Measuring Uncertainty of Correlation and Regression Coefficients

eCornell

7/27/2021

This R Markdown file demonstrates how to use regression analysis to find the best fit line. The example here finds the uncertainty around the correlation coefficient and the regression coefficients of the best fit line that passes through the scatterplot of prestige vs. education. These data are from the Prestige data set.

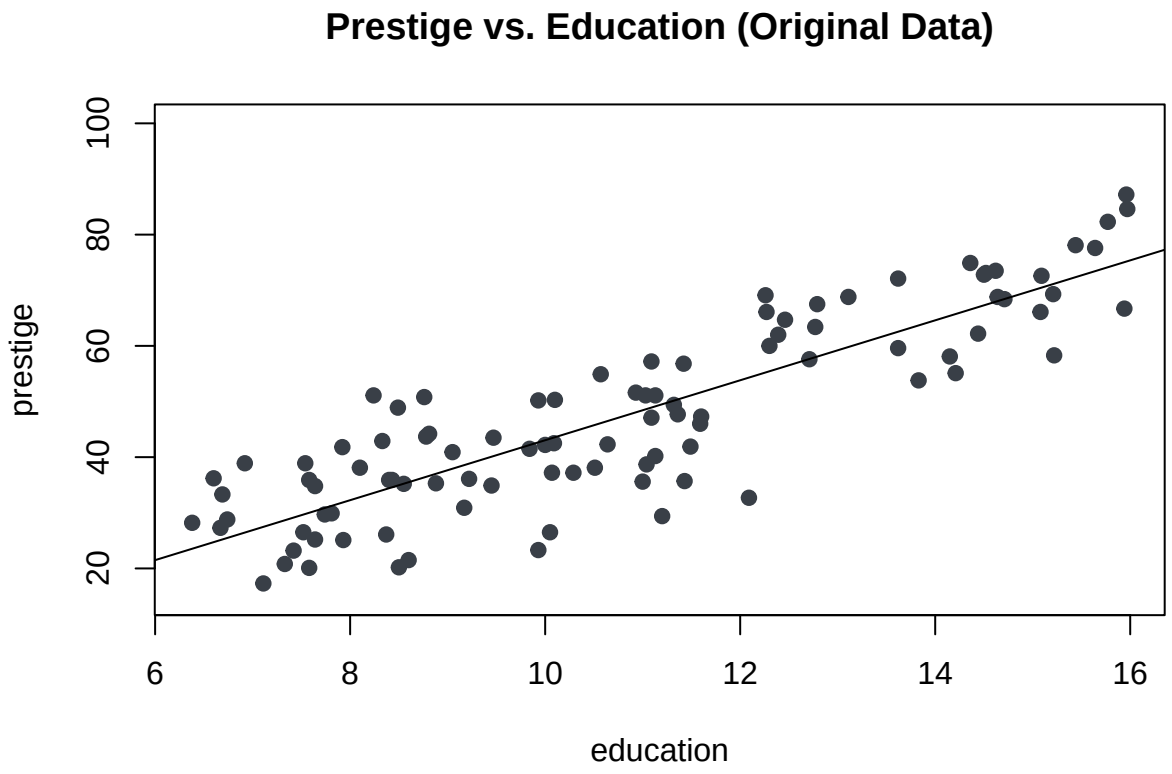
Step 1: Load the data and define colors.

```
knitr::opts_chunk$set(echo = TRUE)
# The Prestige data set is available in the carData library
library(carData)
# Load the Prestige data set
data(Prestige)
# Exclude any observations that do not have an entry in the type column
Prestige = Prestige[!is.na(Prestige$type),]

#eCornell Hex Codes:
crimson = '#b31b1b' #Crimson
lightGray = '#cecece' #lightGray
darkGray = '#606366' #darkGray
skyBlue = '#92b2c4' #skyblue
gold = '#fbb040' #gold
ecBlack = '#393f47' #ecBlack
```

Step 2: Plot the data and regression line.

Create the scatterplot with the regression line through it.

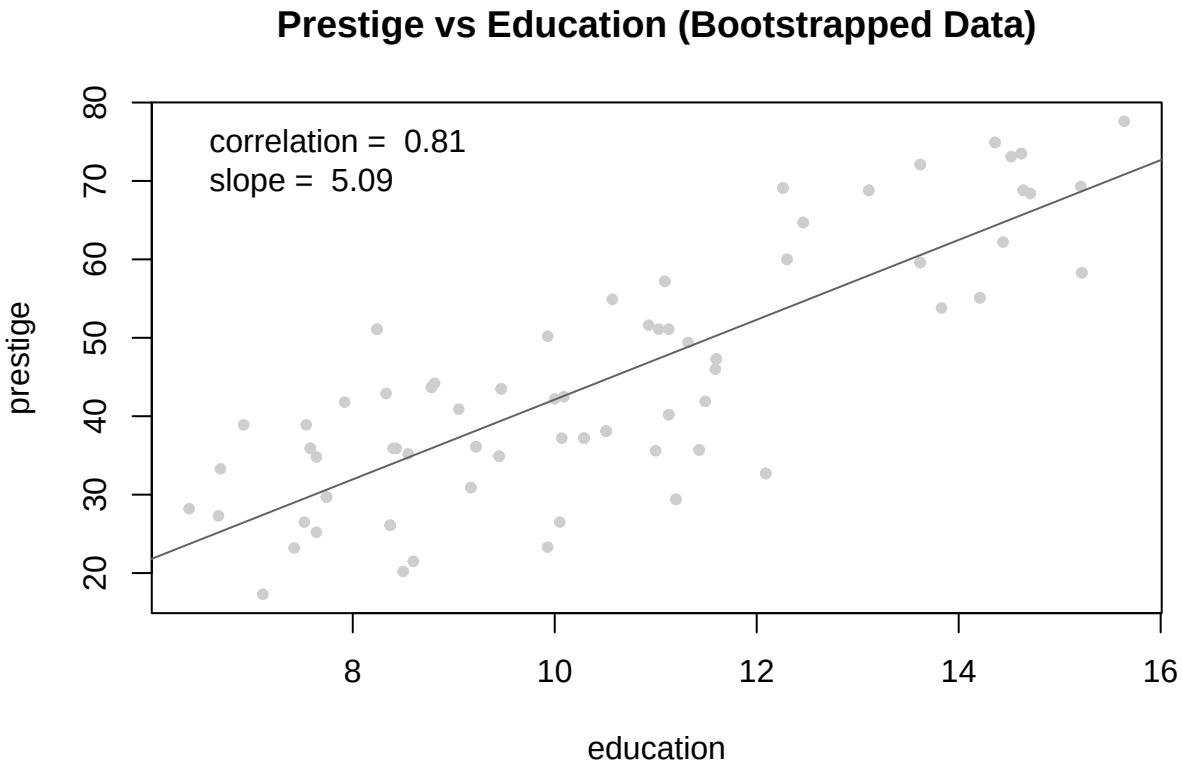
```
plot(prestige ~ education, data = Prestige, pch = 19, col = ecBlack,
     main = 'Prestige vs. Education (Original Data)', ylim = c(15, 100))
fit <- lm(prestige ~ education, data = Prestige)
abline(fit)
```



Step 3: Generate one bootstrapped data set.

Generate a bootstrapped data set, then calculate the correlation and regression coefficients of that data set.

```
# Generate then plot the bootstrapped data set:
boot.id = sample(98, replace = TRUE)
Prestige.boot = Prestige[boot.id,]
plot(Prestige.boot$education, Prestige.boot$prestige,
  pch = 20, col = lightGray,
  xlab = 'education', ylab = 'prestige',
  main = 'Prestige vs Education (Bootstrapped Data)')

# Do a regression on the bootstrapped data:
fit.boot <- lm(prestige ~ education, data = Prestige.boot)

# Calculate the correlation and regression coefficients:
r.boot = round(cor(Prestige.boot$education, Prestige.boot$prestige),2)
b.boot = round(fit.boot$coefficients[2], 2)

# Add the regression line and coefficient values to the plot:
legend('topleft', legend = c(paste('correlation = ', r.boot), paste('slope = ', b.boot)),
         bty = 'n')

abline(fit.boot, lwd = 1, col = darkGray)
```

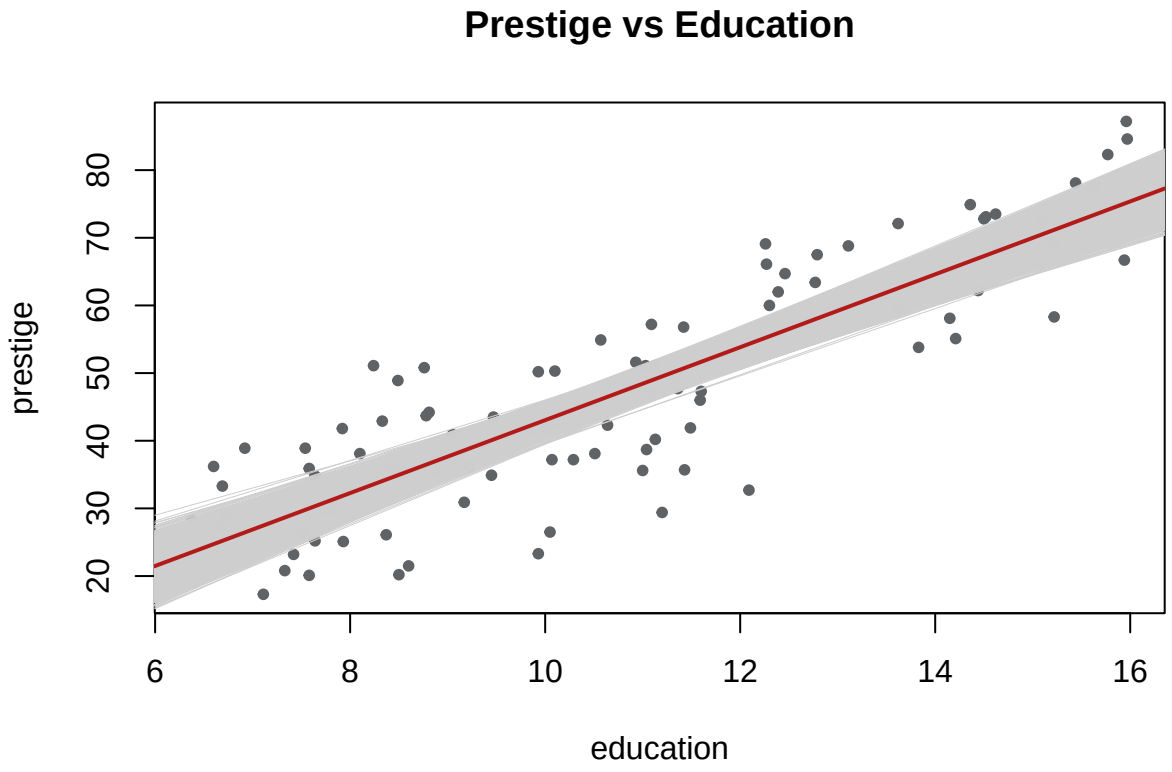


Step 4: Generate many bootstrapped data sets.

In the code below, you'll generate a bootstrapped data set 10,000 times, and store the values of correlations, intercepts, and slopes for each data set.

```
set.seed(1) # Set the seed for reproducibility
B = 10000   # Number of bootstrapped data sets
corr.boot = rep(0, B) # vector to store correlation coefficients
a.boot = rep(0,B)     # vector to store intercept values
b.boot = rep(0, B)    # vector to store slope values

# Create plot of observed data:
plot(Prestige$education, Prestige$prestige,
     pch = 20, col = darkGray,
     xlab = 'education', ylab = 'prestige',
     main = 'Prestige vs Education')

# Use a for loop to generate B bootstrapped data sets:
for (b in 1:B){
  boot.id = sample(98, replace = TRUE)
  Prestige.boot = Prestige[boot.id,]

# Store coefficients on each bootstrapped sample:
  corr.boot[b] = cor(Prestige.boot$education, Prestige.boot$prestige)
  fit.boot <- lm(prestige ~ education, data = Prestige.boot)
  a.boot[b] = fit.boot$coefficients[1]
  b.boot[b] = fit.boot$coefficients[2]

# Visualize each bootstrapped regression line on the plot in gray:
  abline(fit.boot, lwd = 0.5, col = lightGray)
```

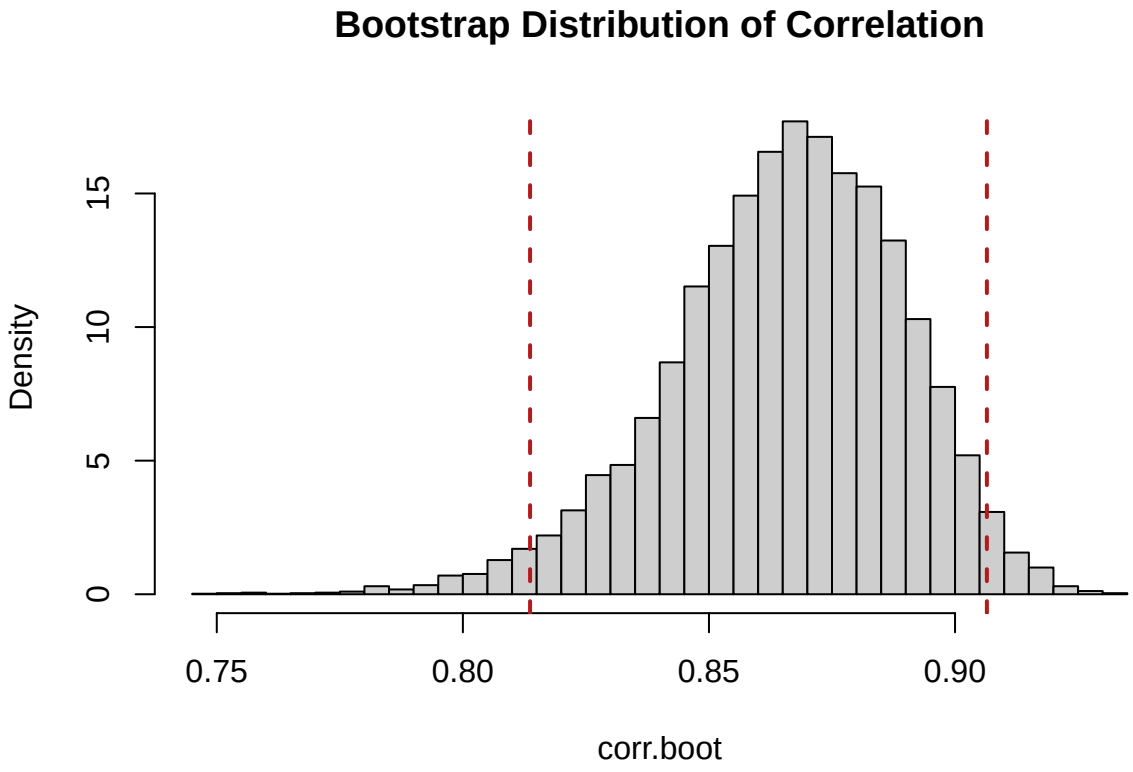
```
}
# Add the regression line for the observed data in red:
abline(fit, col = crimson, lwd = 2)
```



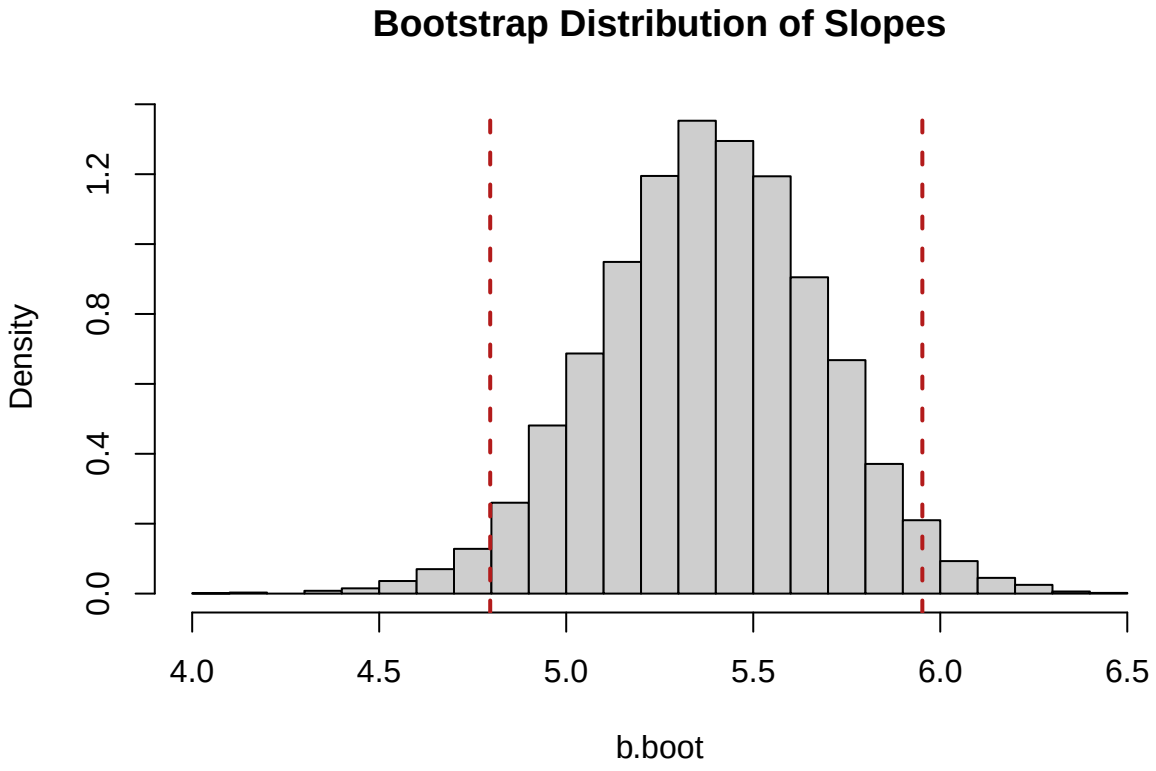
Step 5: Construct and interpret the 95% confidence intervals (CIs).

Use the bootstrap distribution of correlation coefficients to construct 95% CI and test if the population correlation is zero.

```
# Create histogram of correlation coefficients (r):
hist(corr.boot, breaks = 30, freq = FALSE, col = lightGray,
     main = 'Bootstrap Distribution of Correlation')
# Add the 95% CI to the histogram:
abline(v = quantile(corr.boot, probs = c(0.025, 0.975)), lty = 2, col = crimson, lwd = 2)
```




```
# Create a histogram of slopes (b):
hist(b.boot, breaks = 30, freq = FALSE, col = lightGray,
     main = 'Bootstrap Distribution of Slopes')
# Add the 95% CI to the histogram:
abline(v = quantile(b.boot, probs = c(0.025, 0.975)), lty = 2, col = crimson, lwd = 2)
```



Since 0 is not inside this 95% confidence interval, we say that we are 95% confident that the population slope, i.e., the slope of the best fitted line through the scatterplot of ALL jobs in 1971 Canada is non-zero.