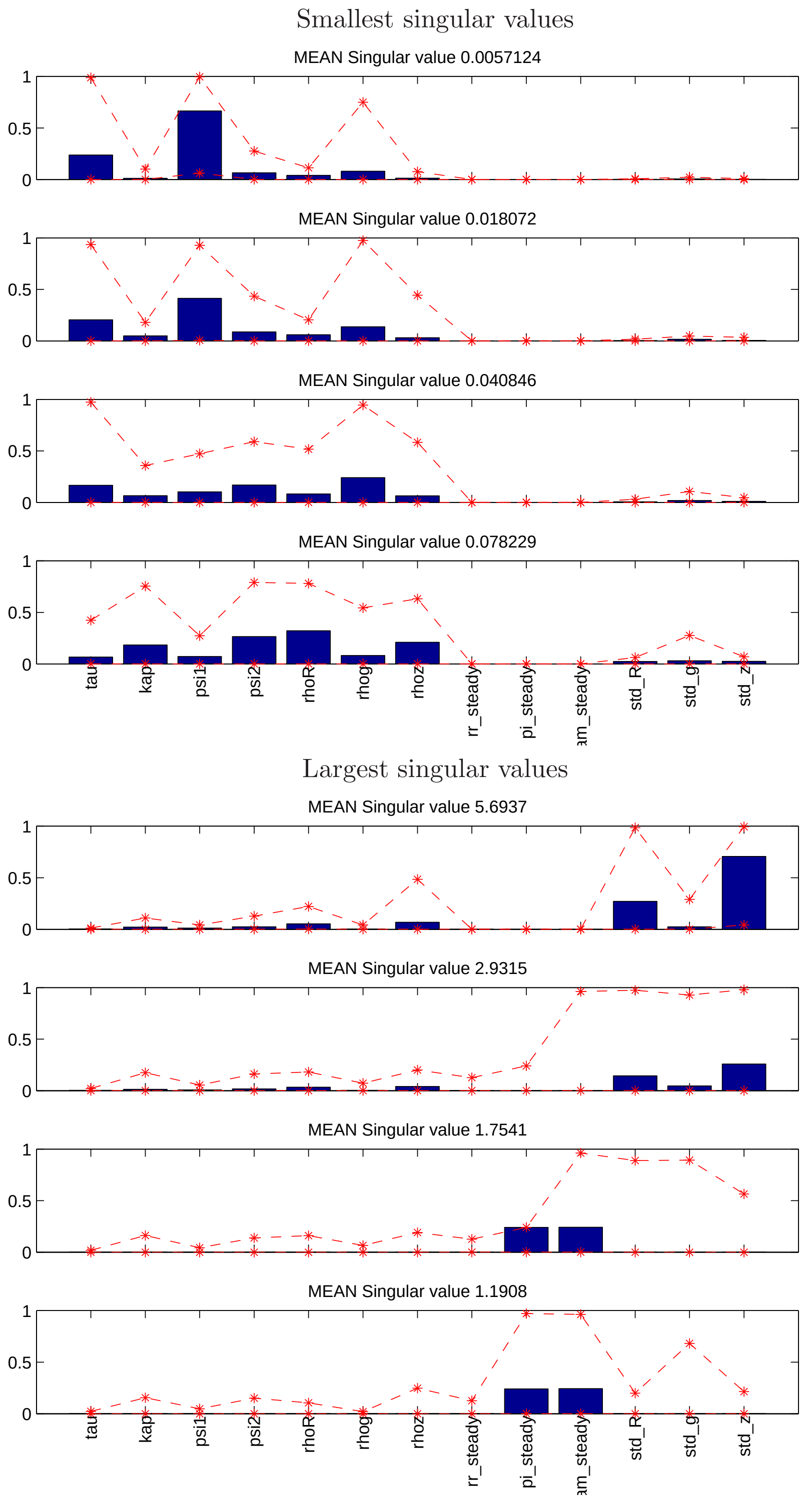


Figure 12: Smirnov tests for the parameters most driving the occurrence of the largest condition numbers in $J(q)$ for An and Schorfheide (2007).

Monte Carlo identification patterns are also displayed, by performing the Singular Valued Decomposition proposed by Andrle (2010) for $J(q)$ in each Monte Carlo sample. This provides a sample of eigenvectors associated to the smallest/largest singular values. The mean and 90% quantile of the distribution of such eigenvectors is plotted as shown in Figure 13. It is interesting to note that, in such plot, the patterns seem to maintain some coherence and consistency across the Monte Carlo sample, suggesting the for this model the identification pattern is quite uniform over the prior space. Also, it is interesting to note that, considering $J(q)$, the mean parameters π^A,r^A and γ^Q display clearly their identifiable effect through the first moments, while using the Hessian (Figure 8), the 'measure' of such effect was less notable.

The final set of diagnostics for the advanced Monte Carlo option, point-estimates are also shown for the parameter combinations associated to the largest and smallest condition number of $J(q)$. We show in Figures 14-15 the identification strength plots for those two special points, where we can see that the identification characteristics are fairly similar to each other and to the prior mean (Figure 5). The main difference distinguishing the 'worst' identifiable case in Figure 14 is that the identification strength of ϕ_1 and ϕ_2 drops significantly.


Smallest singular values

Largest singular values

Figure 13: Smirnov tests for the parameters most driving the occurrence of the largest condition numbers for An and Schorfheide (2007).

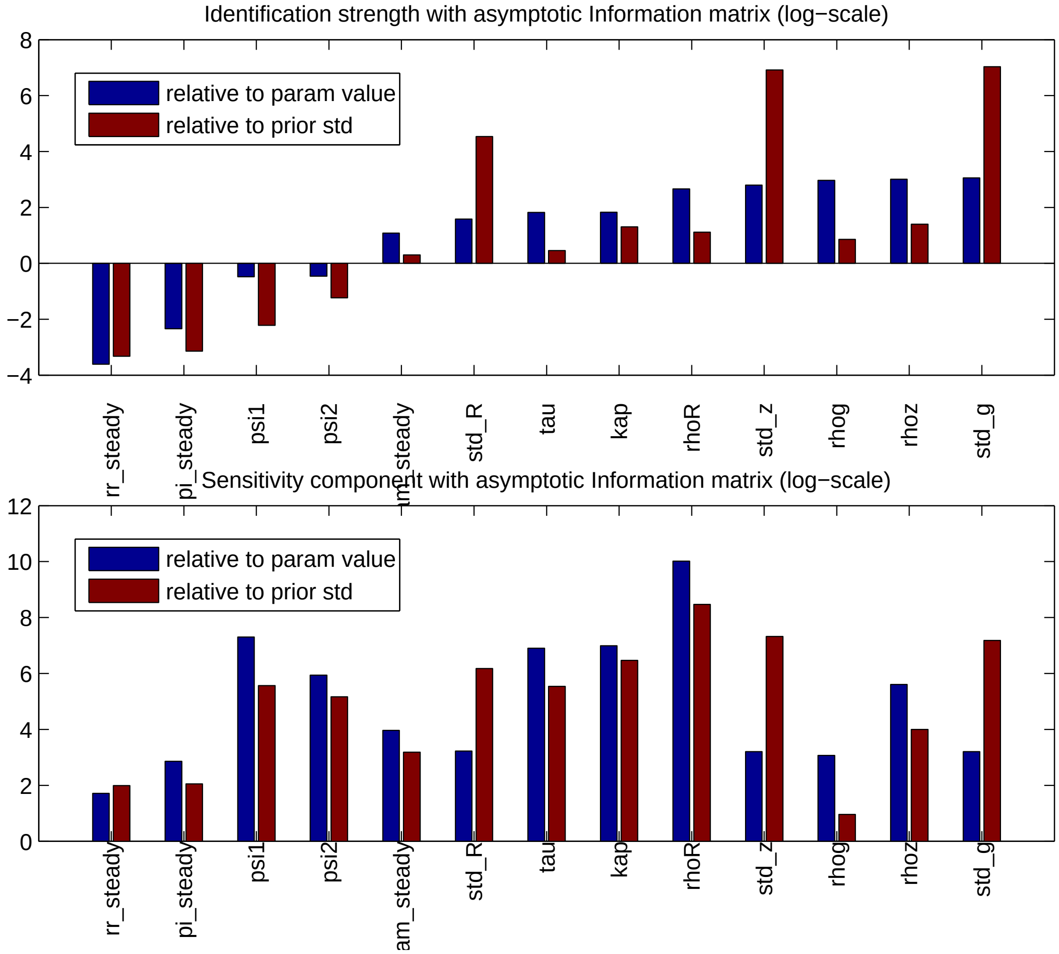


Figure 14: Identification strength for the parameter set associated to the LARGEST condition number of $J(q)$ for An and Schorfheide (2007).

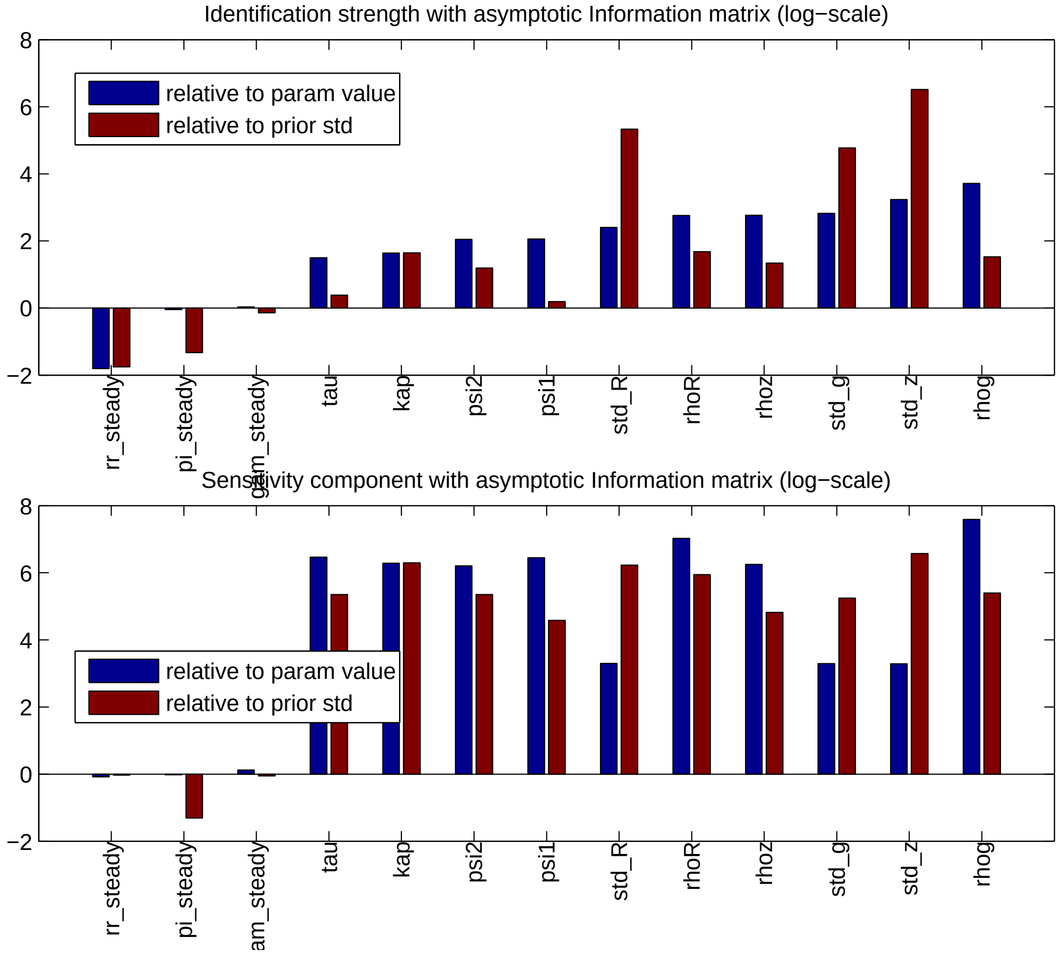


Figure 15: Identification strength for the parameter set associated to the SMALLEST condition number of $J(q)$ for An and Schorfheide (2007).

7 Conclusions

We have described in detail the Identification Toolbox developed for analyzing DSGE models under the DYNARe environment. The toolbox provides a wide set of diagnostic tools to analyze the identification strength of the model. Advanced options allow to inspect identification patterns, that help the analyst in tracking the possibly weakest elements of the model parameter set. Moreover, the Monte Carlo option allow to study how identification features are changed across the entire prior space.

References

An, S. and F. Schorfheide (2007). Bayesian analysis of DSGE models. *Econometric Reviews 26*(2-4), 113–172. DOI:10.1080/07474930701220071.

Andrle, M. (2010). A note on identification patterns in DSGE models (august 11, 2010). ECB Working Paper 1235. Available at SSRN: http://ssrn.com/abstract=1656963.

Canova, F. and L. Sala (2009, May). Back to square one: identification issues in DSGE models. *Journal of Monetary Economics 56*(4).

Cochrane, J. H. (2007, September). Identification with taylor rules: A critical review. NBER Working Papers 13410, National Bureau of Economic Research, Inc.

Gu, C. (2002). *Smoothing Spline* ANOVA *Models*. Springer-Verlag.

Iskrev, N. (2010a). Evaluating the strenght of identification in DSGE models. an a priori approach. unpublished manuscript.

Iskrev, N. (2010b). Local identification in DSGE models. *Journal of Monetary Economics 57*, 189–202.

Kim, J. (2003, February). Functional equivalence between intertemporal and multisectoral investment adjustment costs. *Journal of Economic Dynamics and Control 27*(4), 533–549.

Komunjer, I. and S. Ng (2009). Dynamic identification of DSGE models. unpublished manuscript.

Oakley, J. and A. O'Hagan (2004). Probabilistic sensitivity analysis of complex models: a Bayesian approach. *J. Royal Stat. Soc. B 66*, 751–769.

Ratto, M. (2008). Analysing DSGE models with global sensitivity analysis. *Computational Economics 31*, 115–139.

Ratto, M. and N. Iskrev (2010a). Computational advances in analyzing identification of DSGE models. 6th DYNARE Conference, June 3-4, 2010, Gustavelund, Tuusula, Finland. Bank of Finland, DSGE-net and Dynare Project at CEPREMAP.

Ratto, M. and N. Iskrev (2010b). Identification toolbox for DYNARE. 1st MONFISPOL conference, London, 4-5 November 2010. The London Metropolitan University and the European Research project (FP7-SSH) MONFISPOL.

Ratto, M. and A. Pagano (2010). Recursive algorithms for efficient identification of smoothing spline anova models. *Advances in Statistical Analysis*. In Press.

Saltelli, A., P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola (2010). Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. *Computer Physics Communications 181*(2), 259 – 270.

Sobol', I. M. (1990). Sensitivity estimates for nonlinear mathematical models. *Matematicheskoe Modelirovanie 2*, 112–118. in Russian, translated in English in Sobol' (1993).

Sobol', I. M. (1993). Sensitivity analysis for non-linear mathematical models. *Mathematical Modelling and Computational Experiment 1*, 407–414. English translation of Russian original paper Sobol' (1990).

Storlie, C. B. and J. C. Helton (2007). Multiple predictor smoothing methods for sensitivity analysis: description of techniques. *Reliability Engineering & System Safety 93*, 28–54.

European Commission

EUR 25032 EN – Joint Research Centre – Institute for the Protection and Security of the Citizen
Title: European Community project MONFISPOL (grant agreement SSH-CT-2009-225149) Deliverable 3.1.2:
Algorithms for identification analysis under the DYNARE environment: final version of software.
Author(s): M. Ratto, N. Iskrev
Luxembourg: Publications Office of the European Union
2011 – 58 pp. – 21 x 29.7 cm
EUR – Scientific and Technical Research series – ISSN 1831-9424
ISBN 978-92-79-22035-7
doi:10.2788/97652

Abstract

In this report we document in detail the Identification package developed for the DYNARE environment. The package implements methodologies and collects developed algorithms to assess identification of DSGE models in the entire prior space of model deep parameters, by combining ‘classical’ local identification methodologies and global tools for model analysis, like global sensitivity analysis.

How to obtain EU publications

Our priced publications are available from EU Bookshop (http://bookshop.europa.eu), where you can place an order with the sales agent of your choice.

The Publications Office has a worldwide network of sales agents. You can obtain their contact details by sending a fax to (352) 29 29-42758.

The mission of the JRC is to provide customer-driven scientific and technical support for the conception, development, implementation and monitoring of EU policies. As a service of the European Commission, the JRC functions as a reference centre of science and technology for the Union. Close to the policy-making process, it serves the common interest of the Member States, while being independent of special interests, whether private or national.

LB-NA-25032-EN-N

JRC
EUROPEAN COMMISSION

Publications Office

ISBN 978-92-79-22035-7

9 789279 220357