Regression Inference| R programming language

Test for normality: Data exploration

Script Editor

```
wage1 <- read.csv("C:\\Users\\amalz\\OneDrive\\Desktop\\wage1.csv")
wage1 %>%
  select(wage, educ, exper, tenure, female) %>%
  head(05)
wage1 %>%
  select(wage, educ, exper, tenure, female) %>%
  str
wage1 %>%
  select(wage, educ, exper, tenure, female) %>%
  stargazer(type = "text")
```

Remember to install these packages beforehand:- "tidyverse", "stargazer", "magrittr", "car" . Refer previous notes on ways to set up the same.

Console

```
> wage1 %>%
+   select(wage, educ, exper, tenure, female) %>%
+   head(05)
  wage educ exper tenure female
1 3.10   11     2      0      1
2 3.24   12    22      2      1
3 3.00   11     2      0      0
4 6.00    8    44     28      0
5 5.30   12     7      2      0
> wage1 %>%
+   select(wage, educ, exper, tenure, female) %>%
+   str
'data.frame':	526 obs. of  5 variables:
 $ wage  : num  3.1 3.24 3 6 5.3 ...
 $ educ  : int  11 12 11 8 12 16 18 12 12 17 ...
 $ exper : int  2 22 2 44 7 9 15 5 26 22 ...
 $ tenure: int  0 2 0 28 2 8 7 3 4 21 ...
 $ female: int  1 1 0 0 0 0 0 1 1 0 ...
> wage1 %>%
+   select(wage, educ, exper, tenure, female) %>%
+   stargazer(type = "text")

==========================================
Statistic  N   Mean   St. Dev.  Min    Max
------------------------------------------
wage      526 5.896    3.693   0.530 24.980
educ      526 12.563   2.769     0     18
exper     526 17.017  13.572     1     51
tenure    526 5.105    7.224     0     44
female    526 0.479    0.500     0     1
------------------------------------------
```

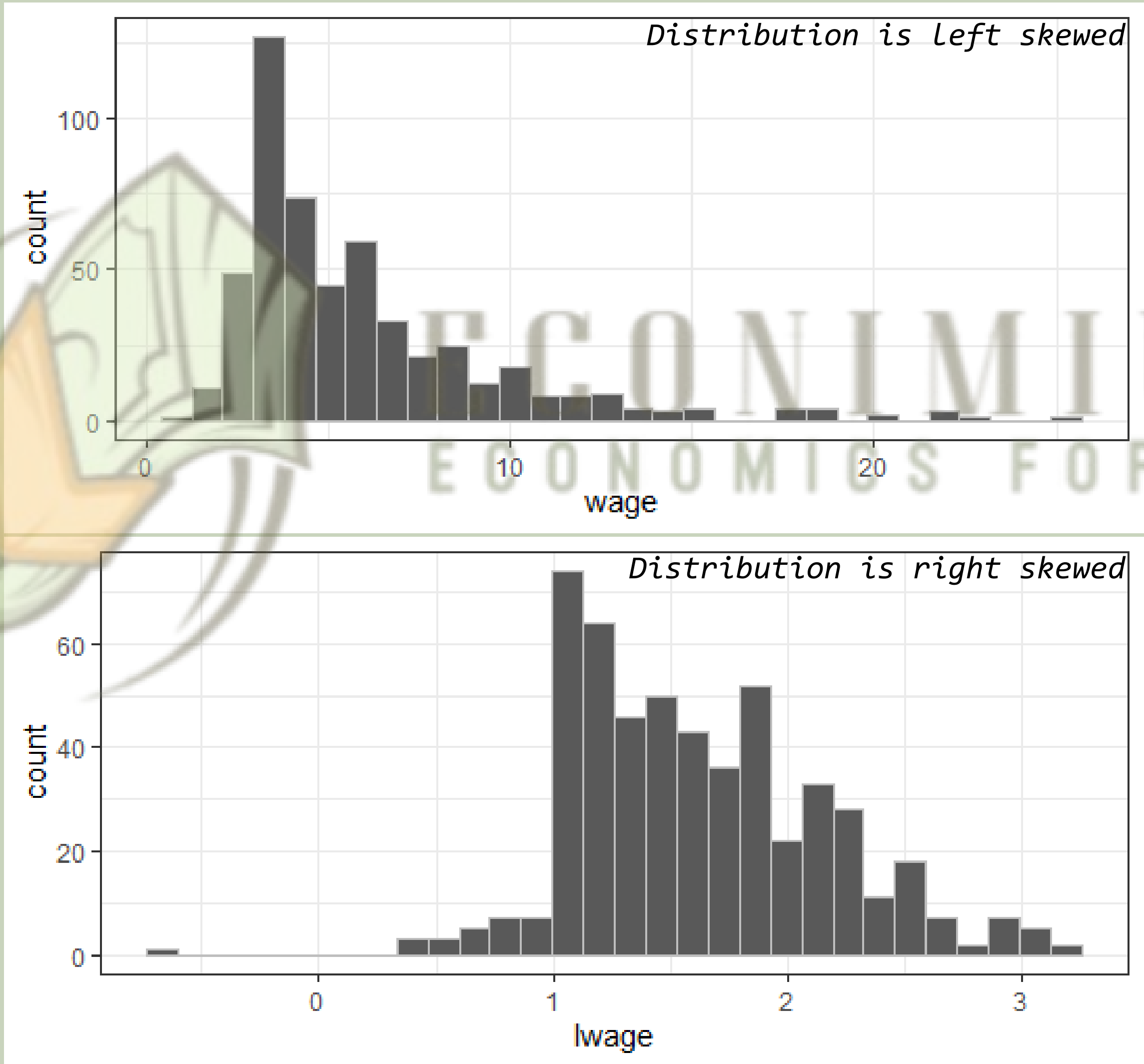
Regression Inference| R programming language

Test for normality: Visual Inspection

Script Editor

```
ggplot(data = wage1) +
  theme_bw() +
  geom_histogram(mapping = aes(x = wage), col = 'grey')

ggplot(data = wage1) +
  theme_bw() +
  geom_histogram(mapping = aes(x = lwage), col = 'grey')
```

Plots pane



Regression Inference| R programming language

Shapiro-Wilk test for normality

 # H0: normality, Ha: non-normality

Script Editor

```
shapiro.test(wage1$wage)
shapiro.test(wage1$lwage)
```

If p-value<0.05, the variable is not normally distributed.

Console

```
        Shapiro-Wilk normality test

data:  wage1$wage
W = 0.80273, p-value < 2.2e-16

> shapiro.test(wage1$lwage)

        Shapiro-Wilk normality test

data:  wage1$lwage
W = 0.96909, p-value = 4.423e-09
```


Hypothesis testing: t-test for coefficient significance

Script Editor

```
# Run Regression
model <- lm(wage ~ educ + exper + tenure + female, wage1)
summary(model)

# Hypothesis testing method 1: compare t-statistic with t-critical value(s)

# Display coefficient
(coefficient <- coef(model)["exper"])

# Display standard error
(se <- vcov(model) %>% # variance-covariance matrix
    diag %>% # extract diagonals
    sqrt %>% # calculate square-roots
    .["exper"]) # S.E. of the regressor "exper"

# Calculate t-statistic = coefficient/standard error
(tstat <- coefficient/se)

# Degrees of freedom (n-k-1)
(df_r <- model$df.residual)

# t-critical value at 5% significance level
qt(p = 0.975, df = df_r, lower.tail = TRUE)
```

If t-statistic is in the rejection region then reject null, the coefficient is significant

Regression Inference| R programming language

Console

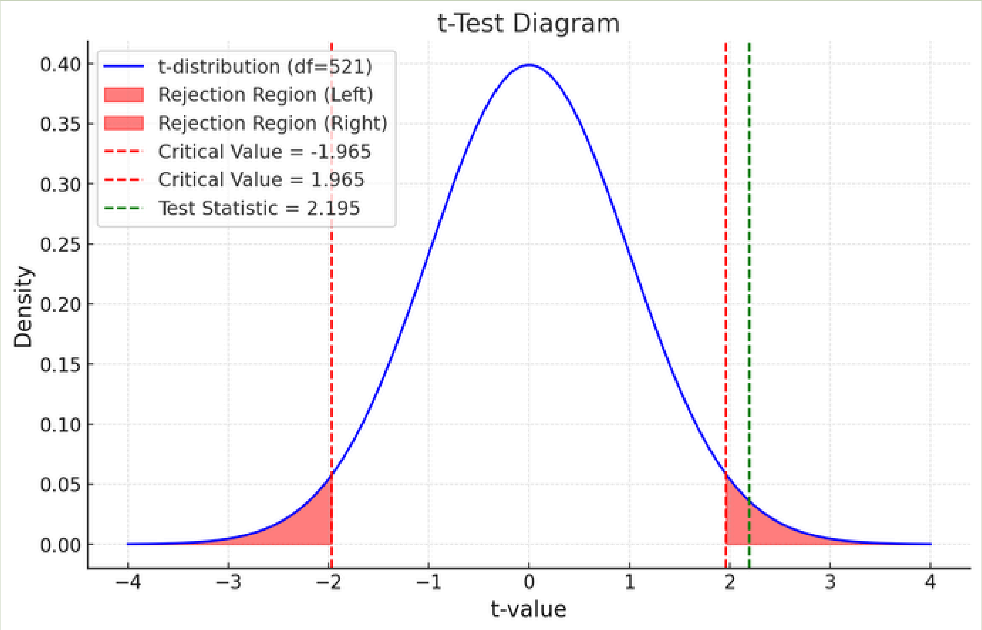
```
> model <- lm(wage ~ educ + exper + tenure + female, wage1)
> summary(model)

Call:
lm(formula = wage ~ educ + exper + tenure + female, data = wage1)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.7675 -1.8080 -0.4229  1.0467 14.0075 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.56794    0.72455  -2.164   0.0309 *  
educ         0.57150    0.04934  11.584  < 2e-16 ***
exper        0.02540    0.01157   2.195   0.0286 *  
tenure       0.14101    0.02116   6.663 6.83e-11 ***
female      -1.81085    0.26483  -6.838 2.26e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.958 on 521 degrees of freedom
Multiple R-squared:  0.3635,	Adjusted R-squared:  0.3587 
F-statistic:  74.4 on 4 and 521 DF,  p-value: < 2.2e-16

> # Display coefficient
> (coefficient <- coef(model)["exper"])
     exper 
0.02539587 
> # Display standard error
> (se <- vcov(model) %>% # variance-covariance matrix
+     diag %>% # extract diagonals
+     sqrt %>% # calculate square-roots
+     .["exper"]) # S.E. of the regressor "exper"
     exper 
0.01156943 
> # Calculate t-statistic = coefficient/standard error
> (tstat <- coefficient/se)
   exper 
2.195083 
> # Degrees of freedom (n-k-1)
> (df_r <- model$df.residual)
[1] 521
> # t-critical value at 5% significance level
> qt(p = 0.975, df = df_r, lower.tail = TRUE)
[1] 1.964528
```

Calculations for exper

The code calculates metrics for the exper coefficient:

1. Coefficient: $\text{Estimate} = 0.02540$.
2. Standard Error: $\text{Std. Error} = 0.01157$.
3. t-statistic: $t = \frac{\text{Estimate}}{\text{Std. Error}} = \frac{0.02540}{0.01157} = 2.195083$.
4. Degrees of Freedom: $df = 521$ (calculated as $n - k - 1$).
5. t-Critical Value: $t_{\alpha/2,df} = qt(0.975, df = 521) = 1.964528$.

Since , t stat> t critical, we reject the null hypothesis for exper, concluding that exper has a statistically significant effect on wage.



Regression Inference| R programming language

Hypothesis testing: Comparing p-value with significance level (usually 5%)

Script Editor

P-value for a two-tailed test of coefficient significance
H0: beta[exper] = 0; H1: beta[exper] not equal to 0

```
2 * pt(q = abs(tstat), df = df_r, lower.tail = FALSE)
pt(q = tstat, df = df_r, lower.tail = FALSE)
pt(q = tstat, df = df_r, lower.tail = TRUE)
```

P-value for an upper one-tailed test of positive coefficient
H0: beta[exper]<=0; H1: beta[exper]>0

P-value for a lower one-tailed test of negative coefficient
H0: beta[exper]>=0; H1: beta[exper]<0

Console

```
> 2 * pt(q = abs(tstat), df = df_r, lower.tail = FALSE)
    exper
0.02859807
> pt(q = tstat, df = df_r, lower.tail = FALSE)
    exper
0.01429904
> pt(q = tstat, df = df_r, lower.tail = TRUE)
   exper
0.985701
```

If p-value < significance level then reject null, coefficient is significant

Hypothesis testing: Calculating confidence intervals

Script Editor

```
qt(p = 0.975, df = df_r)   <-------- Critical value at 5% significance level
coefficient - 1.96 * se     <-------- Lower bound at 95% confidence level
coefficient + 1.96 * se     <-------- Upper bound at 95% confidence level
```

Console

```
> qt(p = 0.975, df = df_r)
[1] 1.964528
> coefficient - 1.96 * se
      exper
0.002719775
> coefficient + 1.96 * se
     exper
0.04807196
```

If confidence interval does not contain 0 then reject null, coefficient is significant

Commonly used z-values for both one-tailed and two-tailed tests at various confidence levels

Confidence Level	Significance Level (α)	Two-Tailed z-Value ($\pm z_{\alpha/2}$)	One-Tailed z-Value (z_{α})
90%	0.10	1.645	1.282
95%	0.05	1.960	1.645
99%	0.01	2.576	2.326
99.9%	0.001	3.291	3.090

The three methods of comparing t-statistic with critical values, p-value with significance level, and confidence intervals lead to the same conclusion.

Regression Inference| R programming language

F-test for single coefficient significance

F-test for single coefficient significance is an alternative to t-test.

[1]*The first step involves estimating the unrestricted model, which includes all the explanatory variables and the restricted model(excluding the variable of interest).*

[2]*Next, we formulate the null hypothesis (H_0) that the coefficient of the variable of interest is equal to zero (no effect). The alternative hypothesis (H_1) suggests that the coefficient is not equal to zero (the variable does have an effect).*

[3]*The next step is to calculate the sum of squared residuals (SSR) of both models.*

[4]*After estimating the model, we calculate the F-statistic, which compares the restricted model with the full model. The F-statistic is computed as the ratio of the difference in the sum of squared residuals between the two models to the number of restrictions, divided by the residual mean square error from the full model.i.e. F-stat = ((ssr_r-ssr_ur)/q) / (ssr_ur/(n-k-1))*

[5]*The final step involves comparing the calculated F-statistic to the critical value from the F-distribution at a given significance level (usually 5%). If the calculated F-statistic exceeds the critical value, we reject the null hypothesis, indicating that the coefficient is significantly different from zero.*

[#] *The steps [1] to [5] can be avoided using the Linear hypothesis test that gives the F test results in a single command.*

Script Editor


```
# Unrestricted model:
# wage = beta0 + beta1*educ + beta2*exper + beta3*tenure + beta4*female + u
model_ur <-  lm(wage ~ educ + exper + tenure + female, wage1)
summary(model_ur)                                                    [1]
# Restricted model:
#   wage = alpha0 + alpha1*educ + alpha3*tenure + alpha4*female + e
model_r1 <- lm(wage ~ educ + tenure + female, wage1)
summary(model_r1)


# SSR for the unrestricted model = ssr_ur, q = number of restrictions and df_denom=n-k-1
(ssr_ur <- sum(resid(model_ur)^2))
(df_ur  <- model_ur$df.residual) # df_ur = n - k - 1
q <- 1   q= number of restrictions (here 1 as just one variable is restricted   [3]
# SSR for the restricted model ssr_r
(ssr_r1 <- sum(resid(model_r1)^2))

# Calculate F-stat using ssr_r and ssr_ur
# F-stat = ((ssr_r-ssr_ur)/q) / (ssr_ur/(n-k-1))                     [4]
(F_stat <- ((ssr_r1 - ssr_ur)/q) / (ssr_ur/df_ur))

# Calculate F-critical value
qf(p = 0.95, df1 = 1, df2 = df_ur)
# If F-stat > F-critical value then reject null, coefficient is significant   [5]

# p-value for F-test
(F_pvalue <- pf(q = F_stat, df1 = 1, df2 = df_ur, lower.tail = F))
# If F p-value<0.05 then reject null, coefficient is significant

# F-test for coefficient significance using R commands
linearHypothesis(model, "exper = 0")                                 [#]
```

Note: The F-statistic is different from the t-statistic for the coeff on exper, but the p-value is same for the F-test and t-test.

Regression Inference| R programming language

Console

```
Call:
lm(formula = wage ~ educ + exper + tenure + female, data = wage1)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.7675 -1.8080 -0.4229  1.0467 14.0075 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.56794    0.72455  -2.164   0.0309 *  
educ         0.57150    0.04934  11.584  < 2e-16 ***
exper        0.02540    0.01157   2.195   0.0286 *  
tenure       0.14101    0.02116   6.663 6.83e-11 ***
female      -1.81085    0.26483  -6.838 2.26e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.958 on 521 degrees of freedom
Multiple R-squared:  0.3635,	Adjusted R-squared:  0.3587 
F-statistic:  74.4 on 4 and 521 DF,  p-value: < 2.2e-16

Call:
lm(formula = wage ~ educ + tenure + female, data = wage1)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.5184 -1.8074 -0.4477  1.0270 14.1229 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -0.84503    0.64774  -1.305    0.193    
educ         0.53799    0.04709  11.425  < 2e-16 ***
tenure       0.16441    0.01835   8.962  < 2e-16 ***
female      -1.78839    0.26559  -6.734 4.38e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

[1]

```
Residual standard error: 2.968 on 522 degrees of freedom
Multiple R-squared:  0.3577,	Adjusted R-squared:  0.354 
F-statistic: 96.88 on 3 and 522 DF,  p-value: < 2.2e-16
> # SSR for the unrestricted model = ssr_ur, q = number of restrictions and df_denom=n-k-1
> (ssr_ur <- sum(resid(model_ur)^2))
[1] 4557.308
> (df_ur  <- model_ur$df.residual) # df_ur = n - k - 1
[1] 521
> q <- 1
> # SSR for the restricted model ssr_r
> (ssr_r1 <- sum(resid(model_r1)^2))
[1] 4599.455
```

[3]

```
> # Calculate F-stat using ssr_r and ssr_ur
> # F-stat = ((ssr_r-ssr_ur)/q) / (ssr_ur/(n-k-1))
> (F_stat <- ((ssr_r1 - ssr_ur)/q) / (ssr_ur/df_ur))
```

[4]

```
[1] 4.818389
> # Calculate F-critical value
> qf(p = 0.95, df1 = 1, df2 = df_ur)
[1] 3.859369
```

[5]

```
> # p-value for F-test
> (F_pvalue <- pf(q = F_stat, df1 = 1, df2 = df_ur, lower.tail = F))
[1] 0.02859807
> # F-test for coefficient significance using R commands
> linearHypothesis(model, "exper = 0")

Linear hypothesis test:
exper = 0

Model 1: restricted model
Model 2: wage ~ educ + exper + tenure + female

  Res.Df    RSS Df Sum of Sq      F Pr(>F)  
1    522 4599.5                             
2    521 4557.3  1    42.148 4.8184 0.0286 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

[#]

Regression Inference| R programming language

F-test for joint coefficient significance

As above, the same can also be done elaborately or (as shown below) can be done using the Linear Hypothesis test.

Script Editor

```
linearHypothesis(model_ur, c("exper = 0", "tenure = 0"))
```

Console

```
> # F-test using R commands
> linearHypothesis(model_ur, c("exper = 0", "tenure = 0"))

Linear hypothesis test:
exper = 0
tenure = 0

Model 1: restricted model
Model 2: wage ~ educ + exper + tenure + female

  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1    523 5307.2
2    521 4557.3  2    749.85 42.862 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

F-test for overall significance of regression

Script Editor

```
linearHypothesis(model_ur, c("educ = 0", "exper = 0",
                             "tenure = 0", "female = 0"))
```

Console

```
Linear hypothesis test:
educ = 0
exper = 0
tenure = 0
female = 0

Model 1: restricted model
Model 2: wage ~ educ + exper + tenure + female

  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1    525 7160.4
2    521 4557.3  4    2603.1 74.398 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Lagrange Multiplier test for coefficient significance

The Lagrange Multiplier (LM) test for coefficient significance is used to test if a particular coefficient in a regression model is significantly different from zero, without estimating the full model. The test begins with the specification of the null hypothesis that the coefficient is zero, and the alternative that it is non-zero. The LM statistic is computed using the difference between the unrestricted model (which includes all variables) and a restricted model (where the coefficient of interest is set to zero). The statistic is then compared to a Chi-square distribution with degrees of freedom equal to the number of restrictions. If the LM statistic exceeds the critical value from the Chi-square distribution, the null hypothesis is rejected, indicating that the coefficient is statistically significant.

Regression Inference| R programming language

```
# Unrestricted model:
# wage = beta0 + beta1*educ + beta2*exper + beta3*tenure + beta4*female + u
# H0: beta[exper]=0 beta[tenure]=0
# Restricted model: wage = alpha0 + alpha1*educ + alpha4*female + e
# Regress dependent variable on the restricted set of independent variables
```

Script Editor

```
model_lmt<-lm(wage ~ educ + female, wage1)
summary(model_lmt)
q <- 2 # 2 restrictions
# Get residuals ehat
wage1$ehat <- residuals(model_lmt)
# Regress residuals ehat on all independent variables
model_ehat <- lm(ehat ~ educ + exper + tenure + female, wage1)
summary(model_ehat)
# LM statistic = n*(R_e)^2 = n*R-squared of above regression
n <- nobs(model_ehat)
R_e2 <- summary(model_ehat)$r.squared
(LM_stat <- n*R_e2)
# Critical value for chi-square distribution
qchisq(p = 0.95, df = q)
# P-value for chi-square distribution
pchisq(LM_stat, df = q, lower.tail = FALSE)
```

Console

```
Call:
lm(formula = wage ~ educ + female, data = wage1)

Residuals:
    Min      1Q  Median      3Q     Max 
-5.9890 -1.8702 -0.6651  1.0447 15.4998 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.62282    0.67253   0.926    0.355    
educ         0.50645    0.05039  10.051  < 2e-16 ***
female      -2.27336    0.27904  -8.147 2.76e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.186 on 523 degrees of freedom
Multiple R-squared:  0.2588,	Adjusted R-squared:  0.256 
F-statistic: 91.32 on 2 and 523 DF,  p-value: < 2.2e-16

> q <- 2 # 2 restrictions
> # Get residuals ehat
> wage1$ehat <- residuals(model_lmt)
> # Regress residuals ehat on all independent variables
> model_ehat <- lm(ehat ~ educ + exper + tenure + female, wage1)
> # LM statistic = n*(R_e)^2 = n*R-squared of above regression
> n <- nobs(model_ehat)
> R_e2 <- summary(model_ehat)$r.squared
> (LM_stat <- n*R_e2)
[1] 74.31901
> # Critical value for chi-square distribution
> qchisq(p = 0.95, df = q)
[1] 5.991465
> # P-value for chi-square distribution
> pchisq(LM_stat, df = q, lower.tail = FALSE)
[1] 7.274985e-17
```

You can directly write the values of n and R_e2 by looking at their corresponding values from the result of the model_ehat (command- summary(model_ehat)) *from the console or you can use the commands* n <- nobs(model_ehat) *and* R_e2 <- summary(model_ehat)$r.squared beforehand as shown here.

 # If p-value < 0.05 then reject null, coefficients are jointly significant
 # If LM-stat > chi2 critical value then reject null, coefficients are jointly significant