Universidad Carlos III de Madrid

MSc in Statistics for Data Science

Bootstrap Project

Simulation and resampling

Authors:

Katherine Fazioli

Diego Perán Vacas

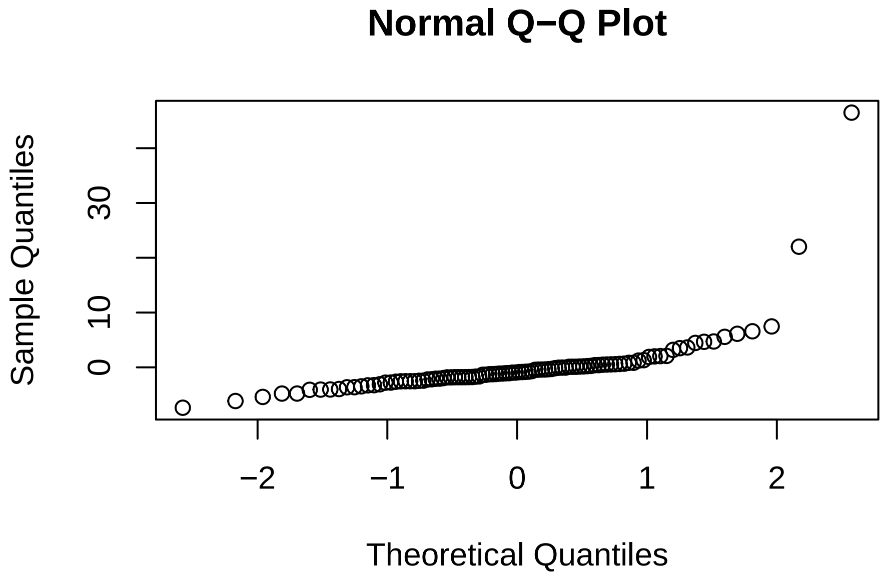
19th March 2021

Introduction

The bootstrap project begins with the extraction of the corresponding database. Our group number is 5, and therefore the data with which we are going to work will be those found in the data_5.csv file.

The first step is to check that the residuals for the OLS model are not normally distributed, which is evidence of the existence of outliers.

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3, data = data5)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.375 -2.184 -0.917  0.486 46.490
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.256      2.494   2.107 0.037688 *
## x1            10.069      3.131   3.216 0.001772 **
## x2            11.412      3.129   3.647 0.000431 ***
## x3            -5.249      3.113  -1.687 0.094945 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.926 on 96 degrees of freedom
## Multiple R-squared:  0.948,  Adjusted R-squared:  0.9464
## F-statistic:   584 on 3 and 96 DF,  p-value: < 2.2e-16
```



It seems clear that these residuals are not normally distributed, and for this reason we will use the confidence intervals estimated later as a criterion for the significance of the covariates.

a) Bootstrap confidence intervals on the regressors (complete model)

First, we bootstrap the coefficients of the complete model (with the three covariates), and with the result we extract the confidence interval of each coefficient.

```
(theta.hat=rlm(y ~ x1 + x2 +x3,data=data5)$coef)
```

```
## (Intercept)          x1          x2          x3
##   4.8571640   4.3401959   5.3280991   0.6466318
```

```
n=length(data5$y)
stat=function(x,xdata){rlm(xdata$y[x] ~ xdata$x1[x] + xdata$x2[x]
                           +xdata$x3[x],data=xdata)$coef}
set.seed(1)
btheta<-bootstrap(x=1:100,nboot=5000,theta=stat,xdata=data5)$thetastar
```

First, we try with the basic bootstrap confidence interval whose formula is:

$$\left[2\hat{\theta} - F^{-1}_{\hat{\theta}^*}(1-\alpha/2)\ ,\ 2\hat{\theta} - F^{-1}_{\hat{\theta}^*}(\alpha/2)\right]$$

therefore for an α equal to 0.05 we have

```
for (i in 2:4){
  print(2*theta.hat[i]-quantile(btheta[i,],c(0.975,0.025)))
}
```

```
##    97.5%     2.5%
## 2.199269 6.606456
##    97.5%     2.5%
## 3.151874 7.702958
##     97.5%      2.5%
## -1.687451  2.825719
```

Next, we are going to use the best intervals in order to study the significance of the regressors, that is, we check with the bias corrected accelerated (BC_a) bootstrap confidence intervals:

```
stat2=function(x,xdata){rlm(xdata$y[x] ~ xdata$x1[x]
                            + xdata$x2[x] +xdata$x3[x],data=xdata)$coef[2]}

bcanon(x=1:100,nboot=5000,theta=stat2,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

```
##      alpha bca point
## [1,] 0.025  2.172617
## [2,] 0.975  6.637041
```

```
stat3=function(x,xdata){rlm(xdata$y[x] ~ xdata$x1[x]
                            + xdata$x2[x] +xdata$x3[x],data=xdata)$coef[3]}

bcanon(x=1:100,nboot=1000,theta=stat3,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

```
##      alpha bca point
## [1,] 0.025  3.100696
## [2,] 0.975  7.969500
```

```
stat4=function(x,xdata){rlm(xdata$y[x] ~xdata$x1[x]
                            + xdata$x2[x]+ xdata$x3[x],data=xdata)$coef[4]}

bcanon(x=1:100,nboot=1000,theta=stat4,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

```
##      alpha bca point
## [1,] 0.025 -2.024593
## [2,] 0.975  2.803878
```

The results in both cases are very similar and therefore the conclusions are the same. It is clear that the least significant regressor is called x3, since its coefficient is the lowest and, observing its confidence intervals, 0 falls within them, so the coefficient could be null. This indicates that x3 is the variable that provides the least information to explain the response variable.

b) Backward elimination

We have to do backward elimination and the criteria to eliminate a covariate from the model are the estimated confidence intervals. Therefore, we check the model by eliminating x3:

```
(theta.hat=rlm(y ~ x1 + x2 ,data=data5)$coef)
```

```
## (Intercept)          x1          x2
##    4.808276    4.983799    5.981283
```

```
n=length(data5$y)
stat=function(x,xdata){rlm(xdata$y[x] ~ xdata$x1[x] + xdata$x2[x] ,data=xdata)$coef}
set.seed(1)
btheta2<-bootstrap(x=1:100,nboot=5000,theta=stat,xdata=data5)$thetastar
```

```
for (i in 2:3){
  print(2*theta.hat[i]-quantile(btheta2[i,],c(0.975,0.025)))}
```

```
##    97.5%     2.5%
## 4.843949 5.113608
##    97.5%     2.5%
## 5.869191 6.098540
```

The length of the confidence intervals of both regressors have been reduced, so we deduce that they are more reliable estimates. Maybe this previous model is the optimal one, but still we continue with the backward elimination, and in this case we discard x1 because it has a lower coefficient.

```
(theta.hat=rlm(y ~ + x2 ,data=data5)$coef)
```

```
## (Intercept)          x2
##   38.264569    6.745086
```

```
stat=function(x,xdata){rlm(xdata$y[x] ~ xdata$x2[x] ,data=xdata)$coef}
set.seed(1)
btheta<-bootstrap(x=1:100,nboot=5000,theta=stat,xdata=data5)$thetastar
```

```
print(2*theta.hat[2]-quantile(btheta[2,],c(0.975,0.025)))
```

```
##    97.5%     2.5%
## 5.852324 7.685125
```

The conclusion is that this model is worse, since the intercepts take a very high value and therefore the only predictor provides little information. In addition, the length of the confidence intervals of the only regressor that remains is greater than the previous one. Therefore, we choose the second model that contains the variables x1 and x2 as regressors as the optimal model and with the most relevant covariates.

c) Confidence intervals on coefficients

We now provide 95% confidence intervals for our coefficients from our optimal model with x1 and x2 as regressors. We report the basic bootstrap confidence interval (as calculated above) as well as the BC_a confidence interval which we caclulate here.

The BC_a confidence interval for x1 is:

```
stat2=function(x,xdata){rlm(xdata$y[x] ~ xdata$x1[x] + xdata$x2[x],data=xdata)$coef[2]}

set.seed(1)
bcanon(x=1:100,nboot=5000,theta=stat2,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

```
##      alpha bca point
## [1,] 0.025  4.848854
## [2,] 0.975  5.118230
```

The BC_a confidence interval for x2 is:

```
stat3=function(x,xdata){rlm(xdata$y[x] ~ xdata$x1[x]+ xdata$x2[x],data=xdata)$coef[3]}

set.seed(1)
bcanon(x=1:100,nboot=1000,theta=stat3,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

```
##      alpha bca point
## [1,] 0.025  5.857875
## [2,] 0.975  6.093488
```

The table below provides a summary with the confidence intervals we have calculated.

	x1		x2	
	0.025	0.975	0.025	0.975
Basic	4.843949	5.113608	5.869191	6.098540
BCa	4.848854	5.11823	5.857875	6.093488

We see very similar results from the basic and BC_a intervals.

d) Confidence intervals on mean response

Now we build confidence intervals on the mean response when (x1,x2,x3)=(14,14,14) using the full model (all three covariates). We also build a confidence interval when just considering (x1,x2)=(14,14) using our optimal model.

We begin with the full model, and find the fitted response when (x1,x2,x3)=(14,14,14) is 149.27.

```
(theta.hat=rlm(y ~ x1 + x2 +x3,data=data5)$coef)
```

```
## (Intercept)          x1          x2          x3
##   4.8571640   4.3401959   5.3280991   0.6466318
```

```
(y_pred=theta.hat[1]+14*theta.hat[2]+14*theta.hat[3]+14*theta.hat[4])
```

```
## (Intercept)
##    149.2661
```

We then bootstrap the mean response and obtain confidence intervals.

```
pred=function(x,xdata){
  rlm_coef=rlm(xdata$y[x] ~ xdata$x1[x] + xdata$x2[x] +
                 xdata$x3[x], data=xdata)$coef
  return(rlm_coef[1]+14*rlm_coef[2]+14*rlm_coef[3]+14*rlm_coef[4])
  }
```

We first obtain the basic bootstrap confidence interval:

```
set.seed(1)
by_pred<-bootstrap(x=1:100,nboot=5000,theta=pred,
                   xdata=data5)$thetastar

2*y_pred-quantile(by_pred,c(0.975,0.025))
```

```
##    97.5%     2.5%
## 118.7235 181.6966
```

Next, we obtain the BC_a confidence interval:

```
set.seed(1)
bcanon(x=1:100,nboot=5000,theta=pred,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

```
##      alpha bca point
## [1,] 0.025  118.2801
## [2,] 0.975  181.8878
```

We see the confidence intervals are very similar for mean response for our full model.

Next, we calculate the confidence intervals for our mean response in our optimal model with x1 and x2 as covariates. We see the fitted response when (x1,x2)=(14,14) is 158.32.

```
(theta.hat=rlm(y ~ x1 + x2,data=data5)$coef)
```

```
## (Intercept)          x1          x2
##    4.808276    4.983799    5.981283
```

```
(y_pred=theta.hat[1]+14*theta.hat[2]+14*theta.hat[3])
```

```
## (Intercept)
##    158.3194
```

We follow the same approach, and first calculate the basic bootstrap confidence intervals:

```
pred=function(x,xdata){
  rlm_coef=rlm(xdata$y[x] ~ xdata$x1[x] + xdata$x2[x],
               data=xdata)$coef
  return(rlm_coef[1]+14*rlm_coef[2]+14*rlm_coef[3])
  }
```

```
set.seed(1)
by_pred<-bootstrap(x=1:100,nboot=5000,theta=pred,
                   xdata=data5)$thetastar
```

```
2*y_pred-quantile(by_pred,c(0.975,0.025))
```

```
##    97.5%     2.5% 
## 157.2351 159.1852
```

We also calculate the BC_a confidence intervals.

```
set.seed(1)
bcanon(x=1:100,nboot=5000,theta=pred,
       alpha=c(0.025,0.975),xdata=data5)$confpoints
```

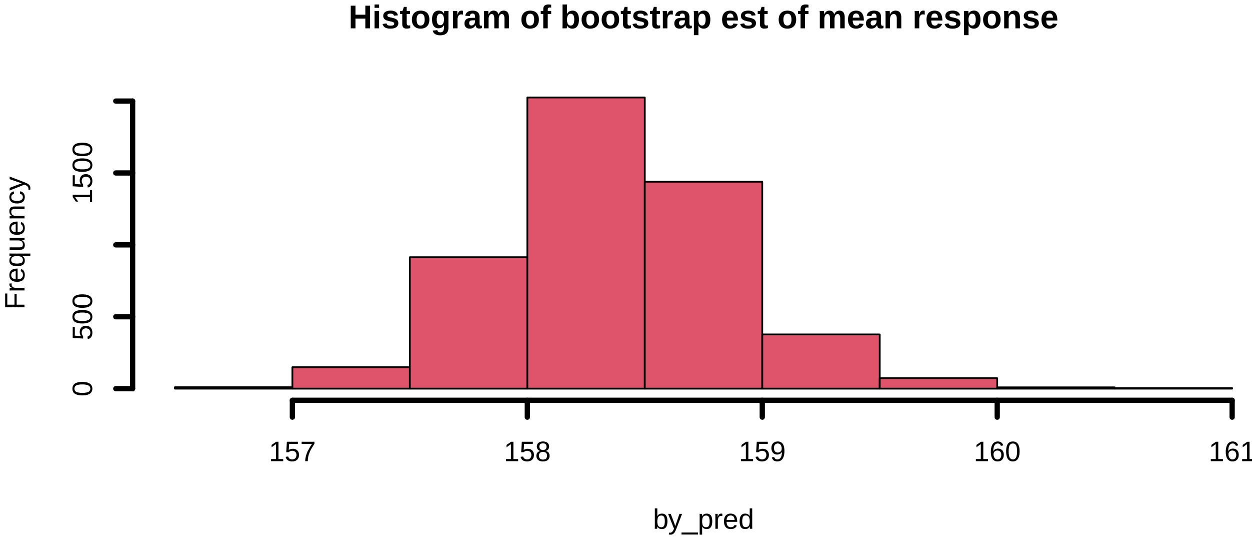
```
##      alpha bca point
## [1,] 0.025  157.3581
## [2,] 0.975  159.2692
```

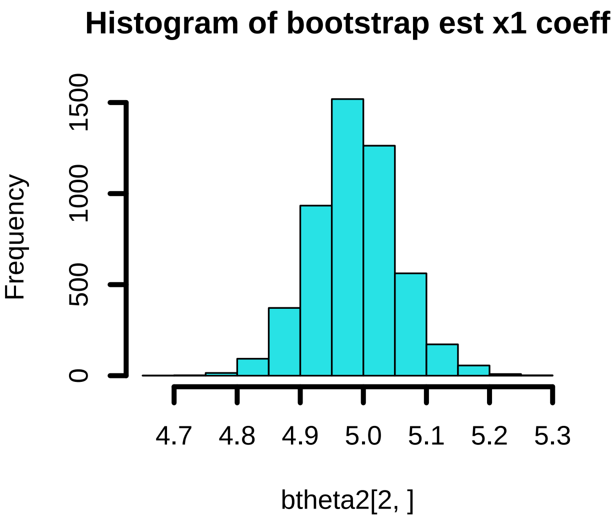
Again, we see the confidence intervals are very similar. However, the range of the confidence interval for mean response for our optimal model is much smaller.

Conclusions

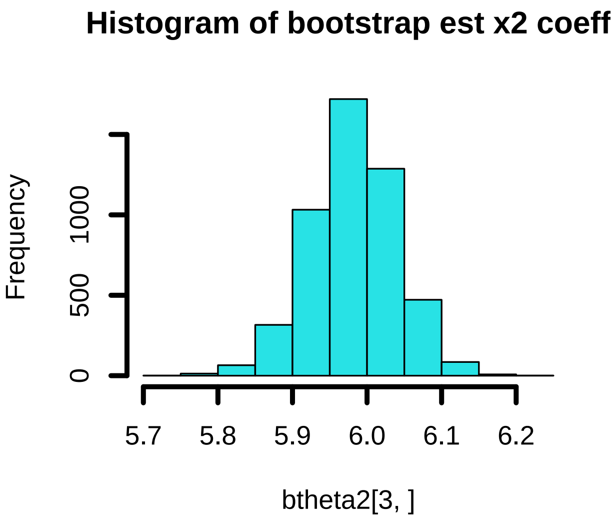
In this project, we have selected an optimal robust regression model using bootstrapping methods. We were able to study the significance of the coefficients by building bootstrap confidence intervals. We used these confidence intervals in our backwards elimination strategy to select the most relevant covariates. We were also able to build confidence intervals for the mean response in both the full model (all three covariates) and our optimal model (with x1 and x2). We saw a much smaller range in our confidence intervals around the coefficients and mean response for our optimal model compared to the full model. This is as expected as we have chosen the model with the most relevant covariates and are able to better to explain response.

Throughout this project, we have seen consistency with the basic and BC_a confidence intervals. In our case, our sample size was sufficient and we did not see much skewness or bias in our distributions of the bootstrap estimates. Below are histograms of the bootstrap estimates of mean response and coefficients from our optimal model. We see in this case, there is not much skew, so we would typically expect consistent estimates for the basic and BC_a confidence intervals as we have observed here.





Histogram of bootstrap est x1 coeff



Histogram of bootstrap est x2 coeff