first is the standard k-fold CV estimate, as in (5.3). The second is a bias-corrected version. On this data set, the two estimates are very similar to each other.

5.3.4 The Bootstrap

We illustrate the use of the bootstrap in the simple example of Section 5.2, as well as on an example involving estimating the accuracy of the linear regression model on the `Auto` data set.

Estimating the Accuracy of a Statistic of Interest

One of the great advantages of the bootstrap approach is that it can be applied in almost all situations. No complicated mathematical calculations are required. Performing a bootstrap analysis in `R` entails only two steps. First, we must create a function that computes the statistic of interest. Second, we use the `boot()` function, which is part of the `boot` library, to perform the bootstrap by repeatedly sampling observations from the data set with replacement.

`boot()`

The `Portfolio` data set in the `ISLR` package is described in Section 5.2. To illustrate the use of the bootstrap on this data, we must first create a function, `alpha.fn()`, which takes as input the (X, Y) data as well as a vector indicating which observations should be used to estimate α. The function then outputs the estimate for α based on the selected observations.

```
> alpha.fn=function(data,index){
+ X=data$X[index]
+ Y=data$Y[index]
+ return((var(Y)-cov(X,Y))/(var(X)+var(Y)-2*cov(X,Y)))
+ }
```

This function *returns*, or outputs, an estimate for α based on applying (5.7) to the observations indexed by the argument `index`. For instance, the following command tells `R` to estimate α using all 100 observations.

```
> alpha.fn(Portfolio,1:100)
[1] 0.576
```

The next command uses the `sample()` function to randomly select 100 observations from the range 1 to 100, with replacement. This is equivalent to constructing a new bootstrap data set and recomputing $\hat{\alpha}$ based on the new data set.

```
> set.seed(1)
> alpha.fn(Portfolio,sample(100,100,replace=T))
[1] 0.596
```

We can implement a bootstrap analysis by performing this command many times, recording all of the corresponding estimates for α, and computing

the resulting standard deviation. However, the `boot()` function automates this approach. Below we produce $R = 1,000$ bootstrap estimates for α. `boot()`

```
> boot(Portfolio,alpha.fn,R=1000)

ORDINARY NONPARAMETRIC BOOTSTRAP

Call:
boot(data = Portfolio, statistic = alpha.fn, R = 1000)

Bootstrap Statistics :
     original       bias      std. error
t1*  0.5758      -7.315e-05   0.0886
```

The final output shows that using the original data, $\hat{\alpha} = 0.5758$, and that the bootstrap estimate for $\text{SE}(\hat{\alpha})$ is 0.0886.

Estimating the Accuracy of a Linear Regression Model

The bootstrap approach can be used to assess the variability of the coefficient estimates and predictions from a statistical learning method. Here we use the bootstrap approach in order to assess the variability of the estimates for β_0 and β_1, the intercept and slope terms for the linear regression model that uses `horsepower` to predict `mpg` in the `Auto` data set. We will compare the estimates obtained using the bootstrap to those obtained using the formulas for $\text{SE}(\hat{\beta}_0)$ and $\text{SE}(\hat{\beta}_1)$ described in Section 3.1.2.

We first create a simple function, `boot.fn()`, which takes in the `Auto` data set as well as a set of indices for the observations, and returns the intercept and slope estimates for the linear regression model. We then apply this function to the full set of 392 observations in order to compute the estimates of β_0 and β_1 on the entire data set using the usual linear regression coefficient estimate formulas from Chapter 3. Note that we do not need the { and } at the beginning and end of the function because it is only one line long.

```
> boot.fn=function(data,index)
+ return(coef(lm(mpg~horsepower,data=data,subset=index)))
> boot.fn(Auto,1:392)
(Intercept) horsepower
   39.936   -0.158
```

The `boot.fn()` function can also be used in order to create bootstrap estimates for the intercept and slope terms by randomly sampling from among the observations with replacement. Here we give two examples.

```
> set.seed(1)
> boot.fn(Auto,sample(392,392,replace=T))
(Intercept) horsepower
  38.739     -0.148
> boot.fn(Auto,sample(392,392,replace=T))
(Intercept) horsepower
  40.038     -0.160
```

Next, we use the `boot()` function to compute the standard errors of 1,000 bootstrap estimates for the intercept and slope terms.

```
> boot(Auto,boot.fn,1000)

ORDINARY NONPARAMETRIC BOOTSTRAP

Call:
boot(data = Auto, statistic = boot.fn, R = 1000)

Bootstrap Statistics :
    original     bias    std. error
t1* 39.936       0.0297  0.8600
t2* -0.158      -0.0003  0.0074
```

This indicates that the bootstrap estimate for $\mathrm{SE}(\hat{\beta}_0)$ is 0.86, and that the bootstrap estimate for $\mathrm{SE}(\hat{\beta}_1)$ is 0.0074. As discussed in Section 3.1.2, standard formulas can be used to compute the standard errors for the regression coefficients in a linear model. These can be obtained using the `summary()` function.

```
> summary(lm(mpg~horsepower,data=Auto))$coef
             Estimate  Std. Error  t value   Pr(>|t|)
(Intercept)    39.936     0.71750     55.7  1.22e-187
horsepower     -0.158     0.00645    -24.5   7.03e-81
```

The standard error estimates for $\hat{\beta}_0$ and $\hat{\beta}_1$ obtained using the formulas from Section 3.1.2 are 0.717 for the intercept and 0.0064 for the slope. Interestingly, these are somewhat different from the estimates obtained using the bootstrap. Does this indicate a problem with the bootstrap? In fact, it suggests the opposite. Recall that the standard formulas given in Equation 3.8 on page 66 rely on certain assumptions. For example, they depend on the unknown parameter σ^2, the noise variance. We then estimate σ^2 using the RSS. Now although the formula for the standard errors do not rely on the linear model being correct, the estimate for σ^2 does. We see in Figure 3.8 on page 91 that there is a non-linear relationship in the data, and so the residuals from a linear fit will be inflated, and so will $\hat{\sigma}^2$. Secondly, the standard formulas assume (somewhat unrealistically) that the x_i are fixed, and all the variability comes from the variation in the errors ϵ_i. The bootstrap approach does not rely on any of these assumptions, and so it is likely giving a more accurate estimate of the standard errors of $\hat{\beta}_0$ and $\hat{\beta}_1$ than is the `summary()` function.

Below we compute the bootstrap standard error estimates and the standard linear regression estimates that result from fitting the quadratic model to the data. Since this model provides a good fit to the data (Figure 3.8), there is now a better correspondence between the bootstrap estimates and the standard estimates of $\mathrm{SE}(\hat{\beta}_0)$, $\mathrm{SE}(\hat{\beta}_1)$ and $\mathrm{SE}(\hat{\beta}_2)$.

```
> boot.fn=function(data,index)
+ coefficients(lm(mpg~horsepower+I(horsepower^2),data=data,
    subset=index))
> set.seed(1)
> boot(Auto,boot.fn,1000)

ORDINARY NONPARAMETRIC BOOTSTRAP

Call:
boot(data = Auto, statistic = boot.fn, R = 1000)

Bootstrap Statistics :
     original     bias    std. error
t1*  56.900    6.098e-03 2.0945
t2* -0.466    -1.777e-04 0.0334
t3*  0.001     1.324e-06 0.0001

> summary(lm(mpg~horsepower+I(horsepower^2),data=Auto))$coef
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       56.9001    1.80043      32 1.7e-109
horsepower        -0.4662    0.03112     -15  2.3e-40
I(horsepower^2)    0.0012    0.00012      10  2.2e-21
```

5.4 Exercises

Conceptual

1. Using basic statistical properties of the variance, as well as single-variable calculus, derive (5.6). In other words, prove that α given by (5.6) does indeed minimize $\mathrm{Var}(\alpha X + (1-\alpha)Y)$.

2. We will now derive the probability that a given observation is part of a bootstrap sample. Suppose that we obtain a bootstrap sample from a set of n observations.

   (a) What is the probability that the first bootstrap observation is *not* the jth observation from the original sample? Justify your answer.

   (b) What is the probability that the second bootstrap observation is *not* the jth observation from the original sample?

   (c) Argue that the probability that the jth observation is *not* in the bootstrap sample is $(1-1/n)^n$.

   (d) When $n = 5$, what is the probability that the jth observation is in the bootstrap sample?

   (e) When $n = 100$, what is the probability that the jth observation is in the bootstrap sample?