**Michael Cai**

**Lab Report 5**



%grid

zmax = 6;

dz = 0.1;

z = [-zmax:dz:zmax]';

%mixture

omega = 0.2; theta = -1; delta = 2;

p1 = exp(-z.^2/2)\*dz/sqrt(2\*pi);

p2 = exp(-(z-theta).^2/(2\*delta))\*dz/sqrt(2\*pi\*delta);

p = (1-omega)\*p1 + omega\*p2

%%

figure(1)

subplot(3,1,1), bar(z,p1,'b')

title('Standard Normal')

subplot(3,1,2), bar(z,p1,'r')

title('Other Normal')

subplot(3,1,3), bar(z,p,'m')

title('Mixture')

%%

mu = 0.026

sig = 0.03

logg = mu + sig\*z

beta = .99; alpha = 10

loggbar = sum(p.\*logg)

stdlogg = sqrt(sum(p.\*(logg-loggbar).^2))

skewlogg = sum(p.\*(logg-loggbar).^3)/stdlogg^3

xkurlogg = sum(p.\*(logg-loggbar).^4)/stdlogg^4 - 3

%%

kernel = beta\*exp(-alpha\*logg)

divz = exp(logg).^3

q1 = sum(p.\*kernel)

r1 = 1/q1

qe = sum(p.\*kernel.\*divz)

Ere = sum(p.\*divz/qe)

Eq\_Prem = Ere-r1

%grid

zmax = 6;

dz = 0.1;

z = [-zmax:dz:zmax]';

%mixture

omega = 0.2; theta = 1; delta = 2;

p1 = exp(-z.^2/2)\*dz/sqrt(2\*pi);

p2 = exp(-(z-theta).^2/(2\*delta))\*dz/sqrt(2\*pi\*delta);

p = (1-omega)\*p1 + omega\*p2

%%

figure(1)

subplot(3,1,1), bar(z,p1,'b')

title('Standard Normal')

subplot(3,1,2), bar(z,p1,'r')

title('Other Normal')

subplot(3,1,3), bar(z,p,'m')

title('Mixture')

%%

mu = 0.026

sig = 0.03

logg = mu + sig\*z

beta = .99; alpha = 10

loggbar = sum(p.\*logg)

stdlogg = sqrt(sum(p.\*(logg-loggbar).^2))

skewlogg = sum(p.\*(logg-loggbar).^3)/stdlogg^3

xkurlogg = sum(p.\*(logg-loggbar).^4)/stdlogg^4 - 3

%%

kernel = beta\*exp(-alpha\*logg)

divz = exp(logg).^3

q1 = sum(p.\*kernel)

r1 = 1/q1

qe = sum(p.\*kernel.\*divz)

Ere = sum(p.\*divz/qe)

Eq\_Prem = Ere-r1

beta = 0.99

alpha = 10

omega = 0.2

theta = 1

delta = 2

sigma = 0

for sig in np.arange(0,2,.0000001):

varlogg = sig\*\*2\*((1-omega)+(omega\*delta)+omega\*(1-omega)\*theta\*\*2)

if .0012249 < varlogg < 0.0012251:

sigma = sig

print "sigma = "

print sigma

break

mu = 0

for m in np.arange(0,2,.00001):

explogg = m + omega\*sigma\*theta

if .01999 < explogg < 0.2001:

mu = m

print "mu = "

print mu

break

#logg = mu + sigma\*z

explog = mu + omega\*sigma\*theta

print "explog = "

print explog

varlogg = sigma\*\*2\*((1-omega)+omega\*delta+omega\*(1-omega)\*theta\*\*2)

print "varlogg = "

print varlogg