Tutorial 9: Heteroskedasticity

Anh Le

1. Test of heteroskedasticity (graphical, BP, and White test)
2. Standard error robust to heteroskedasticity
3. Tips and tricks

- R project

Heteroskedasticity test

Generate heteroskedastic data

For example, we have the following data generating process (DGP):

$$y = 2X + u \tag{1}$$
$$X = N(0, 1) \tag{2}$$
$$u \sim N(0, |x|) \qquad \text{Note how } V(u) = |x|, \text{ instead of being constant} \tag{3}$$

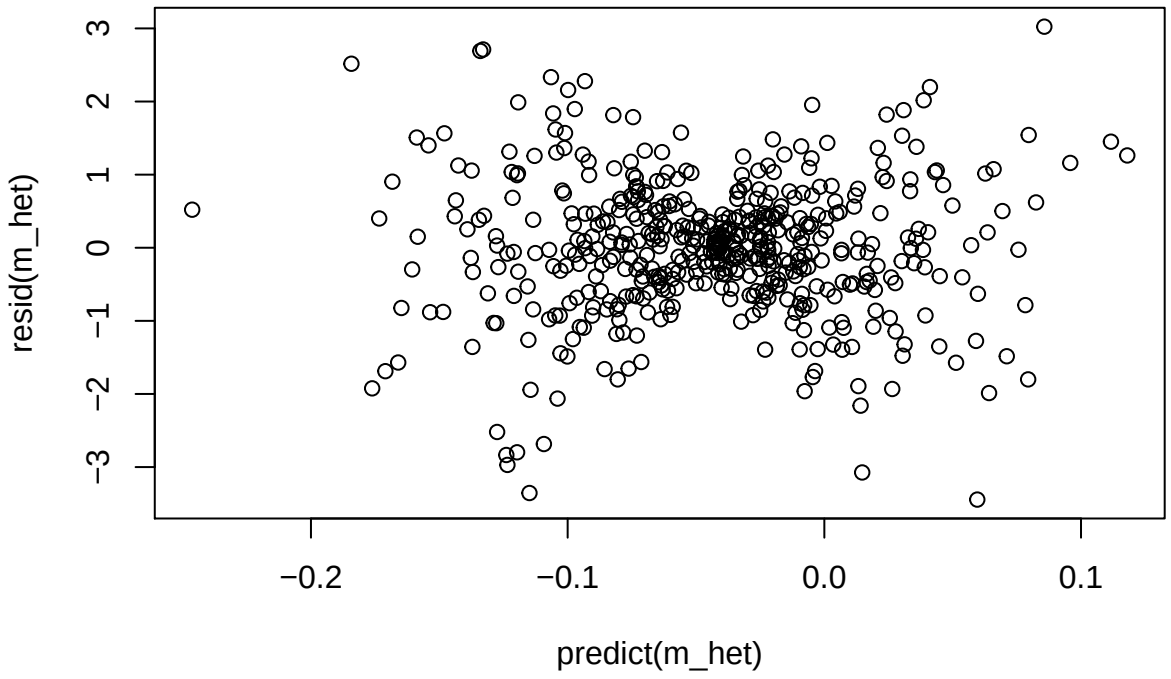
I implement that DGP in R as follows:

```
set.seed(1)
x <- rnorm(500)
u <- rnorm(500, sd = sqrt(abs(x)))
y <- 0.01 * x + u
```

Test for Heteroskedasticity:

Visual test

```
m_het <- lm(y ~ x)
plot(resid(m_het) ~ predict(m_het))
```



Breush-Pagan and White test

```
library(AER) # For the tests
```

BP test

```
bptest(m_het, varformula = ~ x)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m_het
## BP = 1.959, df = 1, p-value = 0.1616
```

White test

```
yhat <- predict(m_het)
bptest(m_het, varformula = ~ yhat + I(yhat^2))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m_het
## BP = 50.287, df = 2, p-value = 1.203e-11
```

Note how the BP test doesn't pick up the heteroskedasticity, but the White test does.

> The default Breusch-Pagan test is a test for linear forms of heteroskedasticity, e.g. as $\hat{y}$ goes up, the error variancesgo up. In this default form, the test does not work well for non-linear forms of heteroskedasticity, such as the hourglass shape we saw before (where error variances got larger as X got more extreme in either direction). The default test also has problems when the errors are not normally distributed
>
> Part of the reason the White test is more general is because it adds a lot of terms to test for

more types of heteroskedasticity. For example, adding the squares of regressors helps to detect nonlinearities such as the hourglass shape. In a large data set with many explanatory variables, this may make the test difficult to calculate. Also, the addition of all these terms may make the test less powerful in those situations when a simpler test like the default Breusch-Pagan would be appropriate, i.e. adding a bunch of extraneous terms may make the test less likely to produce a significant result than a less general test would. (Source)

BP and White test by hand

Recall that our Null hypothesis is thatwe have homoskedasticity

BP test (Wooldridge "Introductory", Testing for heteroskedasticity)

1. Estimate the model `y ~ x1 + x2 + ... + xk` by OLS, as usual. Obtain the squared OLS residuals, $\hat{u}^2$ one for each observation.
2. Run the regression $\hat{u}^2 = \delta_0 + \delta_1 x_1 + \delta_2 x_2 + \cdots + \delta_k x_k$. Keep the R-squared from this regrsesion, $R^2_{\hat{u}^2}$.
3. Form either the F statistic or the LM statistic and compute the p-value (using the $F_{k,n-k-1}$ distribution in the former case and the χ^2_k distribution in the latter case)

$$F = \frac{R^2_{\hat{u}_i^2}/k}{(1 - R^2_{\hat{u}_i^2})/(n-k-1)}; LM = n \times R^2_{\hat{u}_i^2}$$

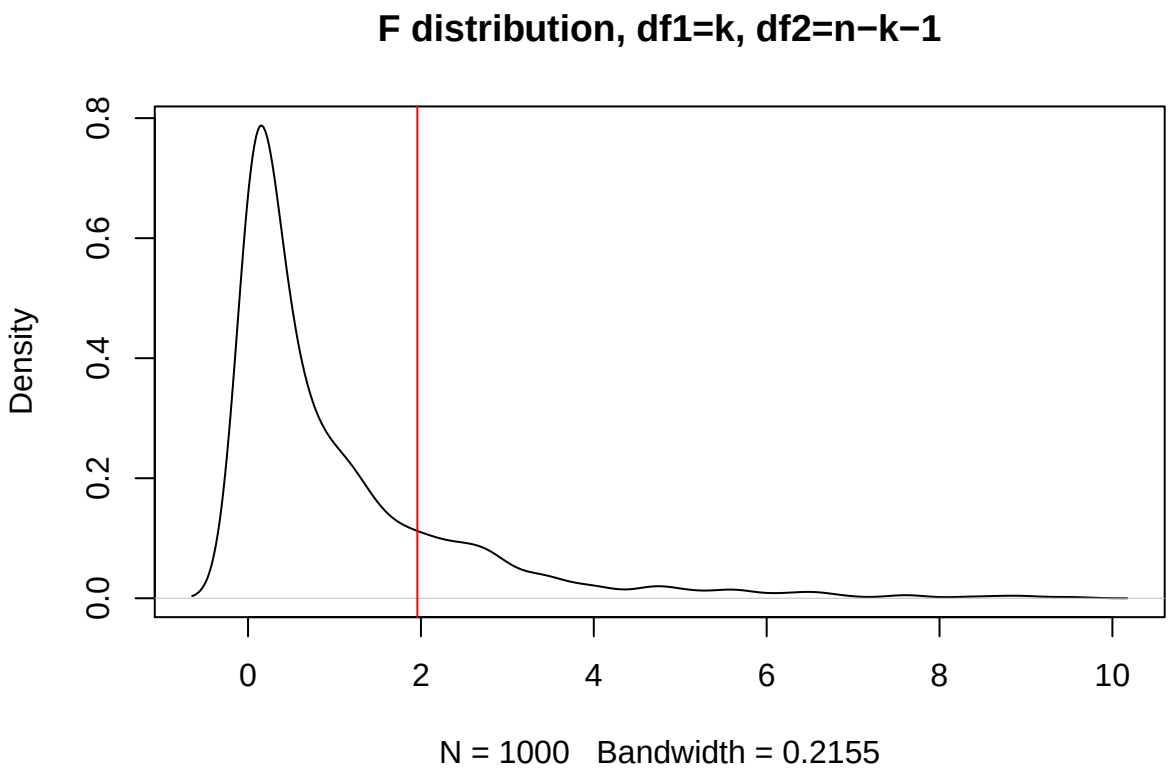
```
m_bp <- lm(y ~ x)
squared_residuals <- resid(m_bp) ** 2
m_bp_stage2 <- lm(squared_residuals ~ x)
R_squared <- summary(m_bp_stage2)$r.squared

n <- length(y) ; k <- 1 # k = 1 because we only have 1 x
(F_statistic <- (R_squared / k) / ((1 - R_squared) / (n - k - 1)))
```

```
## [1] 1.958797
```

Is this F-statistic large or small?

```
plot(density(rf(1000, df1 = k , df2 = n - k - 1)),
     main = "F distribution, df1=k, df2=n-k-1") ; abline(v = F_statistic, col = 'red')
```



Given this F-statistic, what's the p-value?

```
1 - pf(F_statistic, df1 = k, df2 = n - k - 1)
```

```
## [1] 0.1622646
```

```
(t_bp <- bptest(m_het, varformula = ~ x))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m_het
## BP = 1.959, df = 1, p-value = 0.1616
```

White test (Wooldridge "Introductory", Testing for heteroskedasticity) **will be in homework**

1. Estimate the model `y ~ x_1 + x_2 + ... + x_k` by OLS, as usual. Obtain the OLS residual $\hat{u}$ and the fitted values $\hat{y}$. Compute $\hat{u}^2$ and $\hat{y}^2$.
2. Run the regression $\hat{u}^2 = \delta_0 + \delta_1 \hat{y} + \delta_2 \hat{y}^2$.
3. Form either the F or LM statistic and compute the p-value (using the $F_{2,n-3}$ distribution in the former case and the χ^2_2 distribution in the latter case).

Get the correct standard error

White HC standard errors (sandwich standard error)

```
vcovHC(m_het, type = "HC") # from package sandwich, loaded by AER
```

```
##              (Intercept)           x
## (Intercept) 0.0016878425 0.0001226827
## x           0.0001226827 0.0030586553
```

Hypothesis test with White standard error (more precisely, with the heteroskedasticity-consistent estimation of the cov matrix, calculated above). Notice how the coefficients are basically the same as regular OLS. Only the standard error is different.

```
coeftest(m_het, vcov = vcovHC(m_het, type = "HC"))
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.042695   0.041083 -1.0392   0.2992
## x           -0.053430   0.055305 -0.9661   0.3345
```

```
summary(m_het)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4445 -0.4878  0.0170  0.4892  3.0237
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.04270    0.04126  -1.035    0.301
## x           -0.05343    0.04081  -1.309    0.191
##
## Residual standard error: 0.9224 on 498 degrees of freedom
## Multiple R-squared:  0.003431,   Adjusted R-squared:  0.001429
## F-statistic: 1.714 on 1 and 498 DF,  p-value: 0.191
```

Foonote: Etymology of the "sandwich" estimator of the standard error

Formula for standard error is:

$$V(\hat{\beta}_{OLS}|X) = (X'X)^{-1}X'V(\epsilon|X)X(X'X)^{-1}$$

, which reduces to $\sigma^2(X'X)^{-1}$ only in the case of homoskedasticity.

When there's heteroskedasticity, the estimate is (as done in Stata) (notice the "bread" ($(X'X)^{-1}$) and the "meat" ($X'V(\epsilon|X)X$) of the sandwich):

$$V(\hat{\beta}_{OLS}|X) = (X'X)^{-1}X'V(\epsilon|X)X(X'X)^{-1} \tag{4}$$

$$\hat{V}(\hat{\beta}_{OLS}|X) = (X'X)^{-1}X' \begin{pmatrix} \hat{\epsilon}_1^2 & 0 & \dots & 0 \\ 0 & \hat{\epsilon}_2^2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \hat{\epsilon}_n^2 \end{pmatrix} X(X'X)^{-1} \tag{5}$$

$$\hat{V}(\hat{\beta}_{OLS}|X) = \frac{N}{N-K}(X'X)^{-1}\sum_{i=1}^{N}\{X_i X_i'\hat{\epsilon}_i^2\}(X'X)^{-1} \quad \text{The matrix version of slide 36 of your lectures notes} \tag{6}$$

The constant is added since we are estimating the sample variance of the error. In practice, $N >> K$, so it doesn't matter.

Get the correct standard error by hand

From slide 36 of your lectures notes, also Wooldrige "Introductory" - Heteroskedasticity robust inference after OLS estimation

$$\hat{Var}(\hat{\beta}_j) = \frac{\sum_{i=1}^{n} \hat{r}_{ij}^2 \hat{u}_i^2}{SSR_j^2}$$

where $\hat{r}_{ij}^2$ is the ith residual from regressing x_j on all other independent variables, and SSR_j is the sum of squared residuals from this regression

```
m_robust1 <- lm(x ~ 1)
numerator <- sum((resid(m_robust1)**2) * (resid(m_het)**2))
SSR <- sum(resid(m_robust1)**2) ** 2

(var_beta_x <- numerator / SSR)
```

```
## [1] 0.003058655
```

We see that we get exactly the same $\hat{Var}(\hat{\beta}_x)$ as output by R.

```
vcovHC(m_het, type = "HC")
```

```
##               (Intercept)            x
## (Intercept) 0.0016878425 0.0001226827
## x           0.0001226827 0.0030586553
```