Homework 3

Sam Rosenberg

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:
James Blume
Please list what external references you consulted (e.g. articles, books, or websites):

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

(d) (Special case: Correlated errors) Suppose $(\epsilon_1, \ldots, \epsilon_n)$ 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) We know that the traditional least squares estimate for β_0 under the intercept-only model is $\widehat{\beta}_0^{\text{LS}} = \overline{y} = (\mathbf{1}^T\mathbf{1})^{-1}\mathbf{1}^T\boldsymbol{y}$.

Then

$$\begin{aligned}\text{Var}[\widehat{\beta}_0^{\text{LS}}] &= \text{Var}[(\mathbf{1}^T\mathbf{1})^{-1}\mathbf{1}^T\boldsymbol{y}] \\ &= (\mathbf{1}^T\mathbf{1})^{-1}\mathbf{1}^T\text{Var}[\boldsymbol{y}][(\mathbf{1}^T\mathbf{1})^{-1}\mathbf{1}^T]^T \\ &= \frac{1}{n^2}\mathbf{1}^T\text{Var}[\boldsymbol{y}]\mathbf{1} \\ &= \frac{1}{n^2}\sum_{i,j=1}^{n}(\boldsymbol{\Sigma})_{ij}.\end{aligned}$$

Also, we know that the traditional variance estimate is $\widehat{\text{Var}}[\widehat{\beta}_0^{\text{LS}}] = \frac{\widehat{\sigma}^2}{n} = \frac{||\widehat{\boldsymbol{\epsilon}}||^2}{n(n-p)}$. Under (1), we have that

$$\begin{aligned}\mathbb{E}[\widehat{\text{Var}}[\widehat{\beta}_0^{\text{LS}}]] &= \mathbb{E}[\frac{||\widehat{\boldsymbol{\epsilon}}||^2}{n(n-p)}] \\ &= \frac{1}{n(n-p)}\mathbb{E}[\widehat{\boldsymbol{\epsilon}}^T\widehat{\boldsymbol{\epsilon}}] \\ &= \frac{1}{n(n-p)}\mathbb{E}[\text{Tr}(\widehat{\boldsymbol{\epsilon}}^T\widehat{\boldsymbol{\epsilon}})] \\ &= \frac{1}{n(n-p)}\mathbb{E}[\text{Tr}(\widehat{\boldsymbol{\epsilon}}\widehat{\boldsymbol{\epsilon}}^T)] \\ &= \frac{1}{n(n-p)}\mathbb{E}[\sum_{i=1}^{n}\widehat{\epsilon}_i^2] \\ &= \frac{1}{n(n-p)}\sum_{i=1}^{n}\text{Var}[\widehat{\epsilon}_i] \\ &= \frac{1}{n(n-p)}\sum_{i=1}^{n}\sigma_i^2.\end{aligned}$$

We then have that

$$\begin{aligned}\tau_1 &= \frac{\mathbb{E}[\widehat{\text{Var}}[\widehat{\beta}_0^{\text{LS}}]]}{\text{Var}[\widehat{\beta}_0^{\text{LS}}]} \\ &= \frac{n^{-1}(n-p)^{-1}\sum_{i=1}^{n}\sigma_i^2}{n^{-2}\sum_{i,j=1}^{n}(\boldsymbol{\Sigma})_{ij}} \\ &= (\frac{n}{n-p})\frac{\sum_{i=1}^{n}\sigma_i^2}{\sum_{i,j=1}^{n}(\boldsymbol{\Sigma})_{ij}}\end{aligned}$$

When $\tau_1 > 1$, we have that $\mathbb{E}[\widehat{\text{Var}}[\widehat{\beta}_0^{\text{LS}}]] > \text{Var}[\widehat{\beta}_0^{\text{LS}}]$. Then the expected estimated variance of the OLS estimator exceeds that of the true variance. Consequently, we would have that both the Type-I error of the hypothesis test $H_0 : \beta_0 = 0$ and the coverage for the confidence interval for β_0 increase when $\tau > 1$, relative to the case in which we have homoskedasticity (likewise these quantities decrease for $\tau < 1$).

(b) Recall that the general likelihood function for a multivariate normal with mean $\boldsymbol{\beta}$ and covariance $\boldsymbol{\Sigma}$ is

$$L(\boldsymbol{y}) = \frac{1}{\sqrt{(2\pi)^n|\boldsymbol{\Sigma}|}}\exp[-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})].$$

Since $y \sim N(\beta_0, \mathbf{\Sigma})$, the likelihood function for the intercept-only linear model with generalized covariance is

$$L(\boldsymbol{y}) = (2\pi)^{-n/2}|\mathbf{\Sigma}|^{-1/2}\exp[-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{X\beta})^T\mathbf{\Sigma}^{-1}(\boldsymbol{y}-\boldsymbol{X\beta})]$$

and the log-likelihood is

$$-\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}| - \frac{1}{2}[\boldsymbol{y}-\beta_0\mathbf{1})^T\mathbf{\Sigma}^{-1}(\boldsymbol{y}-\beta_0\mathbf{1})].$$

So,

$$\begin{aligned}\frac{\partial\ell}{\partial\beta_0}(\boldsymbol{y}) &= -\frac{1}{2}\frac{\partial}{\partial\beta_0}[(\boldsymbol{y}-\beta_0\mathbf{1})^T\mathbf{\Sigma}^{-1}(\boldsymbol{y}-\beta_0\mathbf{1})] \\ &= -\beta_0(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1}) + \mathbf{1}^T\mathbf{\Sigma}^{-1}\boldsymbol{y}.\end{aligned}$$

Then $\frac{\partial\ell}{\partial\beta_0}(\boldsymbol{y}) = 0$ if and only if $\beta_0 = (\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}\boldsymbol{y}$, so

$$\widehat{\beta}_0^{\text{MLE}} = (\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}\boldsymbol{y}.$$

So,

$$\begin{aligned}\text{Var}[\widehat{\beta}_0^{\text{MLE}}] &= \text{Var}[(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}\boldsymbol{y}] \\ &= [(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}]\text{Var}[\boldsymbol{y}][(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}]^T \\ &= [(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}]\mathbf{\Sigma}[(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\mathbf{\Sigma}^{-1}]^T \\ &= (\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}\mathbf{1}^T[\mathbf{1}^T\mathbf{\Sigma}^{-1}]^T[(\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1}]^T \\ &= (\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-2}\mathbf{1}^T(\mathbf{\Sigma}^{-1})^T\mathbf{1} \\ &= (\mathbf{1}^T\mathbf{\Sigma}^{-1}\mathbf{1})^{-1} \\ &= (\sum_{i,j=1}^{n}(\mathbf{\Sigma}^{-1})_{ij})^{-1}.\end{aligned}$$

We then have that

$$\tau_2 = \frac{\text{Var}[\widehat{\beta}_0^{\text{LS}}]}{\text{Var}[\widehat{\beta}_0^{\text{MLE}}]} = \frac{1}{n^2}(\sum_{i,j=1}^{n}(\mathbf{\Sigma}^{-1})_{ij})(\sum_{i,j=1}^{n}(\mathbf{\Sigma})_{ij}).$$

When $\tau_2 > 1$, we have that $\text{Var}[\widehat{\beta}_0^{\text{LS}}] > \text{Var}[\widehat{\beta}_0^{\text{MLE}}]$. This means that we will have lower variance as compared to the homoskedastic case and thus have higher power. Additionally, the lower variance means that the CI widths will also be narrower as compared to the homoskedastic case.

(c) In the case of heteroskedasticity, we have that that since $\mathbf{\Sigma} = \text{diag}(\sigma_1^2, \ldots, \sigma_n^2)$, $\widehat{\mathbf{\Sigma}} = \text{diag}(\widehat{\sigma}_1^2, \ldots, \widehat{\sigma}_n^2)$, where $\widehat{\sigma}_i^2 = (y_i - \widehat{\beta}_0)^2$, and $\mathbf{\Sigma}^{-1} = \text{diag}(\sigma_1^{-2}, \ldots, \sigma_n^{-2})$.

So,

$$\tau_1 = (\frac{n}{n-p})\frac{\sum_{i=1}^n\sigma_i^2}{\sum_{i=1}^n\sigma_i^2} = \frac{n}{n-p}$$

Also,

$$\tau_2 = \frac{1}{n^2}(\sum_{i=1}^{n}\sigma_i^{-2})(\sum_{i=1}^{n}\sigma_i^2).$$

Note that because $1 \leq p < n$, we have that $n - p < n$, so $\tau_1 > 1$. That means we will have a higher Type-I error as compared to the case in which we have homoskedasticity.

When $(\sum_{i=1}^n\sigma_i^{-2})(\sum_{i=1}^n\sigma_i^2) > n^2$, $\tau_2 > 1$ and we have that power increases relative to the homoskedastic case.

(d) We have

$$\tau_1 = (\frac{n}{n-p})\Big(\frac{\sum_{i=1}^n \sigma_i^2}{\sum_{i,j=1}^n (\mathbf{\Sigma})_{ij}}\Big) = (\frac{n}{n-\rho})(1+\rho(n-1))^{-1}.$$

Note that we can write $\mathbf{\Sigma} = (1-\rho)\boldsymbol{I}_n + \rho\mathbf{1}\mathbf{1}^T$. Then

$$\mathbf{\Sigma}^{-1} = [(1-\rho)(\boldsymbol{I}_n + \frac{\rho}{1-\rho}\mathbf{1}\mathbf{1}^T)]^{-1} = (1-\rho)^{-1}[\boldsymbol{I}_n + (\frac{\rho}{1-\rho}\mathbf{1}\mathbf{1}^T)]^{-1}.$$

Using the Sherman Morrison formula we can simplify find a more explicit solution:

$$\begin{aligned}
\mathbf{\Sigma}^{-1} &= (1-\rho)^{-1}[\boldsymbol{I}_n + (\frac{\rho}{1-\rho}\mathbf{1}\mathbf{1}^T)]^{-1} \\
&= (1-\rho)^{-1}\Big[\boldsymbol{I}_n - \frac{\boldsymbol{I}_n[(\frac{\rho}{1-\rho})\mathbf{1}\mathbf{1}^T]\boldsymbol{I}_n}{1+\mathbf{1}^T\boldsymbol{I}_n(\frac{\rho}{1-\rho})\mathbf{1}}\Big] \\
&= (1-\rho)^{-1}\Big[\boldsymbol{I}_n - \frac{(\frac{\rho}{1-\rho})\mathbf{1}\mathbf{1}^T}{1+(\frac{\rho}{1-\rho})\mathbf{1}^T\mathbf{1}}\Big] \\
&= (1-\rho)^{-1}\Big[\boldsymbol{I}_n - \frac{(\frac{\rho}{1-\rho})\mathbf{1}\mathbf{1}^T}{1+(\frac{\rho}{1-\rho})n}\Big] \\
&= (1-\rho)^{-1}\Big[\boldsymbol{I}_n - \frac{\rho}{1+\rho(n-1)}\mathbf{1}\mathbf{1}^T\Big].
\end{aligned}$$

So,

$$\begin{aligned}
\sum_{i,j=1}^n (\mathbf{\Sigma}^{-1})_{ij} &= (1-\rho)^{-1}\Big[n - \frac{n^2\rho}{1+\rho(n-1)}\Big] \\
&= (1-\rho)^{-1}\Big[\frac{n+n^2\rho-n\rho-n^2\rho}{1+\rho(n-1)}\Big] \\
&= (1-\rho)^{-1}\Big[\frac{n(1-\rho)}{1+\rho(n-1)}\Big] \\
&= \frac{n}{1+\rho(n-1)}.
\end{aligned}$$

Then

$$\begin{aligned}
\tau_2 &= \frac{1}{n^2}(\sum_{i,j=1}^n (\mathbf{\Sigma}^{-1})_{ij})(\sum_{i,j=1}^n (\mathbf{\Sigma})_{ij}) \\
&= \frac{1}{n^2}\Big[\frac{n}{1+\rho(n-1)}\Big][n+n(n-1)\rho] \\
&= 1.
\end{aligned}$$

As before, we have $\frac{n}{n-p} > 1$. Note that $(1+\rho(n-1))^{-1} < 1$, but tends toward 0 as $n \to \infty$. Since τ_1 is simply $\frac{n}{n-p}$ scaled by this quantity, this means that the Type-I error rate decreases relative to the homoskedastic case (and more generally, toward 0) as sample size increases.

Because $\tau_2 = 1$, we have that the power is the same relative to the homoskedastic case.

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.

(a)
```r
pairs_bootstrap <- function(X, y, B){
  # Vector to hold estimates of beta
  beta_b <-
    data.frame(
      matrix(
        NA,
        nrow=length(y),
        ncol=ncol(X)+1))

  # Run bootstrap B times
  for(i in 1:B){
    # Get length(y) rows sampled uniformly
    sample_ind <- sample(x = 1:length(y), size = length(y), replace = TRUE)
    X_b <- data.frame(X[sample_ind, ])
    y_b <- y[sample_ind]

    # Combine sample data into new dataframe
    df_b <- X_b
    df_b[, "y_b"] <- y_b

    # Run lm and add resulting estimates to beta_b dataframe
    lm_b <- lm(y_b ~ ., data = df_b)
```

```
    # Store coefficients
    beta_b[i, ] <- lm_b$coefficients
  }

  # Get estimated covariance matrix
  var_hat_beta_hat <- cov(beta_b)

  return(var_hat_beta_hat)
}
```

(b)
```
huber_white <- function(X, y){
  # Combine X, y
  df_comb <- X
  df_comb$y <- y

  # Get linear model, extract residuals
  lm_hw <- lm(y ~ ., data = df_comb)

  epsilon_hat <- lm_hw$residuals

  # Augment X to add intercept column
  X_aug <- model.matrix(lm_hw)

  # Sandwich estimator is  X^T sigma_hat X
  sigma_hat <- diag(epsilon_hat^2)
  sandwich <- t(X_aug) %*% sigma_hat %*% X_aug
  bread <- solve(t(X_aug) %*% X_aug)

  var_hat_beta_hat <- bread %*% sandwich %*% bread

  return(var_hat_beta_hat)
}
```

(c)
```
# Function for computing confidence intervals
get_ci <- function(x_i, beta_hat, sigma_hat, quantile){
  # Get fitted values
  y_hat <- t(x_i) %*% beta_hat


  # Get SEs
  se <- sqrt(t(x_i) %*% sigma_hat %*% x_i)

  # Compute CI bounds
  ci <- data.frame(cb_lower = y_hat - quantile * se,
                   cb_upper = y_hat + quantile * se)
```

```r
  return(ci)
}

# Function for computing confidence bounds
get_cb <- function(data, beta_hat, sigma_hat, quantile){
  # Get CI bounds for each x_i in data
  ci <- apply(data, MARGIN=1,
              FUN=function(x){get_ci(x, beta_hat, sigma_hat, quantile)})

  # Combine list of bounds into dataframe
  cb <- bind_rows(ci, .id="column_label") %>% select(-column_label)

  return(cb)
}



# Simulation parameters
#  - n: Number of data points
#  - beta_0: Intercept
#  - beta_1: Slope
#  - B: Number of times bootstrap run
#  - quantile: For constructing CIs
n <- 50
beta_0 <- 2
beta_1 <- 3

B <- 500
quantile <- 2


# Sample (x,y)
x <- runif(n = n, min = 0, max = 1)
epsilon_i <- rnorm(n = n, mean = 0, sd = 9 * x)
y <- beta_0 + beta_1 * x + epsilon_i

# Combine simulation data together
df_sim <- data.frame(y=y, x=x)

# Run lm, extract coefficients
lm_sim <- lm(y ~ ., df_sim)
coeff_sim <- lm_sim$coefficients

# Get model matrix
model_mx <- model.matrix(lm_sim$model)


# Get pairs bootstrap SEs
```

```
pairs_sigma_hat <-
  pairs_bootstrap(X=data.frame(x = df_sim$x), y=df_sim$y, B=B)

# Get pairs bootstrap CBs
pairs_cb <-
  get_cb(model_mx, coeff_sim, pairs_sigma_hat, quantile)
colnames(pairs_cb) <- paste0("pairs_", colnames(pairs_cb))

# Get HW SEs
hw_sigma_hat <-
  huber_white(X=data.frame(x = df_sim$x), y=df_sim$y)

# Get HW CBs
hw_cb <-
  get_cb(model_mx, coeff_sim, hw_sigma_hat, quantile)
colnames(hw_cb) <- paste0("hw_", colnames(hw_cb))


# Combine CBs with data
df_sim1 <-
  cbind(df_sim, pairs_cb, hw_cb)

# Make plot
sim_plt <-
  df_sim1 %>%
    ggplot(aes(x=x, y=y, color=NA, fill=NA), alpha=0.2) +
    # Scatter plot points
    geom_point(color="black") +
    # LS line and confidence band
    stat_smooth(aes(color="OLS Line", fill="OLS confidence band"),
                method='lm', formula=y~x, level=0.95) +
  # Pairs bootstrap confidence band
    geom_ribbon(aes(ymin=pairs_cb_lower, ymax=pairs_cb_upper,
                    color="Boostrap confidence band", fill=NA), alpha=0.2) +
  # HW SEs confidence band
    geom_ribbon(aes(ymin=hw_cb_lower, ymax=hw_cb_upper,
                    color="HW confidence band", fill=NA), alpha=0.2) +
    # Labels
    xlab("x") +
    ylab("y") +
    # Manual scales
    scale_color_manual(
      breaks = c("OLS Line",
                 "Boostrap confidence band",
                 "HW confidence band"),
      values = c("black",
                 "green",
```

```
                  "blue")) +
    # Fix legend
    guides(
      color=guide_legend(override.aes=list(fill=NA), order=1, title=NULL),
      fill=guide_legend(override.aes=list(color=NA), order=2, title=NULL)
    )

ggsave(plot = sim_plt, filename = "./figures/sim_plt.png",
       device = "png", width = 9, height = 6)
```

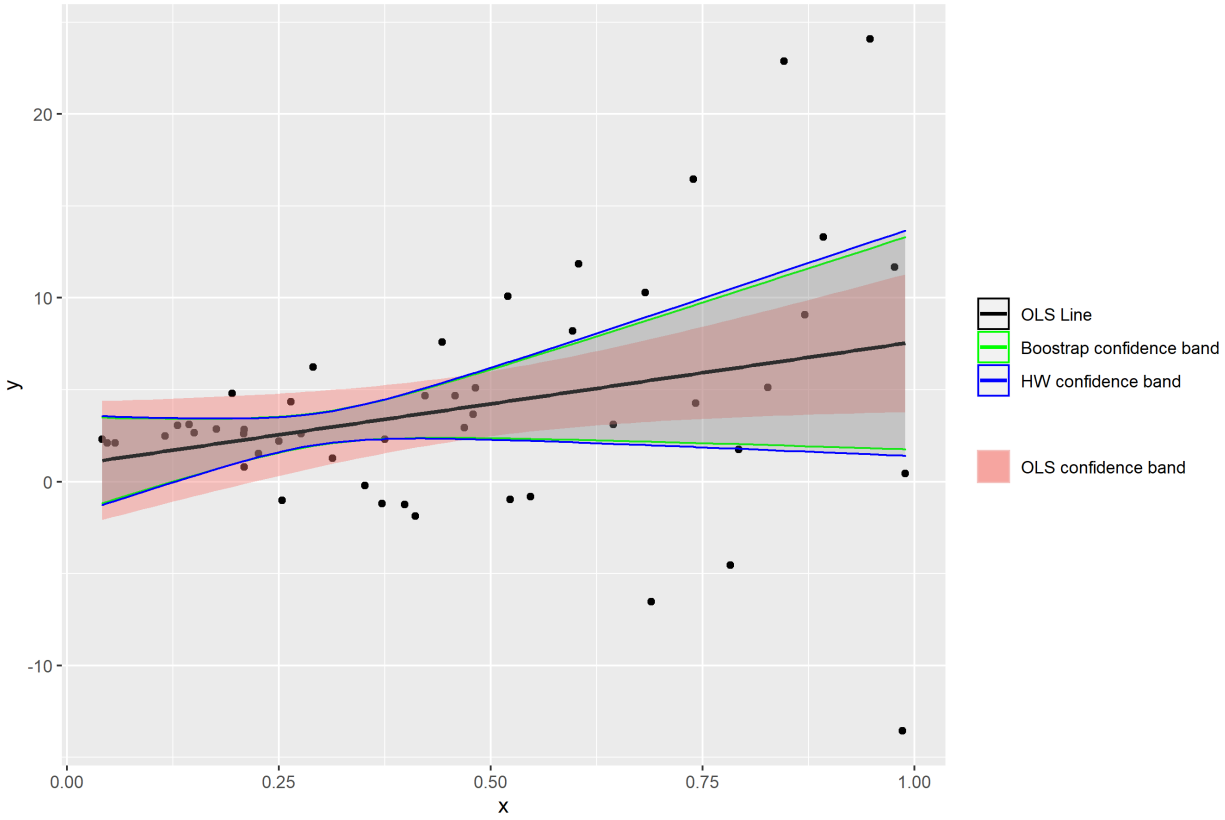

Figure 1: Simulated comparison of confidence bands for robust SE estimates.

We see that the bootstrap and Huber-White confidence bands have similar widths and are narrower than the OLS confidence band up to about $x = 0.6$ and are wider for $x > 0.6$. The width of the OLS confidence band decreases until about $x = 0.65$ and increases thereafter, while the robust methods have bands that decrease in width up to $x = 0.4$ and increase after.

(d)
```
# Function for getting estimated covariance with OLS
ols <- function(X, y){
  # Combine X, y
  df_comb <- X
  df_comb$y <- y
```

```r
  # Get linear model, extract residuals
  lm_hw <- lm(y ~ ., data = df_comb)

  epsilon_hat <- lm_hw$residuals

  # OLS variance estimator is ||epsilon_hat||^2/(n-p)
  sigma_hat_2 <- sum(epsilon_hat^2)/(length(y) - ncol(X))

  # Augment X to add intercept column
  X_aug <- model.matrix(lm_hw)

  # Sandwich estimator
  var_hat_beta_hat <- sigma_hat_2 * solve(t(X_aug) %*% X_aug)

  return(var_hat_beta_hat)
}


# Number of times simulation is run
n_sim <- 100

# Sample x
x <- runif(n = n, min = 0, max = 1)


# Function to run simulation
do_sim <- function(beta_0, beta_1, n, B, quantile){
  # Sample y
  epsilon_i <- rnorm(n = n, mean = 0, sd = 9 * x)
  y <- beta_0 + beta_1 * x + epsilon_i

  # Combine simulation data together
  df_sim <- data.frame(y=y, x=x)

  # Run lm, extract coefficients
  lm_sim <- lm(y ~ ., df_sim)
  coeff_sim <- lm_sim$coefficients

  # Get model matrix
  model_mx <- model.matrix(lm_sim$model)


  # Get OLS SEs
  ols_sigma_hat <-
    ols(X=data.frame(x = df_sim$x), y=df_sim$y)

  # Get OLS CBs
```

```r
ols_cb <-
  get_cb(model_mx, coeff_sim, ols_sigma_hat, quantile)
colnames(ols_cb) <- paste0("ols_", colnames(ols_cb))

# Get pairs bootstrap SEs
pairs_sigma_hat <-
  pairs_bootstrap(X=data.frame(x = df_sim$x), y=df_sim$y, B=B)

# Get pairs bootstrap CBs
pairs_cb <-
  get_cb(model_mx, coeff_sim, pairs_sigma_hat, quantile)
colnames(pairs_cb) <- paste0("pairs_", colnames(pairs_cb))

# Get HW SEs
hw_sigma_hat <-
  huber_white(X=data.frame(x = df_sim$x), y=df_sim$y)

# Get HW CBs
hw_cb <-
  get_cb(model_mx, coeff_sim, hw_sigma_hat, quantile)
colnames(hw_cb) <- paste0("hw_", colnames(hw_cb))


# Combine CBs with data
df_sim1 <-
  cbind(df_sim, ols_cb, pairs_cb, hw_cb)

# Get widths, coverage for each CB
cb_results <-
  df_sim1 %>%
    # Column for computing CB widths
    mutate(
      ols_width = ols_cb_upper-ols_cb_lower,
      ols_covered =
         (y >= ols_cb_lower &
          y <= ols_cb_upper),
      pairs_width = pairs_cb_upper-pairs_cb_lower,
      pairs_covered =
         (y >= pairs_cb_lower &
          y <= pairs_cb_upper),
       hw_width = hw_cb_upper-hw_cb_lower,
       hw_covered =
         (y >= hw_cb_lower &
          y <= hw_cb_upper))


# Get widths in long format
```

```r
  cb_widths <-
    cb_results %>%
      select(x, ols_width, pairs_width, hw_width) %>%
      # Reshape to long format
      pivot_longer(c(ols_width, pairs_width, hw_width),
                   names_to = "method",
                   values_to = "width") %>%
      # Subset of strings in method column
      mutate(method = str_replace(method, "_.*", ""))

  # Get coverage in long format
  cb_covered <-
    cb_results %>%
      select(x, ols_covered, pairs_covered, hw_covered) %>%
      # Reshape to long format
      pivot_longer(c(ols_covered, pairs_covered, hw_covered),
                   names_to = "method",
                   values_to = "covered") %>%
      # Subset of strings in method column
      mutate(method = str_replace(method, "_.*", ""))

  # Combine long formats
  cb_results1 <-
    cb_widths %>%
      inner_join(cb_covered, by=c("x", "method"))

  return(cb_results1)
}


# Get CB average widths and coverage for each simulation run
cb_list <- lapply(1:n_sim,
                  function(x){
                    do_sim(beta_0, beta_1, n, B, quantile)
                  })
# Combine list of simulation results together
cb_df <- do.call(rbind, cb_list)

ci_results <-
  cb_df %>%
    # Group by x, method
    group_by(x, method) %>%
    # Compute average width and coverage for each x, method
    mutate(avg_width = mean(width),
           coverage = mean(covered)) %>%
    # Ungroup
    ungroup() %>%
```

```
    # Drop width, covered columnd
    select(-width, -covered) %>%
    # Take only distinct rows
    distinct() %>%
    # Replace method with full method names
    mutate(
      method =
        str_replace(
          str_replace(
            str_replace(method,
            "pairs", "Pairs bootstrap"),
          "hw", "Huber-White"),
        "ols", "OLS")) %>%
    rename(Method = method)

# plot for average width by method
cb_width_plt <-
  ci_results %>%
    ggplot(aes(x=x, y=avg_width, color=Method)) +
      geom_point() +
      ylab("Average CB width")

# Plot for coverage by method
cb_coverage_plt <-
  ci_results %>%
    ggplot(aes(x=x, y=coverage, color=Method)) +
      geom_point() +
      ylab("CB coverage")

ggsave(plot = cb_width_plt, filename = "./figures/cb_width_plt.png",
       device = "png", width = 6, height = 4)

ggsave(plot = cb_coverage_plt, filename = "./figures/cb_coverage_plt.png",
       device = "png", width = 6, height = 4)
```

Here we have that for the pairs bootstrap and Huber-White robust estimators, the average confidence bound width decreases as x approaches 0.3 and then begins to increase again. A similar pattern occurs with the OLS average confidence bound widths with the switch from decreasing to increasing occurs around $x = 0.5$.

We have that the coverage of the OLS is greater than that of the robust methods up to $x = 0.5$, but the robust methods have generally better coverage for $x > 0.5$.

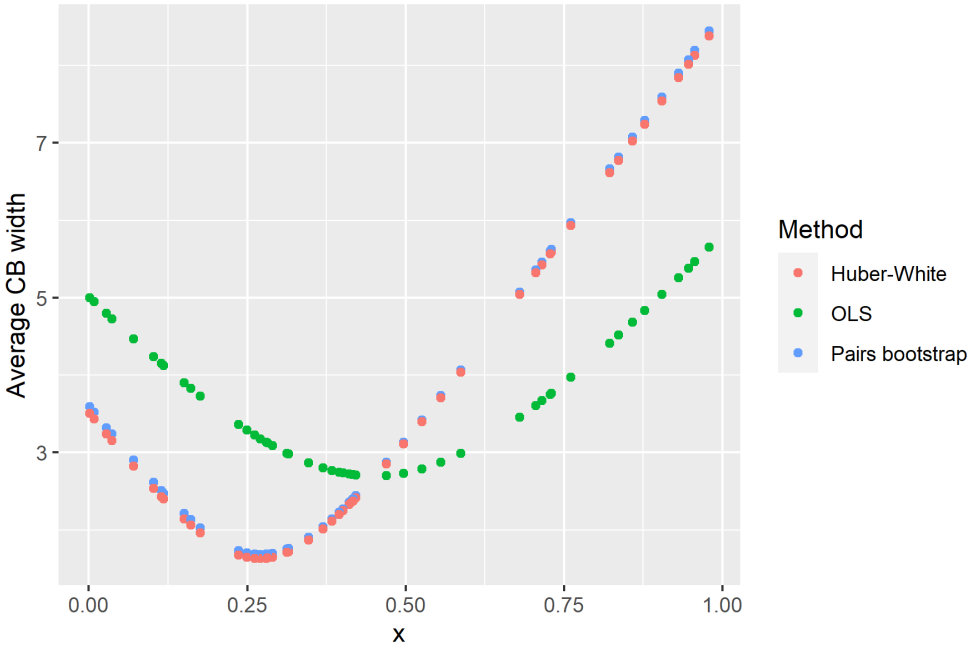

Figure 2: Average CB width as a function of x ($n = 100$).

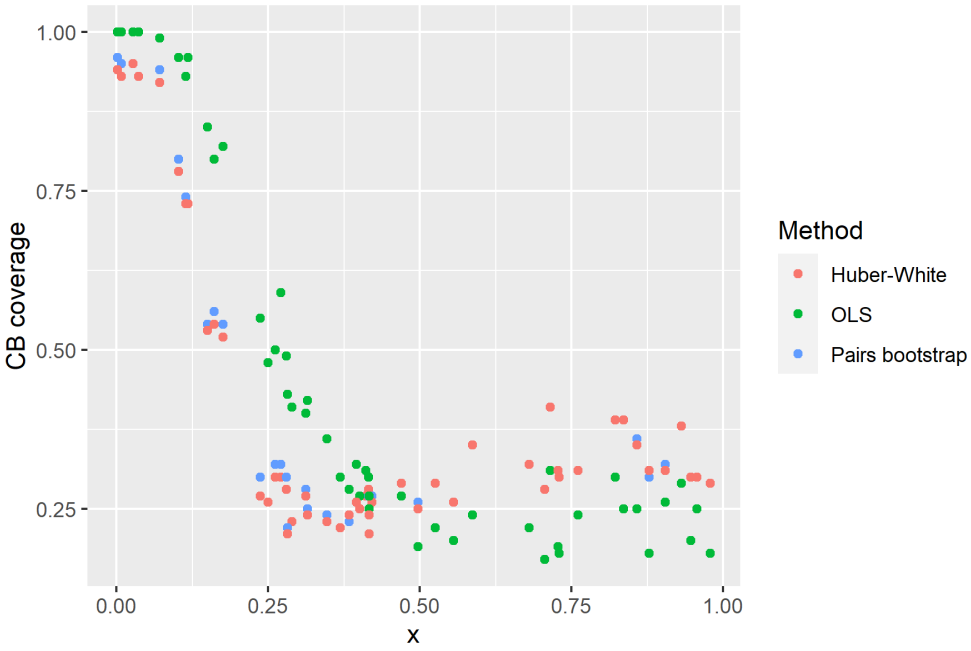

Figure 3: CB coverage as a function of x ($n = 100$).

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).

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

## # A tibble: 200 x 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.

(a)
```
ads_lm <- lm(sales ~ TV + radio + newspaper, ads_data)

png("./figures/residual_plt.png")
plot(ads_lm, which = 3)

png("./figures/qq_plt.png")
plot(ads_lm, which = 2)
dev.off()

png("./figures/leverage_plt.png")
plot(ads_lm, which = 5)
dev.off()
```

In the residual plot we can clearly see that there is heteroskedasticity, where the residuals appear to be approximately a quadratic function of the fitted values $\hat{t}$. Furthermore, the QQ-plot deviates significantly from the QQ-line at the tails, indicating that the residuals are not normally distributed. Based off Cook's distance, it appears that the data may not contain outliers.

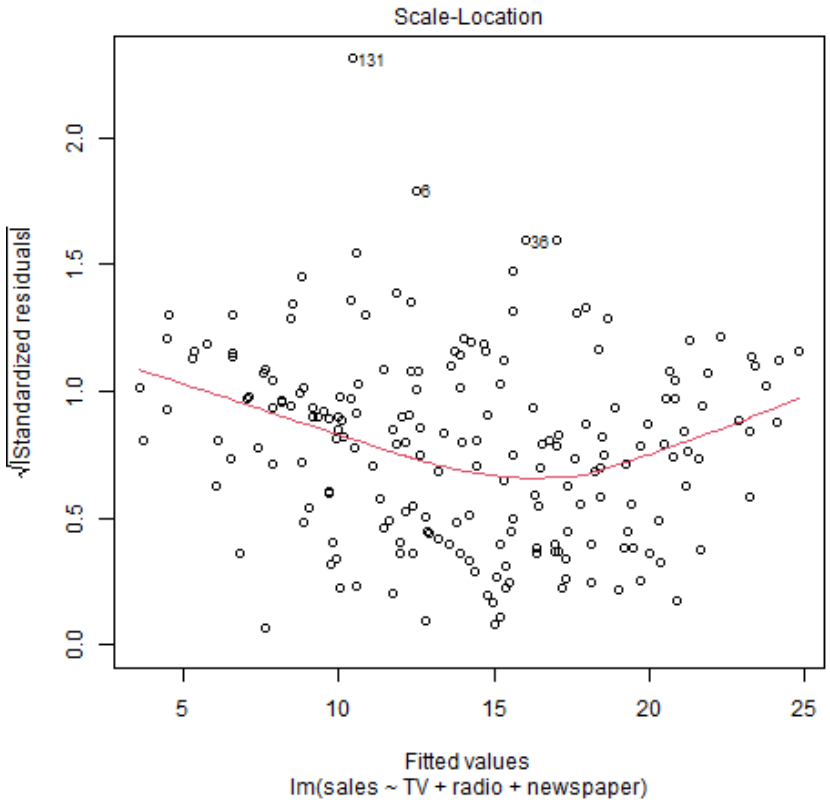

Figure 4: Standardized residuals vs. fitted values.

(b) A quick glance at the pairs plot with the log- and square root-transforms of sales indicates that each of these transforms does little to account for heteroskedasticity when looking at the marginal distribution of each variable of interest.

```
ads_data_aug <-
  ads_data %>%
  mutate(sqrt_sales = sqrt(sales),
         log_sales = log(sales))

pairs_plt <-
  GGally::ggpairs(
    ads_data_aug,
    columnLabels =
      c("TV",
        "Radio",
        "Newspaper",
        "Sales",
        # ggpairs does not provide a way to use LaTeX expressions when changing column names
        #latex2exp::TeX("\\sqrt{\\text{Sales}}"),
        #latex2exp::TeX("\\log{\\text{Sales}}")))
```

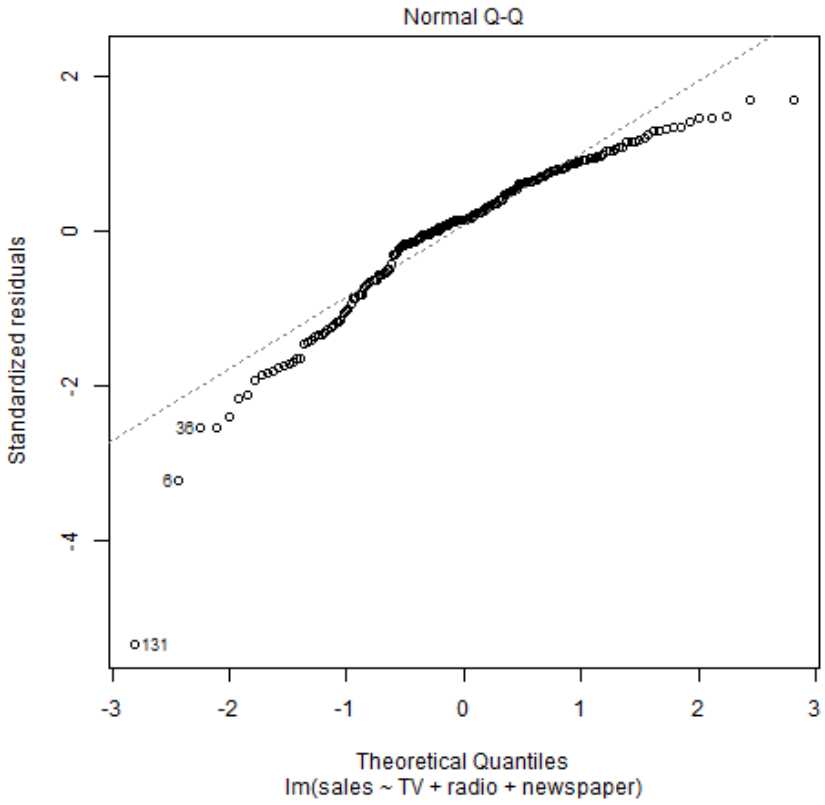

Figure 5: Quantile-quantile plot.

```
        "(Sales)^(1/2)",
        "log(Sales)"))
ggsave("./figures/pairs_plt.png", pairs_plt)
```

We instead perform our statistical inference using standard errors derived from the pairs bootstrap in order to account for apparent the heteroskedasticity.

```
X_ads <- ads_data %>% select(-sales)
sales <- ads_data$sales
B <- 500


# Run lm, extract coefficients
lm_ads <- lm(sales ~ ., ads_data)
coeff_ads <- lm_ads$coefficients

# Get model matrix
ads_model_mx <- cbind(rep(1, length(sales)), X_ads)
colnames(ads_model_mx)[1] <- "(Intercept)"
```

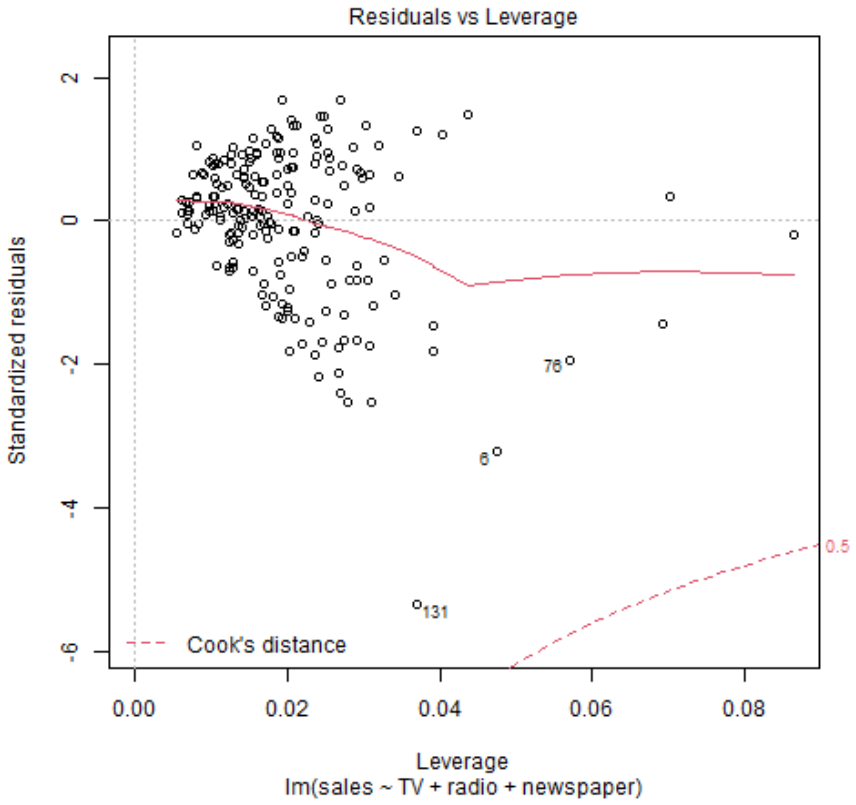

Figure 6: Standardized residuals vs. leverage.

```r
# Get sigma_hat
ads_pairs_sigma_hat <-
  pairs_bootstrap(X_ads, sales, B)

# Extract standard errors
ads_se <-
  ads_pairs_sigma_hat %>%
    diag() %>%
    sqrt()

# Dataframe to hold all estimated values
est_df <-
  # Add beta_hat
  data.frame(beta_i_hat = coeff_ads,
             sigma_i_hat = ads_se) %>%
    # Compute CI bounds, z-statistic
    mutate(CI_lower = beta_i_hat - 2 * sigma_i_hat,
           CI_upper = beta_i_hat + 2 * sigma_i_hat,
           z = beta_i_hat/sigma_i_hat) %>%
```

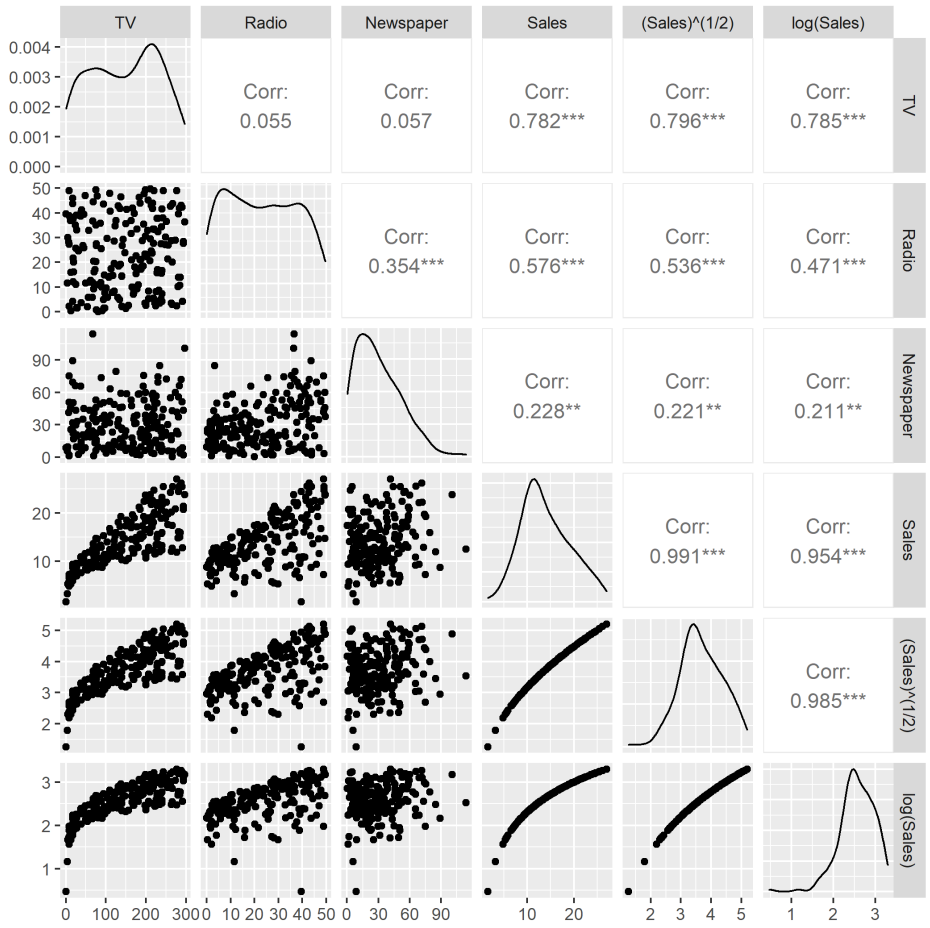

Figure 7: Pairs plot.

```
    # Compute approximate p-value
    mutate(p_val = 2 * pnorm(-abs(z)))

colnames(est_df) <-
  c("$\\widehat{\\beta}_i$",
    "$\\widehat{\\sigma}_i$",
    "CI lower bound",
    "CI upper bound",
    "$z$",
    "p-value")
rownames(est_df) <-
  c("Intercept",
    "TV",
    "Radio",
    "Newspaper")

est_df %>%
  kableExtra::kable(format = "latex", booktabs = TRUE, digits = 4, escape=FALSE) %>%
```

```
kableExtra::save_kable("figures/robust_est_tbl.png")
```

	$\widehat{\beta}_i$	$\widehat{\sigma}_i$	CI lower bound	CI upper bound	z	p-value
Intercept	2.9389	0.3197	2.2995	3.5783	9.1929	0.00
TV	0.0458	0.0019	0.0419	0.0496	23.8342	0.00
Radio	0.1885	0.0109	0.1668	0.2103	17.3421	0.00
Newspaper	-0.0010	0.0063	-0.0137	0.0116	-0.1636	0.87

Table 1: Heteroskedastic robust inference estimates.

The outcomes of the robust inference, including coefficient estimates, CI bounds, and p-values are provided in table 1.

(c) The model suggests that both TV and radio ads are associated with increased sales, though there is no evidence to suggest that newspapers ads are associated with higher sales. Assuming that both TV and radio ads are measured on the same scale, it seems that a one unit increase in radio ads is associated with a larger increase in sales than that of a one unit increase in TV ads.

One thing to note is that this model is not causal and thus looks simply at associations between variables rather than whether one causes another. Additionally, this model assumes a linear relationship between the variables of interest, which may not be correct.