ECOM40006/ECOM90013 Econometrics 3
Department of Economics
University of Melbourne

Week 9 Tutorial Exercise Solutions

Semester 1, 2025

Ask any questions that you may have about the lecture materials, etc. If there is still time then attempt the following questions.

1. Consider the problem of testing several hypotheses jointly. For example, consider a regression model of the form based on a sample of size n:

$$y = \beta_1 + \beta_2 x_1 + \cdots + \beta_{k+1} x_k + u$$

where $u \mid X \sim \mathrm{N}(0, \sigma^2 I_n)$, with $X = [\imath_n, X_1, \ldots, X_k]$ of full column rank such that $\operatorname{plim} n^{-1} X'X = Q > 0$. We shall use $\imath_n$ to denote a column of ones. Now suppose that you wish to test whether or not the model has any explanatory power at all. That is, you wish to test the null hypothesis $H_0 : \beta_1 = \cdots = \beta_k = 0$ against the alternative H_1 : at least one of $\beta_1, \ldots, \beta_k$ differs from zero. In an ideal world one might reach for an F test or any of an LR, LM, or Wald test to do this. (The latter tests all reduce to either an F test or something very close to it.) Unfortunately, you are not in a position to construct an F test and all the information that you have available to you are the t-tests on the individual coefficients. One approach to testing H_0 is to test $H^*_{0,i} : \beta_i = 0$ against $H^*_{1,i} : \beta_i \neq 0$ for $i = 2, \ldots, k+1$ and to accept H_0 if, and only if, you accept all of the $H^*_{0,i}$. In this exercise we wish to evaluate such a strategy. Unless instructed otherwise, we will assume that all tests are performed at the 5% level of significance and we will use critical values from a standard normal distribution. The t-statistics will simply be those computed by R's `summary.lm` command. Similarly, we will compare the procedure described above with performance of the F-statistic generated by R to test H_0. We will compare that statistic with its asymptotic critical values, i.e. $j \times F_{calc} \overset{H_O}{\underset{a}{\sim}} \chi^2_j$, where j is the number of restrictions being tested. Here $j = k$.

   (a) Set $n = 10$.

   (b) Generate the appropriate critical values to be used with the t-tests and F-tests and store them in the variables $cv1$ and $cv2$, respectively.

   Hint: Use the `qnorm` and `qchisq` commands, respectively.

   (c) Generate 3 variables, each with 100 observations, such that $x_1 \sim \mathrm{N}(0, 9)$, $x_2 \sim U[-1, 1]$, i.e., uniform on the interval $[-1, 1]$, and $x_3 + 2\imath_n \sim \chi^2_2$.

   Hint: The functions `rnorm`, `runif`, and `rchisq` are your friends here.

(d) Create 3 new variables called h, f, and F, of length n, and initialize them to be zero vector. Also create an $n \times 4$ array of zeros called T.

Hint: The `rep` function is probably handy here for the vectors and you can create T =`matrix(data=0,nrow=n,ncol=4)`.

(e) Complete the following step 1000 times:

i. Generate a variable $y \sim \mathrm{N}(0, 1)$.

ii. Run the regression $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + u$ and at the j-th iteration, save the 4 t-statistics on the coefficients as the j row of an array called `T`. So, at the end of the process the first column of `T` contains 100 t-statistics for β_0, and so on. Similarly, store the vaules of the F-statistics in the vector F.

Hint: Say you created the variable $mdl.stat$ to contain the output of your summary.lm command, then the t-statistic of β_k is available as $mdl.stat\$coefficients[k, 3]$, where k is the coefficient number, e.g. $k = 1, 2, 3$ here.

iii. If, across the j-th row of T, $H^*_{0,1}$, $H^*_{0,2}$, and $H^*_{0,2}$ are all accepted, set $h(j) = 1$.

iv. For each $F(j)$, if H_0 is accepted, set $f(j) = 1$.

(f) Determine the proportion of times that H_0 is accepted using the t-test method. Compare that with the proportion of times that H_0 is accepted using the F-test method. Is this what you expected? Describe what you have learned.

2. Repeat 1 with $n = 100$.

3. Try and work out what level of significance you should use with each t-test so that the overall size of the t-test approach is 5%. How does this size compare with the 5% used above?

Hint: You will for the most part be able to re-use your code from the Tutorial Exercise. My advice is to start with $n = 10$ but, once things are working, it is a simple matter to see if anything changes when you move to $n = 100$. If it were me then I would adopt a simple bisection method that will work as follows.

As a lower bound, set the size of each t-test to be 1. This means that you always reject $H^*_{0,i}$, and means a critical value of zero. As an upper bound, if the size were set to be zero, that would correspond to a critical value of infinity. Infinity is an ugly number to wrk with, so we can start with something similar to infinity for a $\mathrm{N}(0, 1)$ random variable, like 5. We want an overall size of 0.05, so, divide the interval in half and use 2.5 as your critical value for the individual tests. If the overall size is too big then you need a bigger critical value, so that you reject less frequently. Consequently, replace zero by 2.5 as the lower bound and repeat the process of using as your new critical value the mid-point between the lower and upper bound. Alternatively, if the overall size is too small, then replace the old upper bound by 2.5 and repeat the process of using as your new critical value the mid-point between the lower and upper bound. Once you find a critical value that gives you an overall size of 5% you can find the actual size of the individual tests using the `pnorm` command. Note that you should keep your matrix of results T fixed during all of this, you don't need to repeat that at every step. The bisection method is a slow way to go about this but it is simple and it works.

Solution:

Daniel has provided the program Week9_full.R that does all the work for us. It yields the following:

```
> print("The standard t critical value for this experiment is")
[1] "The standard t critical value for this experiment is"
> print(cv.t)
[1] 1.984723
> print("...and the size using the standard critical value is")
[1] "...and the size using the standard critical value is"
> print(mean(overall.rej))
[1] 0.151
> print("If we used the F-statistic, we get a rejection frequency of")
[1] "If we used the F-statistic, we get a rejection frequency of"
> print(mean(F.stats > cv.F))
[1] 0.052
> print("and if we compared it to the asymptotic chi-squared (3df) we get")
[1] "and if we compared it to the asymptotic chi-squared (3df) we get"
> print(mean(3*F.stats > cv.chisq))
[1] 0.059
> # as a final comment: so basically individual t-tests suck for
joint significance
```

The output says it all really. The one final thing to say is that performing multiple tests does not typically lead to an overall test with anything like the nominal size used to obtain your critical values.

For Question 3, Daniel's program again comes to our rescue. The solution obtained is:

```
> print("The critical value that achieves a 5% size")
[1] "The critical value that achieves a 5% size"
> print(t.crit)
[1] 2.392578
>
> print("But the standard critical value is")
[1] "But the standard critical value is"
> print(cv.t)
[1] 1.984723
> print("...and the size using the standard critical value is")
[1] "...and the size using the standard critical value is"
> print(mean(overall.rej))
[1] 0.151
```

We see that with multiple tests we need to use critical values with much different (smaller) sizes to control the overall size of the joint test.