Assignment 3 - Heteroscedasticity

Aliaksandr Panko, Nadzeya Yakimchyk, Nikolai Agafonov, Ilia Kurennoi

June 3, 2018

Contents

1 Objectives

1. Execute a test for heteroscedasticity.
2. Interpret the results.
3. Compute heteroscedasticity-consistent standard errors (using the R-package 'AER').
4. Make a conclusion.

2 Initial model

Let's start with the regression where all factors in the start of analysis seem to be statistically significant:

```
regr <- lm(log(SalePrice) ~ log(YearBuilt) +
log(LotArea) +
log(LivAreaSF) +
OverallQual +
OverallCond +
NbrCrawfor +
NbrStoneBr +
NbrEdwards +
Fireplaces +
log(X1stFlrSF) +
GarageCars +
BldgTypeDuplex +
BsmtExposureGd +
FoundationPConc
,train)
summary(regr)
```

2.1 Regression Result

```
Call:
lm(formula = log(SalePrice) ~ log(YearBuilt) + log(LotArea) +
    log(LivAreaSF) + OverallQual + OverallCond + NbrCrawfor +
    NbrStoneBr + NbrEdwards + Fireplaces + log(X1stFlrSF) + GarageCars +
    BldgTypeDuplex + BsmtExposureGd + FoundationPConc, data = train)

Residuals:
     Min       1Q   Median       3Q      Max
-1.34787 -0.06324  0.01202  0.07850  0.40134

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -48.143277   4.079612 -11.801  < 2e-16 ***
log(YearBuilt)        7.260822   0.534163  13.593  < 2e-16 ***
log(LotArea)          0.099294   0.012486   7.952 7.16e-15 ***
log(LivAreaSF)        0.329018   0.025892  12.707  < 2e-16 ***
OverallQual           0.074760   0.006916  10.809  < 2e-16 ***
OverallCond           0.054905   0.005675   9.675  < 2e-16 ***
NbrCrawforTRUE        0.151187   0.031137   4.856 1.48e-06 ***
NbrStoneBrTRUE        0.133729   0.042570   3.141 0.001751 **
NbrEdwardsTRUE       -0.071555   0.021447  -3.336 0.000893 ***
Fireplaces            0.023446   0.010669   2.198 0.028296 *
log(X1stFlrSF)        0.124549   0.024102   5.167 3.08e-07 ***
GarageCars            0.059501   0.010043   5.925 4.86e-09 ***
BldgTypeDuplexTRUE   -0.085566   0.028644  -2.987 0.002911 **
BsmtExposureGdTRUE    0.066421   0.021302   3.118 0.001894 **
FoundationPConcTRUE   0.056344   0.015639   3.603 0.000337 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1478 on 715 degrees of freedom
Multiple R-squared:  0.857,	Adjusted R-squared:  0.8542
F-statistic: 306.1 on 14 and 715 DF,  p-value: < 2.2e-16
```

3 Homoscedasticity test

Now let's perform the standard and the studentized Breusch-Pagan test to check the homoscedasticity:

```
car::ncvTest(regr)
lmtest::bptest(regr)
```

3.1 Tests Results

```
> car::ncvTest(regr)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 101.2671    Df = 1     p = 8.038038e-24
> lmtest::bptest(regr)

        studentized Breusch-Pagan test

data:  regr
BP = 76.949, df = 14, p-value = 1.037e-10
```

The tests clearly show that homoscedasticity null hypothesis should be rejected, and we must accept alternative hypothesis of heteroscedasticity. Therefore the estimated standard errors of coefficients are biased and the associated t- and F-statistics are incorrect.

4 Heteroscedasticity-consistent standard errors using the R-package 'AER'

So let's compute heteroscedasticity-consistent standard errors using the R-package 'AER'

```
library(AER)

coefftest <- coeftest(regr, vcov = vcovHC(regr, type = "HC0"))
print(coefftest)

summary(regr)$coefficients[,2]/coefftest[,2]
```

4.1 Coefficients

```
t test of coefficients:

                     Estimate  Std. Error  t value  Pr(>|t|)    
(Intercept)        -48.1432767  3.9227807 -12.2727 < 2.2e-16 ***
log(YearBuilt)       7.2608224  0.5062285  14.3430 < 2.2e-16 ***
log(LotArea)         0.0992942  0.0120457   8.2432 8.014e-16 ***
log(LivAreaSF)       0.3290178  0.0312978  10.5125 < 2.2e-16 ***
OverallQual          0.0747598  0.0064960  11.5087 < 2.2e-16 ***
OverallCond          0.0549048  0.0054592  10.0573 < 2.2e-16 ***
NbrCrawforTRUE       0.1511873  0.0276011   5.4776 5.976e-08 ***
NbrStoneBrTRUE       0.1337294  0.0376288   3.5539 0.0004045 ***
NbrEdwardsTRUE      -0.0715546  0.0395844  -1.8076 0.0710816 .  
Fireplaces           0.0234456  0.0119997   1.9538 0.0511090 .  
log(X1stFlrSF)       0.1245488  0.0246913   5.0442 5.779e-07 ***
GarageCars           0.0595008  0.0142389   4.1788 3.294e-05 ***
BldgTypeDuplexTRUE  -0.0855661  0.0281415  -3.0406 0.0024476 ** 
BsmtExposureGdTRUE   0.0664210  0.0267797   2.4803 0.0133571 *  
FoundationPConcTRUE  0.0563436  0.0136892   4.1159 4.306e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(regr)$coefficients[,2]/coefftest[,2]
        (Intercept)      log(YearBuilt)        log(LotArea)      log(LivAreaSF) 
          1.0399797           1.0551819           1.0365789           0.8272778 
        OverallQual         OverallCond      NbrCrawforTRUE      NbrStoneBrTRUE 
          1.0647135           1.0394703           1.1281101           1.1313210 
     NbrEdwardsTRUE          Fireplaces      log(X1stFlrSF)          GarageCars 
          0.5417922           0.8890750           0.9761486           0.7053153 
 BldgTypeDuplexTRUE  BsmtExposureGdTRUE FoundationPConcTRUE 
          1.0178508           0.7954704           1.1424549 
```

5 Conclusion

It can be seen that standard errors of

- Intercept
- log(YearBuilt)
- log(LotArea)
- OverallQual
- OverallCond
- NbrCrawfor
- NbrStoneBr
- BldgTypeDuplex
- FoundationPConc

were overestimated. But overestimation is not drastic, in most cases by no more than few percent and in any single case by no more than 14% (in case of FoundationPConc).

At the same time standard errors of variables

- log(LivAreaSF)
- NbrEdwards
- Fireplaces
- log(X1stFlrSF)
- GarageCars
- BsmtExposureGd

were underestimated.

Especially heavy underestimation takes place for

- Fireplaces
- log(LivAreaSF)
- BsmtExposureGd
- GarageCars (by almost 30%)
- NbrEdwards (by almost 2 times)

According to recalculation of p-values, variables Fireplaces and NbrEdwards become statistically insignificant. However, p-value for Fireplaces becomes just slightly more than 0.05, and 0.07 for NbrEdwards is also not that drastic.