Bootstrap Example #1

```
> x
 [1] 53 46 44 58 64 45 40 53 73 40 73 47 63 52 82
> (bs <- sample(x,replace=T))
 [1] 64 40 44 52 40 58 73 44 40 52 47 40 53 73 73
> mean(bs)
[1] 52.86667
> (bs <- sample(x,replace=T))
 [1] 53 73 52 44 73 64 53 40 45 53 82 45 40 64 45
> mean(bs)
[1] 55.06667
> bs.means <- numeric(0)
> for (i in 1:1000) {
+   bs <- sample(x,replace=T)
+   bs.means[i] <- mean(bs)
+ }
> mean(x)
[1] 55.53333
> mean(bs.means)-mean(x)
[1] 0.01766667
> sd(bs.means)
[1] 3.215488
> quantile(bs.means,probs=c(.05,.95))
      5%      95%
50.46667 61.13333
>
```

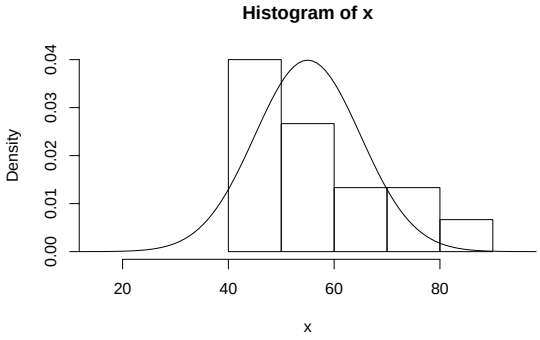
In fact, the sample was generated by:

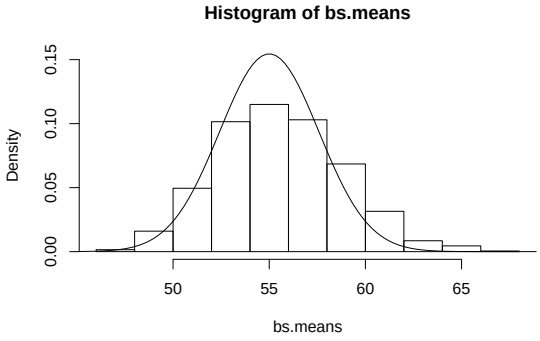
```
> x <- round(rnorm(15,mean=55,sd=10))
```

so $SE(\bar{X}) \approx \sigma/\sqrt{n} = 10/\sqrt{15} = 2.58$. Because the sample happened to have $s = 13.0$, our estimate of $SE(\bar{X})$ was closer to $13.0/\sqrt{15} = 3.36$.

Bootstrap Example #1

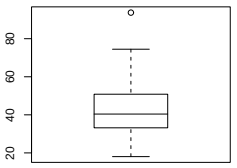
```
> hist(x, freq=F,xlim=c(15,95),breaks=5)
> curve(dnorm(x,55,10),add=T)
> hist(bs.means, freq=F,ylim=c(0,.15))
> curve(dnorm(x,55,10/sqrt(15)),add=T)
```

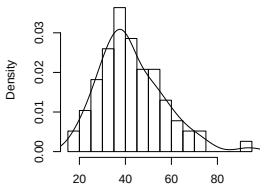




Bootstrap Example #2

Cereal Rating:






```
> bs.means <- numeric(1000)
> bs.trimmeans <- numeric(1000)
> for (b in 1:1000) {
+   bs <- sample(rating,replace=T)
+   bs.means[b] <- mean(bs)
+   bs.trimmeans[b] <- mean(bs,trim=.1)
+ }
> mean(rating)
[1] 42.66570
> mean(bs.means)-mean(rating)
[1] 0.01660235
> sd(bs.means)
[1] 1.529426
> sd(rating)/sqrt(77)
[1] 1.600837
> mean(rating,trim=.1)
[1] 41.66178
> mean(bs.trimmeans)-mean(rating,trim=.1)
[1] 0.02696686
> mean(bs.trimmeans)-mean(rating)
[1] -0.976959
> sd(bs.trimmeans)
[1] 1.529365
>
```


Bootstrap Aspirin Example

```
> bs.as.heartattacks <- rbinom(1000,11037,104/11037)
> bs.pl.heartattacks <- rbinom(1000,11034,189/11034)
> phat.as <- bs.as.heartattacks/11037
> phat.pl <- bs.pl.heartattacks/11034
> relative.risk <- phat.as/phat.pl
> length(relative.risk)
[1] 1000
> (104/11037)/(189/11034)
[1] 0.550115
> mean(relative.risk)-.550115
[1] 0.001446324
> sd(relative.risk)
[1] 0.06653186  # compared to .06675 theoretical approx
> quantile(relative.risk, probs=c(.025,.975))
     2.5%     97.5%
0.4353592 0.6908454    # versus [0.416,0.684] theoretical approx
>
```