Homework 3

Patrick Chao

Due Sunday, October 24 at 11:59pm

1 Instructions

Setup. Pull the latest version of this assignment from Github and set your working directory to `stat-961-fall-2021/homework/homework-3`. Consult the getting started guide if you need to brush up on `R`, `LaTeX`, or `Git`.

Collaboration. The collaboration policy is as stated on the Syllabus:

> "Students are permitted to work together on homework assignments, but solutions must be written up and submitted individually. Students must disclose any sources of assistance they received; furthermore, they are prohibited from verbatim copying from any source and from consulting solutions to problems that may be available online and/or from past iterations of the course."

In accordance with this policy,

Please list anyone you discussed this homework with: Jeffrey Zhang, Dongwoo Kim, Abhinav Chakraborty, Ryan Brill

Please list what external references you consulted (e.g. articles, books, or websites): None

Writeup. Use this document as a starting point for your writeup, adding your solutions between `\begin{sol}` and `\end{sol}`. See the preparing reports guide for guidance on compilation, creation of figures and tables, and presentation quality. Show all the code you wrote to produce your numerical results, and include complete derivations typeset in LaTeX for the mathematical questions.

Programming. The `tidyverse` paradigm for data manipulation (`dplyr`) and plotting (`ggplot2`) are strongly encouraged, but points will not be deducted for using base `R`.

```
> library(tidyverse)
```

Grading. Each sub-part of each problem will be worth 3 points: 0 points for no solution or completely wrong solution; 1 point for some progress; 2 points for a mostly correct solution; 3 points for a complete and correct solution modulo small flaws. The presentation quality of the solution for each problem (as exemplified by the guidelines in Section 3 of the preparing reports guide) will be evaluated out of an additional 3 points.

Submission. Compile your writeup to PDF and submit to Gradescope.

Problem 1. Heteroskedasticity and correlated errors in the intercept-only model.

Suppose that

$$y_i = \beta_0 + \epsilon_i, \quad \text{where } \boldsymbol{\epsilon} \sim N(0, \boldsymbol{\Sigma}) \tag{1}$$

for some positive definite $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$. The goal of this problem is to investigate the effects of heteroskedasticity and correlated errors on the validity and efficiency of least squares estimation and inference.

(a) (Validity of least squares inference) What is the usual least squares estimate $\widehat{\beta}_0^{\text{LS}}$ for β_0 (from Unit 2)? What is its variance under the model (1)? What is the usual variance estimate $\widehat{\text{Var}}[\widehat{\beta}_0^{\text{LS}}]$ (from Unit 2), and what is this estimator's expectation under (1)? The ratio

$$\tau_1 \equiv \frac{\mathbb{E}[\widehat{\text{Var}}[\widehat{\beta}_0^{\text{LS}}]]}{\text{Var}[\widehat{\beta}_0^{\text{LS}}]} \tag{2}$$

is a measure of the validity of usual least squares inference under (1). Write down an expression for τ_1, and discuss the implications of τ_1 for the Type-I error of the hypothesis test of $H_0 : \beta_0 = 0$ and for the coverage of the confidence interval for β_0.

(b) (Efficiency of least squares estimator) Let's assume $\boldsymbol{\Sigma}$ is known. We could get valid inference based on $\widehat{\beta}_0^{\text{LS}}$ by using the variance formula from part (a). Alternatively, we could use the maximum likelihood estimate $\widehat{\beta}_0^{\text{ML}}$ for β_0. What is the variance of $\widehat{\beta}_0^{\text{ML}}$? The ratio

$$\tau_2 \equiv \frac{\text{Var}[\widehat{\beta}_0^{\text{LS}}]}{\text{Var}[\widehat{\beta}_0^{\text{ML}}]} \tag{3}$$

is a measure of the efficiency of the usual least squares estimator under (1), recalling that the maximum likelihood estimator is most efficient. Write down an expression for τ_2, and discuss the implications of τ_2 for the power of the hypothesis test of $H_0 : \beta_0 = 0$ and for the width of the confidence interval for β_0.

(c) (Special case: Heteroskedasticity) Suppose $\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \dots, \sigma_n^2)$ for some $\sigma_1^2, \dots, \sigma_n^2 > 0$. Compute the ratios τ_1 and τ_2 defined in equations (2) and (3), respectively. How do these ratios depend on $(\sigma_1^2, \dots, \sigma_n^2)$, and what are the implications for validity and efficiency?

(d) (Special case: Correlated errors) Suppose $(\epsilon_1, \dots, \epsilon_n)$ are *equicorrelated*, i.e.

$$\Sigma_{j_1 j_2} = \begin{cases} 1, & \text{if } j_1 = j_2; \\ \rho, & \text{if } j_1 \neq j_2. \end{cases} \tag{4}$$

for some $\rho \geq 0$. Compute the ratios τ_1 and τ_2 defined in equations (2) and (3), respectively. How do these ratios depend on ρ, and what are the implications for validity and efficiency?

Solution 1.

(a) The usual least squares estimate is $\hat{\beta}_0^{LS} = \bar{y}$. The variance under the model is

$$\text{Var}[\bar{y}] = \text{Var}\left[\frac{1}{n}\sum_{i=1}^n y_i\right] = \frac{1}{n^2}\text{Var}[\mathbb{1}^T \boldsymbol{Y}] = \frac{1}{n^2}\mathbb{1}^T \boldsymbol{\Sigma} \mathbb{1}.$$

The usual variance estimate is

$$\widehat{\text{Var}}[\hat{\beta}_0^{LS}] = \frac{1}{n}\frac{\|\hat{\varepsilon}\|^2}{n-1} = \frac{\|\boldsymbol{Y} - \bar{y}\mathbb{1}\|^2}{n(n-1)}.$$

Taking the expectation,

$$\begin{aligned}
\mathbb{E}[\widehat{\mathrm{Var}}[\hat{\beta}_0^{LS}]] &= \frac{1}{n(n-1)}\mathbb{E}\left\|\boldsymbol{Y} - \frac{1}{n}\mathbb{1}\mathbb{1}^T\boldsymbol{Y}\right\|^2 \\
&= \frac{1}{n(n-1)}\mathbb{E}\left[\boldsymbol{Y}^T\left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^T\right)\boldsymbol{Y}\right] \\
&= \frac{1}{n(n-1)}\left(\mathrm{Tr}\left(\left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^T\right)\boldsymbol{\Sigma}\right) + \beta_0^2\mathbb{1}^T\left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^T\right)\mathbb{1}\right) \\
&= \frac{1}{n(n-1)}\left(\mathrm{Tr}(\boldsymbol{\Sigma}) - \frac{1}{n}\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1}\right).
\end{aligned}$$

The second inequality follows from the fact that $\left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^T\right)^2 = \left(I - \frac{1}{n}\mathbb{1}\mathbb{1}^T\right)$ and the third equality follows from the expectation of a quadratic form.
We may rewrite τ_1 as

$$\tau_1 = \frac{1}{n-1}\left(\frac{n\,\mathrm{Tr}(\boldsymbol{\Sigma})}{\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1}} - 1\right).$$

Intuitively, τ_1 captures how accurate the estimated variance of $\hat{\beta}_0^{LS}$ actually is. First, note that $\tau_1 \geq 0$ since these variances are both positive. We may furthermore notice that

$$\mathrm{Tr}(\boldsymbol{\Sigma}) \leq \mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1}$$

since the trace of a matrix is less than the sum of all entries. Therefore $0 \leq \tau_1 \leq 1$. Large values of τ_1 are *better*, the estimated variance is close to the true variance, and the Type-I error and coverage are close to the desired values. On the other hand, small values of τ_1 are worse, resulting in poor coverage and inflated Type-I error.

(b) Define $\tilde{y} = \boldsymbol{\Sigma}^{-1/2}y$ and $\tilde{x} = \boldsymbol{\Sigma}^{-1/2}x$. Then $\hat{\beta}_0^{ML}$ is

$$\hat{\beta}_0^{ML} = (\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}})^{-1}\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{Y}} = (\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1})^{-1}\mathbb{1}^T\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Sigma}^{-1/2}\boldsymbol{Y} = \frac{\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{Y}}{\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1}}.$$

The variance follows from the fact that $Y \sim \mathcal{N}(\beta_0\mathbb{1}, \boldsymbol{\Sigma})$.

$$\mathrm{Var}[\hat{\beta}_0^{ML}] = (\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1})^{-2}\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}\mathbb{1} = \frac{1}{\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1}}.$$

Plugging this into τ_2, we have

$$\tau_2 = \frac{1}{n^2}(\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1})(\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1}).$$

We may show that $\tau_2 \geq 1$. Since $\mathbb{1}^T\mathbb{1} = n$,

$$\begin{aligned}
\tau_2 &= \frac{\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1}}{\mathbb{1}^T\mathbb{1}}\frac{\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1}}{\mathbb{1}^T\mathbb{1}} \\
&\geq \min_{\boldsymbol{x}:\|\boldsymbol{x}\|_2=1}(\boldsymbol{x}^T\boldsymbol{\Sigma}\boldsymbol{x})(\boldsymbol{x}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{x}).
\end{aligned}$$

Since Σ is a symmetric positive definition covariance matrix, we can consider the diagonalization $\boldsymbol{\Sigma} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}^{-1}$ with a diagonal matrix $\boldsymbol{D} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ and orthogonal matrix $\boldsymbol{P}$. Let

$\boldsymbol{z} = \boldsymbol{P}\boldsymbol{x}$.

$$\begin{aligned}
\tau_2 &\geq \min_{\boldsymbol{x}:\|\boldsymbol{x}\|_2=1} (\boldsymbol{x}^T\boldsymbol{\Sigma}\boldsymbol{x})(\boldsymbol{x}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{x}) \\
&= \min_{\boldsymbol{z}:\|\boldsymbol{z}\|_2=1} (\boldsymbol{z}^T\boldsymbol{D}\boldsymbol{z})(\boldsymbol{z}^T\boldsymbol{D}^{-1}\boldsymbol{z}) \\
&= \min_{\boldsymbol{z}:\|\boldsymbol{z}\|_2=1} \left(\sum_{i=1}^n \lambda_i z_i^2\right)\left(\sum_{i=1}^n \frac{1}{\lambda_i} z_i^2\right) \\
&\geq \left(\sum_{i=1}^n \frac{\sqrt{\lambda_i}}{\sqrt{\lambda_i}} z_i^2\right)^2 = 1.
\end{aligned}$$

The inequality follows from Cauchy-Schwartz.
Intuitively, since the maximum likelihood estimator is most efficient, $\tau_2 \geq 1$ means that the least squares estimator will always have more variance than the MLE. However, if τ_2 is close to 1, then the least squares estimator does not perform much worse than the MLE. Small values τ_2 are *better*, the hypothesis test has greater power and the width of the confidence interval is smaller.

(c) For $\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \ldots, \sigma_n^2)$, we have

$$\begin{aligned}
\tau_1 &= \frac{1}{n-1}\left(\frac{n\,\text{Tr}(\boldsymbol{\Sigma})}{\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1}} - 1\right) = 1 \\
\tau_2 &= \frac{1}{n^2}(\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1})(\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1}) = \frac{\left(\sum_{i=1}^n \sigma_i^2\right)\left(\sum_{i=1}^n \sigma_i^{-2}\right)}{n^2}.
\end{aligned}$$

In other words, your hypothesis test will still be valid and have proper Type I error, however the test may be inefficient if the values of σ_i^2 are on different scales. If the σ_i^2 are all close to each other, then you will have a more powerful test with narrower intervals.

(d) Note that $\mathbb{1}$ is an eigenvector of Σ with eigenvalue $(1 + (n-1)\rho)$. Therefore

$$\begin{aligned}
\tau_1 &= \frac{1}{n-1}\left(\frac{n\,\text{Tr}(\boldsymbol{\Sigma})}{\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1}} - 1\right) = \frac{1}{n-1}\left(\frac{n^2}{n + (n^2 - n)\rho} - 1\right) = \frac{1-\rho}{1 + (n-1)\rho} \\
\tau_2 &= \frac{1}{n^2}(\mathbb{1}^T\boldsymbol{\Sigma}\mathbb{1})(\mathbb{1}^T\boldsymbol{\Sigma}^{-1}\mathbb{1}) = \frac{1}{n^2}(\mathbb{1}^T(1 + (n-1)\rho)\mathbb{1})(\mathbb{1}^T(1 + (n-1)\rho)^{-1}\mathbb{1}) = 1.
\end{aligned}$$

Regardless of the value of ρ, the least squares estimate is as efficient as the MLE. For small values of ρ, the data is more similar to the homoskedastic case, resulting in increased validity of the hypothesis test. As ρ increases to 1, the validity and coverage drop.

Problem 2. Comparing constructions of heteroskedasticity-robust standard errors.

Suppose that

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \epsilon_i \overset{\text{ind}}{\sim} N(0, \sigma_i^2). \tag{5}$$

Two approaches to obtaining heteroskedasticity-robust standard errors are the pairs bootstrap and Huber-White standard errors. The goal of this problem is to compare the coverage and width of confidence intervals obtained from these two approaches.

(a) Write a function called `pairs_bootstrap`, which inputs arguments $\boldsymbol{X}$, $\boldsymbol{y}$, and B and outputs an estimated $p \times p$ covariance matrix $\widehat{\text{Var}}[\widehat{\boldsymbol{\beta}}]$ based on B resamples of the pairs bootstrap.

(b) Write a function called `huber_white`, which inputs arguments $\boldsymbol{X}$ and $\boldsymbol{y}$ and outputs an estimated $p \times p$ covariance matrix $\widehat{\text{Var}}[\widehat{\boldsymbol{\beta}}]$ based on the Huber-White formula.

(c) Generate $n = 50$ (x, y) pairs by setting x to be equally-spaced values between 0 and 1 and drawing $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$, where $\epsilon_i \overset{\text{ind}}{\sim} N(0, 9x_i^2)$, $\beta_0 = 2, \beta_1 = 3$. Create a scatter plot of these points, the least squares line, and three confidence bands: the standard least squares confidence band as well as those resulting from the pairs bootstrap (with $B = 500$) and the Huber-White formula. Comment on the relative widths of these three bands depending on the value of x.

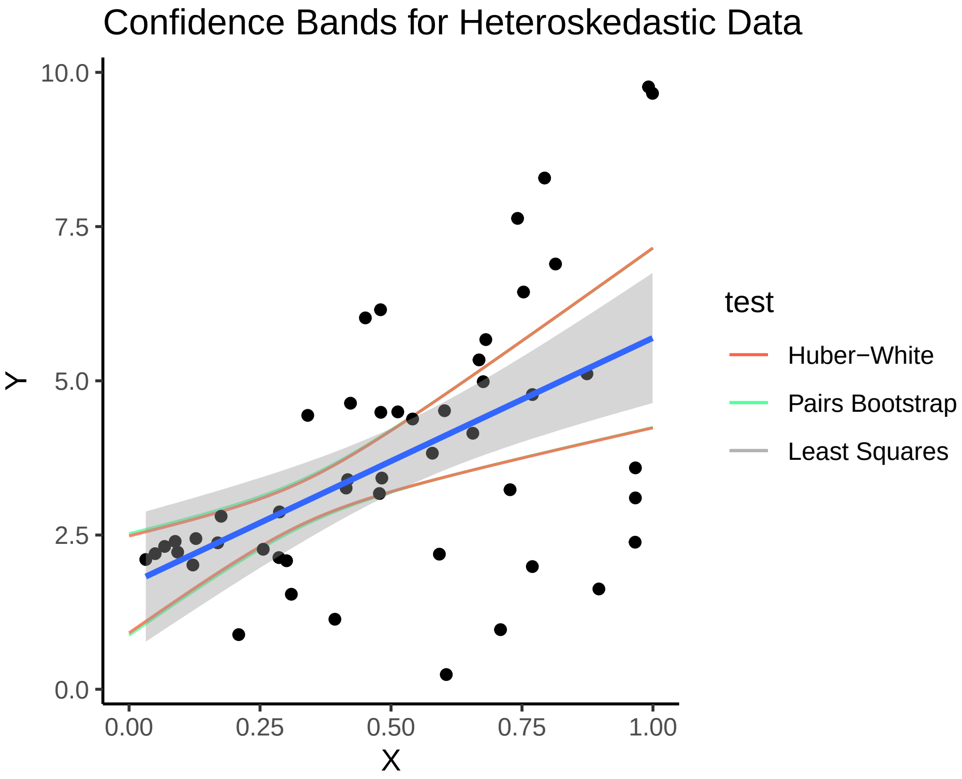
(d) Repeat the experiment from part (c) 100 times to compute the coverage and average width of the three confidence bands for each value of x. Plot these two metrics as a function of x, and comment on the results.

Solution 2.

1.
```
> library(tidyverse)
> pairs_bootstrap <- function(X,Y,B){
+   # Initialize matrix for sampled betas
+   betas <- matrix(ncol=ncol(X),nrow=B)
+   for(i in 1:B){
+     # Construct random sample
+     samples <- sample(nrow(X),nrow(X),replace = T)
+     X_boot <- X[samples,]
+     Y_boot <- Y[samples]
+
+     # Computed bootstrapped beta
+     betas[i,] <- coef(lm(Y_boot˜X_boot-1))
+   }
+   # Return empirical covariance matrix
+   return(cov(betas))
+ }
```

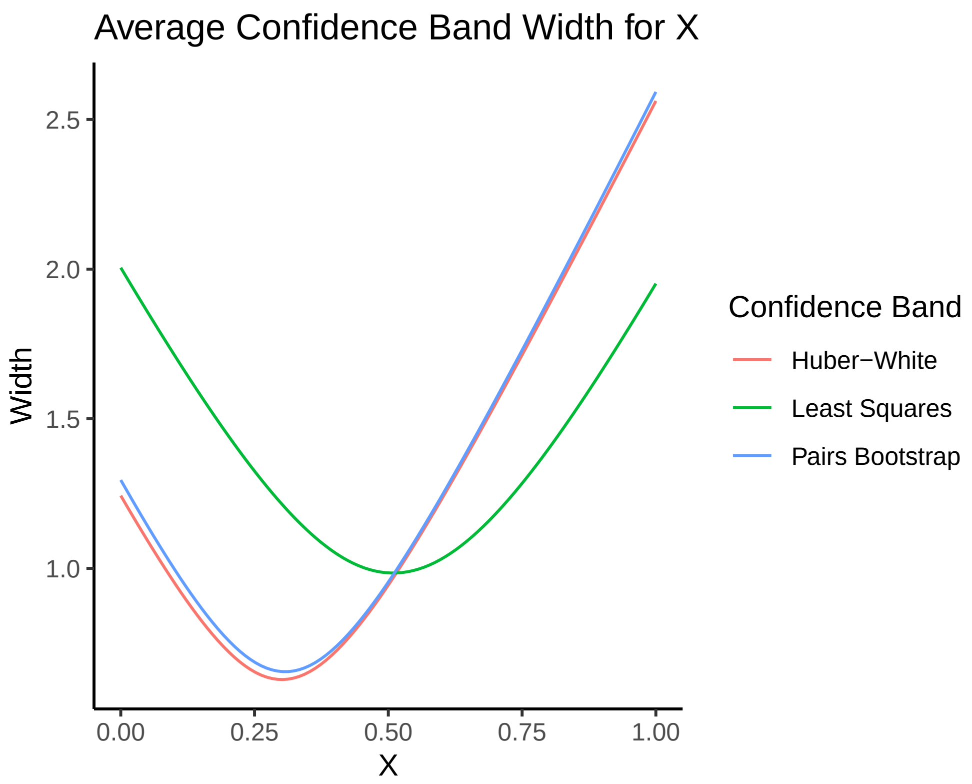
2.
```
> huber_white <- function(X,Y){
+   # Compute beta hat and Huber-White standard errors
+   hatbeta <- coef(lm(Y˜X-1))
+   hat_sigmasq <- as.numeric((Y-X%*%hatbeta)ˆ2)
+   # Compute helper variables
+   hat_sigma <- diag(hat_sigmasq,)
+   gram <- solve(t(X)%*%X)
+   # Output final variance estimate
+   return(gram %*% (t(X)%*% hat_sigma %*% X) %*% gram)
+ }
>
```

3.
```
> n <- 50
> betas <- as.matrix(c(2,3))
> # Data generation function
> generate_data <- function(n,betas){
+   # Construct X and Y matrices
+   X <- as.matrix(data.frame(x0=rep(1,n),x1=runif(n)))
+   Y <- as.numeric(X%*% (betas)) +
+     rnorm(n,mean=rep(0,n),sd=sqrt(X[,2]^2*9))
+   data <- data.frame(x0=rep(1,n),x1=X[,2],y=Y)
+   # Return all objects
+   return(list(X=X,Y=Y,data=data))
+ }
> # Generate Data
> all_vals <- generate_data(n,betas)
> X <- all_vals$X
> Y <- all_vals$Y
> data <- all_vals$data
> # Construct confidence bands for Pairs Bootstrap and
> # Huber-White
> compute_bands <- function(X,Y,B,pts){
+
+   # Compute beta hat
+   hatbeta <-  as.numeric(coef(lm(Y~X-1)))
+   # Compute covariance matrices
+   hw <- huber_white(X,Y)
+   pbs <- pairs_bootstrap(X,Y,B)
+
+   hw_band <- matrix(nrow=npoints,ncol=2)
+   pbs_band <- matrix(nrow=npoints,ncol=2)
+   quants <- qt(c(0.025,0.975),df=n-2)
+
+   # Compute confidence band per value of x
+   for(i in 1:npoints){
+     x_vec <- as.matrix(c(1,pts[i]))
+     # Confidence in terms of t distribution
+     hw_band[i,]  <- quants*sqrt(t(x_vec)%*% hw %*% x_vec)[1]+
+       (t(hatbeta)%*%x_vec)[1]
+     pbs_band[i,] <- quants*sqrt(t(x_vec)%*% pbs %*% x_vec)[1]+
+       (t(hatbeta)%*%x_vec)[1]
+   }
+
+   # Reformulate objects as data frames and rename
+   pbs_band <- data.frame(pbs_band)
+   hw_band <- data.frame(hw_band)
+   colnames(hw_band) <- c("lower","upper")
+   colnames(pbs_band) <- c("lower","upper")
+   hw_band$x <- pts
+   pbs_band$x <- pts
+   return(list(hw_band=hw_band,pbs_band=pbs_band))
+ }
> # Run compute_bands to obtain bands
> npoints <- 100
> B <- 500
> pts <- seq(from=0,to=1,length.out=npoints)
> bands <- compute_bands(X,Y,B,pts)
> hw_band <- bands$hw_band
> pbs_band <- bands$pbs_band
> # Define colors
> colors <- c(
+   "Huber-White" = "tomato",
+   "Pairs Bootstrap" = "seagreen1",
+   "Least Squares" = "gray70"
+ )
> # Plot points, confidence bands, linear model
> ggplot() +
+   geom_line(data=pbs_band,
+       aes(y=upper,x=x,color="Pairs Bootstrap"),alpha=0.8)+
+   geom_line(data=pbs_band,
+       aes(y=lower,x=x,color="Pairs Bootstrap"),alpha=0.8)+
+   geom_line(data=hw_band,
+       aes(y=upper,x=x,color="Huber-White"),alpha=0.8)+
+   geom_line(data=hw_band,
+       aes(y=lower,x=x,color="Huber-White"),alpha=0.8)+
+   geom_point(data=data,aes(x=x1,y=y))+
+   geom_smooth(data=data,
+       aes(x=x1,y=y),alpha=0.4,method=lm)+
+   labs(x="X",y="Y",fill="Confidence Band",
+        title = "Confidence Bands for Heteroskedastic Data")+
+   scale_color_manual(values=colors,name="test")+theme_classic()
>
```

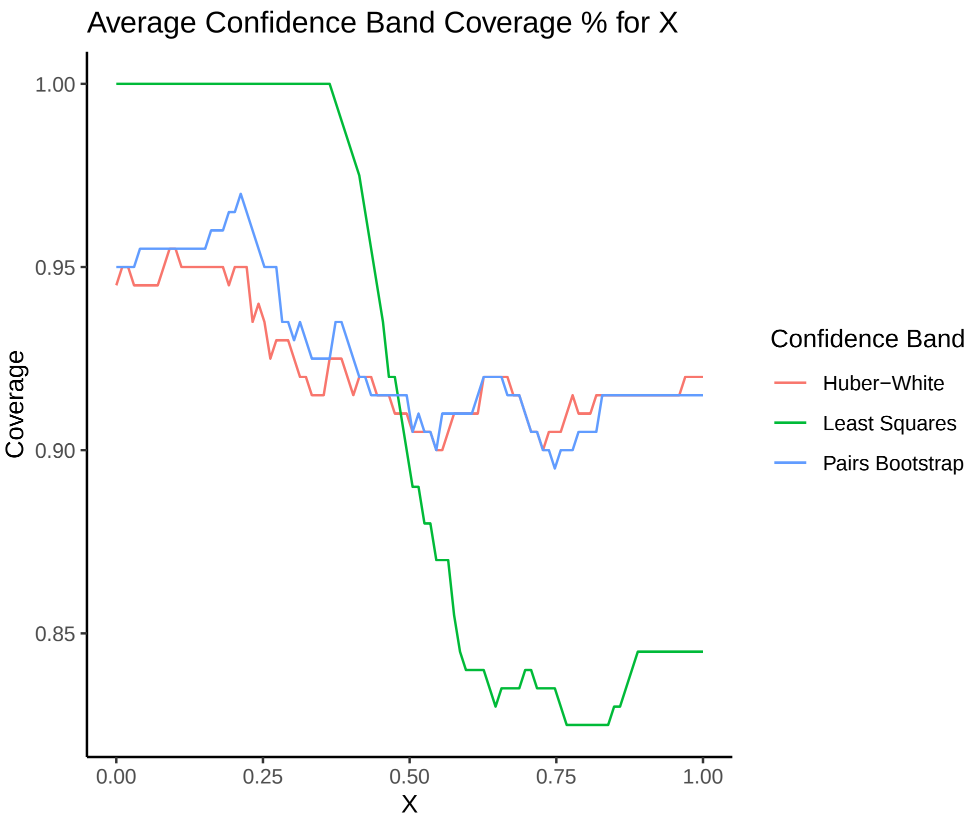


Comparing the relative widths, the Huber-White and pairs bootstraps are almost identical, whereas the least squares band is larger for smaller values of x and smaller for larger values of x. This makes sense as it overestimates the variance of y for x close to 0 and over estimates the variance of y for x close to 1.

4.
```
> # Initialize variables for storing
> npoints <- 100
> pts <- seq(from=0,to=1,length.out=npoints)
> lm_widths <- rep(0,npoints)
> hw_widths <- rep(0,npoints)
> pbs_widths <- rep(0,npoints)
> lm_coverage <- rep(0,npoints)
> hw_coverage <- rep(0,npoints)
> pbs_coverage <- rep(0,npoints)
> # Run simulation 200 times
> ntrials <- 200
> for(i in 1:ntrials){
+   # Generate Data
+   all_vals <- generate_data(n,betas)
+   X <- all_vals$X
+   Y <- all_vals$Y
+   data <- all_vals$data
+
+   # Least Squares Confidence Bands
+   fit <- lm(y~x0+x1-1,data=data)
+   new_x <- data.frame(x0=rep(1,npoints),x1=pts)
+   lm_ints <- data.frame(predict(fit,new_x,interval="confidence"))
+   lm_widths <- lm_widths + lm_ints$upr-lm_ints$lwr
+
+   # Confidence bands for HW and PBS
+   bands <- compute_bands(X,Y,B,pts)
+   hw_band <- bands$hw_band
+   pbs_band <- bands$pbs_band
+   hw_widths <- hw_widths + hw_band$upper-hw_band$lower
+   pbs_widths <- pbs_widths + pbs_band$upper-pbs_band$lower
+
+   # Compute Coverages
+   true_val <-  betas[1]+betas[2]*pts
+   lm_coverage <- lm_coverage +
+     as.numeric((true_val < lm_ints$upr) & (true_val > lm_ints$lwr))
+   hw_coverage <- hw_coverage +
+     as.numeric((true_val < hw_band$upper) & (true_val > hw_band$lower))
+   pbs_coverage <- pbs_coverage +
+     as.numeric((true_val < pbs_band$upper) & (true_val > pbs_band$lower))
+
+ }
> # Divide by number of trials to compute average
> lm_widths <- lm_widths/ntrials
> hw_widths <- hw_widths/ntrials
> pbs_widths <- pbs_widths/ntrials
> lm_coverage <- lm_coverage/ntrials
> hw_coverage <- hw_coverage/ntrials
> pbs_coverage <- pbs_coverage/ntrials
> # Store in data frame
> all_metrics <- data.frame(band=c(rep("Least Squares",npoints),
+                                  rep("Huber-White",npoints),
+                                  rep("Pairs Bootstrap",npoints)),
+            widths=c(lm_widths,hw_widths,pbs_widths),
+            coverage=c(lm_coverage,hw_coverage,pbs_coverage),
+            x=rep(pts,3)
+            )
> # Plot Average Width
> all_metrics%>% ggplot(aes(x=x,y=widths,color=band))+
+   geom_line()+
+   labs(x="X",y="Width",title="Average Confidence Band Width for X",
+        color="Confidence Band")+
+   theme_classic()
>
```



```
> # Plot Average Coverage
> all_metrics%>% ggplot(aes(x=x,y=coverage,color=band))+
+   geom_line()+
+   labs(x="X",y="Coverage",
+        title="Average Confidence Band Coverage % for X",
+        color="Confidence Band")+theme_classic()
```



We see that the pairs bootstrap and Huber-White perform very similarly, with the Huber-White method having a slightly more narrow confidence band. As before, we see that the standard least squares estimate is much wider than necessary for small values of x and not wide enough for large values.

In terms of coverage, as expected the standard least squares estimate covers β about 100% of the time for small values of x due to a superfluously large interval, while dropping to about 85% for large values of x. The pairs bootstrap and Huber-White methods both perform well close to 95% coverage, although they seem to decrease in coverage percentage as X increase, approaching about 93%.

Problem 3. Case study: Advertising data..

In this problem, we will analyze a data set related to advertising spending. It contains the sales of a product (in thousands of units) in 200 different markets, along with advertising budgets (in thousands of dollars) for the product in each of those markets for three different media: TV, radio, and newspaper. The goal is to learn about the relationship between these three advertising budgets (predictors) and sales (response).

```
> ad_data = read_tsv("../../data/Advertising.tsv")
> print(ad_data, n = 5)

# A tibble: 200 × 4
     TV radio newspaper sales
  <dbl> <dbl>     <dbl> <dbl>
1 230.   37.8      69.2  22.1
2  44.5  39.3      45.1  10.4
3  17.2  45.9      69.3   9.3
4 152.   41.3      58.5  18.5
5 181.   10.8      58.4  12.9
# ... with 195 more rows
```

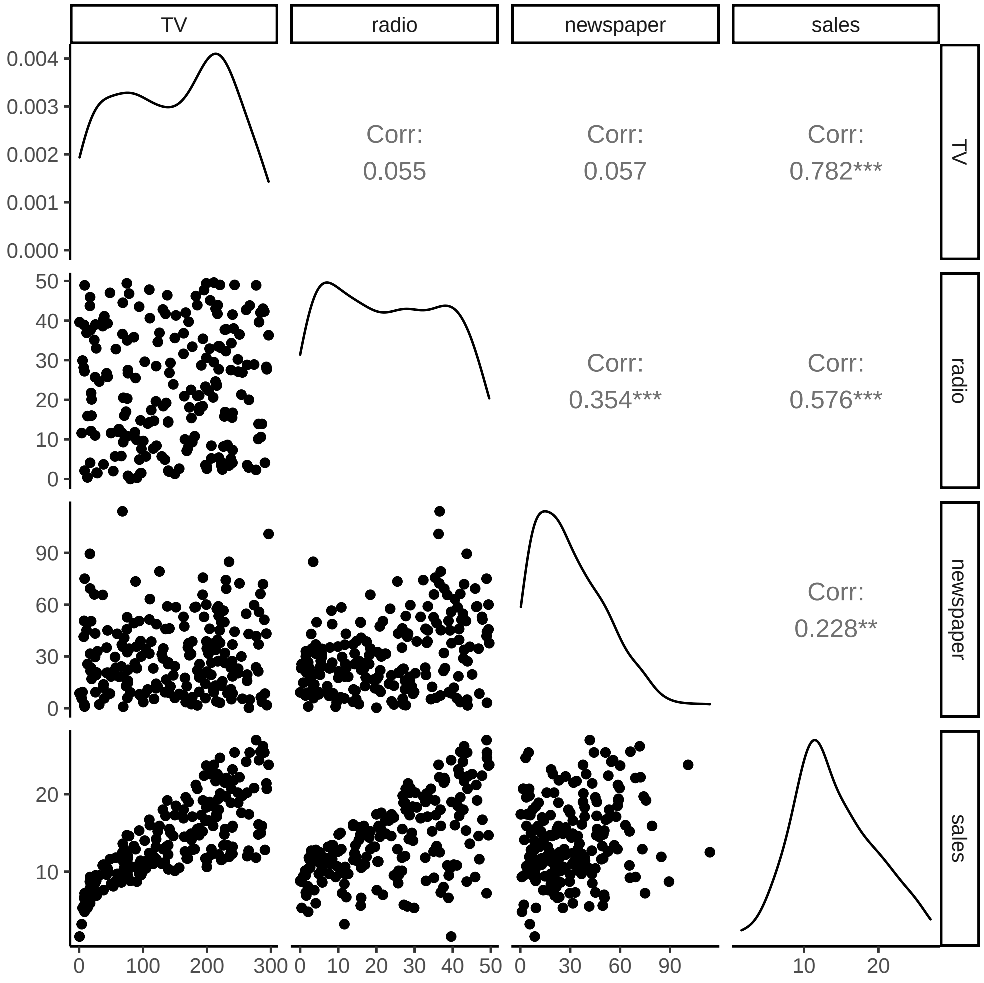
(a) Run a linear regression of `sales` on `TV`, `radio`, and `newspaper`, and produce a set of standard diagnostic plots. What model misspecification issue(s) appear to be present in these data?

(b) Address the above misspecification issues using one or more of the strategies discussed in Unit 3. Report a set of statistical estimates, confidence intervals, and test results you think you can trust.

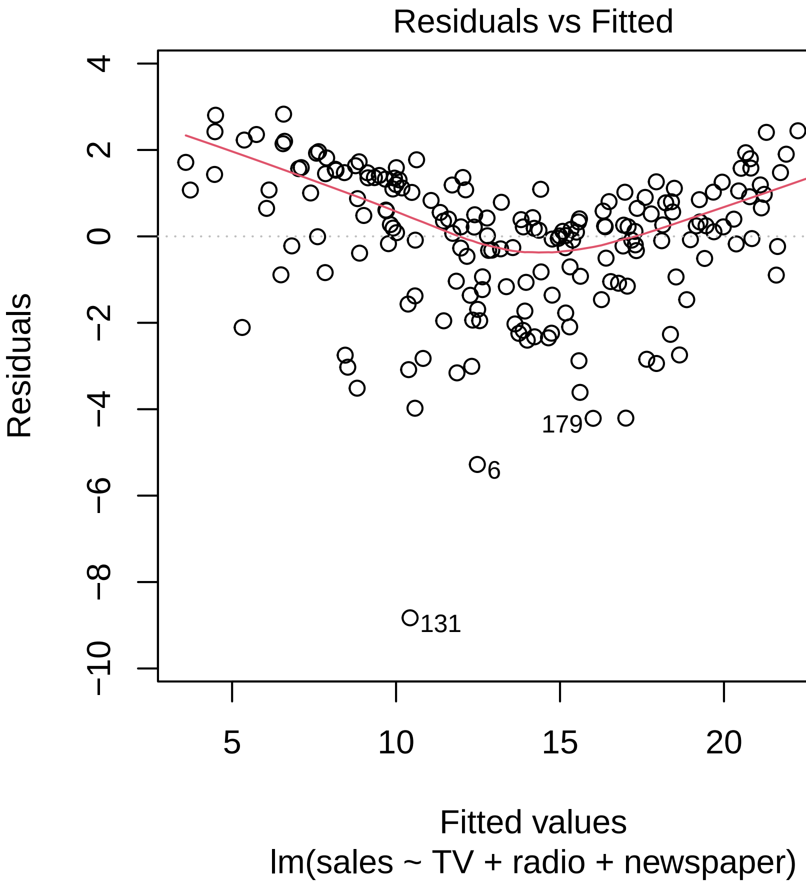
(c) Discuss the findings from part (b) in language that a policymaker could comprehend, including any caveats or limitations of the analysis.

Solution 3.

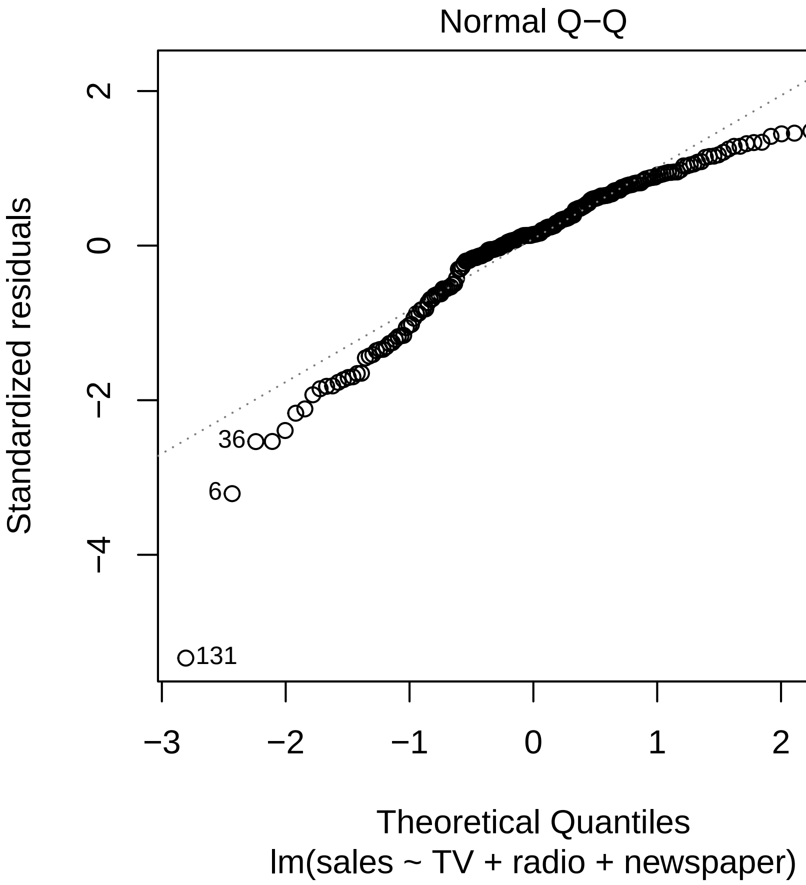
1.
```
> library(GGally)
> GGally::ggpairs(ad_data)+
+     theme_classic()
```



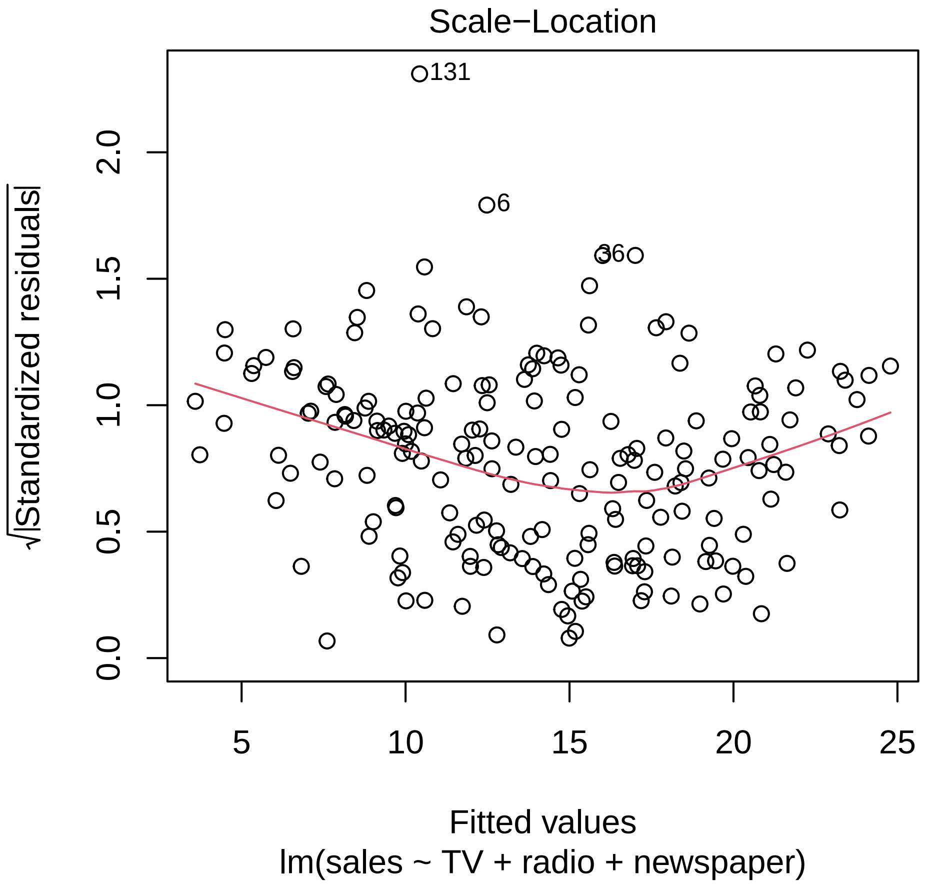
```
> fit <- lm(sales~TV+radio+newspaper,data=ad_data)
> plot(fit,which=1)
```



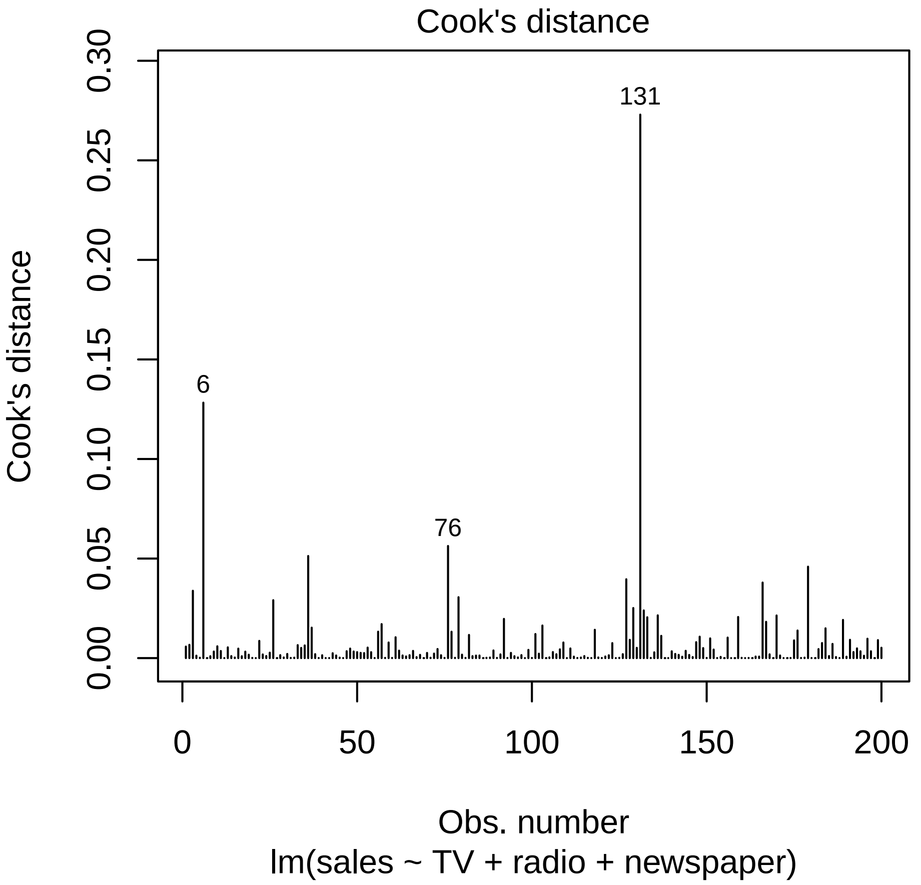
```
> plot(fit,which=2)
```



```
> plot(fit,which=3)
```



```
> plot(fit,which=4)
```



The data exhibits nonnormality of the residuals from the QQ plot, and seems to suggest the presence of outliers. Looking at the individual scatter plots, the variance of the data seems to be heteroskedastic.

2. I propose to remove the observation 131 as it is highly anomalous compared to the other observations. To account for heteroskedasticity, I propose to use the Huber-White estimator.

```
> library(lmtest)
> ad_data_subset <- ad_data[-131,]
> fit_subset <- lm(sales~TV+radio+newspaper,data=ad_data_subset)
> coeftest(fit_subset,voc.=vovHC)

t test of coefficients:

              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.0930655  0.2903323 10.6535   <2e-16 ***
TV           0.0448454  0.0013027 34.4252   <2e-16 ***
radio        0.1939046  0.0080358 24.1301   <2e-16 ***
newspaper   -0.0042519  0.0054702 -0.7773   0.4379
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> coefci(fit_subset,voc.=vovHC)

                  2.5 %      97.5 %
(Intercept)  2.52047095 3.665659979
TV           0.04227623 0.047414582
radio        0.17805638 0.209752833
newspaper   -0.01504017 0.006536367
```

3. Our analysis suggests that radio and television ads have a statistical significant relationship with sales. Our linear model suggests radio ads increase 200 units of sales per thousand dollars of investment, while radio ads increase 44 units of sales per thousand dollars of investment. Therefore our model suggests to invest in radio ads, and then television ads and not newspaper ads.
   This analysis is limited by the fact it only considers a linear relationship between the covariates and response. It may be that there is a diminishing effect in ad spending, as consumers become desensitized or even annoyed at constant bombardment of ads. Furthermore, there may be some form of interaction between advertisements, e.g. if a consumer is exposed to more forms of advertisement, then the consumer is more likely to be affected and adjust their spending habits. This could be remedied with introducing interaction terms or more complex models.