附錄

A SAS 程式碼

本文使用統計軟體 SAS 9.2 中 The MODEL Procedure 參數估計, 附錄爲 skew-t 分配與 AEPD 分配程式碼, SEPD 分配可在設定參數 $p_1 = p_2$ 下估計, EPD 分配可在設定 $p_1 = p_2$, $\alpha = 0.5$ 下估計。

```
/********** Skew-t ********** /
proc model data = taiwan ;
parms arch0 0.1 arch1 0.2 garch1 0.6 ar1 0.5 ar2 0.5 ar3 0.5 v 5 lambda 0.05;
bounds -1<lambda<1, 2<v;
/* mean model */
r =int + ar1*lag1(r) + ar2*lag2(r) +ar3*lag3(r) ;
/* variance model */
if zlag(resid.r ) <0 then
h.r = arch0 + arch1 * xlag(resid.r ** 2, mse.r) + garch1 * xlag(h.r, mse.r) ;
else
h.r = arch0 + arch1 * xlag(resid.r ** 2, mse.r) + garch1 * xlag(h.r, mse.r) +Tgarch* xlag(resid.r **
2, mse.r);
/* specify error distribution */
x= constant('pi');
c=gamma((v+1)/2)/(gamma(v/2)*(x*(v-2))**0.5);
a=4*lambda*c*((v-2)/(v-1));
b=(1+3*(lambda)**2-a**2)**0.5;
if -resid.r/sqrt(h.r)< -(a/b) then obj =-( log(b)+log( gamma((v+1)/2) )-1/2*log(x) -1/2*log((v-2))
-log(gamma(v/2))-log(sqrt(h.r))-(v+1)/2*log(1+1/(v-2)*(((b*(-resid.r/h.r**0.5)+a)/(1-lambda))**2)));
if -resid.r/sqrt(h.r)>= -(a/b) then obj =-(log(b)+log( gamma((v+1)/2) )-1/2*log(x) -1/2*log((v-2))
-log(gamma(v/2))-log(sqrt(h.r))-(v+1)/2*log(1+1/(v-2)*(((b*(-resid.r/h.r**0.5)+a)/(1+lambda))**2)));
errormodel r ~ general(obj , v,lambda);
/* fit the model */
fit r / method=marquardt CONVERGE=0.00001;
run;

/********** AEPD ********** /
proc model data = taiwan ;
parms aa 0.5 arch0 0.1 arch1 .2 garch1 .8 Tgarch 0.1 p1 1 p2 1 ar1 0.5 ar2 0.1 ar3 0.5 ;
bounds 0<aa<1, 0<p1 , 0<p2 ;
/* mean model */
r =int + ar1*lag1(r) + ar2*lag2(r) +ar3*lag3(r) ;
/* variance model */
if zlag( resid.r ) <0 then
h.r = arch0 + arch1 * xlag(resid.r ** 2, mse.r) + garch1 * xlag(h.r, mse.r) ;
else
h.r = arch0 + arch1 * xlag(resid.r ** 2, mse.r) + garch1 * xlag(h.r, mse.r) +Tgarch* xlag(resid.r **
```

```
2, mse.r);
/* specify error distribution */
Kp1=1/(2*p1**(1/p1)*gamma(1+1/p1));
Kp2=1/(2*p2**(1/p2)*gamma(1+1/p2));
B=aa*Kp1+(1-aa)*Kp2;
aastar=aa*Kp1/ (aa*Kp1 +(1-aa)*Kp2);
mean=1/B*( (1-aa)**2*p2*gamma(2/p2)/(gamma(1/p2))**2 - aa**2*p1*gamma(2/p1)/(gamma(1/p1))
**2 ) ;
variance= (1/B**2)*((1-aa)**3*p2**2*gamma(3/p2)/(gamma(1/p2))**3 + aa**3*p1**2*gamma(3/p1)/
(gamma(1/p1))**3- ( (1-aa)**2*p2*gamma(2/p2)/(gamma(1/p2))**2 - aa**2*p1*gamma(2/p1)/(gamma(1/p1))
**2 )**2 );
z=(-resid.r/h.r**0.5*variance**0.5+mean);
if z<0 then obj = -(- log(sqrt(h.r))+log(B)+(-1/p1*abs(z/2/aastar)**p1) +log(variance**0.5));
if z>=0 then obj = -(- log(sqrt(h.r))+log(B)+(-1/p2*abs(z/2/(1-aastar))**p2) +log(variance**0.5)) ;
errormodel r ~ general(obj , aa, p1,p2);
/* fit the model */
fit r / method=marquardt CONVERGE=0.00001;
run;
```