The goal of this project is to get an understanding of random samples and sampling distribution of statistics. As well, we learn some R commands along the way.

1 Obtaining and visualizing random samples

R provides some 15 or so distributions built-in. These include the normal, exponential, uniform, binomial etc. For a distribution we may want to know any of the following:

- Its p.d.f. either $f(k) = P(X = k)$ or $f(x)$ the density.
- Its c.d.f. $F(a) = P(X \leq a)$
- quantiles: Solving for a so than $x = P(X \leq a)$.
- random samples: sampling from a distribution.

In R these are handled by p, d, q, and r functions. The basic idea is each distribution has a family name (norm for normal, exp for exponential, unif for uniform, ...) and to get one of these functions we add p,d,q or r to the front.

For example. For the standard normal we have the density at 1

```
> pnorm(1)
```

```
[1] 0.8413447
```

The area to the left of 1

```
> dnorm(1)
```

```
[1] 0.2419707
```

The value a for which 90% of area is to left

```
> qnorm(0.9)
```

```
[1] 1.281552
```

A random sample of size 10

```
> rnorm(10)

 [1]  0.7267889  0.4366413 -0.7094398 -0.7531484  1.4492867  1.4694527
 [7] -0.2875291  1.0679362  0.2645473 -2.4907817
```

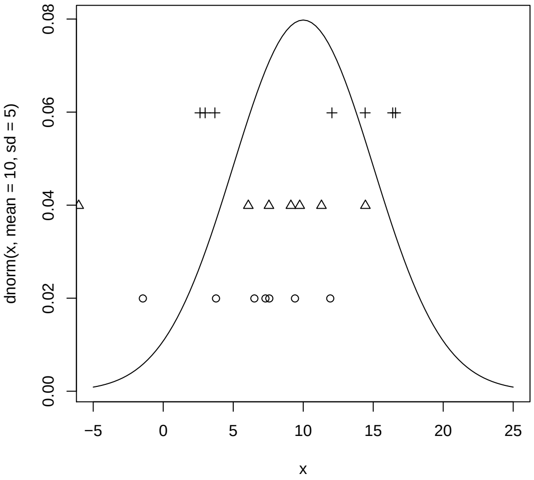
For each family there are parameters. The normal has mean and standard deviation. The defaults are 0 and 1, but to get different values requires us to look at the arguments:

```
> rnorm(5, mean = 100, sd = 15)

[1] 110.03883  88.01936 121.34655  86.60507  82.49320
```

To visualize a sample we can plot the density, and then plot the sample. The density is plotted with the `curve()` function. A single sample can be added with the `rug()` function, or added one at a time using `points()`. For instance, to visualize 3 samples of size 7 from the normal distribution with mean 10, and sd. 5 we have

```
> curve(dnorm(x, mean = 10, sd = 5), -5, 25)
> m = dnorm(10, mean = 10, sd = 5)
> for (i in 1:3) {
+     x = rnorm(7, mean = 10, sd = 5)
+     points(x, m * i/4 + 0 * x, pch = i)
+ }
```



Problem: The exponential distribution has a parameter `lambda=` which is the reciprocal of the mean. Make a graph of the exponential with mean 5. Layer on 3 random samples of size 10.

2 Sampling distributions

We saw in class that the sampling distribution of a statistic can be simulated. We will do this here to get an understanding of the central limit theorem, and to get insight into the sampling distribution of other statistics.

First, the steps we take to investigate the sampling distribution of a statististic:

1. Decide on the parent population.
2. Decide on the size of the sample and the statistic to summarize it
3. Figure out how to compute a single value of the statistics based on a sample
4. Repeat this using a for loop to get enough independent realizations of our statistic that we can understand its sampling distribution.

To illustrate, suppose we see if $n = 10$ is large enough to say the central limit theorem has kicked in when the population is uniform on $[0, 2]$.

The statistic is the mean, and the parent pouplation is uniform. A single realization of the sample mean is found with

```
> mean(runif(10, 0, 2))
```

```
[1] 0.9476854
```

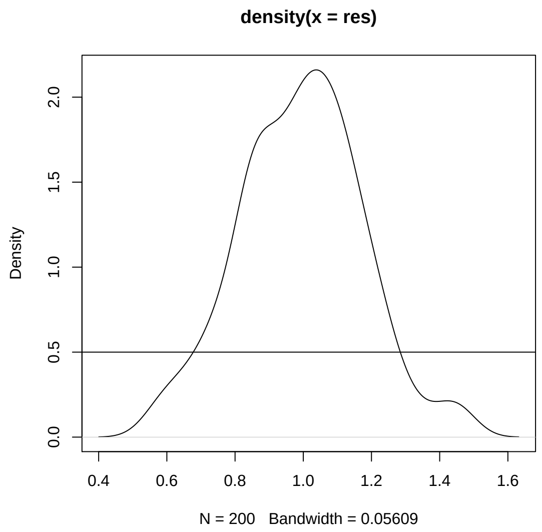
To gather a bunch, say 200, we define a place to store them, and then run a for loop

```
> res = c()
> for (i in 1:200) res[i] = mean(runif(10, 0, 2))
```

Now to visualize what we have. There are lots of ways to do so:

1. With a density

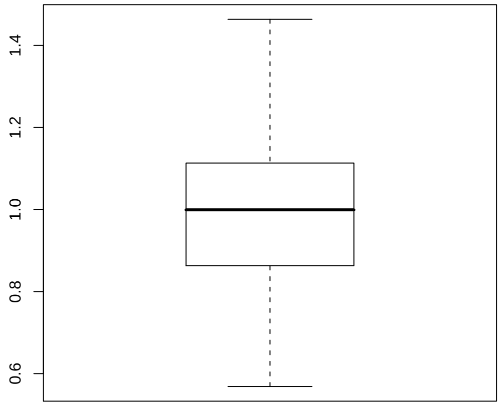
```
> plot(density(res))
> curve(dunif(x, 0, 2), add = TRUE)
```



(I added the original density using `curve()`)

2. With a boxplot to check for skew/symmetry, length of tails

```
> boxplot(res)
```



3. q-q plot to check normality

   ```
   > qqnorm(res)
   ```

What do you think? Does the sampling distribution of the sample mean look normal in this case? More or less. For symmetric, short-tailed distributions the normality of the sample mean happens quite quickly.

Problem: Repeat the above using random samples of size 10 from the exponential distribution with mean 2. (`rexp(10,rate=1/2)`). Does the sample mean look normally distributed?

Problem: Repeat with samples of size 50. Does the sample mean look normally distributed?

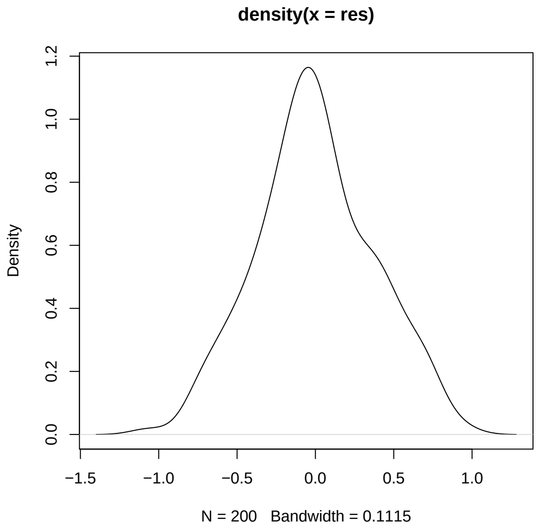
Problem: The log normal distribution is very skewed. For random samples of size 100 is the sample mean normally distributed when `rlnorm(100, meanlog=3)` is used for the random sample? Explain why or why not.

Problem: The cauchy distribution has really long tails. The central limit theorem does not apply. Verify that there are issues using samples of size 50 (`rcauchy(50)`). What are the issues?

Does the sample median have a normal distribution? In class we computed the density of the median, but from that it is hard to tell. Let's check by simulation.

For a normal population we check with samples of size 10:

```
> res = c()
> for (i in 1:200) res[i] = median(rnorm(10))
> plot(density(res))
```



Maybe. What about for other populations?

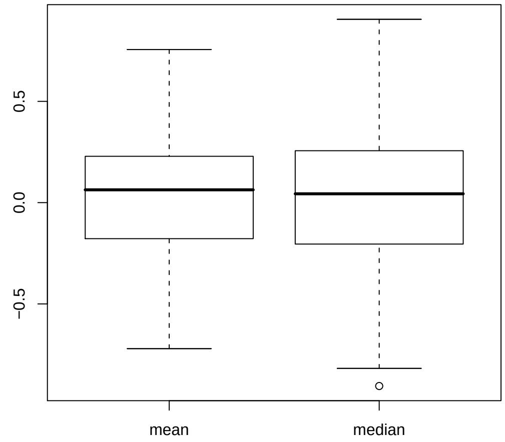
Problem: Let $n = 10$ and the population be exponential with mean 1

Problem: Let $n = 100$ and the population be exponential with mean 1

Problem: Is the mean of the sample median the population mean or the population median?

Comparing distributions of different statistics is informative. Let's compare the median and mean for normal samples.

```
> res.mean = c()
> res.median = c()
> for (i in 1:200) {
+     x = rnorm(10)
+     res.mean[i] = mean(x)
+     res.median[i] = median(x)
+ }
> boxplot(list(mean = res.mean, median = res.median))
```



The side-by-side boxplots show both are centered at the same place, but the sample mean has less spread. This is good, a single realization of the sample mean is likely to be "closer" to 0 than the sample median.

Problem: Repeat the above, only use an exponential random sample with mean 2 (`rexp(10,1/2)`).

Problem: Repeat with a long tailed population (`rlnorm(10,meanlog=3)`).

The central limit theorem says that

$$Z_n = \frac{\bar{X}_n - \mu}{\sigma/\sqrt{n}}$$

has an asymptotic standard normal distribution. In many applications, we actually have the related, but different, random variable

$$T_n = \frac{\bar{X}_n - \mu}{S/\sqrt{n}}$$

where S is the sample standard deviation (σ is the population one).

Problem: For $n = 100$ does the distribution of T_n look normal?

Problem: For $n = 20$ does the distribution of T_n look normal?

Problem: For $n = 4$ does the distribution of T_n look normal?

Computer tip for using R. It is sometimes easier to use an editor to edit R commands, rather than edit at the command line, in analog to using MATLAB with *m*-files. There are two convenient ways to do this in R.

Using .R files If you store your commands (no prompts) in a file called, say, `file.R`, then you can "source" the contents of the file line by line by using the File::Source menu item. Alternatively, you can run the command

```
> source(file.choose())
```

and then select the file you want to read in line by line.

Using a function Suppose you define a function for doing a simulation. For example

```
> sim = function(n) {
+     res = c()
+     for (i in 1:200) res[i] = mean(rnorm(n))
+     res
+ }
```

(This is a template, ask me about input values, return values if you like doing this.)

This function is run with a command like

```
> theresults = sim(10)
```

Suppose you want to edit this function. For instance, to change the parent population, or the statistic. Either of these two commands will open Notebook and allow you make changes to the template.

```
> sim = edit(sim)
```

or, the shorter but harder to remember

```
> fix(sim)
```