Notebook

March 15, 2021

Question 1.a. Set the sample size at 1,000 and generate an error term, u_i, by randomly selecting from a normal distribution with mean 0, and standard deviation 5. Draw an explanatory variable, X_{1i}, from a standard normal distribution, $\mathcal{N}(0,1)$, and then define a second explanatory variable, X_{2i}, to be equal to $e^{X_{1i}}$ for all i. Finally, set the dependent variable to be linearly related to the two regressors plus an additive error term: $y_i = 2 + 4X_{1i} - 6X_{2i} + u_i$. Note that, by construction, the error term of this multivariate linear regression is homoskedastic.

Hint: You may want to refer to how you did this in Problem Set 2. Also, the function `np.exp()` takes a list/array of numbers and applies the exponential function to each element. This is basically the opposite funciton of `np.log()`.

[21]:

```
u = np.random.normal(0, 5, 1000)
X1 = np.random.normal(0, 1 , 1000)
X2 = np.exp(X1)
y = 2 + 4 * X1 - 6 * X2 + u
```

Question 1.b. Regress y on X_1 with homoskedasticity-only standard errors (`statsmodels` does this by default, just don't specify a `cov_type` like we usually do to get robust errors). Do the same analysis for y and X_2. Compare the results with the true data generating process. Explain why differences arise between the population slopes and the estimated slopes, if there are any.

This question is for your code, the next is for your explanation.

[22]:

```
X1_const = sm.add_constant(X1)
model_1b_X1 = sm.OLS(y, X1)
results_1b_X1 = model_1b_X1.fit()
results_1b_X1.summary()
```

[22]:

```
<class 'statsmodels.iolib.summary.Summary'>
"""
                                OLS Regression Results
================================================================================
=======
Dep. Variable:                      y   R-squared (uncentered):
0.183
Model:                            OLS   Adj. R-squared (uncentered):
0.182
Method:                 Least Squares   F-statistic:
223.4
```

```
Date:                  Mon, 15 Mar 2021   Prob (F-statistic):
9.70e-46
Time:                          12:56:13   Log-Likelihood:
-3900.3
No. Observations:                  1000   AIC:
7803.
Df Residuals:                       999   BIC:
7808.
Df Model:                             1
Covariance Type:              nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
x1            -5.3962      0.361    -14.947      0.000      -6.105      -4.688
==============================================================================
Omnibus:                      743.282   Durbin-Watson:                   1.122
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            20813.423
Skew:                          -3.087   Prob(JB):                         0.00
Kurtosis:                      24.481   Cond. No.                         1.00
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 1.c. Explain.

After running the code many times, the estimated slopes were similar to the population slope. Differences may arise because we're taking a simple random sample. There were deviations for both X1 and X2 explanatory variable because we're only taking into account one variable on each simulation.

Question 1.d. Next, regress y on both X_1 and X_2. Compare the estimation results with those you did in part (b/c), especially the model with only the regressor X_1. Examine differences across the three regressions in terms of the coefficient estimates, their standard errors, the R^2, and the adjusted R^2.

This question is for your code, the next is for your explanation.

[24]:

```
X_const = sm.add_constant(np.stack([X1, X2], axis=1)) # This just puts our two␣
 ↪variables together with a const
model_1d = sm.OLS(y, X_const)
results_1d = model_1d.fit()
results_1d.summary()
```

[24]:

```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
```

```
==============================================================================
Dep. Variable:                      y   R-squared:                       0.789
Model:                            OLS   Adj. R-squared:                  0.789
Method:                 Least Squares   F-statistic:                     1867.
Date:                Mon, 15 Mar 2021   Prob (F-statistic):               0.00
Time:                        12:56:13   Log-Likelihood:                -2997.4
No. Observations:                1000   AIC:                             6001.
Df Residuals:                     997   BIC:                             6016.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          1.9744      0.255      7.751      0.000       1.475       2.474
x1             3.8499      0.239     16.101      0.000       3.381       4.319
x2            -5.8852      0.120    -48.843      0.000      -6.122      -5.649
==============================================================================
Omnibus:                        0.499   Durbin-Watson:                   1.933
Prob(Omnibus):                  0.779   Jarque-Bera (JB):                0.586
Skew:                           0.034   Prob(JB):                        0.746
Kurtosis:                       2.903   Cond. No.                         6.15
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 1.e. Explain.

It fits the expectation better because we are taking into account both variables at the same time. As a result, the coefficient estimate is very close to the population coefficient estimate.

Question 1.f. Generate a third regressor: $X_{3i} = 1 + X_{1i} - X_{2i} + v_i$ where v_i is drawn from a normal distribution with mean 0 and standard deviation 0.5. Estimate the model $y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{3i} + w_i$. Compare the result with part (d/e). Do changes in OLS estimates, standard errors, the R^2, and the adjusted R_2 make sense to you? Explain why or why not.

Hint: Think about the concept of "imperfect multicollinearity".

This question is for your code, the next is for your explanation.

[25]:

```
v = np.random.normal(0, 0.5, 1000)
X3 = 1 + X1 - X2 + v

X_const_f = sm.add_constant(np.stack([X1, X2, X3], axis=1))
model_1f = sm.OLS(y, X_const_f)
results_1f = model_1f.fit()
results_1f.summary()
```

[25]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.789
Model:                            OLS   Adj. R-squared:                  0.789
Method:                 Least Squares   F-statistic:                     1243.
Date:                Mon, 15 Mar 2021   Prob (F-statistic):               0.00
Time:                        12:56:13   Log-Likelihood:                -2997.4
No. Observations:                1000   AIC:                             6003.
Df Residuals:                     996   BIC:                             6023.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          2.0053      0.389      5.160      0.000       1.243       2.768
x1             3.8817      0.385     10.083      0.000       3.126       4.637
x2            -5.9166      0.321    -18.407      0.000      -6.547      -5.286
x3            -0.0313      0.296     -0.106      0.916      -0.613       0.550
==============================================================================
Omnibus:                        0.508   Durbin-Watson:                   1.933
Prob(Omnibus):                  0.776   Jarque-Bera (JB):                0.595
Skew:                           0.034   Prob(JB):                        0.742
Kurtosis:                       2.901   Cond. No.                         13.7
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 1.g. Explain.

It does make sense that the OLS estimates are more varied and that the standard errors have increased because adding a third explanatory variable may affect the covariance between the first two explanatory variables. As a result, the R^2 and the adjusted R_2 depends on the effects that the third explanatory variable has on the first two explanatory variables.

Question 2.a. Run a regression of `course_eval` on `beauty` using robust standard errors. What is the estimated slope? Is it statistically significant?

This question is for your code, the next is for your explanation.

[27]:
```
y_2a = ratings['course_eval']
X_2a = ratings['beauty']
model_2a = sm.OLS(y_2a, X_2a)
results_2a = model_2a.fit()
results_2a.summary()
```

[27]:

```
<class 'statsmodels.iolib.summary.Summary'>
"""
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:            course_eval   R-squared (uncentered):
0.001
Model:                            OLS   Adj. R-squared (uncentered):
-0.001
Method:                 Least Squares   F-statistic:
0.3115
Date:                Mon, 15 Mar 2021   Prob (F-statistic):
0.577
Time:                        12:56:13   Log-Likelihood:
-1302.9
No. Observations:                 463   AIC:
2608.
Df Residuals:                     462   BIC:
2612.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
beauty         0.1330      0.238      0.558      0.577      -0.335       0.601
==============================================================================
Omnibus:                       15.399   Durbin-Watson:                   0.026
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               16.405
Skew:                          -0.453   Prob(JB):                     0.000274
Kurtosis:                       2.831   Cond. No.                         1.00
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 2.b. Explain.

The estimated slope is about 0.1330, and that this regression is not statistically significant.

Question 2.c. Run a regression of `course_eval` on `beauty`, including some additional variables to control for the type of course and professor characteristics. In particular, include as additional regressors `intro`, `onecredit`, `female`, `minority`, and `nnenglish`. What is the estimated effect of `beauty` on `course_eval`? Does the regression in (a) suffer from important omitted variable bias (OVB)? What happens with the R^2? Based on the confidence interval from the regression, can you reject the null hypothesis that the effect of beauty is the same as in part (a)? What can you say about the effect of the new variables included?

This question is for your code, the next is for your explanation.

```
[28]: y_2c = y_2a
      X_2c = sm.add_constant(ratings[['beauty', 'intro', 'onecredit', 'female',␣
       ↪'minority', 'nnenglish']])
      model_2c = sm.OLS(y_2c, X_2c)
      results_2c = model_2c.fit()
      results_2c.summary()
```

```
[28]: <class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:            course_eval   R-squared:                       0.155
Model:                            OLS   Adj. R-squared:                  0.144
Method:                 Least Squares   F-statistic:                     13.90
Date:                Mon, 15 Mar 2021   Prob (F-statistic):           1.53e-14
Time:                        12:56:13   Log-Likelihood:                -344.85
No. Observations:                 463   AIC:                             703.7
Df Residuals:                     456   BIC:                             732.7
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          4.0683      0.038    108.364      0.000       3.995       4.142
beauty         0.1656      0.031      5.389      0.000       0.105       0.226
intro          0.0113      0.054      0.208      0.835      -0.096       0.118
onecredit      0.6345      0.111      5.699      0.000       0.416       0.853
female        -0.1735      0.049     -3.520      0.000      -0.270      -0.077
minority      -0.1666      0.076     -2.184      0.029      -0.317      -0.017
nnenglish     -0.2442      0.107     -2.283      0.023      -0.454      -0.034
==============================================================================
Omnibus:                       22.413   Durbin-Watson:                   1.516
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               24.406
Skew:                          -0.555   Prob(JB):                     5.02e-06
Kurtosis:                       3.179   Cond. No.                         5.81
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 2.d. Explain.

The impact that beauty had on course_eval is that the coefficient is now 0.1656, which increased from the previous OLS regression result. The regression in (a) did not incorporate other variables in the OLS regression results, which could cause omitted variable bias. Now that we have more variables into the regression, R^2 has increased to 0.155. From part (a), our beauty coefficient was within the confidence interval that we got from the regression result in part (c). Therefore, we fail to reject the null hypothesis. The effect of the new variables is that the confidence interval range became smaller, which helps condense the variance between beauty and course evaluation.

Question 2.e. Estimate the coefficient on beauty for the multiple regression model in (c) using the three-step process in Appendix 6.3 (the Frisch-Waugh theorem). Verify that the three-step process yields the same estimated coefficient for beauty as that obtained in (c). Comment.

Hint: Recall that if your regression results are called `results`*, you could get the residuals using* `results.resid`*.*

This question is for your code, the next is for your explanation.

[29]:

```
# Do the first step here (regress the outcome variable on covariates)
course_eval = ratings['course_eval']
covariates = sm.add_constant(ratings[['intro', 'onecredit', 'female',␣
 ↪'minority', 'nnenglish']])
model_eval_on_covariates = sm.OLS(course_eval, covariates)
results_eval = model_eval_on_covariates.fit()
eval_residuals = results_eval.resid

# Do the second step here (regress the explanatory variable on covariates)
beauty = ratings['beauty']
model_beauty_on_covariates = sm.OLS(beauty, covariates)
results_beauty = model_beauty_on_covariates.fit()
beauty_residuals = results_beauty.resid

# Do the last step here (regress the outcome variable's residuals on the␣
 ↪explanatory variable's residuals)
model_fw = sm.OLS(eval_residuals, beauty_residuals)
results_fw = model_fw.fit()
results_fw.summary()
```

[29]:

```
<class 'statsmodels.iolib.summary.Summary'>
"""
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                      y   R-squared (uncentered):
0.060
Model:                            OLS   Adj. R-squared (uncentered):
0.058
Method:                 Least Squares   F-statistic:
29.43
Date:                Mon, 15 Mar 2021   Prob (F-statistic):
9.39e-08
Time:                        12:56:14   Log-Likelihood:
```

```
-344.85
No. Observations:                 463   AIC:
691.7
Df Residuals:                     462   BIC:
695.8
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
x1             0.1656      0.031      5.425      0.000       0.106       0.226
==============================================================================
Omnibus:                       22.413   Durbin-Watson:                   1.516
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               24.406
Skew:                          -0.555   Prob(JB):                     5.02e-06
Kurtosis:                       3.179   Cond. No.                         1.00
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 2.f. Explain.

We end up getting the same results as the OLS regression result in part (c) because in the first step, we did a multiple regression on the other variables to clean out the effects that they have for both the independent and dependent variables. Afterwards, the third step performs an OLS regression on the error terms to get the same OLS regression results that is in part (c).

Question 2.g. Professor Smith is a black male with average beauty and is a native English speaker. He teaches a three-credit upper-division course. Predict Professor Smith's course evaluation.

Course_eval = β_0 hat + β_1 hat * beauty + β_2 hat * intro + β_3 hat * onecredit + β_4 hat * female + β_5 hat * minority + β_6 hat * nnenglish + u_i

Course_eval = 4.0683 + 0.1656 * 0 + 0.0113 * 0 + 0.6345 * 0 - 0.1735 * 0 - 0.1666 * 1 - 0.2442 * 0 + u_i

Course_eval = 4.0683 - 0.1666 * 1 + u_i

Course_eval = 3.9017 + u_i

Question 3.a. What do you expect for the sign of the relationship and what mechanism can you think about to explain it?

I would expect that if a person is closer to a college that they would have more years of education completed. This would make sense because the people closer to colleges would probably have both easier accessability to college, and also potentially have cheaper education because they could live at home while studying.

Question 3.b. Run a regression of years of completed education (`yrsed`) on distance to the nearest college (`dist`), measured in tens of miles (For example, dist = 2 means that the distance is 20 miles). What is the estimated slope? Is it statistically significant? Does distance to college explain a large fraction of the variance in educational attainment across individuals? Explain.

This question is for your code, the next is for your explanation.

[31]:
```
y_3b = dist["yrsed"]
X_3b = dist['dist']
model_3b = sm.OLS(y_3b, X_3b)
results_3b = model_3b.fit()
results_3b.summary()
```

[31]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                                 OLS Regression Results
================================================================================
=======
Dep. Variable:                  yrsed   R-squared (uncentered):
0.378
Model:                            OLS   Adj. R-squared (uncentered):
0.378
Method:                 Least Squares   F-statistic:
2304.
Date:                Mon, 15 Mar 2021   Prob (F-statistic):
0.00
Time:                        12:56:15   Log-Likelihood:
-14489.
No. Observations:                3796   AIC:
2.898e+04
Df Residuals:                    3795   BIC:
2.899e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
dist           3.1246      0.065     48.001      0.000       2.997       3.252
==============================================================================
Omnibus:                     2096.656   Durbin-Watson:                   0.136
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            19257.017
Skew:                          -2.495   Prob(JB):                         0.00
Kurtosis:                      12.841   Cond. No.                         1.00
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 3.c. Explain.

The estimated slope is approximately positive 3.12, which is statistically significant. This data leads us to expect that the greater the distance to a college, the more likely they are to have a greater amount of years of education. This could be a skewed result due to where people were sampled. For example, because the sample students were already in highschool, that may block out students who dont even have access to highschool, causing a biased result.

Question 3.d. Now run a regression of `yrsed` on `dist`, but include some additional regressors to control for characteristics of the student, the student's family, and the local labor market. In particular, include as additional regressors: `bytest`, `female`, `black`, `hispanic`, `incomehi`, `ownhome`, `dadcoll`, `cue80`, and `stwmfg80`. What is the estimated effect of `dist` on `yrsed`? Is it substantively different from the regression in (b)? Based on this, does the regression in (b) seem to suffer from important omitted variable bias?

This question is for your code, the next is for your explanation.

[32]:

```
y_3d = dist['yrsed']
X_3d = sm.add_constant(dist[['dist', 'bytest', 'female', 'black', 'hispanic',␣
 ↪'incomehi', 'ownhome', 'dadcoll', 'cue80', 'stwmfg80']])
model_3d = sm.OLS(y_3d, X_3d)
results_3d = model_3d.fit()
results_3d.summary()
```

[32]:

```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                  yrsed   R-squared:                       0.279
Model:                            OLS   Adj. R-squared:                  0.277
Method:                 Least Squares   F-statistic:                     146.3
Date:                Mon, 15 Mar 2021   Prob (F-statistic):         6.94e-260
Time:                        12:56:16   Log-Likelihood:                -7025.9
No. Observations:                3796   AIC:                         1.407e+04
Df Residuals:                    3785   BIC:                         1.414e+04
Df Model:                          10
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          8.8275      0.250     35.271      0.000       8.337       9.318
dist          -0.0315      0.012     -2.550      0.011      -0.056      -0.007
bytest         0.0938      0.003     29.669      0.000       0.088       0.100
female         0.1454      0.051      2.874      0.004       0.046       0.245
black          0.3680      0.071      5.156      0.000       0.228       0.508
hispanic       0.3985      0.074      5.352      0.000       0.253       0.545
incomehi       0.3952      0.061      6.529      0.000       0.277       0.514
ownhome        0.1521      0.067      2.277      0.023       0.021       0.283
dadcoll        0.6961      0.069     10.129      0.000       0.561       0.831
```

```
cue80           0.0232      0.010      2.409      0.016       0.004       0.042
stwmfg80       -0.0518      0.020     -2.608      0.009      -0.091      -0.013
==============================================================================
Omnibus:                      118.266   Durbin-Watson:                   1.924
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               97.867
Skew:                           0.320   Prob(JB):                     5.60e-22
Kurtosis:                       2.543   Cond. No.                         539.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

Question 3.e. Explain.

The estimated effect of dist on yrsed is now a -0.0315, which is substantially different than in part b. The regression from part b does seem to suffer from VERY important ommited variable bias because not only do we have a vast difference in magnitude of the coefficient, but we also have the opposite sign of the coefficient.

Question 3.f. The value of the coefficient on `dadcoll` is positive. What does this coefficient measure? Interpret this effect.

This coefficient measures the association between the father's graduation of college and the number of years of college that the student has taken. This means that the father graduating college would increase the student's years of college by 0.6961.

Question 3.g. Explain why `cue80` and `stwmfg80` appear in the regression. Are the signs of their estimated coefficients what you would have believed? Explain.

The `cue80`regressor is the county unemployment in 1980 and `stwmfg80` regressor is the state hourly wage in manufacturing in 1980. In the county unemployment regressor, it is a positive coefficient, which infers that with a greater unemployment rate, we should see more years of student education. This makes sense because there's higher competition within the county, which would create a greater barrier to entry in the workforce. The state hourly wage in manufacturing has a negative coefficient, which indicates that the greater the hourly wage, the less years of education that students would have. This makes sense because if people had the option to work at a high wage job without an education, then there's a higher opportunity cost to go to college.

Question 3.h. Bob is a black male. His high school was 20 miles from the nearest college. His base-year composite test score (`bytest`) was 58. His family income in 1980 was \$26,000, and his family owned a home. His mother attended college, but his father did not. The unemployment rate in his county was 7.5%, and the state average manufacturing hourly wage was \$9.75. Predict Bob's years of completed schooling using the regression in (d).

$$years of education = 8.8275 + 2.0 * -0.0315 + 58 * 0.0938 + 1 * 0.3952 + 0 * 0.6961 + 0.075 * 0.0232 + 9.75 * -0.0518 + 0 * 0.1454$$

$$years of education = 14.617$$

Bob has completed approximately 15 years of education.

Question 4.a. Why do you think Jaeger and Page estimate their model using only people of a single race and gender (in this particular case the sample consists of white males)?

The purpose of of controlling the estimate model to only a single race and gender to control the differences between race and gender in pay.

Question 4.b. Look at column (3) of the table. In words, interpret the coefficient on the dummy variable "9".

Hint: Note that "12" is the omitted category.

If somebody has only completed 9 years of education, they can expect to get paid less in comparison to someone with a highschool degree because of the negative coefficient in model (3).

Question 4.c. Why do you think the effect of the 14th year of education is larger than that of the 15th?

The effect of a 14th year of education is larger because you would get an Associates degree, whereas you wouldn't get another diploma or degree in the 15th year.

Question 4.d. Now look at column (4). Think about a student who is currently a senior. What is the average difference in the student's wage now and the one that the student could get at the end of the year following graduation?

If you were to assume that were looking at the difference of 15 years of education and 16 years of education, then you would see an approximate log difference of 0.126 in hourly wage.

Question 4.e. Based on the results presented in this column, would you rather choose to complete a PhD or a professional degree? Explain.

I would rather prefer the profession degree because there is a much higher coefficient for a log hourly wage in professional degree than a PhD.

Question 4.f. Using the results from columns (3) and (4), how would you test the presence of a "diploma effect"? Carry out the test at a 5% significance level.

Hint: You may find some of the information you need in the footnote of the table.

To test the presence of a diploma effect, we carry out a hypothesis test. Our null hypothesis is that the diploma regressors would have a combined effect equal to 0. After that, we conduct an F-statistic since we are given the R^2, Adjusted R^2, and the Mean Square error. Afterwards, we determine whether or not the null hypothesis is true at the 5% significance level.

In column (4), the null is:

$$H_0 : \beta_{HS}, \beta_{ND}, \beta_{OA}, \beta_{AA}, \beta_{BA}, \beta_{MA}, \beta_{Prof}, and \beta_{PhD} = 0$$

$$F - Stat = \frac{(R^2_{unrestricted} - R^2_{restricted})/q}{(1 - R^2_{unrestricted})(n - K_{unrestricted} - 1)}$$

$$= \frac{(0.154 - 0.147)/8}{(1 - 0.154)(8957 - 28 - 1)} \approx 9.234$$

With an F-statistic of 9.234, we reject the null at 5% significance level and conclude that there is a diploma effect on the average wage earned.