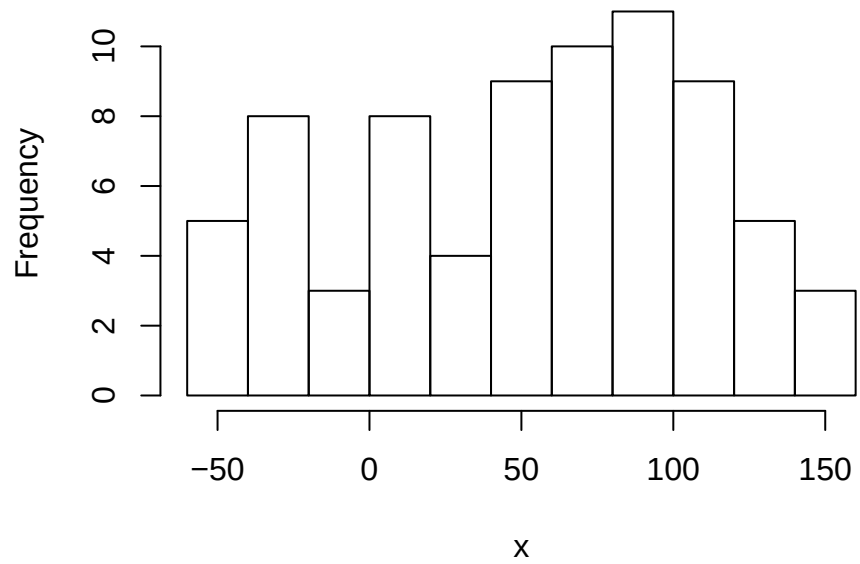
Lab 1 - density histograms and the CLT

This lab will introduce R Notebooks, R Markdown files, and knitR. You will use these formats to keep lab notes and for submitting online. These instructions don't cover these topics. Links to help with this are provided on canvas, and we will go over the basics in depth during lab. As with so many things in R, there is no single best way to do this, and it's just another language that you will need to pick up along the way. Fortunately, it's not as weird as R itself...

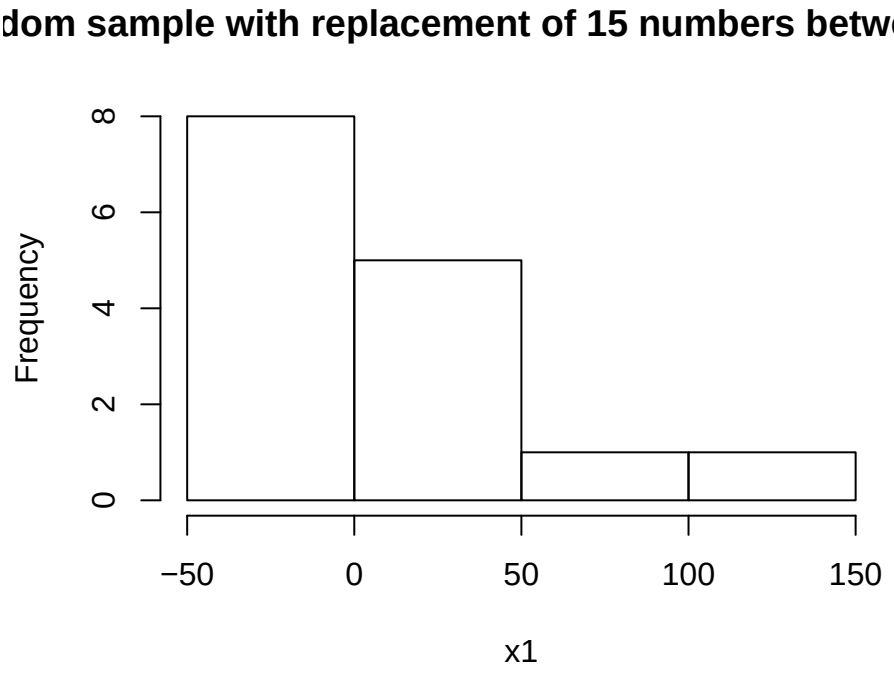
One funny quirk about using R Markdown files and knitR is that the Markdown environment is not directly aware of the R environment (workspace) in yoru active sessions. That means the Markdown file can't see the data you've imported, functions you've written, variales you've named etc. You need to write all the code for replicating that into the markdown file. This is intentional. The markdown files are meant to be used for reproducible work. That is, the document is self-contained by design so that someone else can recreate exactly what you've done using the exact same data. It's a pretty neat feature once you get over the frustration of rectifying the disconnect between your active environment and the R Markdown environment. To make this transition in thinking a bit simpler, we'll write code this week to generate all our data (no importing of csv files yet).

OK, let's learn how to plot histograms using density curves. Density curves are useful as they represent pdfs. Empirical pdfs can be created and plotted from data, and it's important to realize that they are the same thing as the standard block based histogram; the y-axis is rescaled, and the stairsteps are smoothed with a curve. OK, let's make up a datset and take a look:

```
set.seed(1000)
x <- sample(-50:150,75,TRUE) #TRUE is with replacement
x1 <- sample(-50:150,15,TRUE) #smaller sample size for comparison

hist(x,main="random sample with replacement of 75 numbers between -50:150")
```



```
hist(x1,main="random sample with replacement of 15 numbers between -50:150")
```



Question 1

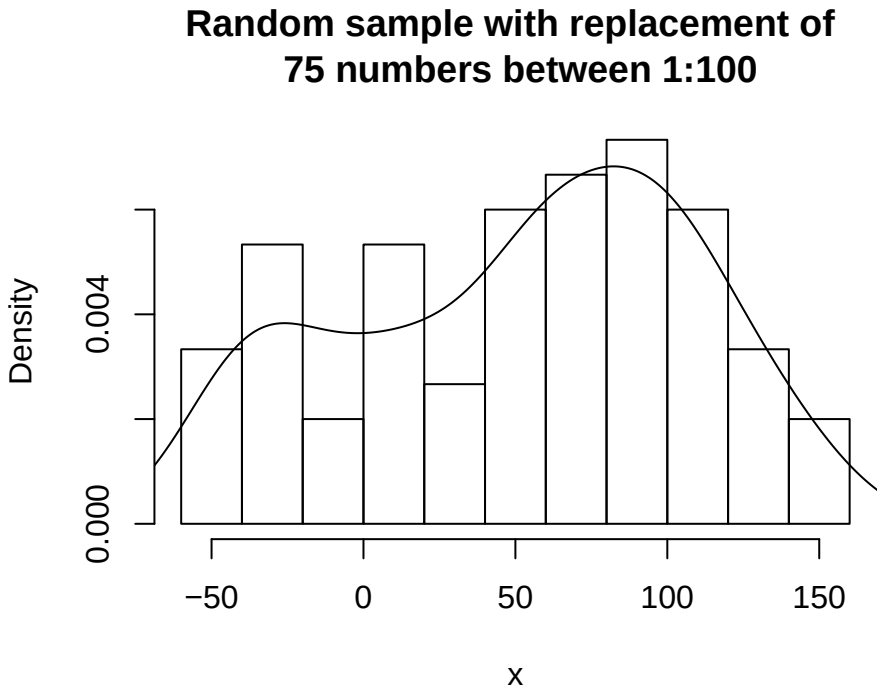
a. What is the scale for the random variable x that we just created?
   - This random variable is in the interval scale (-50 to 150).

b. What is the distriubtion for the random variable x? Hint: look at the code that generated the distribution and think about probabilites of sampling across the domain.
   - The mean of random variable x should be a normal distribution around 50. There is an equal probability of sampling across the domain of -50 to 150.

c. Why does the observed distribution look different than what you might expect for the distribution you named in b?
   - The observed distribution comprise the samples taken, not the mean of the values sampled. The samples will be somewhat flattly distributed within the sample space.

Notice the y-axis label. It tells us the number of times we recorded a number in the interval defined by the x-extent of the histogram bars. We can rescale the y-axis such that the histogram represnets a proper probability distribution. We don't actually have to rescale manually, as R kows this is something that data analysts commonly do - we just set the freq argument to F and we'll get density equivalency on the y-axis. Once we rescale the y-axis, we can add the smoothed density curve.

```
hist(x,main="Random sample with replacement of \n 75 numbers between 1:100",freq=F) # \n insets a carri
lines(density(x))
```

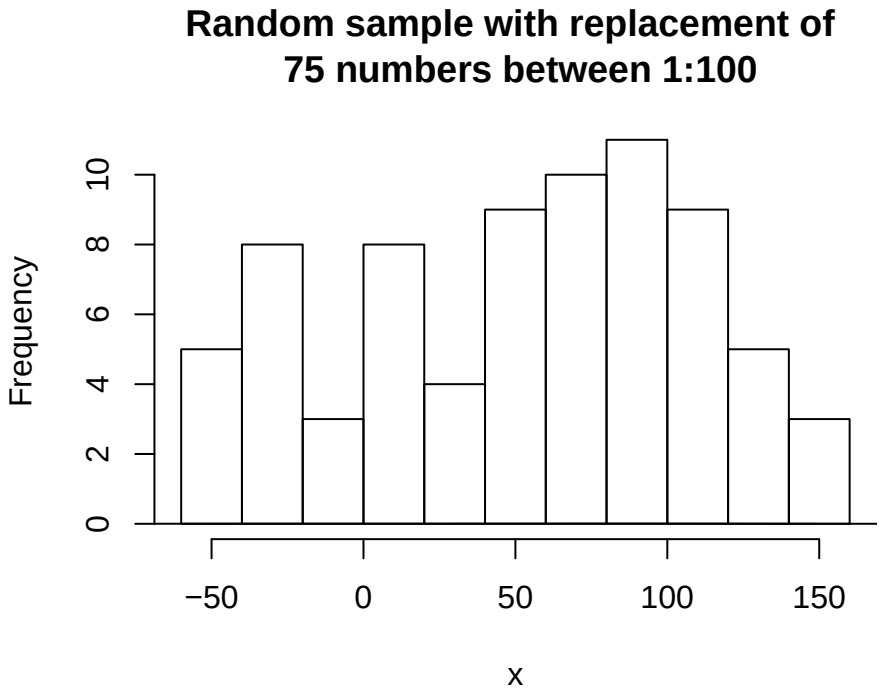


Question 2

a. How does changing the y-axis scale from the frequency of observations to "density" give us a probability distribution? Hint: Think about the definition of a probability distribution, and the area of a rectangle.
   - The density curve is the probability of the sample where the area under the curve equals all probabilities. While the frequency shows the number of times a number within that bin was sampled, the density curve shows the fraction of times a value was sampled divided by the total samples to show the probability of that sample.

b. What do you think would happen if we tried to add the density curve without first rescalign the y-axis? Hint: write a little code to try it!!
   - The frequency y-axis is 0-75 (although not likely all 75 samples will be the same value) and the density y-axis is 0-1. These scales are too far off to visually see on one y-axis

```
hist(x,main="Random sample with replacement of \n 75 numbers between 1:100",freq=TRUE) # \n insets a ca
lines(density(x))
```



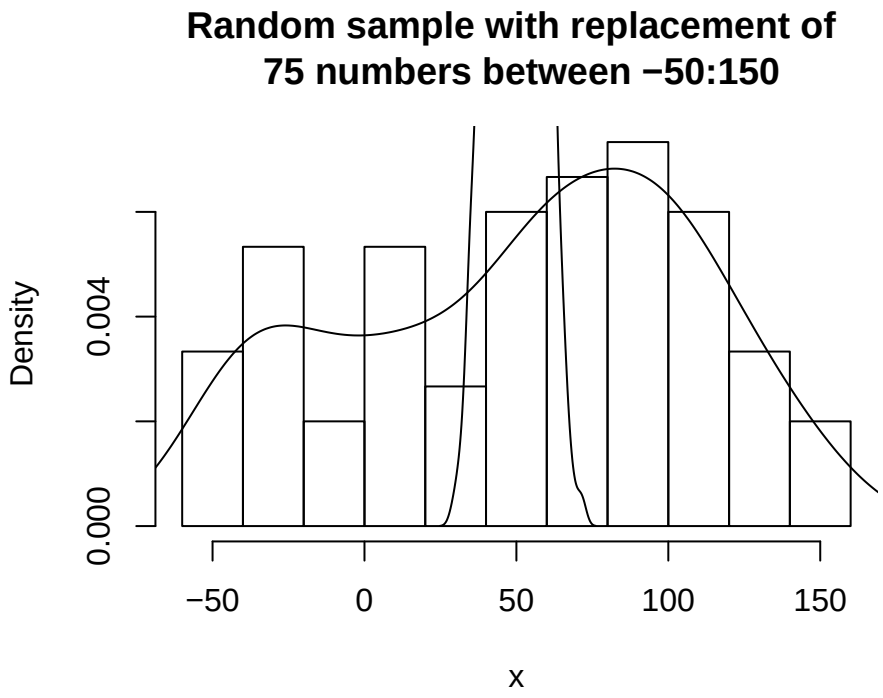
OK, what a weird distribution. Now let's invoke the CLT (Central Limit Theorem) by generating a sampling distribution for the mean, estimated from n=75 data points. The book gives nice examples for looping to do this kind of thing, but here we'll demonstrate a slightly quicker way of achieving the same end, and in the process pick up some skills for data management and organization. Our aim is to generate means for each of 1000 replicated samples of 75 random draws from the set -50:150. Let's make a huge matrix where each column is a replicate, and each row is a sample. This will give us 75x1000 cells, organized such that each set of 75 is a different sample. The trick here comes because we're taking random samples (so the initial organization is arbitrary). Once organized, we can take the column-wise means for plotting. Let's have a look:

```
reps <- 1000
n <- 75
n1 <- 15
pop <- -50:150
set.seed(200)
xMat <- matrix(sample(pop,reps*n,T),nrow=n)
xMat1 <- matrix(sample(pop,reps*n1,T),nrow=n1)

samplingDist <- apply(xMat,2,mean) # I am applying a function to the 2nd dimension (columns) of the mat
samplingDist1 <- apply(xMat1,2,mean)

length(samplingDist) # make sure I have the expected number of means (= to reps above)
```

```
## [1] 1000
```

OK, now let's add the density curve for the sampling distribution of the mean for n=75 to the histogram of our single sample. Markdown environments will hang on to dataframes and named objects and so on, but the code is interpreted sequentially, so if we try to add lines to an existing plot, that plot must have been rendered in the same code chunk. Let's recall it here.

```r
hist(x,main="Random sample with replacement of \n 75 numbers between -50:150",freq=F,cex=0.5)
lines(density(x))
lines(density(samplingDist))
```



Question 3

a. What happened to the top of the curve for the sampling distribution of the mean?

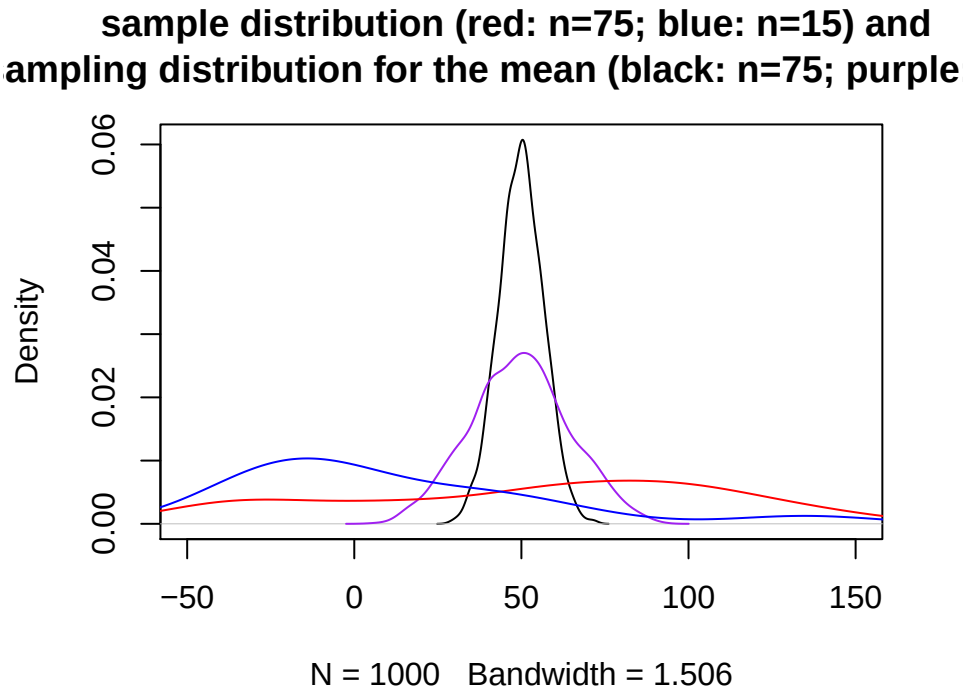
- The y-axis is set by the first plot so the normal distribution is off the scale of that y-axis

b. Yes, it was truncated. Bravo. Now, reason from what you know about the properties of probability distriubtions to explain why it was truncated.

- Sampling 1000 times appraoches the normal distribution while one sample of 75 with replacement will not explore the space enough for a normal distribution. As the sample size increases, this time to 1000 sample means, the probability will tend towards the the true average of the range.

To fix the truncation problem, we just need to specify the plotting parameters in a slightly differnet order.

```r
plot(density(samplingDist), xlim=c(min(pop),max(pop)),main="sample distribution (red: n=75; blue: n=15)
lines(density(samplingDist1), xlim=c(min(pop), max(pop)), col="purple")
lines(density(x), col="red")
lines(density(x1), col="blue")
```

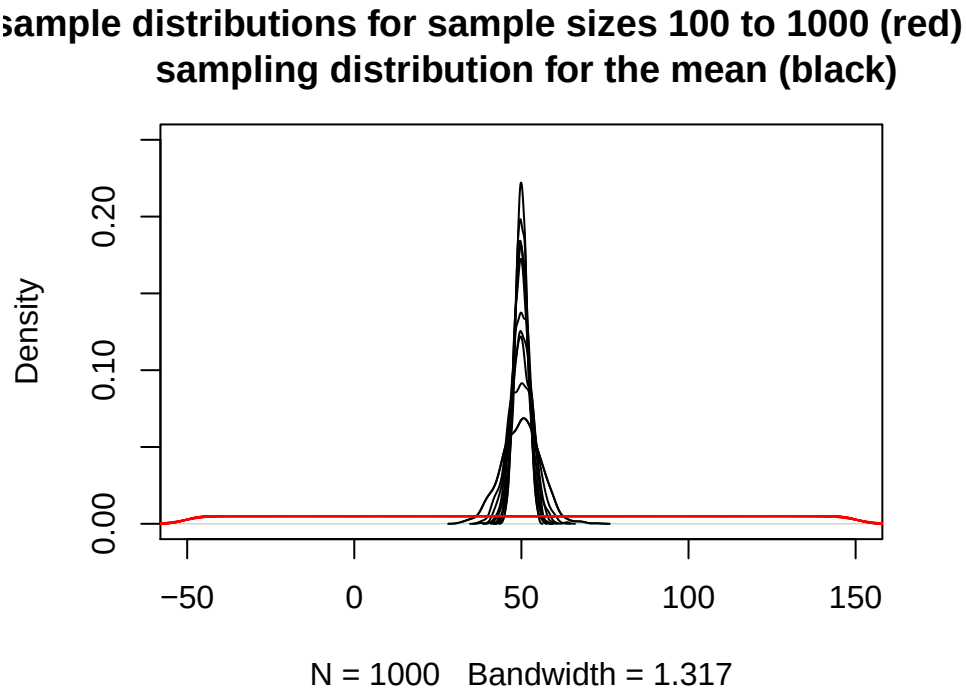


Question 4

a. Why is the sampling distribution of the mean narrower and taller than the distribution for a single sample of n=75? Will this same relationship always hold between the two kinds of ditributions? Explain.

- The average of each sample will be around the middle so the distribution of a sample of means will be near the mean value. The density curve of a set of samples will always be much wider and will not approach a normal distribution even as the sample size is increased.

```
reps <- 1000
n <- seq(100,1000, by=100)
pop <- -50:150
set.seed(200)
xMat <- lapply(n, function(x){
  matrix(sample(pop,reps*x,T),nrow=x)
})
samplingDist <- lapply(xMat, function(xMatnow) apply(xMatnow,2,mean))

plot(density(samplingDist[[1]]), xlim=c(min(pop),max(pop)),ylim=c(0,.25),
     main="sample distributions for sample sizes 100 to 1000 (red) and \n sampling distribution for the
for(i in 1:length(n)){
  lines(density(samplingDist[[i]]), xlim=c(min(pop), max(pop)))
  lines(density(xMat[[i]]), col="red")
  }
```



b. Why is the sampling distriubtion for the mean more bell-shaped than the data distriubtion?

- The data distribution is relatively flat with equal likelihood of sampling anywhere within the sample space. The dsitribution of the mean will always tend towards the true mean of the sample space.

c. Do you think a sampling distriubtion for the mean using n=15 observations will be wider or narrower than the sampling distribution plotted here (based on n=75)? Explain. (Hint: it's OK to add that one to the same plot to help your explaination!!)

- Fewer observations result in a wider sampling distribution due to the skew of the mean with too small of a sample size.