HW4_567

Matthew Stoebe

2024-11-12

#a Use lm() to fit the multiple regression model described above. Provide estimates for the regression parameters β0, β1, β2 by showing the coefficients table from the summary output.

```
## Loading required package: carData
```

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -6.0646629 4.27194117 -1.419650 1.630896e-01
## income       0.5987328 0.11966735  5.003310 1.053184e-05
## education    0.5458339 0.09825264  5.555412 1.727192e-06
```

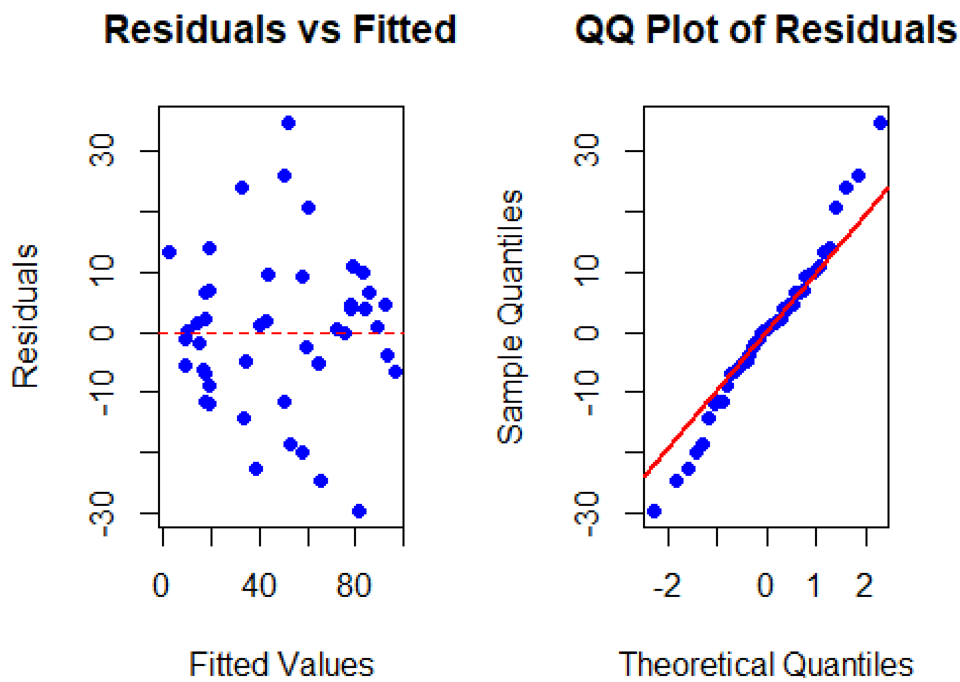
#b

```r
confint(model, level = 0.95)
```

```
##                   2.5 %    97.5 %
## (Intercept) -14.6857892 2.5564634
## income        0.3572343 0.8402313
## education     0.3475521 0.7441158
```

#c

```r
# Set up plotting area
par(mfrow = c(1, 2))  # Two plots side by side

# Residuals vs Fitted Values
plot(model$fitted.values, resid(model),
     main = "Residuals vs Fitted",
     xlab = "Fitted Values",
     ylab = "Residuals",
     pch = 19, col = "blue")
abline(h = 0, lty = 2, col = "red")

# QQ Plot of Residuals
qqnorm(resid(model), main = "QQ Plot of Residuals", pch = 19, col = "blue")
qqline(resid(model), col = "red", lwd = 2)
```

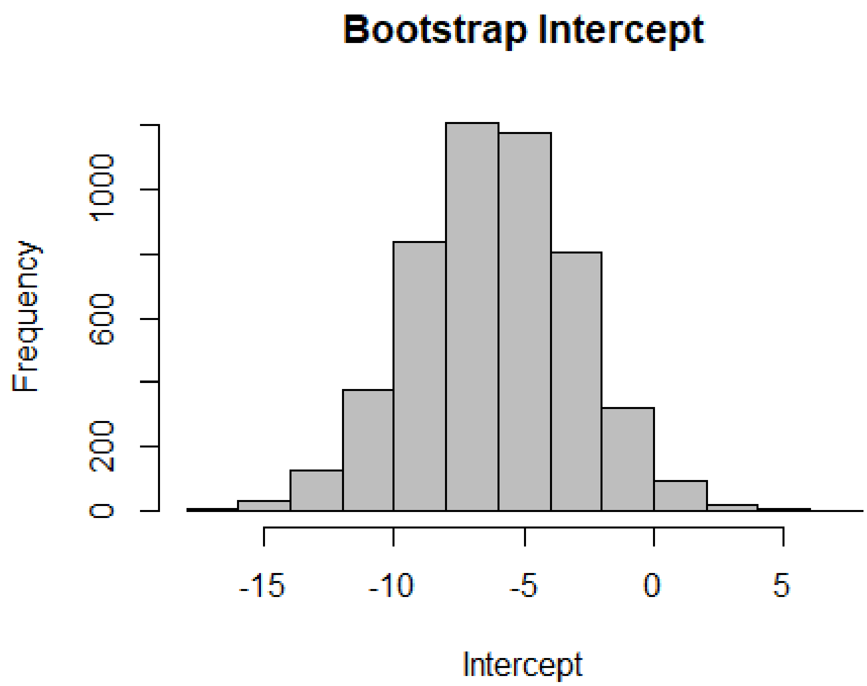


It appears that the variance of the residuals increases from left to right. We also see that for the outer points, the residuals fall off the line of normality. This implies that the assumptions are not satisfied

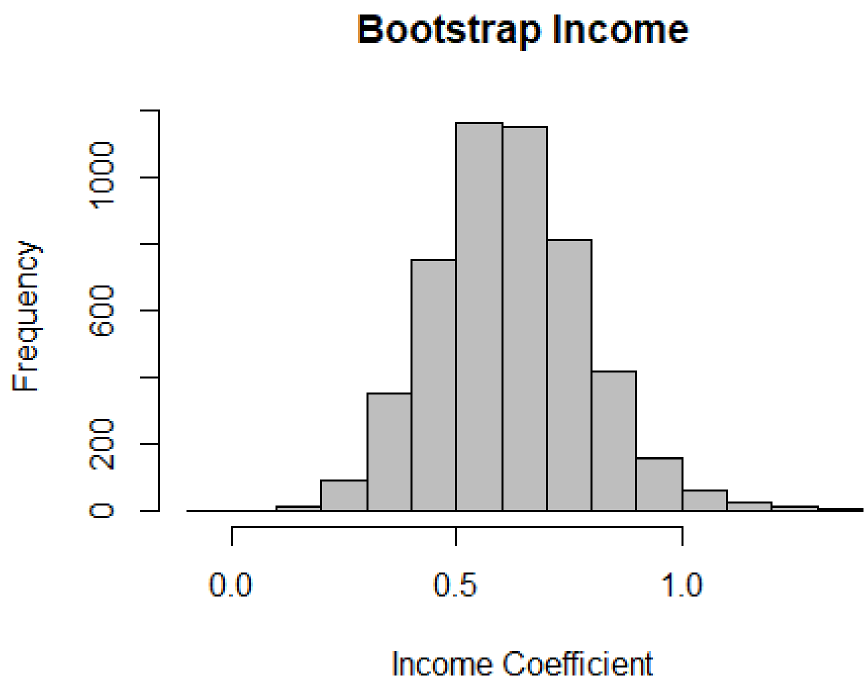
#d

```
library(boot)

## Warning: package 'boot' was built under R version 4.4.2

##
## Attaching package: 'boot'

## The following object is masked from 'package:car':
##
##     logit

boot_fn <- function(data, indices) {
  d <- data[indices, ]
  fit <- lm(prestige ~ income + education, data = d)
  return(coef(fit))
}

set.seed(123)

bootstrap_results <- boot(data = Duncan, statistic = boot_fn, R = 5000)

boot_estimates <- bootstrap_results$t
```

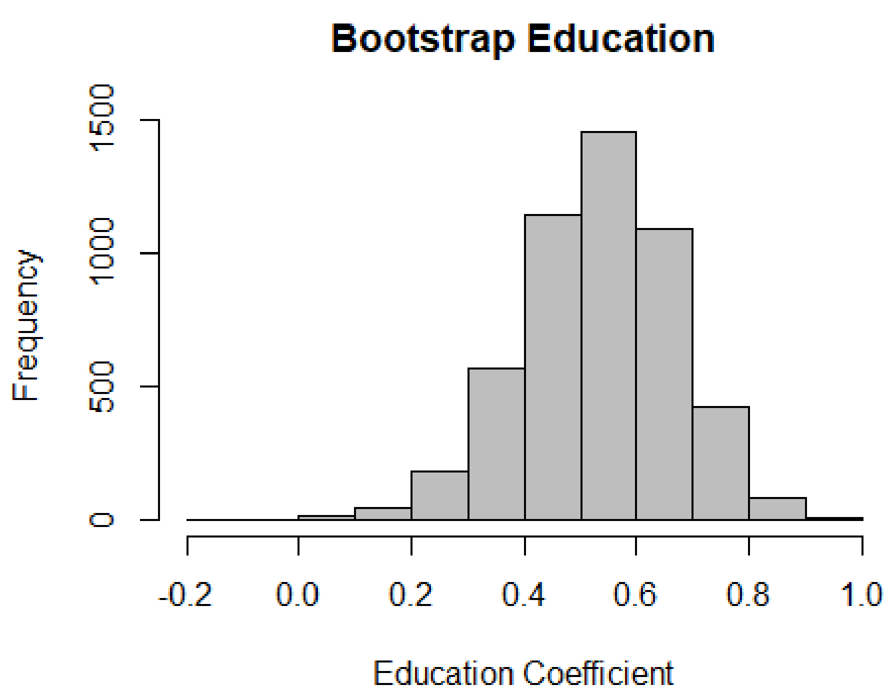
```r
colnames(boot_estimates) <- names(coef(model))

# Histogram for Intercept
hist(boot_estimates[, "(Intercept)"], main = "Bootstrap Intercept",
     xlab = "Intercept", col = "grey", border = "black")
```




```r
# Histogram for Income
hist(boot_estimates[, "income"], main = "Bootstrap Income",
     xlab = "Income Coefficient", col = "grey", border = "black")
```




```r
# Histogram for Education
hist(boot_estimates[, "education"], main = "Bootstrap Education",
     xlab = "Education Coefficient", col = "grey", border = "black")
```



#e

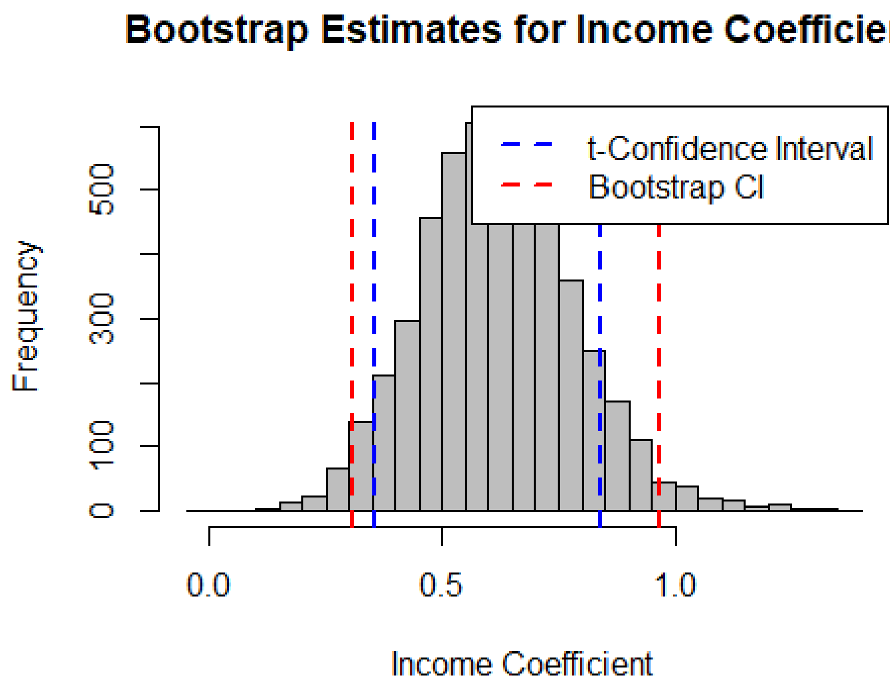
```
original_estimates <- coef(model)
bootstrap_means <- colMeans(boot_estimates)
bias <- bootstrap_means - original_estimates

print(bias)

## (Intercept)      income   education
## -0.09718098  0.01898643 -0.01499215
```

#f

```
percentile_CI <- apply(boot_estimates, 2, function(x) quantile(x, probs =
c(0.025, 0.975)))

# Transpose for better readability
percentile_CI <- t(percentile_CI)
colnames(percentile_CI) <- c("2.5 %", "97.5 %")
percentile_CI

##                   2.5 %      97.5 %
## (Intercept) -12.3272482 -0.08135699
## income        0.3075327  0.96596068
## education     0.2474596  0.78284989
```

#g

```
# Reconstruct histogram for income bootstrap estimates
hist(boot_estimates[, "income"],
     main = "Bootstrap Estimates for Income Coefficient",
     xlab = "Income Coefficient",
     col = "grey",
     border = "black",
     breaks = 30)

# Add t-based confidence interval (from part b)
t_CI <- confint(model, level = 0.95)["income", ]
abline(v = t_CI, col = "blue", lwd = 2, lty = 2)

# Add bootstrap confidence interval (from part f)
boot_CI_income <- percentile_CI["income", ]
abline(v = boot_CI_income, col = "red", lwd = 2, lty = 2)

legend("topright", legend = c("t-Confidence Interval", "Bootstrap CI"),
       col = c("blue", "red"), lty = 2, lwd = 2)
```



Our bootstrap confidence interval is wider than the t confidence interval. this is expected