Lab 6

Zeren Li

Roadmap

- Test for heteroskedasticity (visualization, rvf, BP, and White test)
- Heteroskedasticity-robust standard error
- Weighted Least Squares & Feasible Generalizable Least Squares

Earning Dataset

Wooldridge's notes on the data set: "Notes: I remember entering this data set in the late 1980s, and I am pretty sure it came directly from an introductory econometrics text. But so far my search has been fruitless. If anyone runs across this data set, I would appreciate knowing about it."

```
data("saving", package="wooldridge")  #from wooldridge package

# summary statistics
summary(saving)
```

```
##       sav                inc              size            educ      
##  Min.   :-5577.0   Min.   :  750   Min.   : 2.00   Min.   : 2.00  
##  1st Qu.:  194.5   1st Qu.: 6510   1st Qu.: 3.00   1st Qu.: 9.00  
##  Median :  982.0   Median : 8776   Median : 4.00   Median :12.00  
##  Mean   : 1582.5   Mean   : 9941   Mean   : 4.35   Mean   :11.58  
##  3rd Qu.: 1834.8   3rd Qu.:11903   3rd Qu.: 5.00   3rd Qu.:13.00  
##  Max.   :25405.0   Max.   :32080   Max.   :10.00   Max.   :20.00  
##       age            black           cons       
##  Min.   :26.00   Min.   :0.00   Min.   :-13055  
##  1st Qu.:33.00   1st Qu.:0.00   1st Qu.:  5732  
##  Median :38.50   Median :0.00   Median :  7562  
##  Mean   :38.77   Mean   :0.07   Mean   :  8359  
##  3rd Qu.:44.00   3rd Qu.:0.00   3rd Qu.:  9864  
##  Max.   :54.00   Max.   :1.00   Max.   : 30280
```

Fit a bivariate regression

```
m_het <- lm(sav ~ inc, saving)

# get y_hat
y_hat <- predict(m_het)

# residual
u_hat <- matrix(saving$sav - y_hat)

# variance/covariance matrix
v_cov <-   u_hat %*% t(u_hat)
v_cov[1:5,1:5]
```

```
##          [,1]       [,2]      [,3]       [,4]      [,5]
```

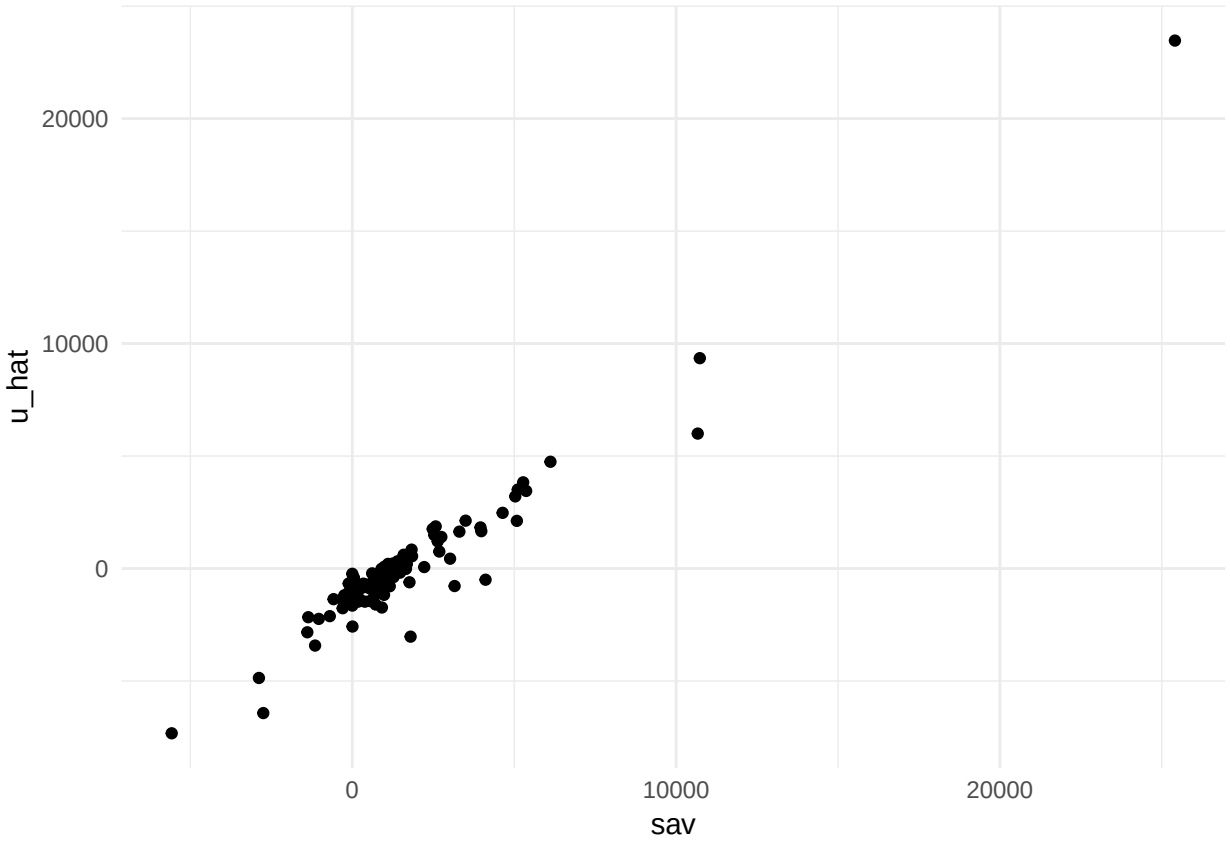
```
## [1,] 141653.5  402516.61 260702.16 -18075.497 329237.88
## [2,] 402516.6 1143774.21 740800.24 -51362.568 935548.47
## [3,] 260702.2  740800.24 479801.87 -33266.533 605936.49
## [4,] -18075.5  -51362.57 -33266.53   2306.498 -42011.94
## [5,] 329237.9  935548.47 605936.49 -42011.939 765230.53
```

Test for Heteroskedasticity

Visualized Analysis

This graph does not show the classic case of fanning-out along with different levels of the dependent variable

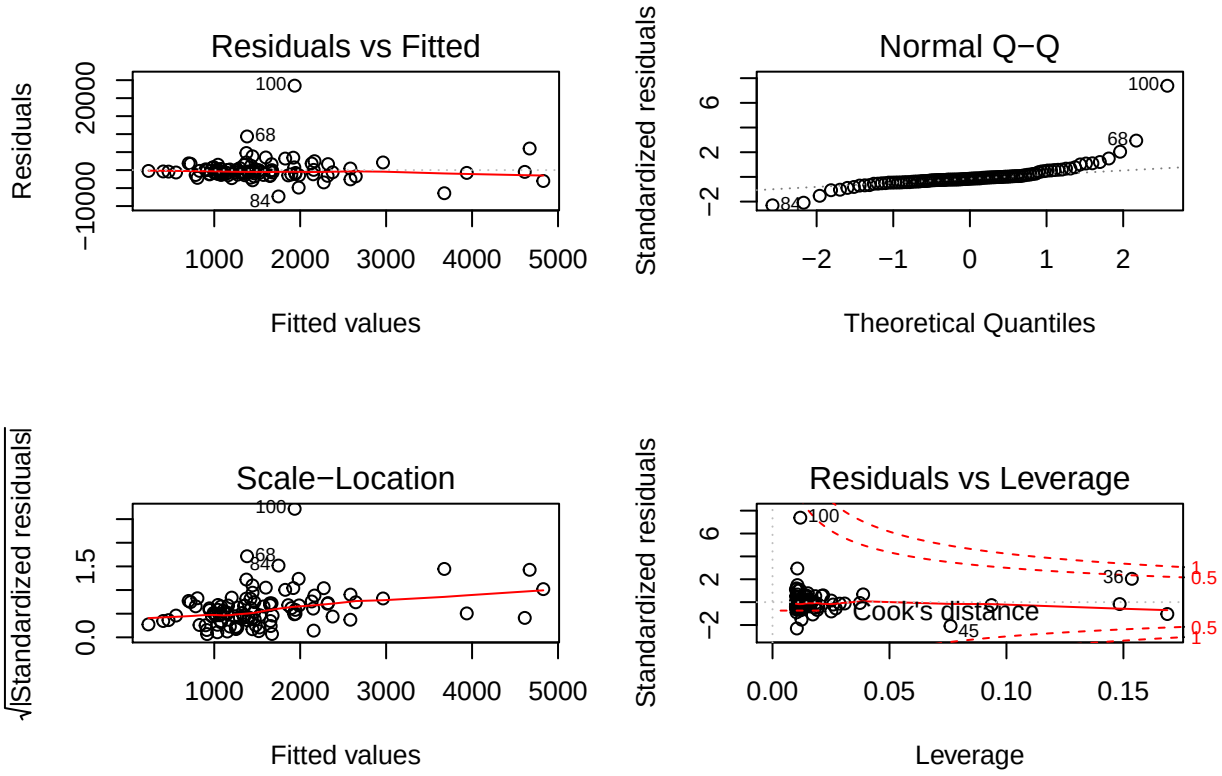
```
df =  data.frame(sav = saving$sav, u_hat = u_hat)

ggplot(df, aes(x = sav, y = u_hat) ) +
  geom_point()
```



Regression Diagnostics

The RVF plot does cause me to worry somewhat. We see very little variance along levels of the fitted values until we hit about $1000 and then the variance increases, declines, and increases again.

```
par(mfrow=c(2,2))
plot(m_het)
```



Breush-Pagan and White test

BP test by hand

Recall that our Null hypothesis is that we have homoskedasticity

*BP test

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)

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

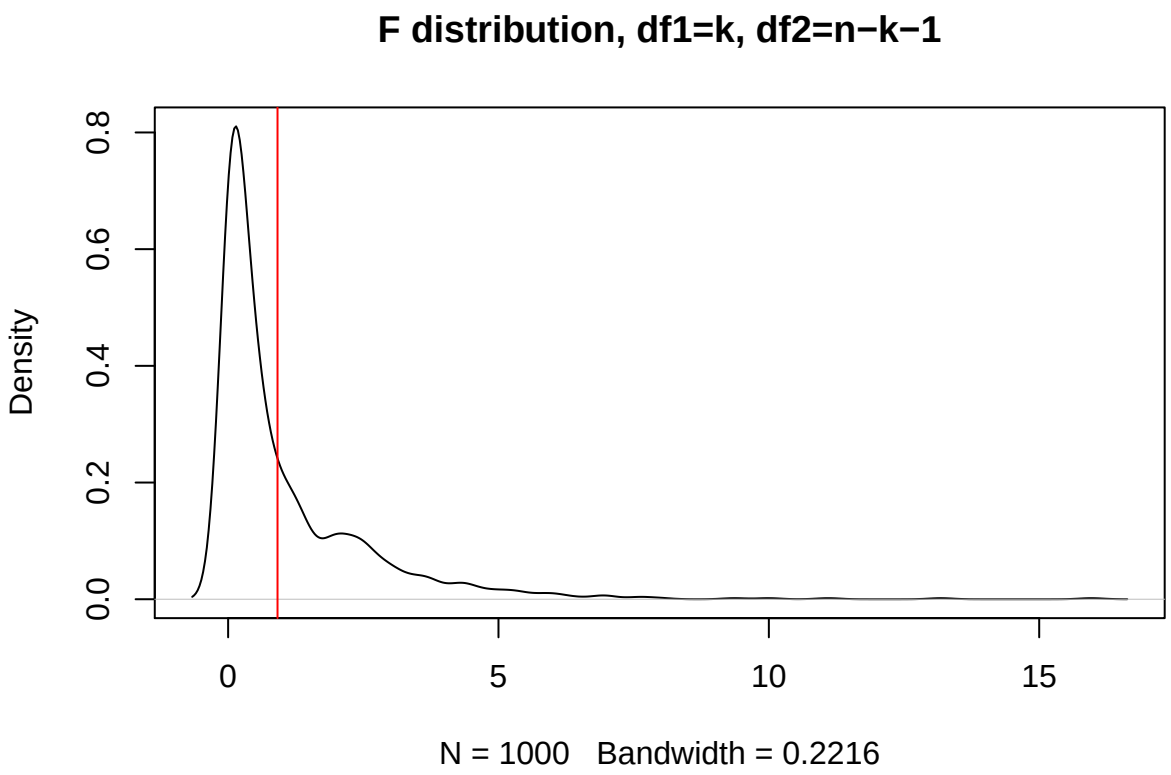
```
y <- saving$sav
x <- saving$inc

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 independent variable
```

Is this F-statistic large or small?

```
(F_statistic <- (R_squared / k) / ((1 - R_squared) / (n - k - 1)))
```

```
## [1] 0.9134505
```

```
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] 0.3415523
```

How about lm statistic?

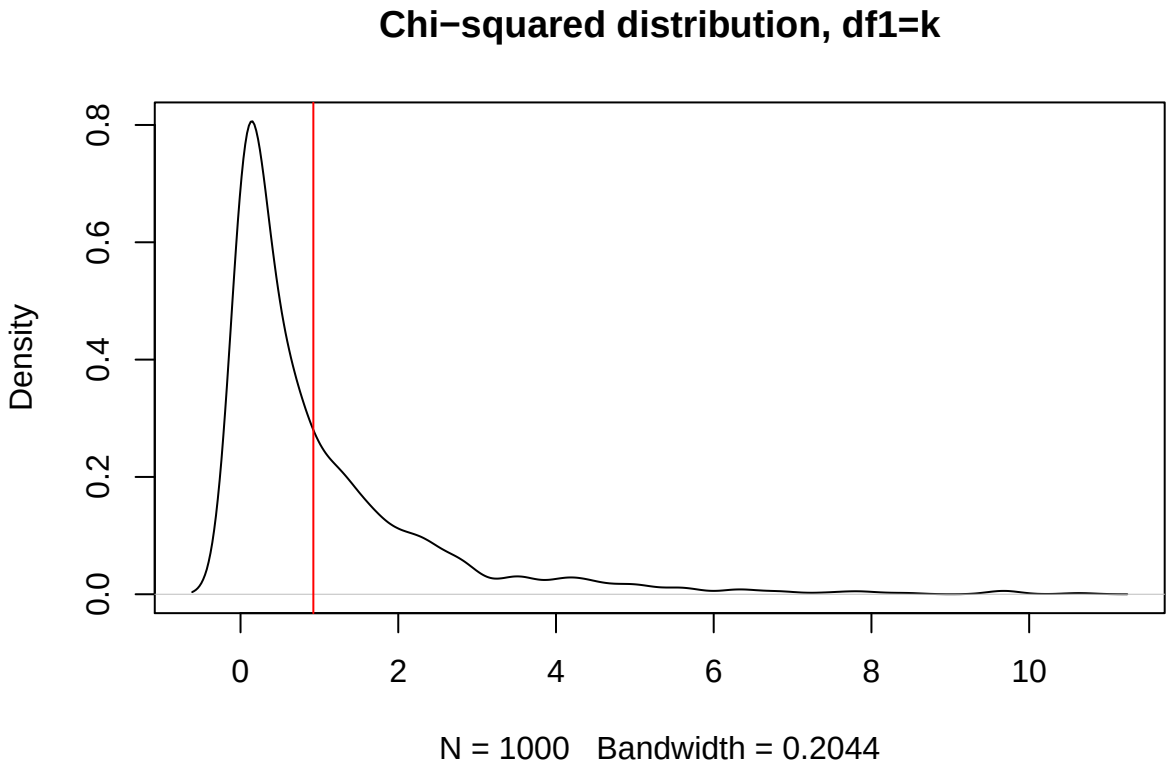
```
(lm_st <- n*(R_squared))
```

```
## [1] 0.9234846
```

```
(1 - pchisq(lm_st, df = k))
```

```
## [1] 0.3365617
```

```
plot(density(rchisq(1000,1)),
     main = "Chi-squared distribution, df1=k") ; abline(v = lm_st, col = 'red')
```




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

```
##
##  studentized Breusch-Pagan test
##
## data:  m_bp
## BP = 0.92348, df = 1, p-value = 0.3366
```

```
# R use lm statistics
t_bp$statistic
```

```
##        BP
## 0.9234846
```

```
t_bp$p.value
```

```
##        BP
## 0.3365617
```

The default Breusch-Pagan test is a test for linear forms of heteroskedasticity, e.g. as $\hat{y}$ goes up, the error variances go 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 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)

White test (Wooldridge Introductory Econometrics*, Chpt 8, Testing for heteroskedasticity)*

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).

```
bptest(m_het, varformula = ~ y_hat + I(y_hat^2))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m_het
## BP = 1.8493, df = 2, p-value = 0.3967
```

Robust 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. Also, notice that we call it "sandwich" as the "bread" ($(X'X)^{-1}$) and the "meat" ($X'V(\epsilon|X)X$).

When there's heteroskedasticity, the standard error is as follows:

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

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

$$\hat{V}(\hat{\beta}_{OLS}|X) = \frac{N}{N-K}(X'X)^{-1}\sum_{i=1}^{N}\{X_i X_i' \hat{\epsilon}_i^2\}(X'X)^{-1} \tag{3}$$

The constant is added since we are estimating the sample variance of the error.

White HC standard errors (sandwich standard error)

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

```
##             (Intercept)          inc
## (Intercept) 273435.2441 -27.044302703
## inc            -27.0443   0.003687837
```

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

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

```
##
## t test of coefficients:
##
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 124.842410 522.910360  0.2387  0.81180
## inc           0.146628   0.060728  2.4145  0.01761 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(m_het)
```

```
##
## Call:
## lm(formula = sav ~ inc, data = saving)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7324.7 -1176.7  -472.1   248.5 23469.3
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 124.84241  655.39312   0.190   0.8493
## inc           0.14663    0.05755   2.548   0.0124 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3197 on 98 degrees of freedom
## Multiple R-squared:  0.06213,    Adjusted R-squared:  0.05256
## F-statistic: 6.492 on 1 and 98 DF,  p-value: 0.01239
```

Regression comparison

```
stargazer(m_het, robust, header = F ,
          column.labels = c("ols", "robust se"))
```

Table 1:

	Dependent variable:	
	sav	
	OLS	*coefficient test*
	ols	robust se
	(1)	(2)
inc	0.147**	0.147**
	(0.058)	(0.061)
Constant	124.842	124.842
	(655.393)	(522.910)
Observations	100	
R^2	0.062	
Adjusted R^2	0.053	
Residual Std. Error	3,197.415 (df = 98)	
F Statistic	6.492** (df = 1; 98)	
Note:	*p<0.1; **p<0.05; ***p<0.01	

Get the robust standard error by hand

From Wooldridge "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] 0.003687837
```

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

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

```
##             (Intercept)           inc
## (Intercept) 273435.2441 -27.044302703
## inc            -27.0443   0.003687837
```

Weighted Least Squares

```
# specifying weight in lm()
wls1 = lm (sav ~ inc, saving,  weights=(1/inc ))
summary(wls1)
```

```
##
## Call:
## lm(formula = sav ~ inc, data = saving, weights = (1/inc))
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -69.893 -12.492  -4.802   4.132 210.642
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -124.95281  480.86061  -0.260  0.79552
## inc            0.17176    0.05681   3.023  0.00319 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.71 on 98 degrees of freedom
## Multiple R-squared:  0.08531,    Adjusted R-squared:  0.07597
## F-statistic:  9.14 on 1 and 98 DF,  p-value: 0.003192
```

```
# the weight is used
wls_bh =  saving %>%
  mutate(weight = 1/(sqrt(inc)),
         hsav = sav*weight,
         hinc = inc*weight ) %>%
  lm(hsav ~ hinc + weight + 0 ,. )

summary(wls_bh)
```

```
##
## Call:
## lm(formula = hsav ~ hinc + weight + 0, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -69.893 -12.492  -4.802   4.132 210.642
##
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## hinc      0.17176    0.05681   3.023  0.00319 **
## weight -124.95281  480.86061  -0.260  0.79552
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.71 on 98 degrees of freedom
## Multiple R-squared:  0.2259, Adjusted R-squared:  0.2101
## F-statistic:  14.3 on 2 and 98 DF,  p-value: 3.55e-06
```

```
# multivariate example
m_ols <- lm ( sav ~ inc + size + educ + age + black, saving )
```

```r
y_hat1 <- predict(m_ols)

# bp test
bptest(m_ols)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m_ols
## BP = 5.5756, df = 5, p-value = 0.3497
```

```r
# white test
bptest(m_ols, varformula = ~ y_hat1 + I(y_hat1^2))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m_ols
## BP = 3.6955, df = 2, p-value = 0.1576
```

```r
# WLS
m_wls <- lm ( sav ~ inc + size + educ + age + black  , saving, weights = (1/inc ) )
```

Feasible Generalizable Least Squares

```
# Run the regression of y on the x's and obtain the residual
wls3 <- lm ( sav ~ inc + size + educ + age + black, saving)
uhat_FGLS <- saving$sav - predict(wls3)

# Take the natural log of the squared residuals
luhat2_FGLS <- log(uhat_FGLS^2)

#  Regress the log of the squared residulas on the independent
fg <- lm(luhat2_FGLS ~ inc + size + educ + age + black, saving)

# Obtain the fitted values
g_hat <- predict(fg)

# Exponentiate the fitted values
h_hat <- exp(g_hat)

# Take the square root of h_hat
hhat_sqrt <- sqrt(h_hat)

fgls_saving = saving %>%
  # add the weight into data
  mutate(hhat_sqrt = hhat_sqrt,
         weight = 1/hhat_sqrt,
         ) %>%
  mutate_at(.vars = c("sav","inc","size","educ","age","black"),
            funs(./hhat_sqrt),
            )

m_fgls_bh <- lm(sav ~ inc + size + educ + age + black + weight + 0, fgls_saving)

summary(m_fgls_bh)
```

```
##
## Call:
## lm(formula = sav ~ inc + size + educ + age + black + weight +
##     0, data = fgls_saving)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8219 -1.2343 -0.4468  0.7849 17.0902
##
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## inc     1.231e-01  7.165e-02   1.718   0.0891 .
## size    7.345e+01  1.422e+02   0.516   0.6067
## educ    1.132e+02  8.460e+01   1.338   0.1840
## age     2.409e+01  3.469e+01   0.694   0.4891
## black   4.502e+02  9.970e+02   0.452   0.6526
## weight -2.248e+03  1.884e+03  -1.193   0.2360
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.74 on 94 degrees of freedom
## Multiple R-squared:  0.2674, Adjusted R-squared:  0.2206
## F-statistic: 5.719 on 6 and 94 DF,  p-value: 4.194e-05
```

```
# double check using built-in function
m_fgls <- lm(sav ~ inc + size + educ + age + black , saving, weights = 1/h_hat)
summary(m_fgls)
```

```
##
## Call:
## lm(formula = sav ~ inc + size + educ + age + black, data = saving,
##     weights = 1/h_hat)
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8219 -1.2343 -0.4468  0.7849 17.0902
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.248e+03  1.884e+03  -1.193   0.2360
## inc          1.231e-01  7.165e-02   1.718   0.0891 .
## size         7.345e+01  1.422e+02   0.516   0.6067
## educ         1.132e+02  8.460e+01   1.338   0.1840
## age          2.409e+01  3.469e+01   0.694   0.4891
## black        4.502e+02  9.970e+02   0.452   0.6526
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.74 on 94 degrees of freedom
## Multiple R-squared:  0.08973,    Adjusted R-squared:  0.04131
## F-statistic: 1.853 on 5 and 94 DF,  p-value: 0.11
```

Regression comparasion

```
stargazer( m_ols ,m_wls,  m_fgls,
           type = "latex",
           header = F,
           column.labels = c("ols", "wls", "fgls")
           )
```

Table 2:

	Dependent variable:		
	sav		
	ols	wls	fgls
	(1)	(2)	(3)
inc	0.109	0.101	0.123*
	(0.071)	(0.077)	(0.072)
size	67.661	−6.869	73.454
	(222.964)	(168.433)	(142.219)
educ	151.824	139.480	113.211
	(117.249)	(100.536)	(84.597)
age	0.286	21.747	24.093
	(50.031)	(41.306)	(34.692)
black	518.393	137.284	450.186
	(1,308.063)	(844.594)	(997.008)
Constant	−1,605.416	−1,854.814	−2,247.544
	(2,830.707)	(2,351.797)	(1,884.279)
Observations	100	100	100
R^2	0.083	0.104	0.090
Adjusted R^2	0.034	0.057	0.041
Residual Std. Error (df = 94)	3,228.598	30.022	2.740
F Statistic (df = 5; 94)	1.697	2.187*	1.853
Note:			*p<0.1; **p<0.05; ***p<0.01