Empirical exercise – Inference – Understanding the F test using the simulated F distribution

1. Aim of the exercise

We are interested in conducting a hypothesis test of multiple linear restrictions on certain population coefficients in a linear regression model. The corresponding test statistic of interest is the F statistic. This statistic has a F distribution. We will simulate this distribution so that we can visualise and work with the distribution itself right in front of us while we are conducting our hypothesis test.

The empirical context is as follows. Mincer (1974) investigates how innate ability affects wages. He estimates a regression model where the dependent variable is the logarithm of wage, and the control variables are IQ score as a proxy for innate ability, experience, education, and a quadratic function of age. The aim is to conduct a hypothesis test of multiple linear restrictions on the population coefficients the quadratic function of age. The data is the wage2 dataset from Wooldridge (2015).

2. Prepare the data and obtain the coefficient estimates

2.1. Load the data

```
clear;
load 'M:\exerciseinferenceFmlr.mat';
clearvars -except lwage IQ exper educ age agesquared;
```

2.2. Create the systematic component of the regression equation

```
y = lwage;
N_obs = size(y,1);
x_0 = ones(N_obs,1);
X = [x_0 IQ exper educ age agesquared];
```

2.3. Obtain the OLS coef. estimates and the S.E. estimates of them

```
LSS = exercisefunction(y,X);
B_hat = LSS.B_hat;
B_hat_VCE = LSS.B_hat_VCE;
```

3. Calculate the F statistic for the hypothesis test of multiple linear restrictions on coefficient estimates

3.1. Matrix of restrictions. df by K matrix.

```
R = [0 0 0 0 1 0; 0 0 0 0 0 1]; % Test whether the two age terms are jointly significant.
```

3.2. Hypothesised value of R*B_hat. df by 1 matrix.

```
q = [0; 0];
```

3.3. Determine the two types of degrees of freedom of the F distribution

```
F_df_restrictions = size(q,1); % Test constraints degrees of freedom.  Also called the numerator degrees of freedom.
F_df_residual = N_obs-size(X,2); % Residual degrees of freedom.  Also called the denominator degrees of freedom.
```

3.4. Calculate the F value and state its distribution derived under the null hypothesis

```
F = (R*B_hat-q)'*inv(R*B_hat_VCE*R')*(R*B_hat-q)/F_df_restrictions; % The Wald statistic is used to test composite linear hypotheses about the estimated parameters of a model.  Replacing the variance of the regression with its estimate gives the F statistic form of the Wald statistic.  F statistic has a F distribution.
F = round(F,2); % This is adjustment for visualisation later in the exercise.
```

3.5. Calculate the one-tailed p value corresponding to the F value

```
p = fcdf(F,F_df_restrictions,F_df_residual,'upper');
```

4. Carry out the (one-tailed) F test using the tabulated F distribution

4.1. Calculate the critical F value considering a significance level of 0.05 and the two degrees of freedom of the F distribution

```
F_c = finv(0.95,F_df_restrictions,F_df_residual); % Reject the null since F > F_c.  Note that F test is always a one-tailed test.
F_c = round(F_c,2); % This is adjustment for visualisation later in the exercise.
```

4.3. Calculate the one-tailed critical F value corresponding to the critical F value

```
p_c = fcdf(F_c,F_df_restrictions,F_df_residual,'upper'); % Reject the null
since p < p_c.
```

5. Carry out the (one-tailed) F test using the simulated F distribution

5.1. Set the number of observations for the data to be used to simulate F distribution

```
N_obs_data_sim_F_dis = 100000; % What do you expect will happen if you increase
this number?
```

5.2. Generate a dataset of a random var. with the distribution and degrees of freedom the test statistic would follow under the null hypothesis

```
data_sim_F_dis = frnd(F_df_restrictions,F_df_residual,[N_obs_data_sim_F_dis,1]);
% The random variable has a F distribution.  Note that each time frnd is
called, different random draws are taken from the F distribution.
```

5.3. Set the number of bins for the histogram to be drawn

```
nbins = 0:0.01:15; % bins start from 0 since the F test is one-tailed.
```

5.4. Plot the histogram of the F distribution

```
histogram(data_sim_F_dis,nbins,'FaceColor','white','EdgeAlpha',0.15);
hold off
title('Fig.  1:  Simulated F distribution')
legend('F disribution')
ylabel('Frequency')
xlabel('F')
```

5.5. Mark the F value

```
histogram(data_sim_F_dis,nbins,'FaceColor','white','EdgeAlpha',0.15);
hold on
line([F F],ylim,'Color','blue') % Mark F value.  Does this seem like a likely
value to observe if in fact this is the true distribution?
hold off
title('Fig.  2:  Simulated F dis.  with F value marked')
legend('F disribution','F value')
ylabel('Frequency')
xlabel('F')
```

5.6. Shade the area where the random values are more extreme than the F value

```
val_below = data_sim_F_dis < F;
val_above_F = data_sim_F_dis >= F;
histogram(data_sim_F_dis(val_below),nbins,'FaceColor','white','EdgeAlpha',0.15);
hold on
histogram(data_sim_F_dis(val_above_F),nbins,'FaceColor','blue','EdgeAlpha',0.15);
% The corresponding area is p_sim.  See below.
hold on
line([F F],ylim,'Color','blue')
hold off
title('Fig.  3:  Simulated F dis.  with F value marked, prob.  area shaded')
legend('Values below','Values above F value','F value')
ylabel('Frequency')
xlabel('F')
```

5.7. Calculate the shaded area associated with the F value which gives the simulated p value

```
F_extreme_value_dummy = data_sim_F_dis > F;
p_sim = mean(F_extreme_value_dummy); % The fraction of the extreme values gives
the probability area associated with the F value under the F distribution.
This gives the simulated p value!  Compare p_sim with p!  If you increase
N_obs_sim_F_dis_data, p_sim gets closer to p.  Why?
```

5.8. Mark the F value, mark the critical F value, shade the area where the random values are more extreme than the F value, and shade the area where the random values are more extreme than the critical F value.

```
val_above_F_c = data_sim_F_dis >= F_c;
histogram(data_sim_F_dis(val_below),nbins,'FaceColor','white','EdgeAlpha',0.15);
hold on
histogram(data_sim_F_dis(val_above_F_c),nbins,'FaceColor','red','EdgeAlpha',0.15);
% Reject the null since p_sim < p_c.
hold on
histogram(data_sim_F_dis(val_above_F),nbins,'FaceColor','blue','EdgeAlpha',0.15);
hold on
line([F_c F_c],ylim,'Color','red') % Mark the critical F value.  Reject the null
since F > F_c.
hold on
line([F F],ylim,'Color','blue')
hold off
title('Fig.  4:  Simulated F dis.  with F and critical F values marked,
prob.  areas shaded')
```

```
legend('Values below','Values above critical F value','Values above F value',
'critical F value','F value')
ylabel('Frequency')
xlabel('F')
```

5.9. Calculate the shaded area associated with the critical F value which gives the simulated critical p value

```
F_extreme_value_dummy_F_c = data_sim_F_dis > F_c;
p_c_sim = mean(F_extreme_value_dummy_F_c); % The fraction of the extreme values
gives probability area associated with the critical F value under the F
distribution!  This gives the simulated critical p value!  Compare
p_criitical_sim with p_c!  If you increase N_obs_sim_F_dis_data, p_c_sim gets
closer to p_c.  Why?
```