### 2TSF book - Chapter 11 - generic gretl scripts ###

### The following scripts are exact replications of what appears in the book (with any errata corrected).

### It is not necessarily the case that they are complete scripts that can be used as-is. Consult the book.

### pp. 278-279: Basic maximum likelihood estimation ###

series Depvar = DepY

list Xlist = RegrListEdRelInc

smpl --no-missing DepY Xlist

ols Depvar Xlist --quiet

matrix bcoef =$coeff

scalar sv=.5

scalar sw = .5

scalar su = .5

set max\_verbose full

#The next three lines make the names of the variables appear in the estimation output#

string varblnames = varname(Xlist)

string errnames = ",sigmav,sigmaw,sigmau"

string allcoefnames =v arblnames ~ errnames

mle logl = check ? likely : NA

series res = Depvar-lincomb(Xlist, bcoef)

series a1 = sv^2/(2\*su^2) + res/su

series b1 =-res/sv-sv/su

series a2 = sv^2/(2\*sw^2)-res/sw

series b2 = res/sv-sv/sw

series likely =-log(sw + su) +log(exp(a1)\*cnorm(b1)+exp(a2)\*cnorm(b2))

scalarcheck=(sv>0)&&(sw>0)&&(su>0)

params bcoef sv sw su

param\_names allcoefnames

end mle --verbose --robust

###############################################################

### pp 280-281-282: Computation of 2TSF individual metrics ###

##additional code before mle command so that the series survive##

series a1 = 0

series b1 = 0

series a2 = 0

series b2=0

series res=0

##... or code after mle ###

series resmle = Depvar-lincomb(Xlist, bcoef)

series a1 = sv^2/(2\*su^2) + resmle/su

series b1 =-resmle/sv-sv/su

series a2=sv^2/(2\*sw^2)-resmle/sw

series b2=resmle/sv-sv/sw

series chi1 = cnorm(b2) + exp(a1-a2)\*cnorm(b1)

series chi2 = exp(a2-a1)\*chi1

scalar lambda = 1/sw + 1/su

series Exp\_w\_neg = (lambda/(1+lambda))\*(1/chi2)\*(cnorm(b1)+exp(a2-a1-b2\*sv+0.5\*(sv^2))\*cnorm(b2-sv))

series M1=1-Exp\_w\_neg

###############################################################

### pp. 282-283: MLE with Heterogeneity of latent forces ###

scalar sv = .5

scalar d1 = .2

scalar d2 = .3

scalar su = .5

string errnames = ",sigmav,d1,d2,sigmau"

string allcoefnames = varblnames ~ errnames

mle logl = check ? likely : NA

series res = Depvar-lincomb(Xlist,bcoef)

series sw = d1\*z1^2+d2\*z2^2

series a1 = sv^2/(2\*su^2) + res/su

series b1 =-res/sv-sv/su

series a2 = sv^2/(2\*sw^2)-res/sw

series b2 = res/sv-sv/sw

series likely =-log(sw + su) + log(exp(a1)\*cnorm(b1) + exp(a2)\* cnorm(b2))

series swneg = (sw<=0)

scalar swposcheck=sum(swneg)

scalar check=(sv>0)&&(swposcheck=0)&&(su>0)

params bcoef sv d1 d2 su

param\_names allcoefnames

end mle --verbose --robust

###############################################################

### p. 284: Combined OLS/GMM estimation (for data see Application in ch. 7)###

series depvar = LWAGE

scalar obsn = $nobs

ols depvar Xlist--quiet

matrix bcoef = $coeff

series res = $uhat

scalar qv = .4

scalar qw = .3

scalar qu = .3

matrix W = I(4)

series eq2 = 0

series eq3 = 0

series eq4 = 0

series eq5 = 0

gmm series res = depvar-lincomb(Xlist,bcoef)

series res2 = res^2

series res3 = res^3

series res4 = res^4

series res5 = res^5 scalar Avres2sq = ((1/obsn)\*sum(res2))^2

scalar Avres2 = (1/obsn)\*sum(res2)

scalar Avres3 = (1/obsn)\*sum(res3)

series eq2 = res2-sqrt(qv)^4-(1-2/$pi)\*(sqrt(qw)^4 + sqrt(qu)^4)

series eq3 = res3-((4-$pi\*sqrt(2)/($pi^(3/2)))\*(qw^3-qu^3)

series eq4 = (res4-3\*Avres2sq)-(8\*($pi-3)/($pi^2))\*(qw^4+qu^4)

series eq5=(res5-10\*Avres2\*Avres3)-(qw^5-qu^5)\*(sqrt(2)/($pi^5/2))\*(96-40\*$pi+3\*$pi^2)

orthog eq2 ; const

orthog eq3 ; const

orthog eq4 ; const

orthog eq5 ; const

weights W

params qv qw qu

end gmm --verbose --iterate

###############################################################

### pp 286-287: Joint GMM estimation (for data see Application in ch. 7)###

series depvar = LWAGE

scalar obsn = $nobs

ols depvar Xlist--quiet

matrix bcoef = $coeff

series res = $uhat

scalar qv = .4

scalar qw = .3

scalar qu = .3

matrix W = I(17)

series eqb0 = 0

series eqb1 = 0

series eqb2 = 0

series eqb3 = 0

series eqb4 = 0

series eqb5 = 0

series eqb6 = 0

series eqb7 = 0

series eqb8 = 0

series eqb9 = 0

series eqb10 = 0

series eqb11 = 0

series eqb12 = 0

series eq2 =0

series eq3 =0

series eq4 =0

series eq5 =0

string varblnames = varname(Xlist)

string errnames = ",sigmav,sigmaw,sigmau"

string allcoefnames = varblnames ~ errnames

gmm series resc = depvar-lincomb(Xlist,bcoef)-sqrt(2/$pi)\*(qw-qu)

series resc2 = resc^2

series resc3 = resc^3

series resc4 = resc^4

series resc5 = resc^5

scalar Avres2sq =((1/obsn)\*sum(resc2))^2scalarAvres2=(1/obsn)\*sum(resc2)

scalar Avres3=(1/obsn)\*sum(resc3)

series eqb0 = resc\*Xlist[1]

series eqb1 = resc\*Xlist[2]

series eqb2 = resc\*Xlist[3]

series eqb3 = resc\*Xlist[4]

series eqb4 = resc\*Xlist[5]

series eqb5 = resc\*Xlist[6]

series eqb6 = resc\*Xlist[7]

series eqb7 = resc\*Xlist[8]

series eqb8 = resc\*Xlist[9]

series eqb9 = resc\*Xlist[10]

series eqb10 = resc\*Xlist[11]

series eqb11 = resc\*Xlist[12]

series eqb12 = resc\*Xlist[13]

series eq2 = resc2-sqrt(qv)^4-(1-2/$pi)\*(sqrt(qw)^4 + sqrt(qu)^4)

series eq3 = resc3-((4-$pi\*sqrt(2)/($pi^(3/2)))\*(qw^3-qu^3)

series eq4 = (resc4-3\*Avres2sq)-(8\*($pi-3)/($pi^2))\*(qw^4 + qu^4)

series eq5 = (resc5-10\*Avres2\*Avres3)-(qw^5-qu^5)\*(sqrt(2)/($pi^5/2))\*(96-40\*$pi+3\*$pi^2)

orthog eqb0 ; const

orthog eqb1 ; const

orthog eqb2 ; const

orthog eqb3 ; const

orthog eqb4 ; const

orthog eqb5 ; const

orthog eqb6 ; const

orthog eqb7 ; const

orthog eqb8 ; const

orthog eqb9 ; const

orthog eqb10 ; const

orthog eqb11 ; const

orthog eqb12 ; const

orthog eq2 ; const

orthog eq3 ; const

orthog eq4 ; const

orthog eq5 ; const

weights W

params bcoef qv qw qu

param\_names allcoefnames

end gmm --verbose --iterate

###############################################################

### pp 290-291: Tug of War 2TSF model with Copula###

function series rankingalt(series X)

matrix a = aggregate(const, X)

matrix b=cum(a[,2])

return replace(X,a[,1],b)

end function

series x\_star = invcdf(N,(1/(obsn+1))\*rankingalt(x1))

series lowbound = 0.000001

series highbound = const-0.000001

ols Depvar Xlist--quiet

matrix bcoef = $coeff

scalar sw = .6

scalar su = .7

scalar rhole =-0.5

set max\_verbose full

#The next three lines make the names of the variables appear in the estimation output

string varblnames=varname(Xlist)

string errnames=",sigmaw,sigmau,rho1e"

string allcoefnames =varblnames ~ errnames

mle logl = check ? likely : NA

series res = Depvar-lincomb(Xlist, bcoef)

series logdensAL =-ln(sw+su) + (res<=0)\*(res/su)-(res>0)\*(res/sw)

series likely =-0.5\*ldet(R2)-0.5\*qRq + 0.5\*((invcdf(N,CDF))^2) + logdensAL

# the following relate to the copula density series

CDFraw = (res<=0)\*(su/(sw+su))\*exp(res/su) + (res>0)\* (1-(sw/(sw+su))\*exp(-res/sw))

list cdflist = lowbound CDFraw highbound

series CDF = median(cdflist)

matrix R2={1,rho1e;rho1e,1}matrixq={x\_star,invcdf(N,CDF)}

matrix qR=q\*inv(R2)

series qRq=sumr(q.\*qR)

scalar check=(abs(rho1e)<1)&&(sw>0)&&(su>0)

params bcoef sw su rho1e

param\_names allcoefnames

end mle

###############################################################

### pp 292-293: The mode as individual predictor of latent forces###

function scalar ulogdensity (const scalar sv, const scalar sw, const scalar su, series res, matrix mod)

scalar k = $t2

scalar a1 = sv^2/(2\*su^2) + res[k] /su

scalar b1 =-res[k] /sv-sv/su

scalar a2 = sv^2/(2\*sw^2)-res[k] /sw

scalar b2 = res[k] /sv-sv/sw

scalar chi1=cnorm(b2)+exp(a1-a2)\*cnorm(b1)

scalar lambda=1/sw+1/su

logdensu=log(lambda)-lambda\*mod+log(cnorm(mod/sv+b2))-log(chi1)

return logdensu

end function

matrix modeu = 0.5\*ones(1,1) #initialize the object we seek to compute.

matrix bounds = {1,1.0e-20,0.9999} #argmax must be between 0 and 1. The bounds are the second and third values.

series modeuvals =0 #declare the series to hold the results

loop m =1..obsn --quiet

smpl 1 m

matrix modeu[1,1] = 0.5 #initial value for the argmax we seek

densval =BFGScmax(&modeu,bounds,ulogdensity(sv,sw,su,resmle,modeu))

smpl 1 obsn

series modeuvals[m] = modeu[1,1]

endloop

###############################################################