Economics 522A, Homework 5 Suggested Solutions

1. The function `hw5_1.m` simulates a draw for y in the normal regression model. Since the variance of y is $\sigma^2 I_n$, we can draw n independent standard normals using `randn`, multiply each by σ, and add to $X\beta$ to get the vector y.

2. See `hw5_3.m`, which also does the later parts of the homework simultaneously.

   By the results in the lecture notes:

$$\hat{\beta}|X \sim N(\beta, \sigma^2(X'X)^{-1}),$$
$$s^2|X \sim \chi^2_{n-k} \cdot \left(\frac{\sigma^2}{n-k}\right),$$

   and s^2 is independent of $\hat{\beta}$. So:

$$E[\hat{\beta}|X] = \beta, \quad E[s^2|X] = \sigma^2,$$
$$V[\hat{\beta}|X] = \sigma^2(X'X)^{-1},$$

   and

$$V[s^2|X] = \left(\frac{\sigma^2}{n-k}\right)^2 V(\chi^2_{n-k}) = \frac{\sigma^4}{(n-k)^2} \cdot 2(n-k) = \frac{2\sigma^4}{n-k}.$$

   In our example, $V[s^2|X] = 8/(200-2) = .0404$. Finally, since they are independent, $Cov(\hat{\beta}, s^2|X) = 0$.

   The sample mean of the estimates and the sample covariance are:

```
Sample Mean under Normality

  0.99970  1.00381  2.00588

True Cov Matrix for Beta

    1.0016e-02   -3.9814e-04
   -3.9814e-04    9.7420e-03

Sample Cov under Normality

    1.0172e-02   -8.4015e-04    4.5621e-04
   -8.4015e-04    9.7506e-03   -1.3053e-04
    4.5621e-04   -1.3053e-04    3.9403e-02
```

   We see that the mean of $\hat{\beta}$ and the mean of s^2 appear to be close to β and σ^2. In addition, the sample covariance matrix appears to be similar to the theoretical predictions.

3. Since $\hat{\beta}$ and s^2 are independent, their covariance should be zero, and this appears to be the case, approximately. There are ways to test for complete statistical independence as well, but we do not explore them here.

4. The function `hw5_2.m` simulates a draw for y under this model. To generate a draw for a χ_1^2 random variable, it generates a draw from a standard normal (using `randn`) and squares it.

   To derive theoretical predictions, first note that $E[\chi_1^2] = 1$, so that we will have $E[y|X] = X\beta$. In addition, we have that

   $$V[y_i|X] = V[v_i|X] = V[\chi_1^2] = 2.$$

   So the covariance matrix of the vector y will be:

   $$V[y|X] = 2 \cdot I_n.$$

   Now, by working through the proofs (which turn out to not depend on normality of y, only the first two moments), we can show that

   $$E[\hat{\beta}|X] = \beta, \quad V[\hat{\beta}|X] = 2(X'X)^{-1},$$

   and

   $$E[s^2|X] = 2.$$

   However, the proofs for independence of $\hat{\beta}$ and s^2, and the proof that s^2 has a scaled chi-square distribution, did rely on normality of y_i. So we have no guarantee that these will in fact hold any more.

   Here are the results using the chi-squared specification:

```
Sample Mean under Chi-Squared

  0.99646  1.00057  1.97007

Sample Cov under Chi-Squared

    1.0252e-02   -5.1156e-04    4.0241e-02
   -5.1156e-04    9.4996e-03    2.9420e-04
    4.0241e-02    2.9420e-04    2.6783e-01
```

   The upper 2×2 part of the sample covariance matrix should be the same as before (approximately), and this seems to be the case.

Matlab Programs

```
hw5_1.m:
function y = hw5_1(X,beta,sigma2)

% assumes X is n x k, and beta is k x 1
% (it would be better to actually check to make sure this holds)

k=length(beta);
n=size(X,1);
sigma = sqrt(sigma2);

y = X*beta + sigma*randn(n,1);
% end hw5_1.m

hw5_2.m:
function y = hw5_2(X,beta)

% simulates y using chi-square distribution, for Q4

n=size(X,1);
chi2draws = randn(n,1).^2;

y = X*beta + chi2draws - 1;
% end hw5_2.m

hw5_3.m:
% main script to run Monte Carlo simulations

load hw5_noheader.dat
X = hw5_noheader;

J = 1000;
k = 2;
n = size(X,1);

beta = [1;1];
sigma2 = 2;
XXinv = inv(X'*X);

b1draws = NaN*ones(J,k);
s1draws = NaN*ones(J,1);

b2draws = NaN*ones(J,k);
s2draws = NaN*ones(J,1);
```

```
for j=1:J,
y1 = hw5_1(X,beta,sigma2);
y2 = hw5_2(X,beta);
b1 = XXinv*X'*y1;
e1 = y1-X*b1;
s1 = e1'*e1/(n-k);

b1draws(j,:) = b1';
s1draws(j,1) = s1;

b2 = XXinv*X'*y2;
e2 = y2-X*b2;
s2 = e2'*e2/(n-k);

b2draws(j,:) = b2';
s2draws(j,1) = s2;
end;

disp('Sample Mean under Normality')
mean( [b1draws s1draws] )

disp('True Cov Matrix for Beta')
sigma2*XXinv

disp('Sample Cov under Normality')
cov( [b1draws s1draws])

disp('Sample Mean under Chi-Squared')
mean( [b2draws s2draws] )

disp('Sample Cov under Chi-Squared')
cov( [b2draws s2draws])
```