Stat 504 HW 4

Q1

a

$E(Y_i|X_i) = \alpha + \beta X_i$. It is linear on X_i as we can observe.

b

$Var(Y_i|X_i) = \sigma^2 + \lambda X_i^2$. It is not constant on X_i since changes in X_i lead to changes in the variance.

c

```
#rm(list = ls())
library(sandwich)
library(estimatr)
```

```
## Warning: package 'estimatr' was built under R version 4.1.2
```

```
library(sandwich)

set.seed(42)
m = 1000

# write simulation function
sim_fun <- function(n = 100, lambda = 0, s2 = 1, a = 0.5, b = 1){

    # simulate data
    x    <- rnorm(n,0,1)
    mean <-  a+b*x
    se <-sqrt(s2 + lambda*x^2)
    y    <- rnorm(n,mean, se)

    # fit ols
    ols <- lm(y ~ x)

    # compute traditional ci
    ci <- confint(ols)

    # check if trad ci covers true b1
    trad.in <- ci["x", 1] <= b & b <= ci["x", 2]

    ols1 <- lm_robust(y ~ x)
    # compute robust ci
    #capture.output(
        ## capture.output serves to suppress printing
        #rob.ci <- Confint(ols, vcov. = vcovHC(ols, "HC0"))
    #)
    ###### CANNOT RUN ##########
```

```r
    #rob.ci <- confint(ols1, vcov. = vcovHC(ols, "HC0"))
    #S(ols, vcov = vcovHC(m, type = "HC0"))

    rob.ci <- confint(ols1)

    # check if rob ci covers true b1
    rob.in <- rob.ci["x", 1] <= b & b <= rob.ci["x", 2]

    # return results
    c(trad = trad.in, rob = rob.in)
}

lambda_n = seq(0,10,1)
sims = expand.grid(lambda_n)


for(i in 1:length(lambda_n)){
    cat("Simulation", i, "of", 10, "\n")
    cat("-params: lambda =", i)

    # simulates 1000 times
    sims_i <- replicate(m, sim_fun(n = 100, lambda = i, s2 = 1, a = 0.5, b = 1))

    # computes coverage
    sims[i, c(2,3)] <- apply(sims_i, 1, mean)
}
```

```
## Simulation 1 of 10
## -params: lambda = 1Simulation 2 of 10
## -params: lambda = 2Simulation 3 of 10
## -params: lambda = 3Simulation 4 of 10
## -params: lambda = 4Simulation 5 of 10
## -params: lambda = 5Simulation 6 of 10
## -params: lambda = 6Simulation 7 of 10
## -params: lambda = 7Simulation 8 of 10
## -params: lambda = 8Simulation 9 of 10
## -params: lambda = 9Simulation 10 of 10
## -params: lambda = 10Simulation 11 of 10
## -params: lambda = 11
```

```r
sims
```

```
##    Var1    V2    V3
## 1     0 0.858 0.939
## 2     1 0.801 0.933
## 3     2 0.806 0.939
## 4     3 0.773 0.924
## 5     4 0.765 0.952
## 6     5 0.770 0.925
## 7     6 0.770 0.932
## 8     7 0.757 0.933
## 9     8 0.767 0.934
## 10    9 0.750 0.929
## 11   10 0.758 0.935
```

d

Our results show that as heteroskedasticity occurs, robust standard errors is a solution to this problem. Compared with the traditional standard error which provides a barely good confidence interval, the confidence interval generated by the robust standard error is fairly "robust", meaning that it is correctly covering the true mean for a good amount of time.

Q2

a

```r
library(haven)
discrimination_df <- read_dta('bm.dta')
summary(lm(call ~ black, data = discrimination_df))
```

```
##
## Call:
## lm(formula = call ~ black, data = discrimination_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.09651 -0.09651 -0.06448 -0.06448  0.93552
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.096509   0.005505  17.532  < 2e-16 ***
## black       -0.032033   0.007785  -4.115 3.94e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2716 on 4868 degrees of freedom
## Multiple R-squared:  0.003466,   Adjusted R-squared:  0.003261
## F-statistic: 16.93 on 1 and 4868 DF,  p-value: 3.941e-05
```

The estimated coefficient is -0.032033, meaning that people with black names tend to have a 0.032033 less chance of being called back than people with white sounding names. For white sound names, the rate is 0.096509. For black sound names, the rate is 0.096509 - 0.032033 = 0.064476.

b

No, we can still run a linear regression and construct a confidence interval for estimation. Because we can use a BLP to explore th correlation between the two variables without constraint.

c

```r
lm_robust(call ~ black, data = discrimination_df)
```

```
##                Estimate  Std. Error   t value     Pr(>|t|)    CI Lower
## (Intercept)  0.09650924 0.005985301 16.124375 5.044644e-57  0.08477535
## black       -0.03203285 0.007784969 -4.114705 3.940803e-05 -0.04729491
##               CI Upper   DF
## (Intercept)  0.1082431 4868
## black       -0.0167708 4868
```

As shown above, the confidence interval using robust standard errors is [-0.04729491, -0.0167708].

d

```
summary(lm(call ~ black + female + yearsexp, data = discrimination_df))
```

```
##
## Call:
## lm(formula = call ~ black + female + yearsexp, data = discrimination_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18498 -0.09225 -0.07790 -0.05694  0.96070
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0647647  0.0106608   6.075 1.33e-09 ***
## black       -0.0320284  0.0077720  -4.121 3.83e-05 ***
## female       0.0077916  0.0092281   0.844    0.399
## yearsexp     0.0032831  0.0007708   4.259 2.09e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2712 on 4866 degrees of freedom
## Multiple R-squared:  0.007367,   Adjusted R-squared:  0.006755
## F-statistic: 12.04 on 3 and 4866 DF,  p-value: 7.551e-08
```

No, it does not change much. Race is potentially a quite important factor (consciously or subconsciously) when hiring people decide whether to hire a person, among all the factors.

e

If our assumption of unconfoundedness only includes gender ands years of experience, yes we can interpret the result causally. But be careful that a few other factors have not been taken into account, such as education and computer skills in the data set. We need to consider those before making a conclusion. But if everything has been adjusted for, we can claim some causal relationship because the researchers were able to intervene in this experiment and this name sounding psuedo race can is the D variable in our causal story.

Q3

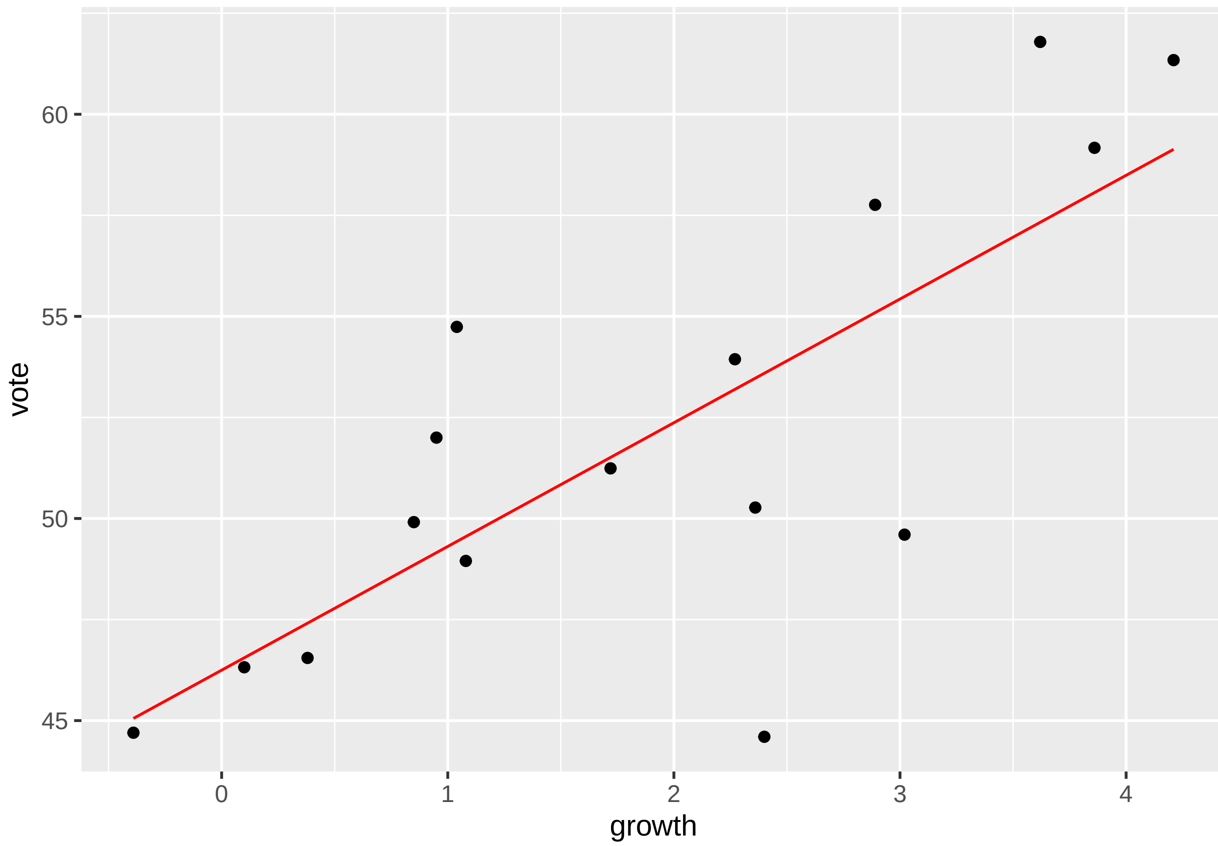
a

```
election_df <- read.csv('hibbs.dat', sep = '', header = T)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
election_model <- lm(vote ~ growth, data = election_df)
ggplot(election_df, aes(x = growth, y = vote)) +
```

```r
    geom_point() +
    geom_line(aes(y = fitted(election_model)), col = "red")
```



b

```r
summary(election_model)
```

```
##
## Call:
## lm(formula = vote ~ growth, data = election_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9929 -0.6674  0.2556  2.3225  5.3094
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  46.2476     1.6219  28.514 8.41e-14 ***
## growth        3.0605     0.6963   4.396  0.00061 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.763 on 14 degrees of freedom
## Multiple R-squared:  0.5798, Adjusted R-squared:  0.5498
## F-statistic: 19.32 on 1 and 14 DF,  p-value: 0.00061
```

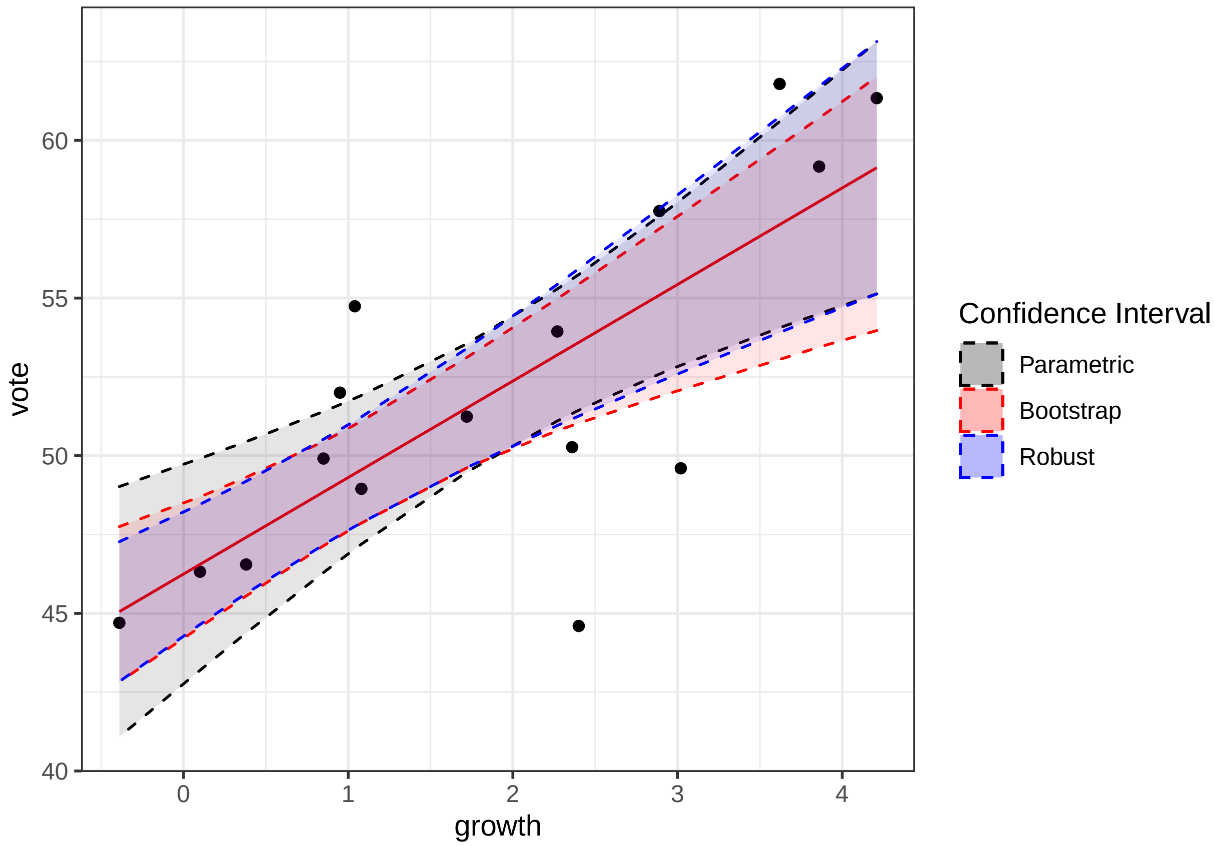
```r
confint(election_model)
```

```
##                 2.5 %    97.5 %
## (Intercept) 42.768951 49.726345
## growth       1.567169  4.553887
```

1% increase in average growth is associated with 3.0605 increase in vote share. If no growth, vote share is predicted to be 46.2476.

c

```r
# confidence level
alpha <- 0.05

# classical parametric confidence interval
param.ci <- predict(election_model, interval = "confidence",newdata = election_df, level = 1-alpha)

# robust parametric confidence interval
election_model1 <- lm_robust(vote ~ growth, data = election_df)
robust.ci <- predict(election_model1, interval = "confidence", newdata = election_df, level = 1-alpha)
robust.ci <- data.frame(robust.ci)
colnames(robust.ci) <- c("robust.fit", "robust.lwr", "robust.upr")

# nonparametric bootstrap confidence interval

# bootstrap function
boot.fun <- function(){
    idx <- sample(nrow(election_df), replace = T)
    ols.boot <- lm(vote ~ growth, data = election_df[idx,])
    yhat.boot <- predict(ols.boot, newdata = election_df)
    return(yhat.boot)
}

# replicate 10,000 times
boot.out <- replicate(10000, boot.fun())

# quantile confidence interval
boot.ci <- t(apply(boot.out, 1, quantile, c(alpha/2, 1-alpha/2)))
colnames(boot.ci) <- c("boot.lwr", "boot.upr")

election <-cbind(election_df, param.ci, robust.ci, boot.ci)

# gpplot
ggplot(election, aes(x = growth, y = vote)) +
            geom_point() +
            geom_line(aes(y = fitted(election_model)), col = "red") +
            geom_ribbon(aes(ymin = lwr, ymax = upr, fill = "Parametric", color = "Parametric"), alpha=0
            geom_ribbon(aes(ymin = boot.lwr, ymax = boot.upr, fill = "Bootstrap", color = "Bootstrap"),
            geom_ribbon(aes(ymin = robust.lwr, ymax = robust.upr, fill = "Robust", color = "Robust"), al
            scale_fill_manual(name = "Confidence Interval", values = c("Parametric" = "black", "Bootstra
            scale_color_manual(name = "Confidence Interval", values = c("Parametric" = "black", "Bootstr
            theme_bw()
```



The confidence intervals are listed below

```
election
```

```
##    year growth  vote inc_party_candidate other_candidate      fit      lwr
## 1  1952   2.40 44.60           Stevenson      Eisenhower 53.59292 51.44004
## 2  1956   2.89 57.76          Eisenhower       Stevenson 55.09257 52.58886
## 3  1960   0.85 49.91               Nixon         Kennedy 48.84910 46.29591
## 4  1964   4.21 61.34             Johnson       Goldwater 59.13247 55.13276
## 5  1968   3.02 49.60            Humphrey           Nixon 55.49044 52.86714
## 6  1972   3.62 61.79               Nixon        McGovern 57.32676 54.05742
## 7  1976   1.08 48.95                Ford          Carter 49.55302 47.19459
## 8  1980  -0.39 44.70              Carter          Reagan 45.05404 41.08652
## 9  1984   3.86 59.17              Reagan         Mondale 58.06129 54.50307
## 10 1988   2.27 53.94           Bush, Sr.         Dukakis 53.19505 51.10191
## 11 1992   0.38 46.55           Bush, Sr.         Clinton 47.41065 44.37629
## 12 1996   1.04 54.74             Clinton            Dole 49.43060 47.04070
## 13 2000   2.36 50.27                Gore       Bush, Jr. 53.47049 51.33770
## 14 2004   1.72 51.24           Bush, Jr.           Kerry 51.51176 49.47656
## 15 2008   0.10 46.32              McCain           Obama 46.55370 43.19553
## 16 2012   0.95 52.00               Obama          Romney 49.15515 46.69063
##         upr robust.fit robust.lwr robust.upr boot.lwr boot.upr
## 1  55.74579   53.59292   51.25705   55.92878 51.02966 55.44190
## 2  57.59628   55.09257   52.34993   57.83522 51.88003 57.19601
## 3  51.40229   48.84910   47.17544   50.52276 47.14314 50.44880
## 4  63.13218   59.13247   55.13013   63.13481 53.96846 62.01252
## 5  58.11375   55.49044   52.63228   58.34861 52.08935 57.66009
## 6  60.59609   57.32676   53.90732   60.74619 53.05283 59.83204
```

```
## 7  51.91144   49.55302   47.87295   51.23308 47.86259 51.09187
## 8  49.02156   45.05404   42.83552   47.27256 42.82184 47.75136
## 9  61.61950   58.06129   54.40762   61.71495 53.44373 60.71002
## 10 55.28818   53.19505   50.95739   55.43270 50.79692 54.97990
## 11 50.44500   47.41065   45.63275   49.18854 45.55160 49.31414
## 12 51.82049   49.43060   47.75442   51.10677 47.74670 50.98193
## 13 55.60329   53.47049   51.16536   55.77563 50.95701 55.29542
## 14 53.54695   51.51176   49.62348   53.40003 49.61503 53.11331
## 15 49.91187   46.55370   44.64768   48.45972 44.56575 48.69481
## 16 51.61967   49.15515   47.48347   50.82683 47.45611 50.72437
```

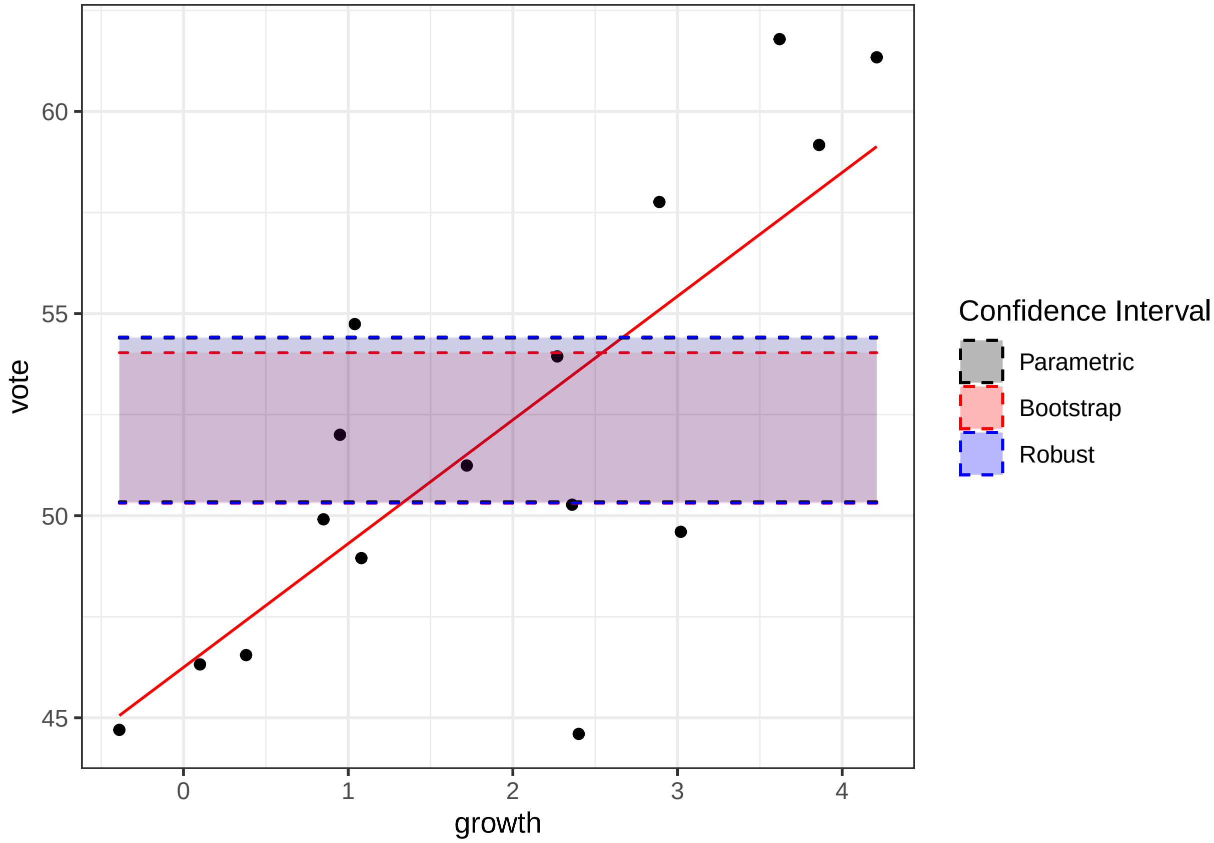
d

```
# Point estimate using election_model
46.248 + 3.061 * 2
```

```
## [1] 52.37
```

```
# = 52.37

# confidence level
alpha <- 0.05

election_df1 = election_df['vote']
election_df1$growth <- rep(2,16)

# classical parametric confidence interval
param.ci <- predict(election_model, interval = "confidence",newdata = election_df1, level = 1-alpha)

# robust parametric confidence interval
election_model1 <- lm_robust(vote ~ growth, data = election_df)
robust.ci <- predict(election_model1, interval = "confidence", newdata = election_df1, level = 1-alpha)
robust.ci <- data.frame(robust.ci)
colnames(robust.ci) <- c("robust.fit", "robust.lwr", "robust.upr")

# nonparametric bootstrap confidence interval

# bootstrap function
boot.fun <- function(){
    idx <- sample(nrow(election_df), replace = T)
    ols.boot <- lm(vote ~ growth, data = election_df[idx,])
    yhat.boot <- predict(ols.boot, newdata = election_df1)
    return(yhat.boot)
}

# replicate 10,000 times
boot.out <- replicate(10000, boot.fun())

# quantile confidence interval
boot.ci <- t(apply(boot.out, 1, quantile, c(alpha/2, 1-alpha/2)))
colnames(boot.ci) <- c("boot.lwr", "boot.upr")

election <-cbind(election_df, param.ci, robust.ci, boot.ci)

# gpplot
```

```
ggplot(election, aes(x = growth, y = vote)) +
            geom_point() +
            geom_line(aes(y = fitted(election_model)), col = "red") +
            geom_ribbon(aes(ymin = lwr, ymax = upr, fill = "Parametric", color = "Parametric"), alpha=0
            geom_ribbon(aes(ymin = boot.lwr, ymax = boot.upr, fill = "Bootstrap", color = "Bootstrap"),
            geom_ribbon(aes(ymin = robust.lwr, ymax = robust.upr, fill = "Robust", color = "Robust"), al
            scale_fill_manual(name = "Confidence Interval", values = c("Parametric" = "black", "Bootstra
            scale_color_manual(name = "Confidence Interval", values = c("Parametric" = "black", "Bootst
            theme_bw()
```



The point estimate is 52.37. The confidence intervals are listed below

```
election
```

```
##    year growth  vote inc_party_candidate other_candidate     fit      lwr
## 1  1952   2.40 44.60           Stevenson      Eisenhower 52.3687 50.34504
## 2  1956   2.89 57.76          Eisenhower       Stevenson 52.3687 50.34504
## 3  1960   0.85 49.91               Nixon         Kennedy 52.3687 50.34504
## 4  1964   4.21 61.34             Johnson       Goldwater 52.3687 50.34504
## 5  1968   3.02 49.60            Humphrey           Nixon 52.3687 50.34504
## 6  1972   3.62 61.79               Nixon        McGovern 52.3687 50.34504
## 7  1976   1.08 48.95                Ford          Carter 52.3687 50.34504
## 8  1980  -0.39 44.70              Carter          Reagan 52.3687 50.34504
## 9  1984   3.86 59.17              Reagan         Mondale 52.3687 50.34504
## 10 1988   2.27 53.94          Bush, Sr.          Dukakis 52.3687 50.34504
## 11 1992   0.38 46.55          Bush, Sr.          Clinton 52.3687 50.34504
## 12 1996   1.04 54.74             Clinton            Dole 52.3687 50.34504
## 13 2000   2.36 50.27                Gore       Bush, Jr. 52.3687 50.34504
```

```
## 14 2004   1.72 51.24           Bush, Jr.           Kerry 52.3687 50.34504
## 15 2008   0.10 46.32              McCain           Obama 52.3687 50.34504
## 16 2012   0.95 52.00               Obama          Romney 52.3687 50.34504
##          upr robust.fit robust.lwr robust.upr boot.lwr boot.upr
## 1  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 2  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 3  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 4  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 5  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 6  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 7  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 8  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 9  54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 10 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 11 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 12 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 13 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 14 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 15 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
## 16 54.39236    52.3687   50.31769   54.41972 50.30632 54.03242
```

Q4

a

```r
house_df <- read.csv('SaratogaHouses.csv')
house_model1 <- lm_robust(price ~ fireplaces, data = house_df)
confint(house_model1)
```

```
##                 2.5 %    97.5 %
## (Intercept) 165679.42 177968.37
## fireplaces   57437.51  75960.13
```

```r
summary(house_model1)
```

```
##
## Call:
## lm_robust(formula = price ~ fireplaces, data = house_df)
##
## Standard error type:  HC2
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|) CI Lower CI Upper   DF
## (Intercept)   171824       3133   54.85 0.00e+00   165679   177968 1726
## fireplaces     66699       4722   14.13 5.94e-43    57438    75960 1726
##
## Multiple R-squared:  0.142 , Adjusted R-squared:  0.1415
## F-statistic: 199.5 on 1 and 1726 DF,  p-value: < 2.2e-16
```

A 95% confidence interval is [57437.51, 75960.13]. The existence of fireplaces seems positively correlated with house prices after adjusting for non-constant variance. The houses with fireplaces can be sold 66699 more.

b

```r
house_model2 <- lm_robust(price ~ bedrooms, data = house_df)
confint(house_model2)
```

```
##                2.5 %   97.5 %
## (Intercept) 41585.01 78140.91
## bedrooms    42234.92 54200.69
```

```r
summary(house_model2)
```

```
##
## Call:
## lm_robust(formula = price ~ bedrooms, data = house_df)
##
## Standard error type:  HC2
##
## Coefficients:
##             Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper   DF
## (Intercept)    59863       9319   6.424 1.716e-10    41585    78141 1726
## bedrooms       48218       3050  15.807 1.146e-52    42235    54201 1726
##
## Multiple R-squared:  0.1603 ,    Adjusted R-squared:  0.1598
## F-statistic: 249.9 on 1 and 1726 DF,  p-value: < 2.2e-16
```

A 95% confidence interval is [57437.51, 75960.13]. The number of bedrooms seems positively correlated with house prices after adjusting for non-constant variance. The houses with one additional bedroom can be sold 48218 more.

c

```r
house_model3 <- lm_robust(price ~ ., data = house_df)
confint(house_model3)
```

```
##                                2.5 %        97.5 %
## (Intercept)            -3.622186e+04  54739.350973
## lotSize                 3.192416e+03  12006.481695
## age                    -2.799696e+02     19.077861
## landValue               7.804355e-01      1.063381
## livingArea              5.880554e+01     81.113473
## pctCollege             -3.763515e+02    156.032975
## bedrooms               -1.363845e+04  -2031.932118
## fireplaces             -6.338209e+03   8411.435710
## bathrooms               1.580555e+04  30419.355538
## rooms                   1.211324e+03   4828.197901
## heatinghot air         -1.868721e+04  18852.117289
## heatinghot water/steam -3.059732e+04   9852.826373
## fuelgas                -7.237740e+03  29100.287525
## fueloil                -1.394359e+04  27044.536325
## sewerpublic/commercial -3.938663e+04  46028.963170
## sewerseptic            -3.748894e+04  47179.152872
## waterfrontYes           7.364957e+04 166738.389880
## newConstructionYes     -5.946916e+04 -31417.684407
## centralAirYes           3.877057e+03  16029.125796
```

```r
summary(house_model3)
```

```
##
```

```
## Call:
## lm_robust(formula = price ~ ., data = house_df)
##
## Standard error type:  HC2
##
## Coefficients:
##                          Estimate Std. Error   t value  Pr(>|t|)   CI Lower
## (Intercept)             9.259e+03  2.319e+04  0.399284 6.897e-01 -3.622e+04
## lotSize                 7.599e+03  2.247e+03  3.382140 7.355e-04  3.192e+03
## age                    -1.304e+02  7.623e+01 -1.711102 8.724e-02 -2.800e+02
## landValue               9.219e-01  7.213e-02 12.781181 8.594e-36  7.804e-01
## livingArea              6.996e+01  5.687e+00 12.301930 2.185e-33  5.881e+01
## pctCollege             -1.102e+02  1.357e+02 -0.811674 4.171e-01 -3.764e+02
## bedrooms               -7.835e+03  2.959e+03 -2.648094 8.169e-03 -1.364e+04
## fireplaces              1.037e+03  3.760e+03  0.275690 7.828e-01 -6.338e+03
## bathrooms               2.311e+04  3.725e+03  6.203952 6.892e-10  1.581e+04
## rooms                   3.020e+03  9.220e+02  3.275103 1.077e-03  1.211e+03
## heatinghot air          8.245e+01  9.570e+03  0.008616 9.931e-01 -1.869e+04
## heatinghot water/steam -1.037e+04  1.031e+04 -1.005862 3.146e-01 -3.060e+04
## fuelgas                 1.093e+04  9.264e+03  1.180036 2.382e-01 -7.238e+03
## fueloil                 6.550e+03  1.045e+04  0.626903 5.308e-01 -1.394e+04
## sewerpublic/commercial  3.321e+03  2.177e+04  0.152524 8.788e-01 -3.939e+04
## sewerseptic             4.845e+03  2.158e+04  0.224476 8.224e-01 -3.749e+04
## waterfrontYes           1.202e+05  2.373e+04  5.064901 4.528e-07  7.365e+04
## newConstructionYes     -4.544e+04  7.151e+03 -6.354789 2.670e-10 -5.947e+04
## centralAirYes           9.953e+03  3.098e+03  3.212873 1.339e-03  3.877e+03
##                          CI Upper   DF
## (Intercept)             54739.351 1709
## lotSize                 12006.482 1709
## age                        19.078 1709
## landValue                   1.063 1709
## livingArea                 81.113 1709
## pctCollege                156.033 1709
## bedrooms                -2031.932 1709
## fireplaces               8411.436 1709
## bathrooms               30419.356 1709
## rooms                    4828.198 1709
## heatinghot air          18852.117 1709
## heatinghot water/steam   9852.826 1709
## fuelgas                 29100.288 1709
## fueloil                 27044.536 1709
## sewerpublic/commercial  46028.963 1709
## sewerseptic             47179.153 1709
## waterfrontYes          166738.390 1709
## newConstructionYes     -31417.684 1709
## centralAirYes           16029.126 1709
##
## Multiple R-squared:  0.6534 ,    Adjusted R-squared:  0.6498
## F-statistic: 101.5 on 18 and 1709 DF,  p-value: < 2.2e-16
```

```
cor(house_df$room, house_df$bedrooms)
```

```
## [1] 0.6718633
```

For bedroom, the confidence interval is [-1.363845e+04, -2031.932118]; for fireplaces it is [-6.338209e+03, 8411.435710].

Yes, the coefficient of fireplace is no longer significant, and the coefficient of bedroom even becomes negative. The results show that possibly some omitted variable bias has occurred. Fireplaces seem uncorrelated with house prices; the increase in the number of bedrooms is associated with lower house prices. The coefficient of bedrooms has changed possibly due to collinearity because the correlation between room and bedroom is as high as 0.67. Since we already have room with a postive significant coefficient, and if we keep only one of the two correlated variables the coefficient will be normal.

Q5

1

No.

2

Yes.