

---

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Examine behavior of the MC under different starting values
% This version with highly correlated data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Set random number generator, start stop watch, open output file
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
rand('state',37); % set arbitrary seed for uniform draws
randn('state',37); % set arbitrary seed for normal draws

% Create data with a strong correlation between x2 and x3
% generate data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
n=10000; % set sample size
x1=ones(n,1);
x2= -1.4+randn(n,1);
x3= 0.8*x2+0.5*randn(n,1); %this induces a correlation of 0.85
% betw. x2 and x3
X=[x1 x2 x3];
k=size(X,2);

btrue=[1.2 0.4 0.8]'; % "true" coefficients

sig2true=1.44; %"true" error variance
eps=sqrt(sig2true)*randn(n,1);
y=X*btrue+eps;

% general elements
r1=2000; % burn-in
r2=10000; % keepers
R=r1+r2;

% priors for beta
mu0=zeros(k,1); %diffuse prior for mean of betas
V0=eye(k)*100; % diffuse prior for varcov of beta

%priors for sig2
v0=1/2;
tau0=1/2; % diffuse prior shape and scale

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% starting values, set 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generic OLS
bols=inv(X'*X)*X'*y;
res=y-X*bols;
s2=(res'*res)/(n-k);
betadraw=bols; %use OLS values as starting draws for Gibbs Sampler
% won't be needed, as before
sig2draw=s2; % use OLS variance as starting draw for Gibbs Sampler

% run GS

```

---

```

[betamat,sig2mat]=...

gs_normal_independent_keepall(X,y,n,k,r1,r2,mu0,V0,betadraw,v0,tau0,sig2draw);
sig2chain1=[sig2mat];
b3chain1=[betamat(end,:)];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% starting values, set 2 - completely arbitrary
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generic OLS
sig2draw=3000;

% run GS
[betamat,sig2mat]=...

gs_normal_independent_keepall(X,y,n,k,r1,r2,mu0,V0,betadraw,v0,tau0,sig2draw);
sig2chain2=[sig2mat];
b3chain2=[betamat(end,:)];

figure(7);
x=(1:1:R);
plot(x,b3chain1,'k',x,b3chain2,'b',x,btrue(3)*ones(R,1),'r');
axis([min(x) max(x) -5 5]);
title('convergence plots for beta3, n=10,000, corr');
xlabel('iteration');
ylabel('drawn values');
legend('chain1','chain2','true value');
hold on;
plot(1,b3chain2(1),'o');
hold off;

to = 'c:\klaus\AAEC6564\mlab\figures\mod2_convergence_plots2_fig7'; %designate
destination
saveas(gcf,to);%"gcf" means "get current figure"

figure(8);
x=(1:1:R);
plot(x,sig2chain1,'k',x,sig2chain2,'b',x,sig2true*ones(R,1),'r');
axis([min(x) max(x) 0 20]);
title('convergence plots for sig2, n=10,000, corr');
xlabel('iteration');
ylabel('drawn values');
legend('chain1','chain2','true value');
hold on;
plot(1,sig2chain2(1),'o');
hold off;

% note: the sample variance for the simulated error is 1.4122
to = 'c:\klaus\AAEC6564\mlab\figures\mod2_convergence_plots2_fig8'; %designate
destination
saveas(gcf,to);%"gcf" means "get current figure"

```

---

---

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Now let's truncate our data to 100 observations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

y=y(1:100);
X=X(1:100,:);
n=length(y);
k=size(X,2);

% general elements
r1=2000; % burn-in
r2=10000; % keepers
R=r1+r2;

% priors for beta
mu0=zeros(k,1); %diffuse prior for mean of betas
V0=eye(k)*100; % diffuse prior for varcov of beta

%priors for sig2
v0=1/2;
tau0=1/2; % diffuse prior shape and scale

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% starting values, set 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generic OLS
bols=inv(X'*X)*X'*y;
res=y-X*bols;
s2=(res'*res)/(n-k);
betadraw=bols; %use OLS values as starting draws for Gibbs Sampler
sig2draw=s2; % use OLS variance as starting draw for Gibbs Sampler
% run GS
[betamat,sig2mat]=...

gs_normal_independent_keepall(X,y,n,k,r1,r2,mu0,V0,betadraw,v0,tau0,sig2draw);
sig2chain1=[sig2mat];
b3chain1=[betamat(end,:)];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% starting values, set 2 - completely arbitrary
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% generic OLS
sig2draw=3000; % use OLS variance as starting draw for Gibbs Sampler

% run GS
[betamat,sig2mat]=...

gs_normal_independent_keepall(X,y,n,k,r1,r2,mu0,V0,betadraw,v0,tau0,sig2draw);
sig2chain2=[sig2mat];
b3chain2=[betamat(end,:)];

figure(9);

```

---

---

```

x=(1:1:R);
plot(x,b3chain1,'k',x,b3chain2,'b',x,btrue(3)*ones(R,1),'r');
axis([min(x) max(x) -5 5]);
title('convergence plots for beta3, n=100, corr');
xlabel('iteration');
ylabel('drawn values')
legend('chain1','chain2','true value')
hold on;
plot(1,b3chain2(1),'o');
hold off;

to = 'c:\klaus\AAEC6564\mlab\figures\mod2_convergence_plots2_fig9'; %designate
destination
saveas(gcf,to);%"gcf" means "get current figure"

figure(10);
x=(1:1:R);
plot(x,sig2chain1,'k',x,sig2chain2,'b',x,sig2true*ones(R,1),'r');
axis([min(x) max(x) 0 20]);
title('convergence plots for sig2, n=100, corr');
xlabel('iteration');
ylabel('drawn values')
legend('chain1','chain2','true value')
hold on;
plot(1,sig2chain2(1),'o');
hold off;

to = 'c:\klaus\AAEC6564\mlab\figures
\mod2_convergence_plots2_fig10'; %designate destination
saveas(gcf,to);%"gcf" means "get current figure"

% note: the sample variance for the simulated error is 1.4122

```

*Published with MATLAB® R2022b*