```
clear all;

%these are my home and office directories
%cd 'C:\WINDOWS\Desktop\203c';
cd 'C:\Documents and Settings\econhist\Desktop\203c';

%set parameter values;
omega=[4.456 -0.274 0.227; -0.274 5.323 0.017; 0.227 0.017 5.247];
mux=[2;2;3];
beta=[1;0.5;-0.5;0.25];
n=100; %total observ;
NS=500; %total samples, simulation draws;

%draw 100 observations for 500 samples;
p=chol(omega)';
psi=randn(3,n,NS); %from standard norm distribution
u=trnd(5,n,NS); %from t-distribution


%create the x and y matrices for all draws
for i=1:n,
   for d=1:NS,
      x_all(:,i,d)=mux+p*psi(:,i,d);
      y_all(i,d)=1*beta(1)+x_all(1,i,d)*beta(2)+x_all(2,i,d)*beta(3)+x_all(3,i,d)*beta(4)
+u(i,d);
   end
end

%options for fimns below;
options(14)=500; %maximum number of iterations;
options(3)=0.001; %tolerance
options(2)=0.001; %tolerance
options(1)=1;


%optimal gmm estimate

for d=1:NS        %loop over all simulation draws
   x=[ones(n,1) x_all(:,:,d)'];     %create x matrix for this draw,nx4
   y=y_all(:,d);        %create y matrix for this draw
   z=[x x(:,2:4).^2];   %create z matrix for this draw, nx7
   beta_gmm_init=inv((x'*z)*(z'*x))*(x'*z)*(z'*y);
   for i=1:n
      w(:,:,i)=(z(i,:)'*(y(i)-x(i,:)*beta_gmm_init))*(z(i,:)'*(y(i)-x(i,:)
*beta_gmm_init))';
   end
   w=1/n*sum(w,3);
   beta_gmm(:,d)=inv((x'*z)*inv(w)*(z'*x))*(x'*z)*inv(w)*(z'*y);     %indexed by d to save
each beta_gmm draw
   for i=1:n
      v(:,:,i)=(z(i,:)'*(y(i)-x(i,:)*beta_gmm))*(z(i,:)'*(y(i)-x(i,:)*beta_gmm))';
   end
   v=1/n*sum(v,3);
   cov_est(:,:,d)=n*inv((x'*z)*inv(v)*(z'*x)); %indexed by d to save
   %sed(:,d)=sqrt(diag(cov_est));         %indexed by d to save


   %wald statistics
   tri=[0 beta_gmm(3,d) beta_gmm(2,d) 1];
   wald(:,d)=n*((beta_gmm(2,d)*beta_gmm(3,d)+beta_gmm(4,d))^2)/(tri*cov_est(:,:,d)*tri');

   %lagrange multiplier statistics
   %this first requires computation the restricted estimates

   global x y z n w_r; %allow the current x y z for this draw to be used by the function

   %restricted intial estimates
```

```
   beta_start=beta_gmm(1:3,d);
   temp_init=fmins('gmm_init_r', beta_start, options);
   beta_gmm_init_r=[temp_init; -temp_init(2)*temp_init(3)];

   %next compute weight matrix using these initial estimates
   for i=1:n
      w_r(:,:,i)=(z(i,:)'*(y(i)-x(i,:)*beta_gmm_init_r))*(z(i,:)'*(y(i)-x(i,:)
*beta_gmm_init_r))';
   end
   w_r=1/n*sum(w_r,3);

   %optimal gmm restricted estimates
   temp=fmins('gmm_r', beta_start, options);
   beta_gmm_r=[temp; -temp(2)*temp(3)];

   %next we need to calculate some additional functions of the restricted estimates
   for i=1:n
      mn_r(:,i)=z(i,:)'*(y(i)-x(i,:)*beta_gmm_r);
      v_r(:,:,i)=(z(i,:)'*(y(i)-x(i,:)*beta_gmm_r))*(z(i,:)'*(y(i)-x(i,:)*beta_gmm_r))';
      der_mn_r(:,:,i)=x(i,:)'*z(i,:);
   end
   mn_r=1/n*sum(mn_r,2);
   v_r=1/n*sum(v_r,3);
   der_mn_r=1/n*sum(der_mn_r,3);


   %the lagrange multplier stat
   lm(:,d)=n*mn_r'*inv(w_r)*der_mn_r'*inv(der_mn_r*inv(v_r)*der_mn_r')*der_mn_r*inv(w_r)
*mn_r;

%likelihood ratio statistics

%we need to calculate the moments at the unrestricted estimates
for i=1:n
   mn_u(:,i)=z(i,:)'*(y(i)-x(i,:)*beta_gmm(:,d));
end
mn_u=1/n*sum(mn_u,2);

lr(:,d)=n*((mn_r'*inv(w_r)*mn_r)-(mn_u'*inv(w)*mn_u));
end

%histograms
hist(wald,50)
hist(lm,50)
hist(lr,50)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%seperate functions

function qn=gmm_r(beta_param)

global x y z n w_r;

%create restricted parameter vector
beta_param=[beta_param; -beta_param(2)*beta_param(3)];

for i=1:n
   mn(:,i)=z(i,:)'*(y(i)-x(i,:)*beta_param);
end
mn=1/n*sum(mn,2);


qn=mn'*inv(w_r)*mn;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function qn_init=gmm_init_r(beta_param)
```

```
global x y z n w_r;

%create restricted parameter vector using V=I

beta_param=[beta_param; -beta_param(2)*beta_param(3)];

for i=1:n
   mn(:,i)=z(i,:)'*(y(i)-x(i,:)*beta_param);
end
mn=1/n*sum(mn,2);

qn_init=mn'*mn;
```