Pol Sci 630: Problem Set 12 Solutions: Heteroskedasticity, Autocorrelation

Prepared by: Anh Le (anh.le@duke.edu)

Due Date: Friday, Nov 20, 2015, 12 AM (Beginning of Lab)

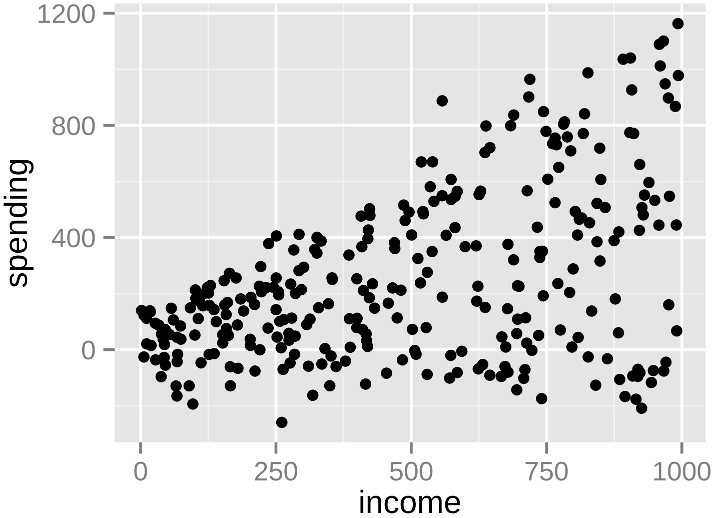
```
rm(list = ls())
library(ggplot2)
```

1 Heteroskedasticity

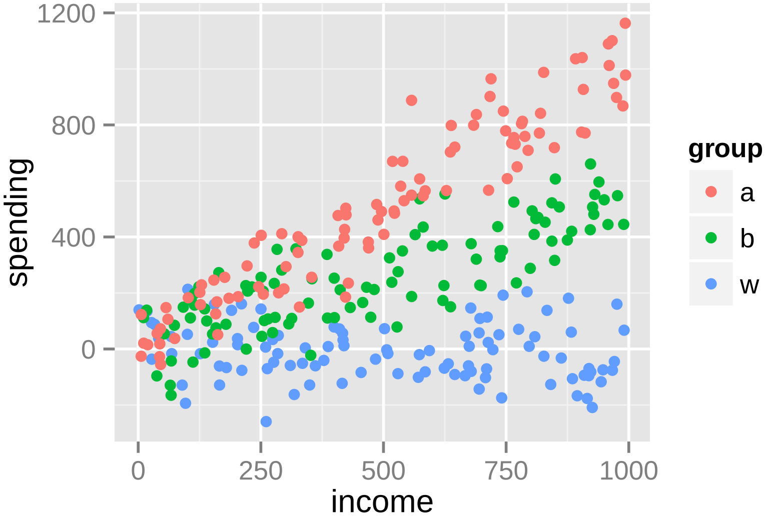
One common cause of heteroskedasticity is that our model does not take into account heterogenous effect across sub-populations. For example, we have a model of spending (dependent var) as a function of income (independent var), and the propensity to spend differs across ethnic groups. Formally,

$$spending = \beta_{ethnic} income + \epsilon \tag{1}$$

where $\beta ethnic$ takes a different value for white, black, and asian. If we don't know about this heterogeneity of propensity to spend across ethnic groups, the graph will show heteroskedasticity:



Buf if we are smart researcher, we'll realize the underlying cause of the heterogeneity, as shown in the following plot:



The take-home point is that heteroskedasticity could be a signal of underlying model specification, and we should think hard about the cause of heteroskedasticity instead of applying a quick fix.

1.1 Simulating

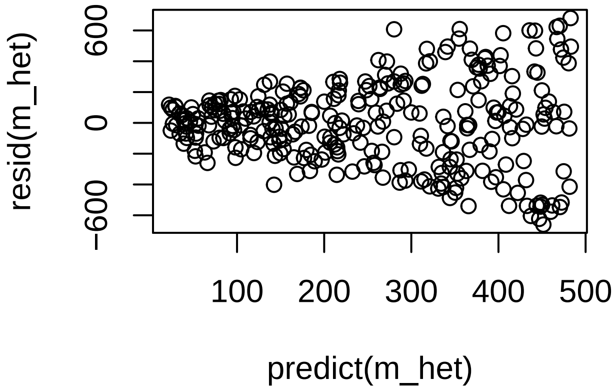
Simulate the spending and income pattern for three ethnic groups as described above. Re-create the two plots above. The numbers don't have to be the same – just make sure that your data has heteroskedasticity due to underlying heterogenous effect across ethnic groups as described in the example above. Note: Don't look at my code.

1.2 Diagnostics: Visual

Using the simulated data above, regress spending on income, plot the residual against the predicted value.

Solution

```
m_het <- lm(spending ~ income, data = d)
plot(predict(m_het), resid(m_het))
```



1.3 Diagonistics: Hypothesis test

Conduct BP test and White test. Why do the tests reach the same conclusion here, unlike in the lab tutorial?

Solution

```
library(AER)
bptest(m_het, varformula = ~ d$income)

##
```

```
##  studentized Breusch-Pagan test
##
## data:  m_het
## BP = 98.029, df = 1, p-value < 2.2e-16

bptest(m_het, varformula = ~ d$income + I(d$income^2))

##
##  studentized Breusch-Pagan test
##
## data:  m_het
## BP = 104.89, df = 2, p-value < 2.2e-16
```

The test reaches the same conclusion because the variance of the error terms is a linear function of *income* (not of $income^2$, for example), so both the BP and the White tests are able to detect this.

1.4 Fixing: robust standard error

Run hypothesis test without and with robust standard error. What's the conclusion?

Solution

```
summary(m_het)

##
## Call:
## lm(formula = spending ~ income, data = d)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -660.3 -175.0    6.6  149.0  680.3
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21.22530   30.19268   0.703    0.483
## income       0.46486    0.05246   8.860   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 269 on 298 degrees of freedom
## Multiple R-squared:  0.2085,Adjusted R-squared:  0.2059
## F-statistic: 78.51 on 1 and 298 DF,  p-value: < 2.2e-16

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

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 21.225300  20.346658  1.0432    0.2977
## income       0.464863   0.057242  8.1210 1.236e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Both regressions show that income has a positive and significant impact on spending

1.5 Fixing: FGLS

Conduct FGLS. Hint: For stability, log transform $residual^2$ in the auxiliary regression, then exponentiate the predicted value of the auxiliary regression to get the weight.

Solution

```
auxiliary_FGLS <- lm(I(log(resid(m_het)^2)) ~ d$income)
w <- exp(predict(auxiliary_FGLS))
m_het_wls <- lm(spending ~ income, weights = 1 / w, data = d)
summary(m_het_wls)

##
## Call:
## lm(formula = spending ~ income, data = d, weights = 1/w)
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -4.4095 -1.3527  0.0594  1.4140  4.0674
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 24.35206   15.11258   1.611    0.108
## income       0.45591    0.04491  10.152   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.621 on 298 degrees of freedom
## Multiple R-squared:  0.257,Adjusted R-squared:  0.2545
## F-statistic: 103.1 on 1 and 298 DF,  p-value: < 2.2e-16
```

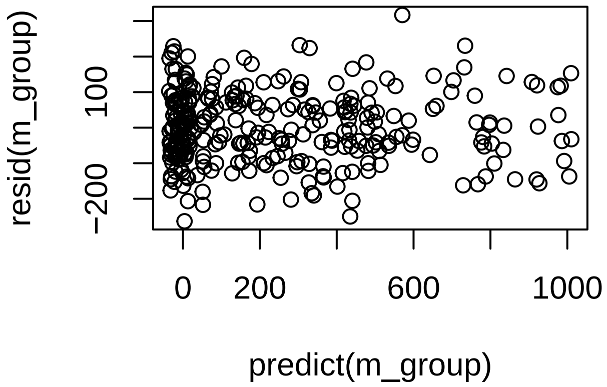
FGLS also confirms that income has a positive and significant impact on spending.

1.6 Fixing: Provide a correct model

Specify a regression model that takes into account heterogenous effect of income on spending across ethnic groups. Show that there's no longer heteroskedasticity.

Solution

```
m_group <- lm(spending ~ income + group + income:group, data = d)
plot(resid(m_group) ~ predict(m_group))
```



As shown in the diagnostics plot, there's no longer heteroskedasticity

2 Multicollinearity

2.1 Diagnosing with VIF

Using dataset `Prestige`, run regression of prestiage against income, education, and women. Calculate VIF. Interpret the largest VIF.

Solution

```
vif(lm(prestige ~ income + education + women, data=Prestige))

##    income education     women
##  2.282038  1.845165  1.526593
```

The largest VIF is 2.28 for income, meaning that by including other variables, the variance of the coefficient for income is inflated 2.28 times.

2.2 Dealing with multicollinearity

If you are concerned that the VIF is causing your SEs to be pretty big. What should you do to address this issue?

Solution

Note to grader: Just grade on how the submission makes an argument. We don't expect everyone to hit all the points below.

We should either collect more data (and hopefully the multicollinearity only exists in the previous sample and not in population), or to live with imprecision in our estimates, as omitting variables would likely lead to OVB.

It's important to note that there's no magic statistical fix for multicollinearity. If two variables are highly correlated, it's simply impossible to vary one and control for the other.

One thing one may do is to conduct dimension reduction (e.g. factor analysis, PCA) so that we recover the latent factor that drives all of these highly correlated variables. For example, in a survey of governance, we may see that the measurement of road quality, electricity quality, water quality are all highly correlated. Perhaps there's a common and latent factor of "public provision" that drives all of them.

3 Diagnosing autocorrelation

3.1 Generating autocorrelated data

Similar to the lab, generate data (i.e. e, X, Y) that follow an AR(2) process, i.e.:

$$v(t) \sim N(0,1) \tag{2}$$

$$e(t) = a_1 e(t-1) + a_2 e(t-2) + v(t) \qquad \text{Important: } a_1 + a_2 < 1 \tag{3}$$

$$Y(t) = X(t) + e(t) \tag{4}$$

Solution

```r
T <- 100 # Num of time periods

# Generate autocorrelated e
e <- vector(mode = 'numeric', length = T)
e[1] <- rnorm(1)
e[2] <- 0.4 * e[1] + rnorm(1)
for (t in 3:T) {
  e[t] <- 0.4 * e[t - 1] + 0.2 * e[t - 2] + rnorm(1)
}

X <- rnorm(T)
Y <- X + e
```

3.2 Diagnostics: Visual

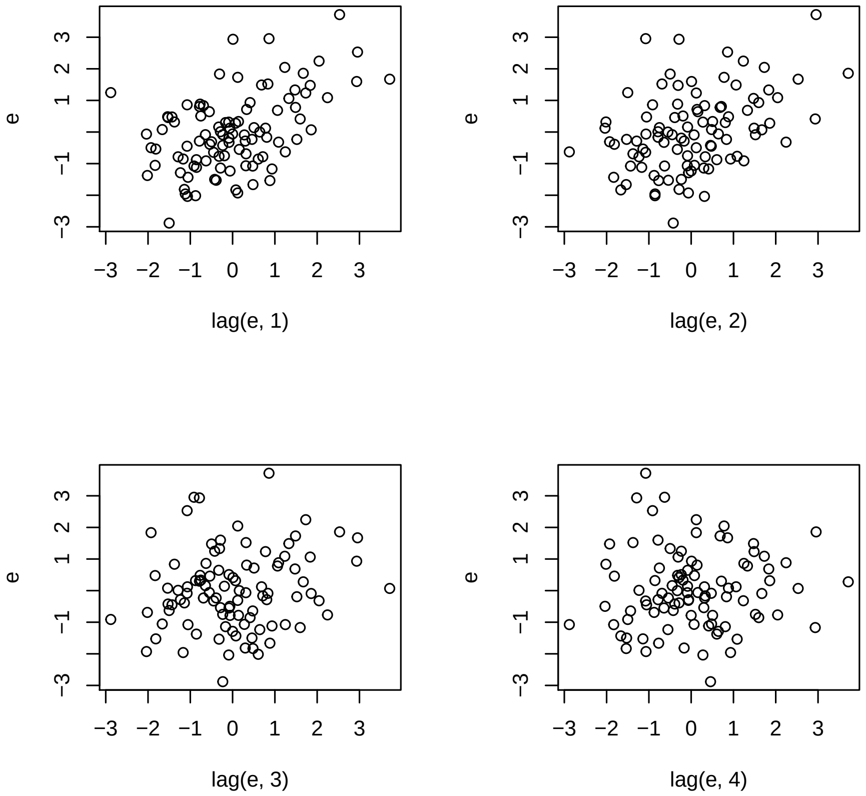
Plot residual against time and against lagged , up to 4 lags (e.g. residual lag-1 residual, residual lag-2 residual, etc. up to 4 plots) How does the correlation look across the four plots?

Solution

```
lag <- function(x, lag_period) {
  return(c(rep(NA, lag_period), x[1:(length(x) - lag_period)]))
}

m_auto <- lm(Y ~ X)
e <- resid(m_auto)

par(mfrow = c(2, 2))
plot(lag(e, 1), e)
plot(lag(e, 2), e)
plot(lag(e, 3), e)
plot(lag(e, 4), e)
```



We see that he error autocorrelation diminishes the further the lag is (See how the relationship is very strong in the first plot, but not so much in the fourth?).

3.3 Diagnostics: Hypothesis testing

Regress residuals against X and lag1 and lag2 residuals, and then doing an F test for joint significance in the lagged residuals.

Solution

```
lag1_e <- lag(e, 1)
lag2_e <- lag(e, 2)

# Reg residual against X and lagged residuals
m_autotest <- lm(e ~ X + lag1_e + lag2_e)

# Doing an F test
```

```r
library(car) # to run F-test
linearHypothesis(m_autotest, c("lag1_e", "lag2_e"))

## Linear hypothesis test
##
## Hypothesis:
## lag1_e = 0
## lag2_e = 0
##
## Model 1: restricted model
## Model 2: e ~ X + lag1_e + lag2_e
##
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     96 147.25
## 2     94 105.30  2    41.952 18.725 1.43e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We reject the null of no autocorrelation. Notice how we can do the F-test on more than just one lag to detect autocorrelation. In real research, you would use your judgement to guess the autocorrelation structure (i.e. how far back does the autocorrelation go?) and test it.