# Econ 583 Lab 6
# Solutions for Nonlinear GMM: Empirical Exercise

Eric Zivot

Winter 2013

## 1 Nonlinear GMM: Empirical Exercise

In this exercise, you will estimate a typical Euler equation asset pricing model. The data for the exercise consists of monthly observations on gross consumption growth, $C_t/C_{t-1}$, and returns, $R_{jt}$, for ten size sorted portfolios and T-Bills (risk free rate). (see Zivot and Wang 2005, chapter 21 for an example using S-PLUS).

Consider an Euler equation asset pricing model that allows the individual to invest in $J$ risky assets with returns $R_{j,t+1}$ ($j = 1, \ldots, J$), as well as a risk-free asset with certain return $R_{f,t+1}$. Assuming power utility over consumption $C$

$$\begin{aligned} U(C) &= \frac{C^{1-\alpha_0}}{1-\alpha_0} \\ \alpha_0 &= \text{risk aversion parameter} \end{aligned}$$

the Euler equations may be written as the system of $J+1$ nonlinear equaitons

$$\begin{aligned} E\left[(1+R_{f,t+1})\beta_0\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0}|I_t\right]-1 &= 0, \\ E\left[(R_{j,t+1}-R_{f,t+1})\beta_0\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha_0}|I_t\right] &= 0,\ j=1,\ldots,J, \end{aligned}$$

where $I_t$ represents information available at time $t$ and $\beta_0$ represents the time discount parameter. The parameter vector to be estimated $\boldsymbol{\theta}_0 = (\beta_0, \alpha_0)'$.

1. Using the instrument $x_t = 1$, GMM estimation may be performed using the $J+1$ vector of moments

$$\mathbf{g}(\mathbf{w}_{t+1}, \boldsymbol{\theta}) = \begin{pmatrix} (1+R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha}-1 \\ (R_{1,t+1}-R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \\ \vdots \\ (R_{J,t+1}-R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \end{pmatrix}$$

Using these moments, estimate $\boldsymbol{\theta}$ using the iterated efficient GMM estimator (with a White-type HC covariance estimator) as well as the continuous updating (CU) efficient GMM estimator. Use $\boldsymbol{\theta} = (1, 5)'$ as the starting values for the optimization for the iterated GMM estimator, and use the iterated GMM estimates as the starting values for the CU GMM estimator. Do the estimates look economically plausible? Does the J-statistic reject the overidentifying restrictions at the 5% significance level?

**Solution**: For this question I will use the R package gmm to do the estimation. Details are given in the script file `econ583lab6Solutions.R`[1]. To use the `gmm()` function, you must first write a function to compute the moment conditions $\mathbf{g}(\mathbf{w}_{t+1}, \boldsymbol{\theta})$ which returns a matrix containing $\mathbf{g}(\mathbf{w}_{t+1}, \boldsymbol{\theta})$ for $t = 1, \ldots, T$ at a given value of $\boldsymbol{\theta}$. One such a function is

```
eulerJ.moments <- function(parm,x=NULL) {
  # parm = (beta,alpha)
  # data = (C(t+1)/C(t),1+Rf(t+1),R1(t+1)-Rf(t+1),...,
  #         RJ(t+1)-Rf(t+1))
  n.col = ncol(x)
  sdf = parm[1]*x[,1]^(-parm[2])
  d1 = sdf*x[,2] - 1
  d2 = as.matrix(rep(sdf,(n.col-2))*x[,3:n.col])
  return(cbind(d1,d2))
}
```

The function has two arguments, `parm` and `x`, where `parm` is a $2 \times 1$ vector $\boldsymbol{\theta} = (\beta, \alpha)'$ and `x` is a $T \times (J+2)$ data matrix whose $t$th row is

$$\mathbf{w}_t = (c_t/c_{t-1}, 1 + R_{f,t}, R_{1t} - R_{ft}, \ldots, R_{Jt} - R_{ft})$$

The function returns a $T \times (J+1)$ data matrix whose $t$th row is $\mathbf{g}(\mathbf{w}_t, \boldsymbol{\theta})'$.

The data for estimation is created as follows

```
> pricing.df = read.csv(file="C:/Users/ezivot/Dropbox/econ583/pricing.csv")
> rownames(pricing.df) = pricing.df[,1]
> pricing.df = pricing.df[,-1]
> pricing.mat = as.matrix(pricing.df)
> n.col = ncol(pricing.mat)
> excessRet.mat = apply(pricing.mat[,2:(n.col-1)],2,
+                       function(x,y){x-y},
+                       pricing.mat[,"RF"])
> eulerDataJ.mat = cbind(pricing.mat[,"CONS"],1+pricing.mat[,"RF"],
```

[1] I could use Eviews for the iterated efficient estimator, but not for the CU estimator (Eviews can only do the CU estimator for single equation GMM).

```
+                         excessRet.mat)
> colnames(eulerDataJ.mat)[1] = "CONS"
> colnames(eulerDataJ.mat)[2] = "RF"
```

The iterated efficient GMM estimator using the White estimate of $\mathbf{S}$ and initial values $\boldsymbol{\theta}_1 = (1, 5)'$ is computed using

```
> start.vals = c(1,5)
> names(start.vals) = c("beta","alpha")
> eulerJ.gmm.fit = gmm(g=eulerJ.moments, x=eulerDataJ.mat,
+                       t0=start.vals, type="iterative",
+                       vcov="iid", optfct="optim")
> summary(eulerJ.gmm.fit)

Call:
gmm(g = eulerJ.moments, x = eulerDataJ.mat, t0 = start.vals,
    type = "iterative", vcov = "iid", optfct = "optim")


Method:  iterative

Coefficients:
       Estimate    Std. Error  t value     Pr(>|t|)
beta   8.2736e-01  1.1616e-01  7.1228e+00  1.0578e-12
alpha  5.7395e+01  3.4221e+01  1.6772e+00  9.3504e-02

J-Test: degrees of freedom is 9
                J-test   P-value
Test E(g)=0:    5.76329  0.76336
```

Here, $\hat{\boldsymbol{\theta}}_{iter}\left(\hat{\mathbf{S}}_{HC}^{-1}\right) = (0.827, 57.395)'$ with $\widehat{SE}(\hat{\beta}_{iter}) = 0.116$ and $\widehat{SE}(\hat{\alpha}_{iter}) = 34.221$. The estimate of $\beta$ looks reasonable, but the estimate of $\alpha$ is much too large. A more economically plausible estimate would be around 3. However, the standard error of $\hat{\alpha}_{iter}$ is quite big and the J-test does not reject so the model specification is not rejected. Note, $\hat{\boldsymbol{\theta}}_{iter} = (0.827, 57.395)'$ is almost identical to what Eviews gives.

The CU efficient GMM estimator using the White estimate of $\mathbf{S}$ and initial values $\boldsymbol{\theta}_1 = \hat{\boldsymbol{\theta}}_{iter}$ is computed using

```
> eulerJ.cugmm.fit = gmm(g=eulerJ.moments, x=eulerDataJ.mat,
+                       t0=coef(eulerJ.gmm.fit), type="cue",
+                       vcov="iid", optfct="optim")
> summary(eulerJ.cugmm.fit)
```

```
Call:
gmm(g = eulerJ.moments, x = eulerDataJ.mat, t0 = coef(eulerJ.gmm.fit),
    type = "cue", vcov = "iid", optfct = "optim")

Method:  cue

Coefficients:
       Estimate    Std. Error  t value     Pr(>|t|)
beta   6.9741e-01  1.1886e-01  5.8676e+00  4.4212e-09
alpha  9.6193e+01  3.1617e+01  3.0425e+00  2.3466e-03

J-Test: degrees of freedom is 9
                J-test   P-value
Test E(g)=0:    5.14334  0.82164
```

Here, $\hat{\boldsymbol{\theta}}_{cu}(\hat{\mathbf{S}}_{CU-HC}^{-1}) = (0.697, 96.193)'$ with $\widehat{SE}(\hat{\beta}_{cu}) = 0.119$ and $\widehat{SE}(\hat{\alpha}_{cu}) = 31.617$. The estimate of $\beta$ looks reasonable, but the estimate of $\alpha$ is much too large. However, the standard error of $\hat{\alpha}_{iter}$ is quite big and the J-test does not reject so the model specification is not rejected.

2. Now consider using the instrument $\mathbf{x}_t = (1, C_t/C_{t-1})'$. What are the $2(J+1)$ moment conditions to be used for GMM estimation? Using these moments, estimate $\boldsymbol{\theta}$ using the iterated efficient GMM estimator (with a White-type HC covariance estimator). Do the estimates look economically plausible? Does the J-statistic reject the overidentifying restrictions at the 5% significance level?

**Solution**. The moment vector is now

$$\mathbf{g}(\mathbf{w}_{t+1}, \boldsymbol{\theta}) = \begin{pmatrix} (1+R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1 \\ (R_{1,t+1} - R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \\ \vdots \\ (R_{J,t+1} - R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \\ \left(\frac{C_t}{C_{t-1}}\right)\left((1+R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} - 1\right) \\ \left(\frac{C_t}{C_{t-1}}\right)(R_{1,t+1} - R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \\ \vdots \\ \left(\frac{C_t}{C_{t-1}}\right)(R_{J,t+1} - R_{f,t+1})\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\alpha} \end{pmatrix}$$

A function to compute the moments is

```
euler2J.moments <- function(parm,x=NULL) {
  # parm = (beta,gamma)
  # data = (C(t+1)/C(t),1+Rf(t+1),R1(t+1)-Rf(t+1),...,
  #         RJ(t+1)-Rf(t+1),C(t)/C(t-1))
  n.col = ncol(x)
  sdf = parm[1]*x[,1]^(-parm[2])
  d1 = sdf*x[,2] - 1
  d2 = as.matrix(rep(sdf,(n.col-3))*x[,3:(n.col-1)])
  d3 = d1*x[,n.col]
  d4 = d2*x[,n.col]
  return(cbind(d1,d2,d3,d4))
}
```

The iterated efficient GMM estimator using the White estimate of $\mathbf{S}$ and initial values $\boldsymbol{\theta}_1 = (1,5)'$ is computed using

```
> start.vals = c(1,5)
> names(start.vals) = c("beta","alpha")
> euler2J.gmm.fit = gmm(g=euler2J.moments, x=eulerData2J.mat,
+                       t0=start.vals, type="iterative",
+                       vcov="iid", optfct="optim")
> summary(euler2J.gmm.fit)

Call:
gmm(g = euler2J.moments, x = eulerData2J.mat, t0 = start.vals,
    type = "iterative", vcov = "iid", optfct = "optim")


Method:  iterative

Kernel:  Quadratic Spectral

Coefficients:
       Estimate     Std. Error   t value      Pr(>|t|)
beta    9.9809e-01   8.8343e-04   1.1298e+03   0.0000e+00
alpha  -1.2500e+00   3.7575e-01  -3.3268e+00   8.7855e-04

J-Test: degrees of freedom is 20
                J-test     P-value
Test E(g)=0:    30.207110   0.066567
```

Here, $\hat{\boldsymbol{\theta}}_{iter}\left(\hat{\mathbf{S}}_{HC}^{-1}\right) = (0.998, -1.250)'$ with $\widehat{SE}(\hat{\beta}_{iter}) = 0.008$ and $\widehat{SE}(\hat{\alpha}_{iter}) = 0.376$. The estimate of $\beta$ looks reasonable, but the estimate of $\alpha$ is negative! Also, J-test rejects at the 6.6% level so the model specification is suspect.

The CU efficient GMM estimator using the White estimate of $\mathbf{S}$ and initial values $\boldsymbol{\theta}_1 = \hat{\boldsymbol{\theta}}_{iter}$ is computed using

```
> euler2J.cugmm.fit = gmm(g=euler2J.moments, x=eulerData2J.mat,
+                         t0=coef(euler2J.gmm.fit), type="cue",
+                         vcov="iid", optfct="optim")
> summary(euler2J.cugmm.fit)

Call:
gmm(g = euler2J.moments, x = eulerData2J.mat, t0 = coef(euler2J.gmm.fit),
    type = "cue", vcov = "iid", optfct = "optim")

Method:  cue

Kernel:  Quadratic Spectral

Coefficients:
       Estimate     Std. Error   t value      Pr(>|t|)
beta    9.9806e-01   8.7397e-04   1.1420e+03   0.0000e+00
alpha  -1.2370e+00   3.7160e-01  -3.3288e+00   8.7224e-04

J-Test: degrees of freedom is 20
                J-test     P-value
Test E(g)=0:    30.206323   0.066579
```

The estimates are almost identical to the iterated GMM estimates.