Eryk Wdowiak
11 Nov 2024

simple illustration of a non-stationary residual
and comparison of estimation with OLS and AR

linear model:
y[i] = alpha + beta * x[i] + res[i]

with non-stationary residual,
which follows an autoregressive process:
res[i] = rho * res[i-1] + wn[i]

where:
-1 < rho < 1

for simplicity, we'll set the true values to:
alpha = 0
beta = 1

and we'll examine two "symmetrically opposite" paths of the residual:

- scenario A -- "boom then bust"
- scenario B -- "bust then boom"

```
(%i1) load(draw)$
```

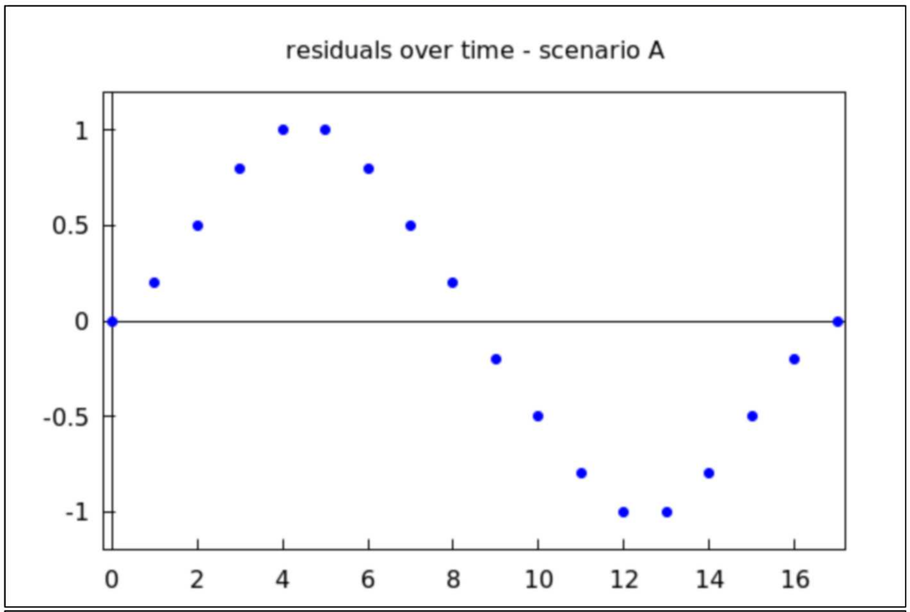
loadfile: failed to load /usr/share/maxima/5.42.1/share/draw/draw.lisp
-- an error. To debug this try: debugmode(true);

first, we'll generate some data where
Y is a function of X and an autoregressive residual

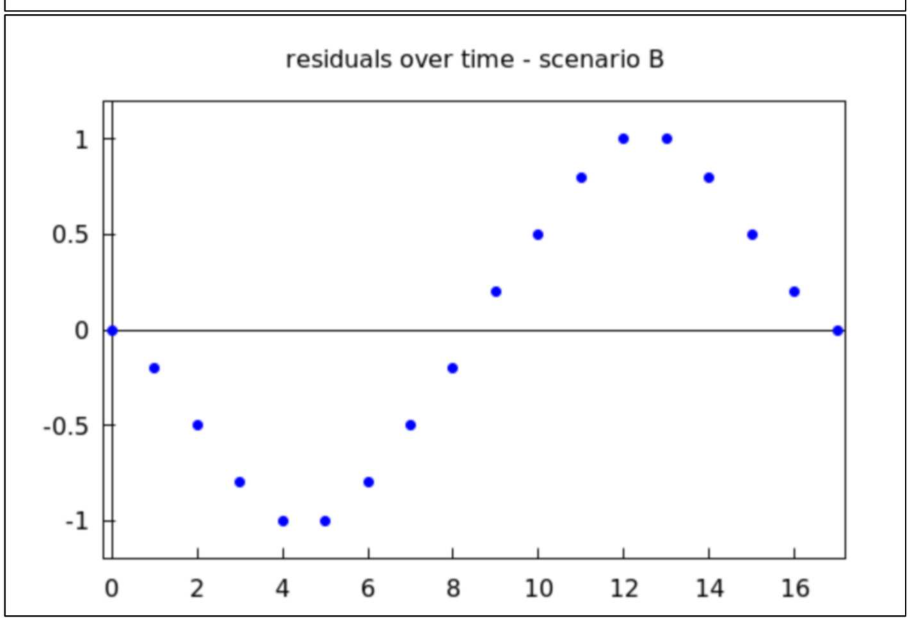
```
(%i7) time : [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17]$
xx : [0,−8,−7,−6,−5,−4,−3,−2,−1, 0, 1, 2, 3, 4, 5,6,7,8]$
res_sca : [0, 0.2,0.5, 0.8,1,1,0.8,0.5,0.2,−0.2,−0.5,−0.8,−1,−1,−0.8,−0.5,−0.2, 0]$
res_scb : −1 · res_sca$
time_error_sca : transpose (matrix (time, res_sca))$
time_error_scb : transpose (matrix (time, res_scb))$
```

```
(%i9) wxdraw2d(
  title="residuals over time − scenario A",
  point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
  points(time_error_sca),
  xrange=[−0.2,17.2],   yrange=[−1.2,1.2])$
wxdraw2d(
  title="residuals over time − scenario B",
  point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
  points(time_error_scb),
  xrange=[−0.2,17.2],   yrange=[−1.2,1.2])$
```

(%t8)



(%t9)

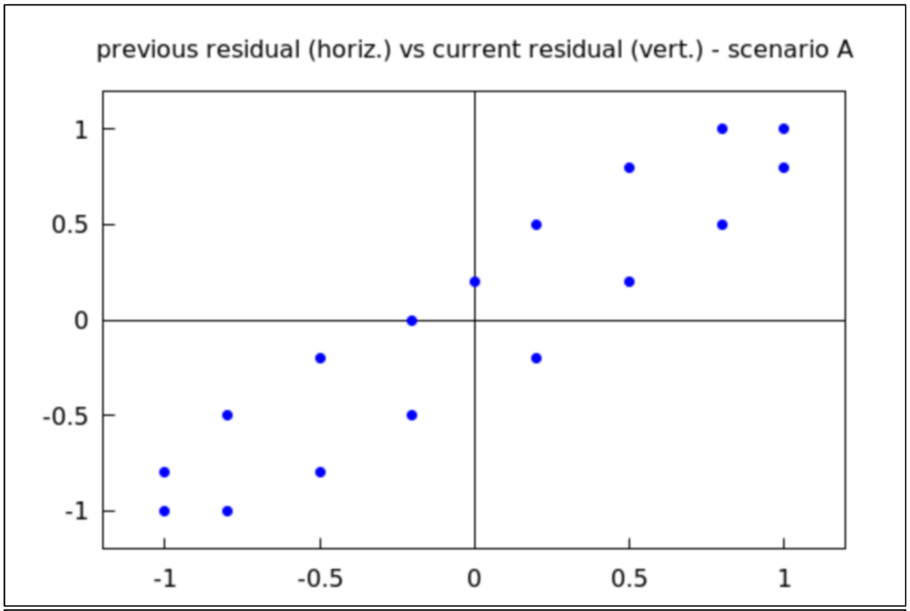


the serial correlation of the residuals is a violation of Gauss-Markov

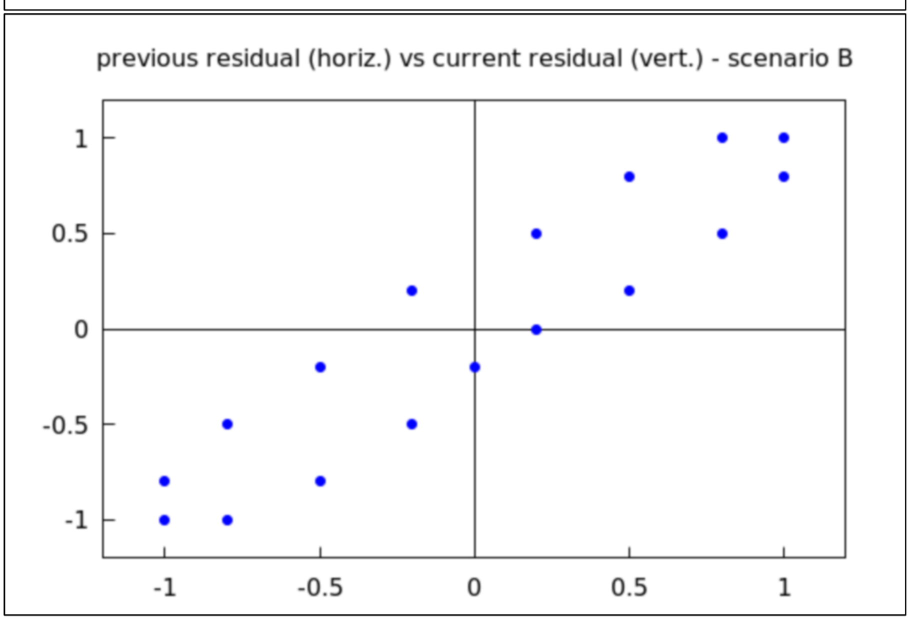
```
(%i11) scatter_res_sca: apply (matrix, makelist ([res_sca[i−1], res_sca[i]], i, 2, length(xx)))$
scatter_res_scb: apply (matrix, makelist ([res_scb[i−1], res_scb[i]], i, 2, length(xx)))$
```

```
(%i13) wxdraw2d(
    title="previous residual (horiz.) vs current residual (vert.) − scenario A",
    point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
    points(scatter_res_sca),
    xrange=[−1.2,1.2],   yrange=[−1.2,1.2])$
  wxdraw2d(
    title="previous residual (horiz.) vs current residual (vert.) − scenario B",
    point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
    points(scatter_res_scb),
    xrange=[−1.2,1.2],   yrange=[−1.2,1.2])$
```

(%t12)



(%t13)

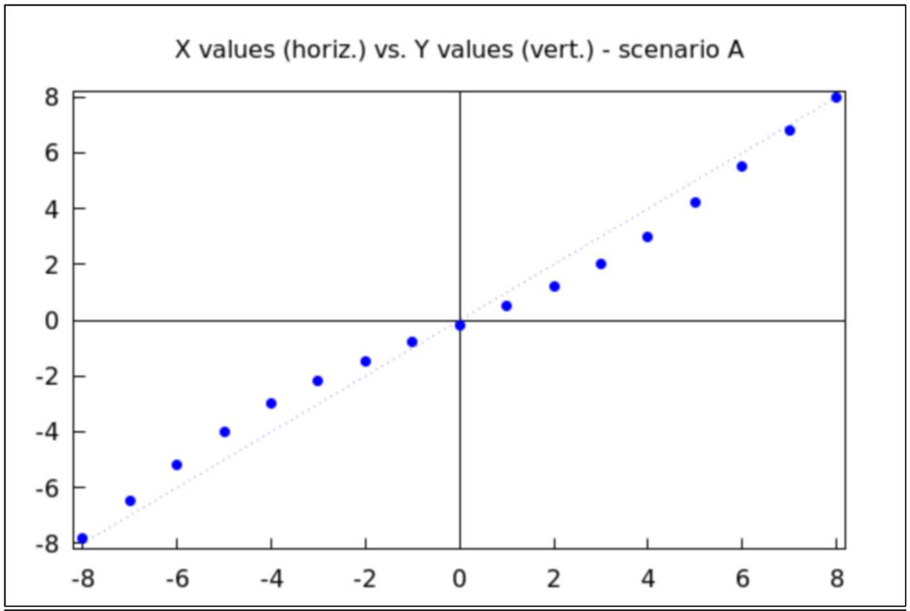


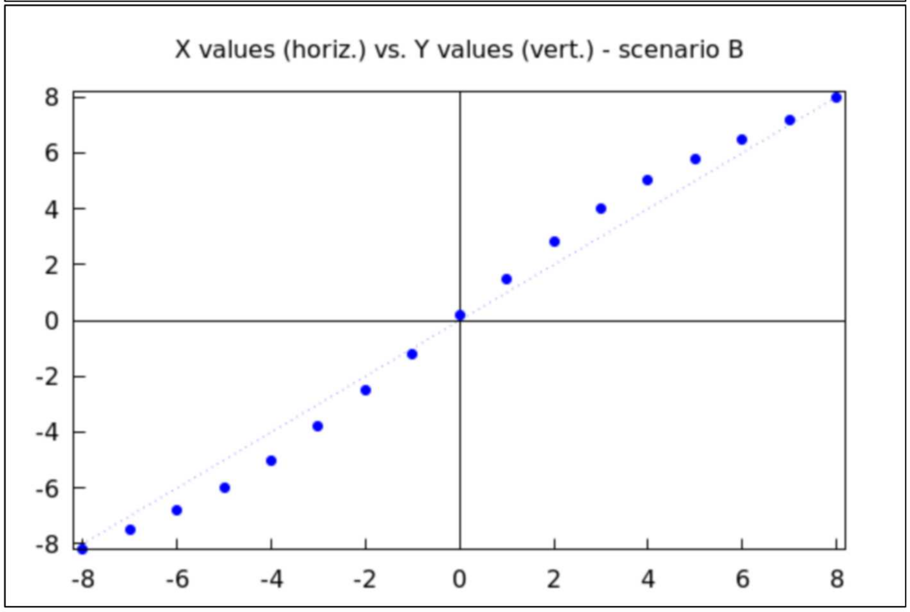
now that we have generated autoregressive residuals
let's compute the observed values of Y:
y[i] = alpha + beta * x[i] + res[i]

for simplicity:
alpha = 0
beta = 1
so we can just add the residuals to X in each scenario

```
(%i17) yy_sca : xx + res_sca $
yy_scb : xx + res_scb $
scatter_sca: apply (matrix, makelist ([xx[i], yy_sca[i]], i, 2, length(xx)))$
scatter_scb: apply (matrix, makelist ([xx[i], yy_scb[i]], i, 2, length(xx)))$
```

```
(%i19) wxdraw2d(
    title="X values (horiz.) vs. Y values (vert.) − scenario A",
    point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
    points(scatter_sca),
    line_type=dots, explicit(x,x,−8,8),
    xrange=[−8.2,8.2],   yrange=[−8.2,8.2])$
  wxdraw2d(
    title="X values (horiz.) vs. Y values (vert.) − scenario B",
    point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
    points(scatter_scb),
    line_type=dots, explicit(x,x,−8,8),
    xrange=[−8.2,8.2],   yrange=[−8.2,8.2])$
```

(%t18)


(%t19)


now that we have generated some data

let's estimate the model parameters with OLS
and add the regression lines to the scatterplots

```
(%i31) ols_sca(alpha,beta) := sum( (yy_sca[i] − alpha − beta · xx[i] )^2 , i , 2, length(xx) );
ols_scb(alpha,beta) := sum( (yy_scb[i] − alpha − beta · xx[i] )^2 , i , 2, length(xx) );
sol_ols_sca:lbfgs(ols_sca(alpha,beta),'[alpha,beta],[0.01,1.0],0.0001,[−1,0])$
sol_ols_scb:lbfgs(ols_scb(alpha,beta),'[alpha,beta],[0.01,1.0],0.0001,[−1,0])$
beta_sca : subst(sol_ols_sca[2],beta)$
beta_scb : subst(sol_ols_scb[2],beta)$
print("")$
print("OLS estimates in scenario A:")$
print(sol_ols_sca)$
print("")$
print("OLS estimates in scenario B:")$
print(sol_ols_scb)$
```

(%o20) $$\mathrm{ols\_sca}(\alpha,\beta) := \sum_{i=2}^{\mathrm{length}(xx)} \left(yy\_sca_i - \alpha + (-\beta)\, xx_i\right)^2$$

(%o21) $$\mathrm{ols\_scb}(\alpha,\beta) := \sum_{i=2}^{\mathrm{length}(xx)} \left(yy\_scb_i - \alpha + (-\beta)\, xx_i\right)^2$$

OLS estimates in scenario A:

$$[\alpha = -4.345482307321902\ 10^{-16}, \beta = 0.9019607843137253]$$

OLS estimates in scenario B:

$$[\alpha = -1.353084311261909\ 10^{-16}, \beta = 1.098039215686274]$$

```
(%i34) print("")$
       print("scenario A − OLS underestimates beta: ", 1.0 · beta_sca)$
       print("scenario B − OLS overestimates beta: ", 1.0 · beta_scb)$
```

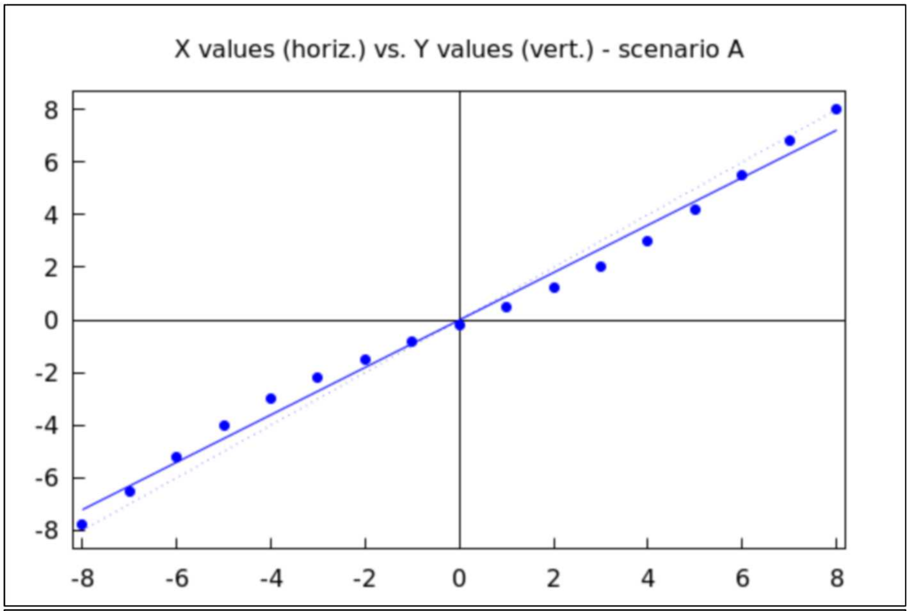
scenario A − OLS underestimates beta: 0.9019607843137253
scenario B − OLS overestimates beta: 1.098039215686274

now let's draw the same scatterplots as above, but with the OLS regression lines
so that we can visually see:

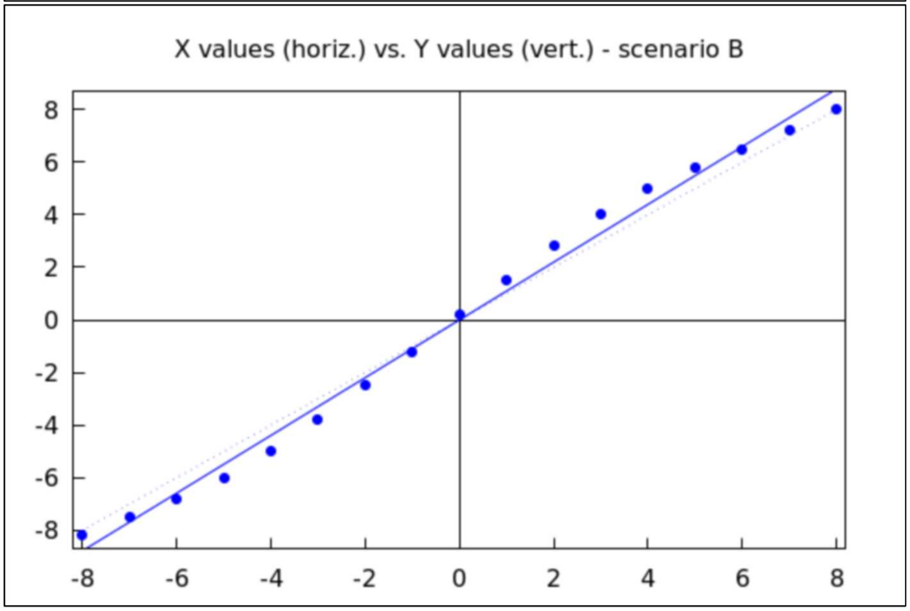
- the underestimated slope (scenario A)
- the overestimated slope (scenario B)

```
(%i36) wxdraw2d(
    title="X values (horiz.) vs. Y values (vert.) − scenario A",
    explicit( beta_sca·x , x , −8, 8 ),
    point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
    points(scatter_sca),
    line_type=dots, explicit(x,x,−8,8),
    xrange=[−8.2,8.2],   yrange=[−8.7,8.7])$
  wxdraw2d(
    title="X values (horiz.) vs. Y values (vert.) − scenario B",
    explicit( beta_scb·x , x , −8, 8 ),
    point_type=filled_circle,   xaxis=true,yaxis=true,xaxis_type=solid,yaxis_type=solic
    points(scatter_scb),
    line_type=dots, explicit(x,x,−8,8),
    xrange=[−8.2,8.2],   yrange=[−8.7,8.7])$
```

(%t35)



(%t36)



now let's account for the non-stationarity

we'll assume that the residual follows an autoregressive process:
res[i] = rho * res[i-1] + wn[i]

and estimate the model parameters with the following model:

y[i] = alpha + beta * x[i] + res[i]
y[i] = alpha + beta * x[i] + rho * res[i-1] + wn[i]
y[i] = alpha + beta * x[i] + rho * (y[i-1] - alpha + beta * x[i-1]) + wn[i]

```
(%i46) ar_sca(alpha,beta,rho) := sum( (yy_sca[i] − alpha − beta · xx[i] − rho·( yy_sca[i−1]
       ar_scb(alpha,beta,rho) := sum( (yy_scb[i] − alpha − beta · xx[i] − rho·( yy_scb[i−1]
       sol_ar_sca:lbfgs(ar_sca(alpha,beta,rho),'[alpha,beta,rho],[0.01,1.0,0.5],0.0001,[−1,0
       sol_ar_scb:lbfgs(ar_scb(alpha,beta,rho),'[alpha,beta,rho],[0.01,1.0,0.5],0.0001,[−1,0
       print("")$
       print("parameter estimates in scenario A:")$
       print(sol_ar_sca)$
       print("")$
       print("parameter estimates in scenario B:")$
       print(sol_ar_scb)$
```

(%o37) $$ar\_sca(\alpha,\beta,\rho):=\sum_{i=2}^{length(xx)} (yy\_sca_i-\alpha+(-\beta)\, xx_i+(-\rho)\,(yy\_sca_{i-1}-\alpha+(-\beta)\, xx$$

(%o38) $$ar\_scb(\alpha,\beta,\rho):=\sum_{i=2}^{length(xx)} (yy\_scb_i-\alpha+(-\beta)\, xx_i+(-\rho)\,(yy\_scb_{i-1}-\alpha+(-\beta)\, xx$$

parameter estimates in scenario A:
$[\alpha=0.1228337268250304,\beta=0.9750616243067297,\rho=0.912784747086664 2]$

parameter estimates in scenario B:
$[\alpha=-0.1228354305786306,\beta=1.024938424098319,\rho=0.912786327319693]$

finally, let's compare the estimates of beta from the OLS models to those from the AR models

```
(%i60) beta_hat_sca:subst(sol_ar_sca[2],beta)$
beta_hat_scb:subst(sol_ar_scb[2],beta)$
print("")$
print("scenario A")$
print("OLS model estimate of beta: ", 1.0 · beta_sca)$
print("AR model estimate of beta: ", beta_hat_sca)$
print("")$
print("scenario B")$
print("OLS model estimate of beta: ", 1.0 · beta_scb)$
print("AR model estimate of beta: ", beta_hat_scb)$
print("")$
print("Notice that accounting for the non−stationary residual")$
print("pushes the estimates of beta towards the true value of 1")$
print("")$
```

scenario A
OLS model estimate of beta: 0.9019607843137253
AR model estimate of beta: 0.9750616243067297

scenario B
OLS model estimate of beta: 1.098039215686274
AR model estimate of beta: 1.024938424098319

Notice that accounting for the non−stationary residual
pushes the estimates of beta towards the true value of 1

using maximum likelihood, we can estimate the parameters
and compute the standard errors of our estimates

```
(%i73) N : 17$

ll_ols_sca(alpha,beta,gamma):= −(N/2)·log(gamma) − (N/2)·log(2·%pi) − (1/(2·gamr
ll_ols_scb(alpha,beta,gamma):= −(N/2)·log(gamma) − (N/2)·log(2·%pi) − (1/(2·gamr
ll_ar_sca(alpha,beta,rho,gamma):= −(N/2)·log(gamma) − (N/2)·log(2·%pi) − (1/(2·ga
ll_ar_scb(alpha,beta,rho,gamma):= −(N/2)·log(gamma) − (N/2)·log(2·%pi) − (1/(2·ga

define( info_ols_sca(alpha,beta,gamma) , −1· invert(hessian( ll_ols_sca(alpha,beta,ga
define( info_ols_scb(alpha,beta,gamma) , −1· invert(hessian( ll_ols_scb(alpha,beta,ga
define( info_ar_sca(alpha,beta,rho,gamma) , −1· invert(hessian( ll_ar_sca(alpha,beta
define( info_ar_scb(alpha,beta,rho,gamma) , −1· invert(hessian( ll_ar_scb(alpha,beta

sol_ll_ols_sca : lbfgs( −ll_ols_sca(alpha,beta,gamma) , '[alpha,beta,gamma], [0.01,0.
sol_ll_ols_scb : lbfgs( −ll_ols_scb(alpha,beta,gamma) , '[alpha,beta,gamma], [0.01,0.
sol_ll_ar_sca : lbfgs( −ll_ar_sca(alpha,beta,rho,gamma) , '[alpha,beta,rho,gamma], [0
sol_ll_ar_scb : lbfgs( −ll_ar_scb(alpha,beta,rho,gamma) , '[alpha,beta,rho,gamma], [0

(%i87) alpha_ols_sca : subst(sol_ll_ols_sca[1],alpha)$
beta_ols_sca : subst(sol_ll_ols_sca[2],beta)$
gamma_ols_sca : subst(sol_ll_ols_sca[3],gamma)$

alpha_ols_scb : subst(sol_ll_ols_scb[1],alpha)$
beta_ols_scb : subst(sol_ll_ols_scb[2],beta)$
gamma_ols_scb : subst(sol_ll_ols_scb[3],gamma)$

alpha_ar_sca : subst(sol_ll_ar_sca[1],alpha)$
beta_ar_sca : subst(sol_ll_ar_sca[2],beta)$
rho_ar_sca : subst(sol_ll_ar_sca[3],rho)$
gamma_ar_sca : subst(sol_ll_ar_sca[4],gamma)$

alpha_ar_scb : subst(sol_ll_ar_scb[1],alpha)$
beta_ar_scb : subst(sol_ll_ar_scb[2],beta)$
rho_ar_scb : subst(sol_ll_ar_scb[3],rho)$
gamma_ar_scb : subst(sol_ll_ar_scb[4],gamma)$
```

```
(%i101) sea_ols_sca : sqrt( info_ols_sca(alpha_ols_sca,beta_ols_sca,gamma_ols_sca)[1,1] )$
        seb_ols_sca : sqrt( info_ols_sca(alpha_ols_sca,beta_ols_sca,gamma_ols_sca)[2,2] )$
        seg_ols_sca : sqrt( info_ols_sca(alpha_ols_sca,beta_ols_sca,gamma_ols_sca)[3,3] )$

        sea_ols_scb : sqrt( info_ols_scb(alpha_ols_scb,beta_ols_scb,gamma_ols_scb)[1,1] )$
        seb_ols_scb : sqrt( info_ols_scb(alpha_ols_scb,beta_ols_scb,gamma_ols_scb)[2,2] )$
        seg_ols_scb : sqrt( info_ols_scb(alpha_ols_scb,beta_ols_scb,gamma_ols_scb)[3,3] )$

        sea_ar_sca : sqrt( info_ar_sca(alpha_ar_sca,beta_ar_sca,rho_ar_sca,gamma_ar_sca)[1
        seb_ar_sca : sqrt( info_ar_sca(alpha_ar_sca,beta_ar_sca,rho_ar_sca,gamma_ar_sca)[2
        ser_ar_sca : sqrt( info_ar_sca(alpha_ar_sca,beta_ar_sca,rho_ar_sca,gamma_ar_sca)[3
        seg_ar_sca : sqrt( info_ar_sca(alpha_ar_sca,beta_ar_sca,rho_ar_sca,gamma_ar_sca)[4

        sea_ar_scb : sqrt( info_ar_scb(alpha_ar_scb,beta_ar_scb,rho_ar_scb,gamma_ar_scb)[1
        seb_ar_scb : sqrt( info_ar_scb(alpha_ar_scb,beta_ar_scb,rho_ar_scb,gamma_ar_scb)[2
        ser_ar_scb : sqrt( info_ar_scb(alpha_ar_scb,beta_ar_scb,rho_ar_scb,gamma_ar_scb)[3
        seg_ar_scb : sqrt( info_ar_scb(alpha_ar_scb,beta_ar_scb,rho_ar_scb,gamma_ar_scb)[4
```

```
(%i117) print("")$
        print("OLS:  y[i] =",alpha,"+",beta,"· x[i] + res[i]")$
        print(" AR:  y[i] =",alpha,"+",beta,"· x[i] + ",rho,"· ( y[i−1] − ",alpha,"+",beta,"· x[i−
        print("")$
        print("scenario A − OLS estimates")$
        print(alpha, ": ",alpha_ols_sca," se: ",sea_ols_sca)$
        print(beta, ": ",beta_ols_sca," se: ",seb_ols_sca)$
        print(rho, ": ",0)$
        print(gamma, ": ",gamma_ols_sca," se: ",seg_ols_sca)$
        print("")$
        print("scenario A − AR estimates")$
        print(alpha, ": ",alpha_ar_sca," se: ",sea_ar_sca)$
        print(beta, ": ",beta_ar_sca," se: ",seb_ar_sca)$
        print(rho, ": ",rho_ar_sca," se: ",ser_ar_sca)$
        print(gamma, ": ",gamma_ar_sca," se: ",seg_ar_sca)$
        print("")$
```

OLS: y[i] = α + β ** x[i] + res[i]*

AR: y[i] = α + β ** x[i] +* ρ ** (y[i−1] −* α + β ** x[i−1]) + wn[i]*

scenario A − OLS estimates

α *:* $-5.248630704976216\, 10^{-7}$ *se:* 0.1146445217050405

β *:* 0.901960794005034 *se:* 0.0234017149985364

ρ *:* 0

γ *:* 0.2234372280680667 *se:* 0.07663836732112402

scenario A − AR estimates

α *:* 0.1228251409996585 *se:* 0.7115808225984115

β *:* 0.975062107785824 *se:* 0.02836025756107919

ρ *:* 0.9127869426470371 *se:* 0.1028548663109951

γ *:* 0.06073412079967149 *se:* 0.02083163260950503

```
(%i133) print("")$
        print("OLS:  y[i] =",alpha,"+",beta,"· x[i] + res[i]")$
        print(" AR:  y[i] =",alpha,"+",beta,"· x[i] + ",rho,"· ( y[i−1] − ",alpha,"+",beta,"· x[i−1
        print("")$
        print("scenario B − OLS estimates")$
        print(alpha, ": ",alpha_ols_scb," se: ",sea_ols_scb)$
        print(beta, ": ",beta_ols_scb," se: ",seb_ols_scb)$
        print(rho, ": ",0)$
        print(gamma, ": ",gamma_ols_scb," se: ",seg_ols_scb)$
        print("")$
        print("scenario B − AR estimates")$
        print(alpha, ": ",alpha_ar_scb," se: ",sea_ar_scb)$
        print(beta, ": ",beta_ar_scb," se: ",seb_ar_scb)$
        print(rho, ": ",rho_ar_scb," se: ",ser_ar_scb)$
        print(gamma, ": ",gamma_ar_scb," se: ",seg_ar_scb)$
        print("")$
```

OLS: *y[i]* = α + β * *x[i]* + *res[i]*

AR: *y[i]* = α + β * *x[i]* + ρ * (*y[i−1]* − α + β * *x[i−1]*) + *wn[i]*

scenario B − OLS estimates

α *:* $1.40672503596235\ 10^{-8}$ *se:* 0.1146444741168652

β *:* 1.098039211157605 *se:* 0.02340170528466944

ρ *:* 0

γ *:* 0.2234370425740535 *se:* 0.07663824007328432

scenario B − AR estimates

α *:* −0.1228248085161045 *se:* 0.7115797276449234

β *:* 1.024937878849125 *se:* 0.02836026165807854

ρ *:* 0.9127868117643136 *se:* 0.102854875692645 4

γ *:* 0.06073413409487149 *se:* 0.02083164172993915