RLisbonaMSDS6306\_Week4\_BootstrapSampling

Randy Lisbona

June 4, 2016

## Create the normal and exponential sample datasets

Use rnorm and rexp to create some sample datasets

Normal.dataset.n30mu10s1 <- rnorm(30,10,1)  
Normal.dataset.n10mu10s1 <- rnorm(10,10,1)  
Normal.dataset.n30mu15s1 <- rnorm(30,15,1)  
  
Exponential.dataset.n30lambda1.0 <- rexp(30,1)  
Exponential.dataset.n30lambda0.5 <- rexp(30,.5)  
Exponential.dataset.n30lambda0.25 <- rexp(30,.25)

## Print the datasets, inlcude the first 10 records from each dataset

head(Normal.dataset.n30mu10s1, n= 10)

## [1] 9.846861 10.219528 7.470806 10.313265 10.524513 10.839622 10.328684  
## [8] 8.995837 10.242922 10.243880

head(Normal.dataset.n10mu10s1, n= 10)

## [1] 10.358554 10.939801 9.837444 11.321651 9.617259 10.020284 10.430956  
## [8] 12.669333 10.016772 10.973681

head(Normal.dataset.n30mu15s1, n= 10)

## [1] 15.25614 15.61198 16.35781 15.58329 17.32629 14.27892 15.76382  
## [8] 14.49370 15.90921 15.34560

head(Exponential.dataset.n30lambda1.0, n= 10)

## [1] 0.33974682 0.08608573 3.56117106 0.54700153 0.47522199 1.26253700  
## [7] 1.21728125 0.87540114 0.13899423 1.99566989

head(Exponential.dataset.n30lambda0.5, n= 10)

## [1] 0.05837207 1.07512710 0.13888295 0.04818014 2.36488169 1.61764789  
## [7] 0.87782402 0.26045594 0.18408790 0.58006724

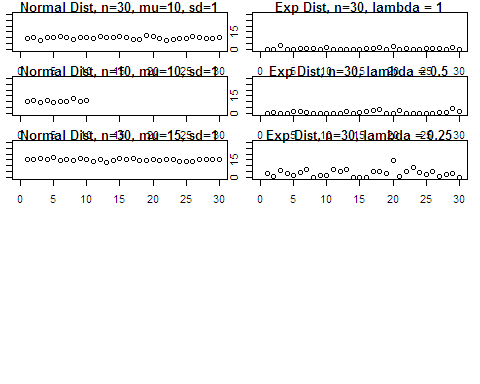
head(Exponential.dataset.n30lambda0.25, n= 10)

## [1] 3.5717112 0.8765647 6.2594288 3.6445118 1.8601272 4.1172647 7.4246006  
## [8] 0.7055461 1.9282375 2.2950124

## Explore the data with Plot of the normal and exponential sample datasets

Use plot and hist to compare the datasets set x and y limits to make it easier to compare plots

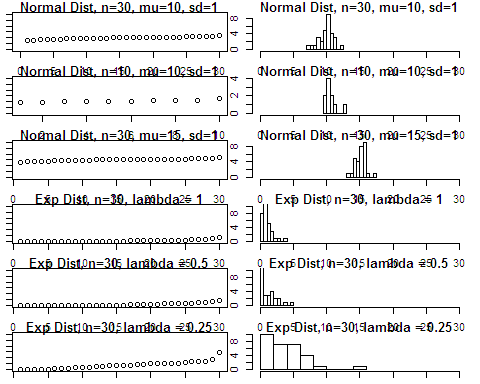
par(mfrow=c(6,2),mar=c(1,1,1,1))  
  
plot(Normal.dataset.n30mu10s1,xlim=c(0,30), ylim=c(0,30), main="Normal Dist, n=30, mu=10, sd=1")  
plot(Exponential.dataset.n30lambda1.0,xlim=c(0,30), ylim=c(0,30), main="Exp Dist, n=30, lambda = 1")  
  
plot(Normal.dataset.n10mu10s1,xlim=c(0,30), ylim=c(0,30), main="Normal Dist, n=10, mu=10, sd=1")  
plot(Exponential.dataset.n30lambda0.5,xlim=c(0,30), ylim=c(0,30), main="Exp Dist, n=30, lambda = 0.5")  
  
plot(Normal.dataset.n30mu15s1,xlim=c(0,30), ylim=c(0,30), main="Normal Dist, n=30, mu=15, sd=1")  
plot(Exponential.dataset.n30lambda0.25,xlim=c(0,30), ylim=c(0,30), main="Exp Dist, n=30, lambda = 0.25")



## Sort the records ascending and plot again. Include histograms

set x and y limits to make it easier to compare plots

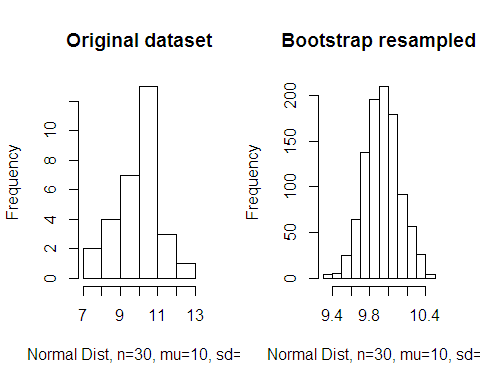
par(mfrow=c(6,2),mar=c(1,1,1,1))  
  
plot(sort(Normal.dataset.n30mu10s1), xlim=c(0,30), ylim=c(0,30), main="Normal Dist, n=30, mu=10, sd=1")  
hist(Normal.dataset.n30mu10s1,breaks = 10,xlim=c(0,30), main="Normal Dist, n=30, mu=10, sd=1")  
  
plot(sort(Normal.dataset.n10mu10s1), ylim=c(0,30), main="Normal Dist, n=10, mu=10, sd=1")  
hist(Normal.dataset.n10mu10s1,breaks = 10,xlim=c(0,30), main="Normal Dist, n=10, mu=10, sd=1")  
  
plot(sort(Normal.dataset.n30mu15s1), ylim=c(0,30), main="Normal Dist, n=30, mu=15, sd=1")  
hist(Normal.dataset.n30mu15s1,breaks = 10,xlim=c(0,30), main="Normal Dist, n=30, mu=15, sd=1")  
  
plot(sort(Exponential.dataset.n30lambda1.0), ylim=c(0,30), main="Exp Dist, n=30, lambda = 1")  
hist(Exponential.dataset.n30lambda1.0,breaks = 10,xlim=c(0,30), ylim=c(0,10), main="Exp Dist, n=30, lambda = 1")  
  
plot(sort(Exponential.dataset.n30lambda0.5), ylim=c(0,30), main="Exp Dist, n=30, lambda = 0.5")  
hist(Exponential.dataset.n30lambda0.5,breaks = 10, xlim=c(0,30),ylim=c(0,10), main="Exp Dist, n=30, lambda = 0.5")  
  
plot(sort(Exponential.dataset.n30lambda0.25), ylim=c(0,30), main="Exp Dist, n=30, lambda = 0.25")  
hist(Exponential.dataset.n30lambda0.25,breaks=10, xlim=c(0,30),ylim=c(0,10), main="Exp Dist, n=30, lambda = 0.25")

 ## Resample the datasets, compare original histogram to resampled histogram # notice that the resampled histograms more closely resemble a normal distribution, illustrating the central limit theorem

R <- 1000  
Normal.dataset.n30mu10s1.resample <- numeric(R)  
Normal.dataset.n10mu10s1.resample <- numeric(R)  
Normal.dataset.n30mu15s1.resample <- numeric(R)  
Exponential.dataset.n30lambda1.0.resample <- numeric(R)  
Exponential.dataset.n30lambda0.5.resample <- numeric(R)  
Exponential.dataset.n30lambda0.25.resample <- numeric(R)  
  
  
x=Normal.dataset.n30mu10s1  
mean(x)

## [1] 9.936169

for (i in 1:R) {  
 sample <- sample(x, size = length(x), replace = TRUE)  
 Normal.dataset.n30mu10s1.resample[i] = mean(sample)  
}  
  
par(mfrow = c(1,2))  
hist(Normal.dataset.n30mu10s1, main="Original dataset", xlab="Normal Dist, n=30, mu=10, sd=1")  
hist(Normal.dataset.n30mu10s1.resample, main="Bootstrap resampled", xlab="Normal Dist, n=30, mu=10, sd=1")



describe(Normal.dataset.n30mu10s1)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## 1 1 30 9.94 1.04 10.22 10 0.92 7.47 12.01 4.54 -0.52 0.04  
## se  
## 1 0.19

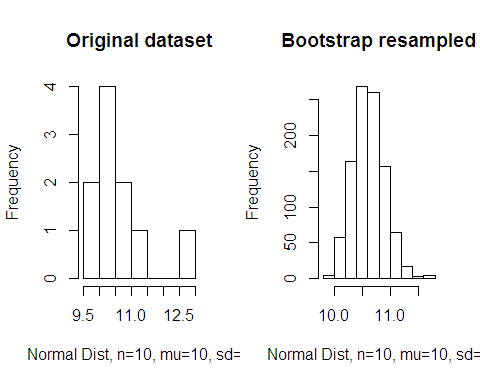
describe(Normal.dataset.n30mu10s1.resample)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## 1 1 1000 9.93 0.19 9.93 9.93 0.18 9.34 10.47 1.13 0.04 -0.08  
## se  
## 1 0.01

############################################################  
x=Normal.dataset.n10mu10s1  
mean(x)

## [1] 10.61857

for (i in 1:R) {  
 sample <- sample(x, size = length(x), replace = TRUE)  
 Normal.dataset.n10mu10s1.resample[i] = mean(sample)  
}  
  
par(mfrow = c(1,2))  
hist(Normal.dataset.n10mu10s1, main="Original dataset", xlab="Normal Dist, n=10, mu=10, sd=1")  
hist(Normal.dataset.n10mu10s1.resample, main="Bootstrap resampled", xlab="Normal Dist, n=10, mu=10, sd=1")



describe(Normal.dataset.n10mu10s1)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## 1 1 10 10.62 0.9 10.39 10.49 0.82 9.62 12.67 3.05 0.96 -0.07  
## se  
## 1 0.29

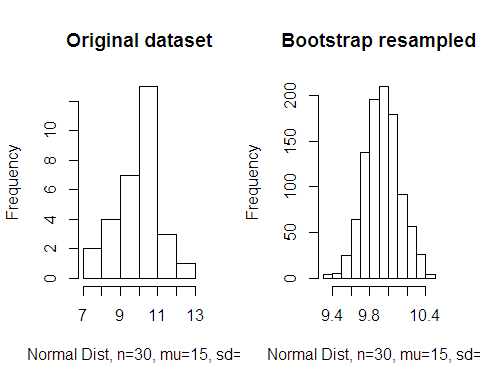
describe(Normal.dataset.n10mu10s1.resample)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## 1 1 1000 10.62 0.28 10.6 10.61 0.27 9.94 11.71 1.77 0.37 0.29  
## se  
## 1 0.01

################################################################  
x=Normal.dataset.n30mu15s1  
mean(x)

## [1] 15.1329

for (i in 1:R) {  
 sample <- sample(x, size = length(x), replace = TRUE)  
 Normal.dataset.n30mu15s1.resample[i] = mean(sample)  
}  
  
par(mfrow = c(1,2))  
hist(Normal.dataset.n30mu10s1, main="Original dataset", xlab ="Normal Dist, n=30, mu=15, sd=1")  
hist(Normal.dataset.n30mu10s1.resample, main="Bootstrap resampled", xlab ="Normal Dist, n=30, mu=15, sd=1")



describe(Normal.dataset.n30mu15s1)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## 1 1 30 15.13 0.84 15.17 15.14 0.75 13.26 17.33 4.07 0.05 0.2  
## se  
## 1 0.15

describe(Normal.dataset.n30mu15s1.resample)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## 1 1 1000 15.14 0.15 15.13 15.13 0.14 14.62 15.67 1.05 0.13 0.33  
## se  
## 1 0