Advanced Time Series

Gauss Exercises

1st GAUSS assignment ATS (cannot be handed in for grading!)

1. Create a standard normally-distributed random variable
(Compare page 23-29 in the introductory slides)

1. Generate a sequence of T=1000 standard normal random variables and write the realizations in one column (GAUSS hint: `rndn(r,c)`). Produce a graph of the resulting series. For the x-axis value you need to generate a sequence (Hint: `seqa(1,1,T)`). Plot the series against this sequence.

2. Compute sample means (GAUSS hint: `meanc`) and sample standard deviations (GAUSS hint: `stdc`) of the series generated in step 1. Interpret the results.

3. Repeat the steps, i.e. let the program run another time. Compare your results. What happens to your graph?

4. What is the function of a *seed*? Repeat the steps 1 to 3, but this time use a seed.

5. Generate a second series that contains the cumulated standard normal random variables generated in step 1 and plot the series. How do we call such a process?

2. Simulate an AR(1) process

1. Simulate a first order AR (AR(1)) process that is defined as follows:

   $y_t = \phi y_{t-1} + \varepsilon_t$, where $\varepsilon_t \sim \mathrm{N}(0, 1)$

   The number of observations is T = 100.
   Draw the vector of residuals ε_t from a standard normal distribution, initialize the vector y and define its starting value $y_0 = 0$, which is not part of the final vector y (so the first element of y is y_1). The elements of y are generated successively :

   $y_1 = \phi y_0 + \varepsilon_1$
   $y_2 = \phi y_1 + \varepsilon_2$
   $y_3 = \phi y_2 + \varepsilon_3$
   and so on.

   Generate the series by using a *loop* statement!

2. Plot the realization of the process.

3. Write a procedure that delivers one realization (T = 100) of an AR(1) process. Hint: Use as input variables ϕ and the number T, as output the resulting AR(1) realization.

4. Simulate the process for different parameter values: $\phi = 0$ (white noise) $\phi =$ -0.5 $\phi =$ 0.5 and $\phi = 1$ (random walk) by calling on your procedure.

5. Plot the realizations of the process for the different parameters into one graphic window (Hint: `window(r,c,t)`). Give each graph a title (see page 27 on the introductory slides).

2nd GAUSS assignment ATS (can be handed in for grading!)

1. Simulate an AR(1) process -continued

1. Use your procedure from the first assignment to simulate an AR(1) process with $\phi =$ 0 (white noise), $\phi =$ -0.5, $\phi =$ 0.5, and $\phi =$ 1 (random walk) and $T =$ 100 (number of observations).

2. Analyze the stationarity properties of this AR(1) process by using simulations: Compute ensemble means and variances for 30 realizations of your process (for every of the four parameter alternatives). Hint: Use again a loop to generate a matrix which contains the different realizations of the process in its columns. Visualize the means and variances in a plot and label the axis. Interpret the graph.

3. Compute empirical autocorrelations (first order) by using the 30 realizations (Ensembles), i.e compute Corr(y_1, y_2) and Corr(y_2, y_3),..., Corr(y_{99}, y_{100})). Hint: You could use the Gauss command `corrx`. It computes the correlation between columns of a matrix and returns the corresponding correlation matrix, i.e. if y is a $N \times T$ matrix then `corrx(y)` returns a $T \times T$ (symmetric) matrix of correlations. The ith row, jth column element of the correlation matrix gives the correlation between column i and column j. You need to read out the correlations you want to have. There are different ways to do this. (You could use a help matrix: If, for instance, you want to read out the diagonal elements of a matrix, you could multiply the matrix with an identity matrix. Then all the off diagonal elements are zero and you can use `sumc` to sum up the columns, which then results in a vector that contains the diagonal elements of the initial matrix. *This is only an example, we do not need the diagonal elements here, but...!*

   Plot the resulting sequence of autocorrelations in a graph and title it. Interpret the graph.

4. Compute for *one* arbitrary realization of the AR(1) process (for every of the four parameter alternatives) the empirical autocorrelation of first order (GAUSS hint;`acf`) and compare this to the theoretical value (print them to the output window) . Interpret the result.

5. Given your results of the previous questions: Evaluate if the AR(1) process is (weakly) stationary. Does the stationarity depend on the choice of the parameter ϕ? For which parameter value is the AR(1) process stationary?

3. Simulate a MA(1) process

1. Simulate a first order moving average (MA(1)) process that is defined as follows:

   $y_t = \theta\varepsilon_{t-1} + \varepsilon_t$, $\varepsilon_t \sim \mathrm{N}(0, 1)$

   $T =$ 100. Simulate the process for different parameter values: θ =0.5, θ =-0.5, θ =2, θ =-10 Plot the realizations of the process in a graph.

2. Write a procedure that delivers one realization of a MA(1) process. Hint: Use as input variables θ and the number T, as output the resulting MA(1) realization.

3. Evaluate whether the requirements for weak stationarity are fulfilled. Compute ensemble means and variances for 30 realizations of your process (for every of the four parameter alternatives). Visualize the means and variances in a plot and label the axis. Interpret the graph.

4. Compute empirical autocorrelations (first order) by using the 30 realizations (Ensembles). Plot the resulting sequence of autocorrelations in a graph and title it. Interpret the graph.

5. Compute for *one* arbitrary realization of the MA(1) process (for every of the four parameter alternatives) the empirical autocorrelation of first order (GAUSS hint;`acf`) and compare this to the theoretical value (print them to the output window) . Interpret the result.

6. Given your results of the previous questions: Evaluate if the MA(1) process is (weakly) stationary. Does the stationarity depend on the choice of the parameter θ?

3rd GAUSS assignment ATS (can be handed in for grading!)

1. Estimate an AR(1) process using CML

1. Use the AR simulation procedure of the 1st set of assignment. Simulate an AR(1) process with $\phi = 0.5$, $c = 0$, $\sigma^2 = 1$ and $T = 100$.

   $y_t = c + \phi y_{t-1} + \varepsilon_t$, $\varepsilon_t \sim \mathrm{N}(0, \sigma^2)$

2. The log likelihood function of an AR(1) process conditioning on the first observation is given by:

$$\ln L = \; - \; \left[\frac{T-1}{2}\right] \ln(2\pi) - \left[\frac{T-1}{2}\right] \ln(\sigma^2) - \sum_{t=2}^{T} \left[\frac{\varepsilon_t^2}{2\sigma^2}\right],$$

   where $\varepsilon_t = y_t - c - \phi y_{t-1}$.

   Write a procedure that estimates the parameters (c, ϕ, σ) of an AR(1) process by using the Gauss library `cml` (do not forget to inlcude the library into your program file). The cml call is:

   { x,f,g,cov,retcode } = CML(dataset,vars,&fct,start)

   &fct is the procedure that computes the log likelihood contributions. In our case the one of an AR(1) process given above. Input arguments for the procedure are *always* a *column* vector of the parameters to be estimated and a data matrix. Output argument is the vector of likelihood *contributions of each observation.*

   In the body of the procedure you have to read out the parameters from the input vector, then calculate the sequence (vector) of ε_t and compute the loglikelihood. Hint: `lagn` gives the n-th lag of matrix. Compute the lag of the series as a global variable (not inside of the procedure). Also note that you loose one observation by taking a lag of the series.

   When you call `cml`, you need to provide starting values for the parameter vector.

3. Estimate ϕ, c, and σ^2 and discuss your results.

4. Compute standard errors of the estimators. Test the null hypothesis $H_0 : parameter = 0$. Computer p-values of the test-statistic. Gauss hint: cdfnc. Interpret your results.

5. Simulate the process for different values of $T = 30$, $T = 50$, $T = 100$, $T = 200$. Estimate ϕ and discuss your results.

6. Write a likelihood function for only two parameters σ and ϕ. Plot the likelihood as a function of σ and ϕ in a three dimensional plot (GAUSS Graph Hint: xyz/surface). Let σ run from 0.8 to 1.5 in steps of 0.05 and let ϕ run from 0 to 0.9 in steps of 0.01. Use the graph to explain the general idea of maximum likelihood.

4th GAUSS assignment ATS (can be handed in for grading!)

1. Estimate a MA(1) process

1. Use the MA simulation procedure of the 1st set of assignment. Simulate an MA(1) process with $\theta = 0.5$, $c = 0$, $\sigma^2 = 1$, and $T = 100$.

   $y_t = c + \theta\varepsilon_{t-1} + \varepsilon_t$, $\varepsilon_t \sim \mathrm{N}(0, \sigma^2)$

2. The conditional log likelihood function (assuming that $\varepsilon_0 = 0$) of an MA(1) processis given by:

$$\ln L = \quad - \quad \frac{T}{2}\ln(2\pi) - \frac{T}{2}\ln(\sigma^2) - \sum_{t=2}^{T}\left[\frac{\varepsilon_t^2}{2\sigma^2}\right],$$

   where $\varepsilon_t = y_t - c - \theta\varepsilon_{t-1}$.

   Write a procedure that estimates the parameters (c, ϕ, σ) of an MA(1) process by using the Gauss library `cml` (do not forget to inlcude the library into your program file). Remember: Input arguments for the likelihood procedure are *always* a *column* vector of the parameters to be estimated and a data matrix. Output argument is the vector of likelihood *contributions of each observation.*

3. Compute standard errors of the estimators. Test the null hypothesis $H_0 : parameter = 0$. Computer p-values of the test-statistic. Gauss hint: cdfnc.

4. Simulate the process for different parameter values of θ = -0.5, $\theta = 1$, $\theta = 0.3$ and $T = 100$. Estimate θ. Discuss the maximum likelihood estimator $\hat{\theta}$.

5. Simulate the process for different values of $T = 30, 50, 100, 200$. Estimate θ. Discuss the maximum likelihood estimator $\hat{\theta}$ and its estimated standard error. How do your results change?

6. Write the likelihood function for only two parameters σ and θ. Plot the likelihood as a function of σ and θ in a three dimensional plot. GAUSS Graph Hint: xyz.

5th GAUSS assignment ATS (can be handed in for grading!)

Testing for Stationarity - Simulating the Dickey-Fuller Test Statistics:

- Stationarity test with null hypothesis: unit root (non stationarity), i.e. that $\rho = 1$
- Non-standard asymptotic distribution of unit root processes
- Inference requires simulation of asymptotic distribution

Procedure for the Dickey-Fuller test statistic simulation:

1. Simulate a random walk
2. Conduct an OLS regression and calculate the t-statistic for the null hypothesis that the true value of ρ equals 1
3. Simulate the test statistic: Run Step 1 and 2 n=10000 times and sort the t-values into quantiles

True Process:

$$y_t = y_{t-1} + \varepsilon_t$$

Case 1

Estimated Regression:

$$y_t = \rho y_{t-1} + \varepsilon_t \tag{1}$$

Case 2

Estimated Regression:

$$y_t = \alpha + \rho y_{t-1} + \varepsilon_t \tag{2}$$

Case 4

Estimated Regression:

$$y_t = \alpha + \rho y_{t-1} + \delta t + \varepsilon_t \tag{3}$$

Programming - Case 1:

1. Simulate a random walk

1. Simulate a random walk that is defined as follows:

$$y_t = \alpha + \rho y_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim \mathrm{N}(0, \sigma^2) \tag{4}$$

   $T = 100$. Simulate the process for the following parameter values: ρ =1, α =0, $\sigma^2 = 1$.

2. Write a procedure that delivers one realization of the random walk process. Hint: Use as input variables ρ, α and T, as output the resulting random walk realization.

2. Estimate the unknown parameters with OLS

1. Compute the estimator $\widehat{\rho}$ in Equation (1). First, implement an OLS estimator: $\beta = (X'X)^{-1}(X'Y)$;. $X = y_{t-1}$ and Y contains the dependent variable y_t. For estimation do not use the starting values of y_t and you loose one observation by taking a lag of the series y_t.

2. Estimate the standard error of your parameter $\widehat{\rho}$ and calculate the t-statistic for the null hypothesis that the true value of ρ equals 1.

3. Write a procedure around the OLS estimation. The OLS procedure has two input variables. A $(T \times K)$ matrix containing K exogenous regressors (here $X = y_{t-1}$) and a $(T \times 1)$ matrix containing the dependent variable, i.e. $Y = y_t$. As output specify a row **vector** that contains the estimate for $\widehat{\rho}$, the corresponding standard error and the t-statistic associated with the null hypothesis that the true value $\rho = 1$.

3. Simulation of the test statistic

1. Simulate the two procedures of Step 1 and 2 n=10000 times. In order to do so, write a loop around the two procedures. Collect the parameter estimates for ρ and the corresponding standard errors and t-values in a matrix. Hint: Initialize a matrix of dimension $n \times 3$. Fill in the results from the simulation.

2. Sort the estimated t-values into quantiles (0.01, 0.025, 0.05, 0.1, 0.9, 0.95, 0.975, and 0.99). Hint: Use the GAUSS command `quantiled`.

3. Create a critical values table for the Dickey-Fuller Test for Case 1 for sample size T = 25 and T = 100 and for the 0.01, 0.025, 0.05, 0.1, 0.9, 0.95, 0.975, and 0.99 Quantile.

Programming - Case 2:

1. Use the procedure from Case 1 and simulate the random walk (see Equation (4)) with $T = 100$, ρ =1, α =0, $\sigma^2 = 1$.

2. Estimate the unknown ρ and α parameters in Equation (2) with OLS. Hint: Order the matrix that contains your regressors as follows: $X = y_{t-1}$ ∼vector of constants.

3. Simulate the previous two steps n=10000 times. Store the parameter estimates for ρ and the corresponding standard errors and t-values in a matrix.

4. Sort the estimated t-values into quantiles.

5. Create a critical values table for the Dickey-Fuller Test for Case 2 for sample size T = 25 and T = 100 and for the 0.01, 0.025, 0.05, 0.1, 0.9, 0.95, 0.975, and 0.99 Quantile.

Programming - Case 4:

1. Use the procedure from Case 1 and simulate a random walk (see Equation (4)) with $T = 100$, ρ =1, α =0, $\sigma^2 = 1$.

2. Estimate the unknown ρ, δ and α parameters in Equation (3) with OLS. Hint: Order the matrix that contains your regressors as follows: $X = y_{t-1}$ ∼vector of constants∼time trend.

3. Simulate the two steps n=10000 times. Store the parameter estimates for ρ and the corresponding standard errors and t-values in a matrix.

4. Sort the estimated t-values into quantiles.

5. Create a critical values table for the Dickey-Fuller Test for Case 4 for sample size T = 25 and T = 100 and for the 0.01, 0.025, 0.05, 0.1, 0.9, 0.95, 0.975, and 0.99 Quantile.

Interpretation:

A researcher wants to conduct a Dickey-Fuller test to test an economic time series for a unit root. He estimates a regression model of the form:

$$y_t = \alpha + \rho y_{t-1} + \delta t + \varepsilon_t$$

The researcher works under the null hypothesis that the true data generating process is given by:

$$y_t = y_{t-1} + \varepsilon_t$$

Running the regression the researcher computed the estimate $\hat{\rho} = 0.78$. The estimated OLS standard error $s.e.(\hat{\rho}) = 0.021$. The sample size is 100. Use your simulated critical values and interpret the result.

6th GAUSS assignment ATS (can be handed in for grading!)

1. Loading Data and Data Description

1. First, load the data into a GAUSS matrix. The data_var.txt file contains Swiss seasonal adjusted macroeconomic data: quarterly Interest Rate in % (first column), quarterly Consumer Price Index (second column), quarterly Gross Domestic Product in Swiss (BIP) in Mio CHF (third column) and quarterly Money Stock M1 in Mio CHF (fourth column). The first observation is from the first quarter of 1974 and the last observation is from the first quarter of 2002. T = 113. Gauss hint: loadm all[] = data_var.txt; all2 =reshape(all,113,4);

2. Plot the four time series and describe the plot. Decide whether the time series includes
   a) a constant
   b) a constant and a time trend
   c) no constant and no time trend.

2. Conducting Dickey Fuller Tests

1. Conduct a Dickey Fuller test of non stationarity for all four time series. Estimate a regression model.

   Generally, the regression model is written as:

   $$\Delta y_t = \beta_1 + \beta_2 y_{t-1} + \beta_3 t + \varepsilon_t$$

   with $\Delta y_t = y_t - y_{t-1}$, i.e. first differences of y_t, t as time trend and β_1 as constant.

   Based on your observations, if you decided that the series contains a constant, include β_1 in the regression. Else $\beta_1 = 0$. Based on your observations, if you decided that the series contains a time trend, include $\beta_3 \times t$ in the regression. Else $\beta_3 = 0$. For the first parameter estimate $\widehat{\beta_2}$ use the standard error to compute the t-statistic for $H_0 : \beta_2 = 0$.

   Compare your t-value with the corresponding Dickey-Fuller distribution and decide whether you can reject or not reject the null hypothesis of a unit root. Interpret your result.

2. Compute log differences of the series, i.e. $\tilde{y}_t = \Delta log(y_t) = log(y_t) - log(y_{t-1})$.

3. Plot the series $\tilde{y}_t$ and describe the plot. Decide whether the time series includes
   a) a constant
   b) a constant and a time trend
   c) no constant and no time trend.

4. Conduct a Dickey Fuller test on $\tilde{y}_t$ (for all four time series) and discuss your result. Does $\tilde{y}_t$ contain a unit root?

7th GAUSS assignment ATS (can be handed in for grading until Tue, 22nd Dec 2009!)

Estimate a GARCH(1,1) for a return series using cml

1. First, load the return data into a GAUSS matrix `garch.fmt`. You can load this files with the `load` command (look it up in the GAUSS Help).

2. Plot the return time series. Describe the stylized facts you observe.

   A simple model to account for these stylized facts ist the GARCH model.
   Mean equation (y_t is a log return time series):

$$y_t = c + \varepsilon_t$$

   In the following we assume c = 0. Argue why this is a sensible assumption. Therefore,

$$\varepsilon_t = y_t$$

   Variance equation:

$$h_t = \omega + \alpha\varepsilon_{t-1}^2 + \beta h_{t-1} \tag{5}$$

   Writing the conditional likelihood function GARCH(1,1):

$$f(y_t|y_{t-1}, ...y_0; \boldsymbol{\theta}) = \frac{1}{\sqrt{2\pi h_t}} \exp\left[\frac{-(y_t - c)^2}{2h_t}\right]$$

   Conditional log likelihood function:

$$\begin{aligned} \mathscr{L}(\boldsymbol{\theta}) &= \sum_{t=1}^{T} \log f(y_t|y_{t-1}, ...y_0; \boldsymbol{\theta}) \\ &= -\frac{T}{2}\log(2\pi) - \frac{1}{2}\sum_{t=1}^{T}\log(h_t) - \frac{1}{2}\sum_{t=1}^{T}\frac{(y_t - c)^2}{h_t} \end{aligned}$$

3. Write a procedure that estimates the parameters of an GARCH(1,1) process. Hint: First read all following remarks. Then work through them step by step.

   First build the h_t of GARCH(1,1) model. Hint: You need starting values for h, ω and α, β. The starting value of the h sequence is an appropriate guess of ω and α, β for computing the unconditional expectation of the GARCH(1,1) process in Equation (5), i.e. $h = \omega/(1 - \alpha - \beta)$ with $\omega = 0.1$, $\alpha = 0.1$, and $\beta = 0.6$.

Second compute the value of the log likelihood function by plugging in ε_t and the conditional variance sequence of h_t.

Now write the rest of the procedure around the procedure-body. Hint: Do not forget to define locals. Input variables of the procedure are the data y and the parameters that are estimated. (ω, α, β). The parameters enter the procedure as one column vector!) Remember: Output variable is a column vector that contains the likelihood contribution of each observation.

Before you call cml make modify your parameters: In a GARCH estimation you have to ensure that the left hand side variable of Equation (5) is always positive. You have to impose restrictions on your parameters. Modify your procedure the following way: $parameter = \exp(parameter^*)$. The estimated equation is then

$$h_t \quad = \quad \exp(\omega^*) + \exp(\alpha^*)\varepsilon_{t-1}^2 + \exp(\beta^*)h_{t-1}$$

4. Use CML to estimate the unknown parameters; ω^*, α^*, β^*. Take as starting values the log of the starting values given above. Choose as global variables:
   _cml_Algorithm =1;
   _cml_LineSearch=2;
   _cml_DirTol = 1e-5;
   _cml_CovPar = 2;

5. After Estimation. Transfer the estimated parameter back by taking $parameter = \exp(parameter^*)$.

8th GAUSS assignment ATS (can be handed in for grading!)

1. Estimate a GARCH(1,1) for a return series using cml

Estimate a GARCH (1,1) using the return data given in `garch.fmt` (for details see last assignment sheet!) for:

1. unrestricted parameters
2. parameters restricted by $parameter = \exp(parameter^*)$
3. parameters that are directly restricted (to be larger than zero) by cml globals: `_cml_c` and `_cml_d`. These are matrices that define inequalities, i.e, throughout the optimization using cml the following inequality has to be always fulfilled:
   `cml_c*parametervector .gt _cml_d`

Compare your results from the differnet estimations. What difficulties do you encounter?

2. Calculate standard erros using the delta method

1. To compute standard errors of the estimators ω, α, β, when estimated with the restriction $parameter = \exp(parameter^*)$, the delta method is required:

   First, write a procedure that returns you the econmic parameters from the estimated ones, i.e.$parameter = \exp(parameter^*)$. Input variable is one column vector θ^* containing ω^*, α^*, β^*. Output variable is one column vector θ containing ω, α, β.

   Second, write a procedure that returns the estimated variance covariance matrix of $\theta = \omega, \alpha, \beta$. Input variables are estimated θ^* and the estimated variance covariance matrix of θ^*. Within the procedure use the Gauss function `gradp` and calculate standard errors according to the delta method summarized below. Describe shortly, how this function works.

2. Test the null hypothesis $H_0 : \beta = 0.7$. $H_A : \beta \neq 0.7$. Compute p-values of the test-statistic. Interpret.

Delta Method

Suppose that $\{\mathbf{x}_n\}$ is a sequence of K-dimensional random vectors such that $\mathbf{x}_n \xrightarrow{p} \boldsymbol{\beta}$ and

$$\sqrt{n}(\mathbf{x}_n - \boldsymbol{\beta}) \xrightarrow{d} N(\mathbf{0}, \boldsymbol{\Sigma})$$

then

$$\sqrt{n}(\mathbf{a}(\mathbf{x}_n) - \mathbf{a}(\boldsymbol{\beta})) \xrightarrow{d} N(\mathbf{0}, \mathbf{A}(\boldsymbol{\beta})\boldsymbol{\Sigma}\mathbf{A}(\boldsymbol{\beta})')$$

where $\mathbf{A}(\boldsymbol{\beta})$ is the matrix of continuous first derivatives of $\mathbf{a}(\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}$:

$$\mathbf{A}(\boldsymbol{\beta}) = \frac{\partial \mathbf{a}(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}'}.$$

Hint: Look in the GAUSS help how the `gradp` function works.

9th GAUSS assignment ATS (can be handed in for grading!)

1. Loading Data and Data Description

1. First, load the data into a GAUSS matrix. The .txt file contains Swiss seasonal adjusted macroeconomic data: quarterly Interest Rate in % (first column), quarterly Consumer Price Index (second column), quarterly Gross Domestic Product in Swiss (BIP) in Mio CHF (third column) and quarterly Money Stock M1 in Mio CHF (fourth column). The first observation is from the first quarter of 1974 and the last observation is from the first quarter of 2002. T = 113. Gauss hint: `loadm all[] = data_var.txt; all2 =reshape(all,113,4);`

- m = money stock M1
- r = quarterly average of three month Swiss franc LIBOR rate of interest
- p = Consumer price index
- y = GDP in 1990 Swiss francs

Write a procedure that creates the differences and log differences of some vector or matrix. Use it to get following series: $\tilde{m}_t$, $\tilde{r}_t$, $\tilde{p}_t$, $\tilde{y}_t$
Denote: $\tilde{m}_t = \Delta \log m_t$; $\tilde{r}_t = \Delta r_t$; $\tilde{p}_t = \Delta \log p_t$; $\tilde{y}_t = \Delta \log y_t$;
($\Delta \log m_t$, Δr_t, $\Delta \log p_t$, $\Delta \log y_t$) with $\Delta \log p_t = \log p_t - \log p_{t-1}$
Plot the levels of all variables in one window. Give the plots sensible titles. Repeat the same for the series in differences.

2. Estimation of a structural VAR

You are interested to model the effects of Swiss Monetary policy. You consider the following four ($k = 4$) Swiss variables for your analysis: the consumer price index, the GDP in 1990 Swiss francs, the money stock M1 variable, and the quarterly average of three month Swiss franc LIBOR rate of interest.
The GAUSS code for this assignment sheet uses a procedure collection written for VAR estimation (`var_code.src`) by Paul Fackler. Include the source file in your program file.

Structural Vector Autoregression (SVAR) in primitive form

$Bx_t = \Gamma_0 + \Gamma_1 x_{t-1} + \varepsilon_t$

$$E(\varepsilon_t \varepsilon_\tau') = \begin{cases} D & \text{for} \quad t = \tau \\ 0 & \text{otherwise.} \end{cases}$$

$$\begin{aligned}
\tilde{m}_t &= \gamma_{10} - \phantom{b_{21}\tilde{m}_t -} b_{12}\tilde{r}_t - b_{13}\tilde{p}_t - b_{14}\tilde{y}_t + \gamma_{14}\tilde{m}_{t-1} + \gamma_{12}\tilde{r}_{t-1} + \gamma_{13}\tilde{p}_{t-1} + \gamma_{14}\tilde{y}_{t-1} + \varepsilon_{1t} \\
\tilde{r}_t &= \gamma_{20} - b_{21}\tilde{m}_t - \phantom{b_{12}\tilde{r}_t -} b_{23}\tilde{p}_t - b_{24}\tilde{y}_t + \gamma_{24}\tilde{m}_{t-1} + \gamma_{22}\tilde{r}_{t-1} + \gamma_{23}\tilde{p}_{t-1} + \gamma_{24}\tilde{y}_{t-1} + \varepsilon_{2t} \\
\tilde{p}_t &= \gamma_{30} - b_{31}\tilde{m}_t - b_{32}\tilde{r}_t - \phantom{b_{13}\tilde{p}_t -} b_{34}\tilde{y}_t + \gamma_{34}\tilde{m}_{t-1} + \gamma_{32}\tilde{r}_{t-1} + \gamma_{33}\tilde{p}_{t-1} + \gamma_{34}\tilde{y}_{t-1} + \varepsilon_{3t} \\
\tilde{y}_t &= \gamma_{40} - b_{41}\tilde{m}_t - b_{42}\tilde{r}_t - b_{43}\tilde{p}_t \phantom{- b_{14}\tilde{y}_t} + \gamma_{41}\tilde{m}_{t-1} + \gamma_{42}\tilde{r}_{t-1} + \gamma_{43}\tilde{p}_{t-1} + \gamma_{44}\tilde{y}_{t-1} + \varepsilon_{4t}
\end{aligned} \tag{1}$$

Writing the VAR in standard form "solves" the system

$x_t = B^{-1}\Gamma_0 + B^{-1}\Gamma_1 x_{t-1} + B^{-1}\varepsilon_t$
$x_t = A_0 + A_1 x_{t-1} + e_t$

$$\begin{aligned}
\tilde{m}_t &= a_{10} + a_{11}\tilde{m}_{t-1} + a_{12}\tilde{r}_{t-1} + a_{13}\tilde{p}_{t-1} + a_{14}\tilde{y}_{t-1} + e_{1t} \\
\tilde{r}_t &= a_{20} + a_{21}\tilde{m}_{t-1} + a_{22}\tilde{r}_{t-1} + a_{23}\tilde{p}_{t-1} + a_{24}\tilde{y}_{t-1} + e_{2t} \\
\tilde{p}_t &= a_{30} + a_{31}\tilde{m}_{t-1} + a_{32}\tilde{r}_{t-1} + a_{33}\tilde{p}_{t-1} + a_{34}\tilde{y}_{t-1} + e_{3t} \\
\tilde{y}_t &= a_{40} + a_{41}\tilde{m}_{t-1} + a_{42}\tilde{r}_{t-1} + a_{43}\tilde{p}_{t-1} + a_{44}\tilde{y}_{t-1} + e_{4t}
\end{aligned} \tag{2}$$

The innovations of a VAR in standard form are, by construction, contemporaneously correlated (composite innovations/shocks).
⇒ Estimate the VAR in standard form (see Equation (2)) using one lag and the procedure given in the source file:

```
ar = var(x,lags,z,nameout);
```

Read the documentation of the procedure code carefully. What are the input arguments and what are the output arguments? Which global settings do you need? Read out the vector containing the estimated constants and a matrix containing the estimated coefficients of the lags and print them to the output window. Make sure you know how to interpret the output matrices.

To obtain the idiosyncratic shocks (ε_t) from the composite shocks (e_t) we need the structural parameters, the matrix B

Covariance matrix of e_t:

$$\mathbb{E}(e_t e_t') = \Omega$$

relation between shocks: $e_t = B^{-1}\varepsilon_t$

$$\begin{aligned}
\mathbb{E}(e_t e_t') &= B^{-1}\mathbb{E}(\varepsilon_t \varepsilon_t') \left[B^{-1}\right]' \\
&= B^{-1} D \left[B^{-1}\right]'
\end{aligned}$$

To identify the structural parameters B, we decompose the variance covariance matrix of composite innovations (Cholesky-Decomposition of Ω). Note: Ordering of the variables is important!!

⇒ Compute estimates of B and D.

some maths:
For any real symmetric positive definite matrix Ω exists a lower triangular matrix A with ones along the principal diagonal and a unique diagonal matrix D with positive elements on the principal diagonal

$$\Omega \quad = \quad ADA' = PP'$$

i.e. $A = B^{-1}$
Ω: real symmetric positive definite matrix
A: lower triangular matrix with ones along the principal diagonal
D: diagonal matrix with positive elements

$$\Omega = ACC'A' \text{ with } C = \begin{bmatrix} \sqrt{d_1} & 0 & 0 & \dots & 0 \\ 0 & \sqrt{d_2} & 0 & \dots & 0 \\ 0 & 0 & \sqrt{d_3} & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & \sqrt{d_k} \end{bmatrix}$$

$$D = CC' = \begin{bmatrix} d_1 & 0 & 0 & \dots & 0 \\ 0 & d_2 & 0 & \dots & 0 \\ 0 & 0 & d_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & d_k \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ a_{21} & 1 & 0 & \dots & 0 \\ a_{31} & a_{32} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{k1} & a_{k2} & a_{k3} & \dots & 1 \end{bmatrix}$$

and consequently: $P = AC$ is a lower triangular matrix.

Get the matrices B and D from P!
GAUSS hint:
`load omega = varcv;`
either compute P by hand (look `chol` up in the reference!):
or compute p via the program code:
`{P_inv,P'} = reca0(omega,ind);`

Which restrictions are implied by the Cholesky decomposition? Why do we need the Cholesky decomposition at all?

10th GAUSS assignment ATS (can be handed in for grading!)

Impulse Response and Variance Decomposition

1. Loading Data and Data Description

You are interested to model the effects of Swiss Monetary policy. The data set data_var.txt contains the consumer price index, the GDP in 1990 Swiss francs, the money stock M1 variable, and the quarterly average of three month Swiss franc LIBOR rate of interest.

2. Estimating Vector Autoregression Models

The GAUSS code for this assignment sheet uses a procedure collection written for VAR estimation (`var_code.src`) by Paul Fackler. We estimate a VAR model with GAUSS. In the 7th assignment you find the model written in primitive and standard form.

1. First, use the procedure `var()` to estimate the VAR model. (This should be already done in the 7th assignment. Read the documentation of the procedure code to apply the correct global settings.) GAUSS hint: e.g. `ar = var(x,lags,z,nameout);`

2. Use the procedure `reca0()` to perform Cholesky decomposition of the innovation covariance matrix. GAUSS hint: e.g. `{p_inv,P'} = reca0(omega,ind);`

3. Produce impulse response functions

1. To compute impulse response functions you need the coefficient matrices of the VMA representation. Use the procedure `vma()` to back out the VMA parameters. Report the MA coefficient matrix for the first lag. GAUSS hint: e.g. `ma = vma(ar,lags,malags);` Denote `malags` as the number of lags used in the VMA. `malags` is 10. The last row of the `ma` matrix contains the constant. The first k rows contain the matrix of parameter estimates for the first lag. The second k rows contain the matrix of parameter estimates for the second lag. Etc...

2. In order to examine the effect of a shock in one variable on all variables in the VAR compute impulse response functions with the procedure `impulse()`. Plot your results in a comprehensive way, i.e. plot the response of one variable to its own shock and the shock in the other three variables.
GAUSS hint: e.g. `irf = impulse(ma,y_inv,outcode);`
Note: The `irf` collects in the rows the time and in the columns the IR function values. The first column corresponds to the responds of the first variable to a shock in the first variable. The second column corresponds to the responds in the second variable to a shock in the first variable. etc... The fifth column corresponds to the responds in the first variable to a shock in the second variable. The sixth column corresponds to the responds in the second variable to a shock in the second variable. etc...

3. Describe and compare patterns of impulse response functions by answering following questions: How big/small are the responses to shocks, i.e. the response of one variable to its own and shocks in the other variables? How persistent are these shocks? To

answer the questions make one plot for every variable containing it responses to the different shocks. Plot all four graphs into one window. In order to make some nice graphs, look the following options up in the User Guide:
`_plegstr, _pmcolor, _pcolor, _paxht, _pnumht, _plwidth, _pltype, _plegctl, _ptek`
Save the graph to your folder using `tkf2eps` (look it up in the language reference!)

4. Conduct a variance decomposition

1. Conduct a variance decomposition with the procedure `fedecomp` for each economic variable and plot your results in a comprehensive way.
   GAUSS hint: e.g. `varcom = fedecomp(irf,outcode);`
   Note: The `varcom` collects in the rows the time and in the columns the Variance Decomposition values. The first column corresponds to percent of the variance in the first variable due to the first variable. The second column corresponds to percent of the variance in the first variable due to the second variable. etc... The fifth column corresponds to percent of the variance in the second variable due to the first variable. The sixth column corresponds to percent of the variance in the second variable due to the second variable. Plot the variance decomposition of each variable into one graph and all four graphs into one window.

2. Discuss the results and draw a conclusion from the plots of the variance decomposition. In particular analyze how a shock in the GDP affects the other variables e.g. How is the proportion of the movements in a sequence due to its 'own' shocks versus shocks?? to the other variables?

11th GAUSS assignment ATS: Cointegration - Engle and Granger Method (can be handed in for grading!)

1. Loading Data and Data Description

Load the fmt-file data_abx.fmt. The file contains data on a Canadian Stock (ABX- Barrick Gold Corporation), which is traded on the Toronto Stock Exchange and the New York Stock Exchange simultaneously. The data includes the first two hours of trading, ranging from 1st January 2004 to 31st March 2004, sampled at 5 minute intervals.
The first column of the data matrix consist of the time series of the log home market (TSX) midquotes ((bid+ask)/2), the second contains the log foreign market (NYSE) midquotes, the log CAD/USD exchange rate is included in the third column, and in the last column you find a dummy variable indicating the first observation of a day (i.e., indicator = 1, if it is the first observation of the day, otherwise zero)

2. Preliminary Analysis

1. After loading the data into Gauss, convert the NYSE midquote series into Canadian dollars (note: we are talking about log series here!).
2. Plot the home market and the converted foreign market midquote series into one graph. Interpret the graph. Which cointegration vector would you suggest?

3. Engle-Granger Method, Step 1: Auxiliary Regression

1. In order to estimate the normalized cointegration vector, run a regression of the TSX midquotes (y_{1t}) on the NYSE midquotes (y_{2t}), back out the estimated parameter and the residual series. (If you are unsure whether or not to include a constant in the regression, have look again at the graph you've made above).
2. What is the estimated cointegration vector? What does *normalized* mean in this context?
3. Test the residual series for stationarity. Interpret and report the result.

4. Engle-Granger Method, Step 2: VECM Estimation

1. Create return series for the midquotes (i.e. first differences). Since we want to exclude overnight returns from our estimation, the first observation of each day has to be deleted. To do so, make use of the first day indicator and the *delif* comand (Look it up in the command reference). Note: You have to apply the same selection to the residual series to keep both series at the same dimension!
2. Estimate the VECM using the residual series as the error correction term. (i.e. the matrix of the two return series is the dependent variable matrix, the regressor matrix consists of ones (constant), the residual series and the first lag of the return series):

$$\begin{aligned} \Delta y_{1t} &= a_{10} + \gamma_1 \hat{\varepsilon}_{t-1} + a_{11}\Delta y_{1t-1} + a_{12}\Delta y_{2t-1} + u_{1t} \\ \Delta y_{2t} &= a_{20} + \gamma_2 \hat{\varepsilon}_{t-1} + a_{21}\Delta y_{1t-1} + a_{22}\Delta y_{2t-1} + u_{2t} \end{aligned} \quad (6)$$

3. Back out the estimated parameters. Interpret and report estimated equations of the VECM in Equation (6).
4. What could be a sensible interpretation of the correction terms coefficients (γ)?

12th GAUSS assignment ATS
(can be handed in for grading!)

1. Theoretical Background

Using US data on consumption C_t, investment I_t, and output, Y_t, we want to estimate the parameters of a Vector Error Correction model by Maximum Likelihood. Assuming that three variables follow a VECM of order one that can be written as:

$$\Delta y_t = \alpha + \zeta_0 y_{t-1} + \zeta_1 \Delta y_{t-1} + \varepsilon_t,$$

where $y_t = \begin{pmatrix} C_t \\ I_t \\ Y_t \end{pmatrix}$, ζ_1 is a (3×3) parameter matrix, α a (3×1) vector of constants and

$\zeta_0 = -BA'$.

We know that there exist two cointegration relations between those three variables, and therefore B and A are (3×2) matrices, where B contains the adjustment coefficients and A denotes the cointegration matrix, which if normalized as proposed by Phillips (see Hamilton page 576) is of the form :

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ a_{31} & a_{32} \end{pmatrix}$$

The disturbances ε_t are Gaussian and the conditional log-likelihood function is:
$\log L(\Omega, \zeta_1, \alpha, \zeta_0)$

$$= (-\tfrac{Tn}{2}) \ln(2\pi) - (\tfrac{T}{2}) \ln|\Omega|$$
$$-\tfrac{1}{2} \sum_{t=1}^{T} [(\Delta y_t - \zeta_1 \Delta y_{t-1} - \alpha - \zeta_0 y_{t-1})' \Omega^{-1} (\Delta y_t - \zeta_1 \Delta y_{t-1} - \alpha - \zeta_0 y_{t-1})]$$

where Ω denotes the covariance matrix of the error terms, n is the number of variables and T the number of observations.

2. Data Processing

1. Load the data contained in ciy.txt into Gauss. The file contains quarterly US data for the period 1947Q1 - 1988Q4 for the three variables log consumption (column one), log investment (column two), and log output (column three), the number of observations for each variable is 168.

2. Generate the following four matrices: the first contains the three variables as they have been read in from the data file, the second includes the lag of the three variables, the third the first differences, and the fourth the first lag of the differences. Make sure that there are no missings in any of these matrices and that all have the same dimensions.

3. Computing the log-likelihood Function

We will estimate the system by the Gauss procedure cml and therefore we need a procedure that returns the log-likelihood contribution of each observation. Before you start, please read all points listed below and work through them step by step.

1. Read in the vector contained in x1.fmt. It contains the starting values for $x0$.

2. All parameters needed for the log-likelihood will be passed into the procedure by a column vector. Consequently, we have to read out the respective elements from this column vector and reshape them into their former dimensions. The vector of parameters is assumed to be structured as follows:

$$x0 = \begin{pmatrix} \zeta_{11} \\ \zeta_{12} \\ \zeta_{13} \\ \zeta_{21} \\ \zeta_{22} \\ \zeta_{23} \\ \zeta_{31} \\ \zeta_{32} \\ \zeta_{33} \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \omega_{11} \\ \omega_{22} \\ \omega_{33} \\ \omega_{21} \\ \omega_{31} \\ \omega_{32} \\ a_{31} \\ a_{32} \\ b_{11} \\ b_{21} \\ b_{31} \\ b_{12} \\ b_{22} \\ b_{32} \end{pmatrix}$$

Read out the elements of ζ_1 and reshape them into a (3×3) matrix.
Next build a column vector that contains the constants $(\alpha_1, \alpha_2, \alpha_3)'$
Then rebuild the normalized cointegration matrix A and the matrix B by addressing the according elements in the vector.

As far as the covariance matrix Ω is concerned, we will use a little trick to make sure that the covariance matrix will always stay positive definite. We define: $\Omega = PP'$ (Cholesky decomposition). Build a matrix P as given below:

$$P = \begin{pmatrix} \omega_{11} & 0 & 0 \\ \omega_{21} & \omega_{22} & 0 \\ \omega_{31} & \omega_{32} & \omega_{33} \end{pmatrix}$$

and get Ω.

3. Compute a column vector that contains the log-likelihood contribution of each observation, i.e. compute the log-likelihood function.

4. Write a procedure around the reshaping commands and the log-likelihood. Output argument is the value of the log-likelihood function. Input arguments are the parameter vector $x0$ and the data set (in our case we pass the data into the procedure via globals, but cml demands those two input arguments). Do not forget to define local variables in the likelihood procedure.

4. CML Estimation

1. Use for staring values $x0$ the vector contained in x1.fmt.

2. Further use the following options in cml:

```
cmlset;
_cml_GridSearch=1;
_cml_algorithm=3;
_cml_linesearch=2;
_cml_covpar=2;
_cml_GradMethod=1;
_cml_DirTol=1e-4;
__output=1;
```

   Since we estimate 26 parameters it might be of help to define labels for the estimated coefficients. This can be done by:

```
_cml_parnames=
"zeta11"|"zeta12"|"zeta13"|
"zeta21"|"zeta22"|"zeta23"|
"zeta31"|"zeta32"|"zeta33"|
"const1"|"const2"|"const3"|
"omega11"|"omega22"|"omega33"|
"omega12"|"omega31"|"omega32"|
"a31"|"a32"|
"b11"|"b21"|"b31"|
"b12"|"b22"|"b32";
```

3. Call cml and estimate the parameters. Report the estimated parameters and interpret your results.

5. Likelihood Ratio Test

1. Estimate the unrestricted model, i.e. ζ_0 is no more equal $-BA'$, but simply a (3×3) parameter matrix. Report the estimated parameters. The starting values are contained in x2.fmt in the following order:

$$x0 = \begin{pmatrix} \zeta_{11}^1 \\ \zeta_{12}^1 \\ \zeta_{13}^1 \\ \zeta_{21}^1 \\ \zeta_{22}^1 \\ \zeta_{23}^1 \\ \zeta_{31}^1 \\ \zeta_{32}^1 \\ \zeta_{33}^1 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \omega_{11} \\ \omega_{22} \\ \omega_{33} \\ \omega_{21} \\ \omega_{31} \\ \omega_{32} \\ \zeta_{11}^0 \\ \zeta_{12}^0 \\ \zeta_{13}^0 \\ \zeta_{21}^0 \\ \zeta_{22}^0 \\ \zeta_{23}^0 \\ \zeta_{31}^0 \\ \zeta_{32}^0 \\ \zeta_{33}^0 \end{pmatrix}$$

Use the following cml settings:

```
cmlset;
_cml_GridSearch=1;
_cml_algorithm=1;
_cml_linesearch=2;
_cml_covpar=2;
_cml_GradMethod=1;
_cml_DirTol=1e-4;
__output=1;
```

2. Conduct a Likelihood Ratio Test: $LR = 2(L(\hat{\theta}) - L(\tilde{\theta}))$

   where $L(\hat{\theta})$ denotes the the value of the log-likelihood function at the unrestricted estimates and $L(\tilde{\theta})$ the value of the log-likelihood function at the restricted estimates (Note: cml returns the mean log-likelihood, so it has to be multiplied by T to obtain $L(\hat{\theta})$ and $L(\tilde{\theta})$.) The test statistic's distribution does not follow a standard distribution, but requires simulation. Hamilton's Table B.10 on page 767 contains the quantiles of the simulated distribution. (Here we need a 5% level, case 2, $g = n - h = 3 - 2 = 1$, critical value = 8.083) Can we reject the restrictions? Report and interpret your result.

13th GAUSS assignment ATS (cannot be handed in for grading!)

Identifying a structural VECM model with long-run and short-run restrictions

1. Preliminaries

We are revisiting Beaudry and Lucke (2009)'s study "Letting different views about business cycles compete" to analyze the US business cycle. We are interested which effect total factor productivity (TFP), investment specific technology (IST), asset prices and the interest rate have on hours worked. The data set NIPA_h.txt contains quarterly data from 1955:1 to 2007:2 with log TFP, log of the real price of investment goods, log per capita S&P500 stock prices index, log hours per capita worked and the log of the Federal Funds nominal interest rate.

1.1 Load and organize the data. GAUSS hint:

```
load data[]= NIPA_h.txt; data2 = reshape(data, 210,5);
```

The econometric model is a structural VAR

$$\mathbf{y}_t = \mathbf{c} + \mathbf{\Phi}_1 \mathbf{y}_{t-1} + \mathbf{\Phi}_2 \mathbf{y}_{t-2} + \ldots + \mathbf{\Phi}_p \mathbf{y}_{t-p} + \mathbf{W}\mathbf{u}_t$$

where

$$\mathbf{y}_t = \begin{bmatrix} y_{1t} \\ y_{2t} \\ y_{3t} \\ y_{4t} \\ y_{5t} \end{bmatrix} = \begin{bmatrix} \text{total factor productivity}_t \\ \text{real price of investment goods}_t \\ \text{stock prices}_t \\ \text{hours worked}_t \\ \text{Federal Funds interest rate}_t \end{bmatrix}.$$

The structural innovations $\mathbf{u}_t$ are assumed to be uncorrelated and have standardized variances, so that $\mathbb{E}(\mathbf{u}_t \mathbf{u}_t') = \mathbf{I_n}$.
In order to identify the structural innovations via the $\mathbf{W}$ matrix, Beaudry and Lucke (2009) make the following assumptions:

(1) only TFP shocks can have a contemporary effect on TFP

(2) monetary shocks do not have a contemporaneous effect on hours

(3) IST, hours and monetary shocks have no long run effect on TFP

(4) hours and monetary shocks have no long run effect on the real price of investment

We use the 2step procedure described in the lecture to estimate the structural model.

2. Estimating the reduced form VECM

Since the variables are cointegrated, the parameters are appropriately estimated in a VECM framework. As in Beaudry and Lucke (2009), estimate the model with 6 lags and 3 linearly independent cointegrating relations!

2.1 Estimate the reduced form VECM

$$\Delta \mathbf{y}_t = \mathbf{c} + \mathbf{B}\mathbf{A}'\mathbf{y}_{t-1} + \boldsymbol{\zeta}_1 \Delta \mathbf{y}_{t-1} + \cdots + \boldsymbol{\zeta}_{p-1}\mathbf{y}_{t-p+1} + \boldsymbol{\varepsilon}_t$$

GAUSS hint: use the procedure *SVECM_rFORM_estimation*

$(\hat{\mathbf{B}},\ \hat{\underline{\mathrm{A}}},\ \hat{\boldsymbol{\zeta}}_0,\ \hat{\boldsymbol{\Omega}},\ \hat{\boldsymbol{\zeta}},\ \hat{\boldsymbol{\Psi}}) =$ red_form_estimation(l, r, data)

INPUT: data is an $(T \times n)$ matrix of variables, l is the laglength and r is the cointegrating rank

OUTPUT:

$\hat{\mathbf{B}}$: $(n \times r)$ matrix of adjustment coefficient

$\hat{\mathbf{A}}$: $(n \times r)$ matrix of linearly independent cointegrating relations

$\hat{\boldsymbol{\zeta}}_0 = \hat{\mathbf{B}}\hat{\mathbf{A}}'$

$\hat{\boldsymbol{\zeta}} = [\hat{\boldsymbol{\zeta}}_1, \hat{\boldsymbol{\zeta}}_2, \cdots, \hat{\boldsymbol{\zeta}}_{p-1}, \hat{\mathbf{c}}]'$, the remaining VECM parameters

$\hat{\boldsymbol{\Psi}} = \hat{\mathbf{A}}_\perp(\hat{\mathbf{B}}'_\perp(\mathbf{I}_n - \sum_{i=1}^{p-1}\hat{\boldsymbol{\zeta}}_{\mathbf{i}})\hat{\mathbf{A}}_\perp)^{-1}\hat{\mathbf{B}}'_\perp$ is the long run impact matrix of the moving average representation of the VECM

2.1 Look at your results. What is the rank of $\hat{\boldsymbol{\Psi}}$, what is the rank of $\hat{\boldsymbol{\zeta}}_0$? What should hold for $\hat{\mathbf{A}}'\hat{\boldsymbol{\Psi}}$? Check and explain intuitively!

3. Implement the identifying restrictions

3.1 Estimate $\mathbf{W}$ using the maximum likelihood method. Therefore, maximize the concentrated log likelihood function

$$lnL(\mathbf{W}) = \text{constant} - \frac{T}{2}|\mathbf{W}|^2 - \frac{T}{2}\text{trace}(\mathbf{W}'^{-1}\mathbf{W}^{-1}\hat{\boldsymbol{\Omega}}) \tag{7}$$

subject to the restrictions implied by the assumptions (1) - (4).

Hint:

$$\mathbf{W} = \begin{bmatrix} * & 0 & 0 & 0 & 0 \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & 0 \\ * & * & * & * & * \end{bmatrix} \quad \text{and} \quad \mathbf{L} = \boldsymbol{\Psi}\mathbf{W} = \begin{bmatrix} * & 0 & * & 0 & 0 \\ * & * & * & 0 & 0 \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}$$

GAUSS hint:

Use the Gauss procedure CML and implement the long-run restrictions with the NONLINEAR EQUALITY option.

First build a procedure that returns the value of the concentrated likelihood function. Input should be the 20 free parameters of the $\mathbf{W}$-matrix in vector form, $\mathbf{W}_f = [a_{11}, a_{21}, a_{22}, \cdots, a_{55}]'$. In the procedure, construct the $\mathbf{W}$-matrix from these parameters. For the likelihood function, use the estimated covariance matrix $\hat{\mathbf{\Omega}}$ as a global variable. You do not need additional data.

To implement the long-run restrictions, use the CML option EqProc. The option allows you to solve a constraint maximization problem.

CML Guide:

Nonlinear equality constraints are of the form:

$$G(\Theta) = 0$$

where Θ is the vector of parameters, and $G(\Theta)$ is an arbitrary, user-supplied function. Nonlinear equality constraints are specified by assigning the pointer to the user-supplied function to the **GAUSS** global, **_cml_EqProc**.

For example, suppose you wish to constrain the product of two parameters to be equal to 1:

```
proc eqp(b);
retp(b[1]*b[2]-1);
endp;
_cml_EqProc = &eqp;
```

In our case, you will have to build a procedure that specifies those elements of the $[\mathbf{\Psi W}]$ matrix that are zero. Give this procedure the name *eqproc*. Read in $\mathbf{W}_f$ and build the $\mathbf{W}$-matrix as above. Compute $\mathbf{L} = \mathbf{\Psi W}$ using the estimated $\hat{\mathbf{\Psi}}$ matrix as a global variable. Return a vector containing the elements of $\mathbf{L}$ that are zero, e.g.

retp($\mathbf{L}[1,1] \mid \mathbf{L}[1,4] \mid \cdots \mid \mathbf{L}[2,5]$);

Before estimation, set the following globals:

_cml_Algorithm = 4;

_cml_LineSearch = 2;

_cml_DirTol = 1e-10;

_cml_CovPar = 2;

_cml_switch = { 3 3, 1e-9 1e-9, 20 20, 1e-6 1e-6};

_cml_NumObs = rows(data2)-6; @ number of observations @

_cml_EqProc = &eqproc;

Choose appropriate starting values and make sure you define both procedures and the globals before calling CML!

3.2 Compute the $\hat{\mathbf{W}}$-matrix. How can you check if convergence was successful?

3.3 $\mathbf{W}$ is only identified locally. For example, for a given estimate $\hat{\mathbf{W}}$, $-\hat{\mathbf{W}}$ will also be a solution to the maximization problem, equation (1). Generally, we can reverse the signs of each column of $\hat{\mathbf{W}}$ without changing the likelihood. Since we want positive elements on the main diagonal, reverse the sign for each column of $\hat{\mathbf{W}}$ that has a negative main diagonal element.

Example:

$$\hat{\mathbf{W}} = \begin{bmatrix} a & x & \cdots \\ -b & -y & \cdots \\ c & -z & \cdots \\ -d & v & \cdots \\ -e & -w & \cdots \end{bmatrix} \Rightarrow \begin{bmatrix} a & -x & \cdots \\ -b & y & \cdots \\ c & z & \cdots \\ -d & -v & \cdots \\ -e & w & \cdots \end{bmatrix}$$

4. Interpret the results

4.1 Plot the resulting Impulse Response Functions and Variance Decompositions and interpret them. Compare the results to those of Beaudry and Lucke (2009), page 15 (identification scheme ID2).

Gauss hint: Define the following globals:

l = # of lags in the VAR

A = $\hat{\mathbf{W}}$

vd_plot = 0 or 1 (if 1, variance decomposition will be plotted, else IRFs)

malags = # of lags for IRFs (32 in the paper)

gammas = $\hat{\boldsymbol{\zeta}}$

PImat = $\hat{\boldsymbol{\zeta}}_0$

After that, include the pre-written source file `results_plot.src`.