**Lab Assignment Week 06**

**I. Objectives**: The objectives are to practice methods for diagnosing and resolving the problem of non-constant error variance, also known as heteroskedasticity. There exist other, simpler methods for both tasks, but they were designed for (and are most easily implemented in) bivariate regression; we will focus instead on methods that are applicable to multivariate regression.

**II. Datasets**: *abortions.dta*, *populations.dta*

**III. Packages**: *car*, *foreign*, *lmtest*, *sandwich*

**IV. Preparation**

1) Open RStudio by double-clicking the icon or selecting RStudio from the Windows Start menu.

2) Clear any data in memory by typing rm(list=ls())

3) Download datasets *abortions.dta* and *populations.dta* and place them in a working directory

4) Download R scripts “Lab06.R” and “white-test.R” and place them in a working directory.

5) Open the R script by typing *Ctrl*+*O* or by clicking on File in the upper-left corner, using the dropdown menu, and navigating to the script in your working directory.

6) Run lines 3–6 in the R script to load four packages that you will need. At least two of these you have used before and do not need to install. If any package is not already installed, you need to install it using the **Packages** tab or by running the lines of code shown in line 5 and line 6.

**V. Instructions for Lab 06**

In this lab, you will explore states’ abortion rates (number performed per 1,000 women between the ages of 15 and 44) in 1992. Begin by loading the datasets by running lines 8–9 or typing the following lines, changing the directory if needed:

> abdta <- read.dta("C:/abortions.dta")

> popdta <- read.dta("C:/populations.dta")

The dependent variable is named *abortion*.

There are four explanatory variables:

*price* (average price of a procedure in non-hospital facilities),

*income* (dollars per capita in 1992),

*picket* (percent of survey respondents who reported experiencing protests at abortion clinics that involved physical contact or blocking of patients), and

*funds* (a dummy variable, = 1 if state health funds can pay for abortion procedures).

Run line 11 to regress of states’ abortion rate on the four explanatory variables. Then, write the coefficients and standard errors in the space provided on page 6.

> base<-lm(abortion~price+income+picket+funds, data=abdta)

> summary(base)

**A. Diagnosing Heteroskedasticity**

Run line 12 to generate the added variable plots, which illustrate the estimated relationships between each explanatory variable and the dependent variable:

> avPlots(base, pch = 19)

Added variable plots might provide some insight into heteroskedasticity on a variable-by-variable basis. Does the residuals’ variance appear to be systematically related to any of these variables?

Line 13 plots the residuals versus fitted values (this is the first of four plots generated by typing plot(base) and then typing <return>); you might examine the plot for any deviation from an even, linear pattern.

Run line 14 to generate residuals (*uhat*) and fitted values (*yhat*) and add them to the data frame:

> abdta$uhat <- base$residuals; abdta$yhat <- base$fitted.values

I included code for some optional plots: line 15 plots predicted value of abortion rates against income, and line 16 plots the residual against income. These can be useful if you believe one explanatory variable, such as income, is the primary source of heteroskedasticity.

> plot(abdta$income, abdta$yhat); abline(lm(yhat~income, abdta))

> plot(abdta$income, abdta$uhat); abline(h=0, col="red")

Run lines 19–20 to perform the Breusch-Pagan test, which performs one auxiliary regression of the squared residual on all four explanatory variables. The first step is to generate the squared residuals (line 19); we then estimate the linear model (line 20). Examine the F statistic which is shown if you use the summary(bp) command after running the regression model.

> abdta$uhatsq <- abdta$uhat^2

> bp<-lm(uhatsq ~ price + income + picket + funds, abdta)

> summary(bp)

We could also use either a “score statistic,” a.k.a. the LM test statistic, which is distributed chi-squared and is the product of N and the R2 from this regression (and is computed in lines 22–23)

> rsq <- summary(bp)$r.squared

> nrsq = NROW(abdta$uhat)\*rsq

The LM test statistic follows a chi-square distribution with degrees of freedom equal to the number of explanatory variables, so you can use the following two lines of code (lines 24–25) to save the product of N and *R*2 and then to perform the chi-square test.

> dfbp = bp$rank-1

> pchisq(nrsq, dfbp, ncp=0, lower.tail=FALSE, log.p=FALSE)

Line 28 provides code for a shortcut method of performing the Breusch-Pagan test; is it reporting the LM statistic or the F statistic?

> bptest(base)

The null hypothesis that we are seeking to reject with the Breusch-Pagan test is that the errors are homoskedastic. Based on the results of the test, do we reject (*p* < .05) or retain (*p* ≥ .05) the null hypothesis?

Line 29 performs a similar test – the Cook-Weisberg Test – which you can learn a little more about by typing *?ncvTest* in the console window.

> ncvTest(base)

Run lines 32–36 to perform Wooldridge’s special form of White’s test, which regresses the squared residual on and 2. The test statistic is the product of N and the R2 from this regression (i.e., the LM statistic), but you can also examine the F statistic to see whether *p* < .05.

> abdta$yhatsq = (abdta$yhat)^2

> whitetest<-lm(uhatsq ~ yhat + yhatsq, abdta); summary(whitetest)

> nr2 = NROW(whitetest$residuals)\*summary(whitetest)$r.squared

> dfwhite = whitetest$rank-1

> pchisq(nr2, dfwhite, ncp=0, lower.tail=FALSE, log.p=FALSE)

Although the F-statistic allows you to reject the null hypothesis of homoscedasticity, neither coefficient for *yhat* or for *yhatsq* is statistically significant. Why do you think that might be? After pondering this question, check for multicollinearity (between *yhat* and *yhatsq*).

Open the R script “white-test.R” and select every line of code, from line 1 to line 65. Run the script, then return to the R script “Lab06.R” where lines 39–40 perform the original version of White’s test. To perform the original version of the test, you regress the squared residual on every explanatory variable, the square of every explanatory variable, and the interaction of every explanatory variable with every other explanatory variable. Even with only four explanatory variables, this test consumes 14 degrees of freedom (with an N of just 50).

> white.test(base)

> rm(white.test)

Is the *p* value under the .05 threshold?

**B. Resolving Heteroskedasticity**

In the simplest form of the Weighted Least Squares method for resolving heteroskedasticity, one theorizes that *one* explanatory variable is interfering with homoscedasticity, and then one divides all variables – independent and dependent – by the square root of that variable. This process makes the most sense when the error has a variance proportional to *one* explanatory variable. For example, when we are using aggregate data collected from geographic units of varying populations, the error variance may be inversely proportional to population size.

Run line 43 to merge two datasets – the active dataset plus one with states’ populations as measured in the 1990 census. Line 44 divides states’ populations by 1000, so that the scale is the same as the dependent variable.

> mdata <- merge(abdta, popdta, by="state")

> mdata$pop1990k <- mdata$pop1990/1000

Line 45 plots the residual against states’ populations; does the error variance appear larger for the states with smaller populations? (California and New York, the two largest states, also have high residuals, but the remaining states’ residuals do seem to fit a nice ‘cone’ pattern.)

> plot(mdata$pop1990k, mdata$uhat, pch=19); abline(h=0, col="red")

Run lines 48–60 to perform ‘Feasible Generalized Least Squares’ the *loooooong* way. First, lines 48–49 generate the squared population and squared residuals, and line 50 regresses the squared residuals on population and squared population.

> mdata$pop1990ksq <- mdata$pop1990k^2

> mdata$uhatsq <- (mdata$uhat)^2

> fglsmod <- lm(mdata$uhatsq ~ mdata$pop1990k + mdata$pop1990ksq)

Lines 51−52 generate the weights (*w*) as the square root of the predicted residuals:

> mdata$wsq <- predict(fglsmod)

> mdata$w <- sqrt(mdata$wsq)

Next, the constant and all the variables must all be divided by the weight (lines 53–58):

> mdata$one\_w <- 1/mdata$w

> mdata$abortion\_w <- mdata$abortion/mdata$w

> mdata$price\_w <- mdata$price/mdata$w

> mdata$income\_w <- mdata$income/mdata$w

> mdata$picket\_w <- mdata$picket/mdata$w

> mdata$funds\_w <- mdata$funds/mdata$w

Finally, run the regression on the weighted variables (line 59 or line 60). You must modify the command to ensure that the constant is dropped, because we replace the constant with 1/*w*.

> fglsmod.1 <- lm(abortion\_w ~ one\_w + price\_w + income\_w + picket\_w + funds\_w - 1, data=mdata)

> fglsmod.2 <- lm(abortion\_w ~ 0 + one\_w + price\_w + income\_w + picket\_w + funds\_w, data=mdata)

Earlier I asked you to write down the coefficients and standard errors from the ordinary least squares regression; write down these values for the weighted least squares regression on page 6.

Line 63 provides a simpler way to perform Feasible Generalized Least Squares by using the analytic weights function:

> wlsmod <- lm(abortion ~ price + income + picket + funds, data=mdata, weights = 1/wsq)

The coefficients should be identical to those from the long version of WLS, so you could have skipped directly from line 52 to line 64. (But, where’s the fun in that?)

Notice, however, the measures of model fit are very different, because we had changed the dependent variable to abortion\_w in line 54 and lines 59–60, whereas we use the original dependent variable, abortion, in line 11 and in line 63.

Lines 66–70 run Wooldridge’s version of WLS, which uses the log of the squared residuals in the process of generating the weights. Note that I could have shortened the code by one line (line 69) by using 1/mdata$woolfit as the weights in line 70.

> mdata$luhatsq <- log(mdata$uhatsq)

> wooldridge <- lm(luhatsq ~ price + income + picket + funds + pop1990k + pop1990ksq, data=mdata)

> mdata$woolfit <- exp(wooldridge$fitted.values)

> mdata$wool.w = 1/mdata$woolfit

> wooldridge.wls <- lm(abortion~price+income+picket+funds, data=mdata, weights=mdata$wool.w)

Write down the estimated coefficients and standard errors from the model wooldridge.wls on page 6; how do the results of this model compare to the results shown after line 63?

Run lines 73 and either line 74 or line 75 to run a regression with heteroskedasticity-robust standard errors, and write down the coefficients and the standard errors in the appropriate space on page 6. Compare the coefficients, standard errors, and *t* statistics to the earlier results.

> lm.mod <- lm(abortion ~ price + income + picket + funds, mdata)

> robse<-vcovHC(lm.mod, type="HC1"); coeftest(lm.mod,robse)

A slightly different version of this test uses the *hccm*() command in the *car* package:

> robust<-hccm(lm.mod); coeftest(lm.mod,robust)

You can compare the variance-covariance matrices of the estimators by typing their names; in order to round to four decimal points (for easier viewing), you can type the following:

> round(vcov(lm.mod), digits=4)

> round(robse, digits=4)

> round(robust, digits=4)

Remember that the estimated variances of the estimated coefficients are on the main diagonal. Does any version of the variance-covariance matrix contain systematically larger variances?

Does any version of the variance-covariance matrix contain systematically smaller variances?

Finally, run lines 78–79 to regress the natural log of the abortion rate on the same four variables. The topic we are taking up next Thursday is nonlinear models, of which the log-linear model (which uses the natural log of *y* as the dependent variable, but maintains all *x* in their original form) is one variation.

Plot the residuals against the fitted values using line 80, and plot the residuals against population (which was not among the regressors) using line 81. After examining the plot, conduct a test of heteroskedasticity using one of the commands or procedures in part A of the lab; one option is shown in line 82. Does transforming the dependent variable resolve the heteroskedasticity problem?

To clear the **Environment**, type rm(list=ls()) or click on the broom icon.

To clear the **Console** window, type Ctrl-*l*

Space is provided on the here for you to write down some estimates of the coefficients and standard errors, based on four different methods for estimating the multiple regression model.

Ordinary Least Squares estimates:

|  |  |  |
| --- | --- | --- |
| ***explanatory variable*** | ***coefficient*** | ***standard error*** |
| *constant* |  |  |
| *price* |  |  |
| *income* |  |  |
| *picket* |  |  |
| *funds* |  |  |

Weighted Least Squares estimates (main method in line 59, 60, or 63):

|  |  |  |
| --- | --- | --- |
| ***explanatory variable*** | ***coefficient*** | ***standard error*** |
| *constant* |  |  |
| *price* |  |  |
| *income* |  |  |
| *picket* |  |  |
| *funds* |  |  |

Weighted Least Squares estimates (another method in line 70):

|  |  |  |
| --- | --- | --- |
| ***explanatory variable*** | ***coefficient*** | ***standard error*** |
| *constant* |  |  |
| *price* |  |  |
| *income* |  |  |
| *picket* |  |  |
| *funds* |  |  |

Ordinary Least Squares estimates with robust standard errors (either line 74 or line 75):

|  |  |  |
| --- | --- | --- |
| ***explanatory variable*** | ***coefficient*** | ***standard error*** |
| *constant* |  |  |
| *price* |  |  |
| *income* |  |  |
| *picket* |  |  |
| *funds* |  |  |