# Central Limit Theorem Visualized

Casey Moroney

11/17/2022

## Intro

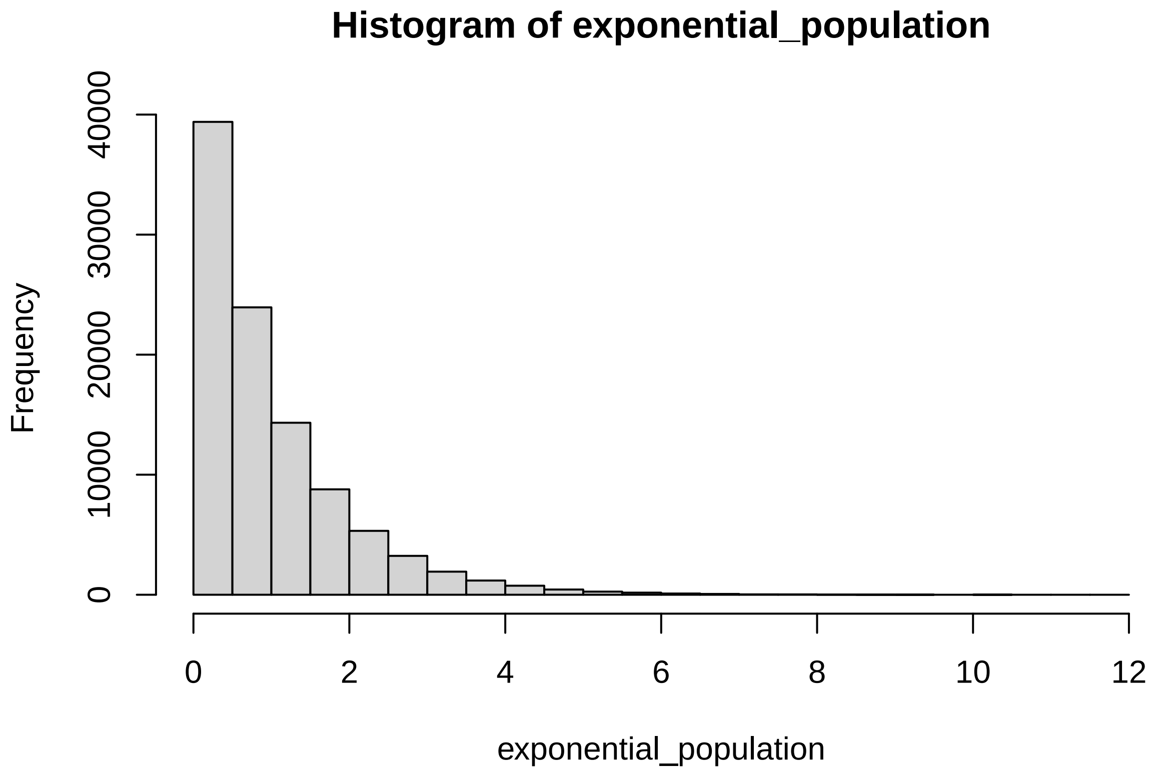
The central limit theorem is a primary concept in probability theory that has applications in many areas of statistics.

> The central limit theorem states that if you have a population with mean $\mu$ and standard deviation $\sigma$ and take sufficiently large random samples from the population with replacement, then the distribution of the sample means will be approximately normally distributed. This will hold true regardless of whether the source population is normal or skewed, provided the sample size is sufficiently large (usually n > 30). (LaMorte, 2016)

Put simply, it means that if you take a large enough sample size you can use assumptions from the normal distribution to make inferences, even if the population you are sampling from is not normally distributed.

We can visualize this using simulations. Let's generate a sample from an exponential distribution.

```r
exponential_population <- rexp(100000)

# Create histogram
hist(exponential_population)
```



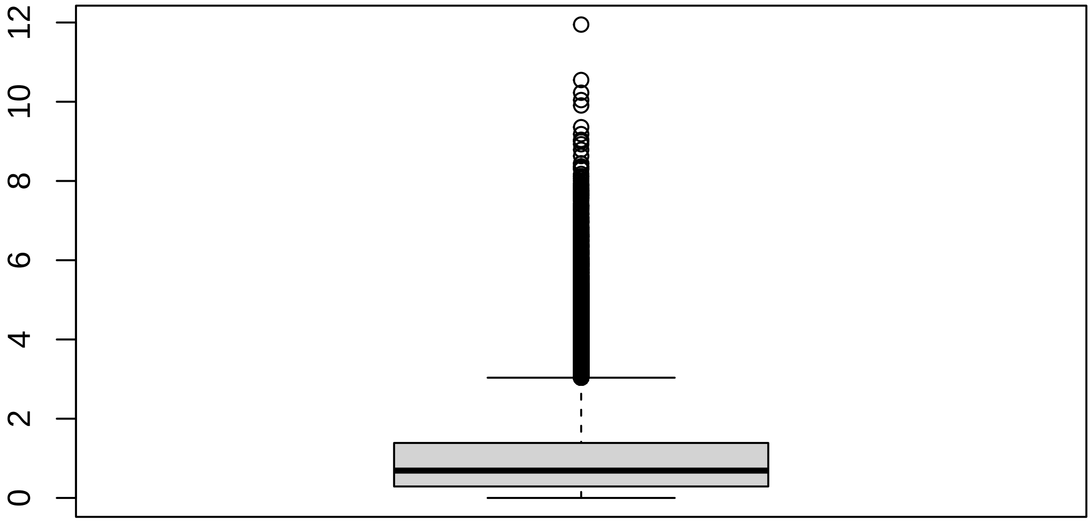
```
# View the first 100 values
exponential_population[1:100]
```

```
##   [1] 0.93573749 0.05261421 1.18102811 1.22980856 0.34399910 0.52564489
##   [7] 0.08368051 0.35414520 0.59480416 0.52095983 0.68389481 1.33613542
##  [13] 1.09380669 0.05953209 2.11425605 0.83554934 0.28144654 0.87104041
##  [19] 0.73485567 0.33976451 2.06698739 1.34751969 0.31568739 0.56511922
##  [25] 1.51660029 1.48577928 0.46351059 0.26052170 1.62332860 1.53503534
##  [31] 0.27605129 0.42711631 0.50988923 0.05083895 0.52912187 0.56451549
##  [37] 0.40815090 1.30761291 0.81125717 2.34587708 0.51407080 0.16060543
##  [43] 0.10363084 0.72391243 0.30593316 1.09019146 0.37462331 0.80479737
##  [49] 1.50085217 0.28272848 0.17444082 0.33330527 0.37203369 0.62319325
##  [55] 0.27043222 0.53592943 0.84440638 2.60926167 0.27504692 0.09944014
##  [61] 0.58770711 1.98645467 0.27477303 1.52513069 0.51614307 0.26252911
##  [67] 0.89408020 0.26714943 1.88447777 3.65349110 2.56838630 0.39301870
##  [73] 0.31919184 0.01912128 0.02663364 0.78699168 1.41197590 0.04771593
##  [79] 0.78259558 1.79195838 1.99513650 1.19526964 1.35319994 2.77893499
##  [85] 1.28526416 0.94675372 0.25076402 0.75872007 0.91289220 0.21981512
##  [91] 0.45723873 4.05377733 1.98975633 0.63160238 0.40801641 1.60900366
##  [97] 0.72174328 3.48828207 1.57214873 2.18081089
```

```
# Summary stats
summary(exponential_population)
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.2892  0.6894  1.0001  1.3876 11.9474
```

```
boxplot(exponential_population)
```



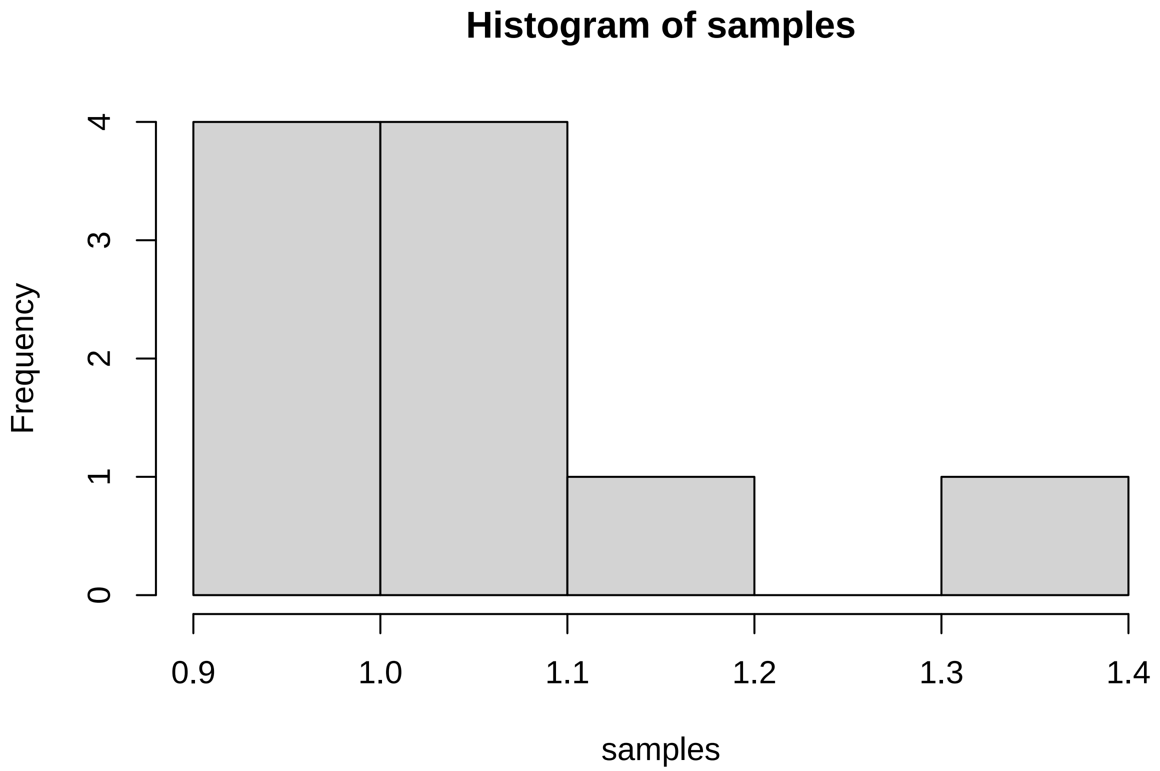
We can see that the distribution has a long tail, mean of 1, and median 0.69.

According to the CLT if we take repeated samples with replacement from this population, the sample means will be normally distributed (provided the sample size is sufficient). We will start with samples of 100 random observations repeated 10 times then observe the sample means.

```
samples <- array()

for(i in 1:10){
  s <- sample(exponential_population, 100)
  mu <- mean(s)
  samples[i] <- mu
}

samples
```

```
##  [1] 1.3014075 1.0842214 1.0107146 0.9301292 1.0382455 0.9267549 1.0841420
##  [8] 1.1743088 0.9438751 0.9438515
```

```
hist(samples)
```

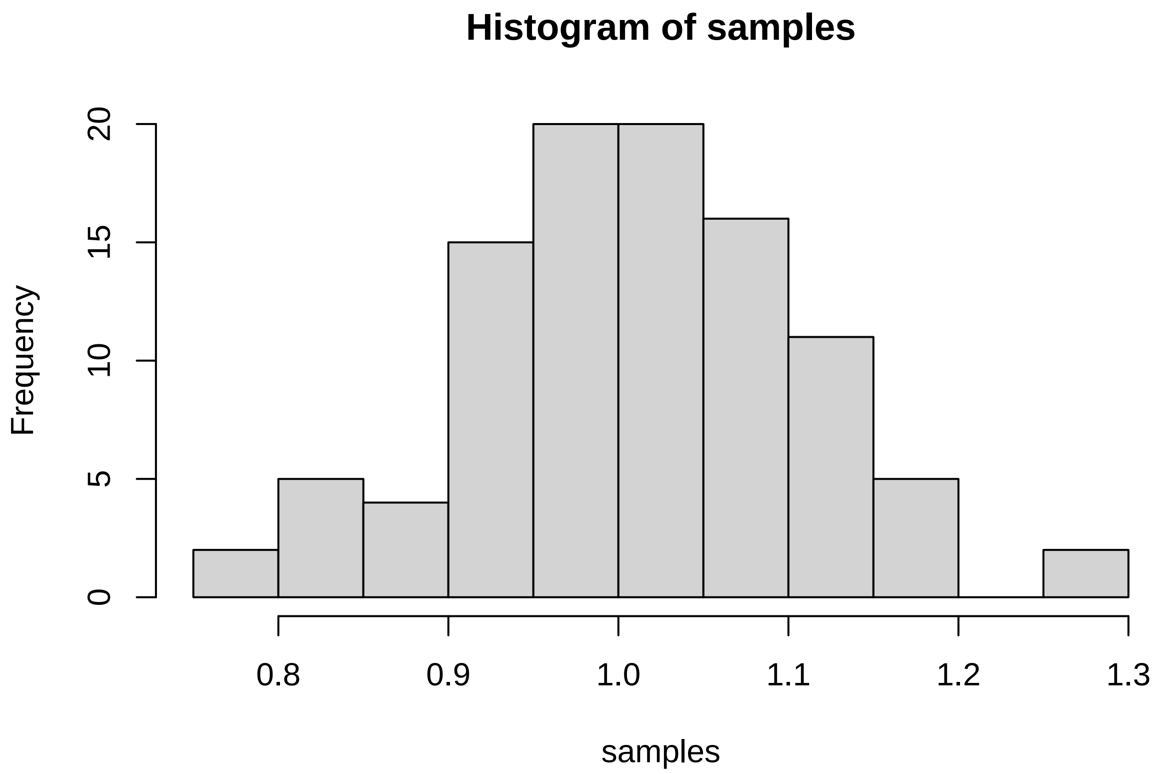


The distribution looks more normal than exponantial, but not quite there yet. In order for CLT to hold, we need to sample the population a *sufficiently large* number of times. Let's try 100 samples:

```
samples <- array()

for(i in 1:100){
  s <- sample(exponential_population, 100)
  mu <- mean(s)
  samples[i] <- mu
}

samples
```

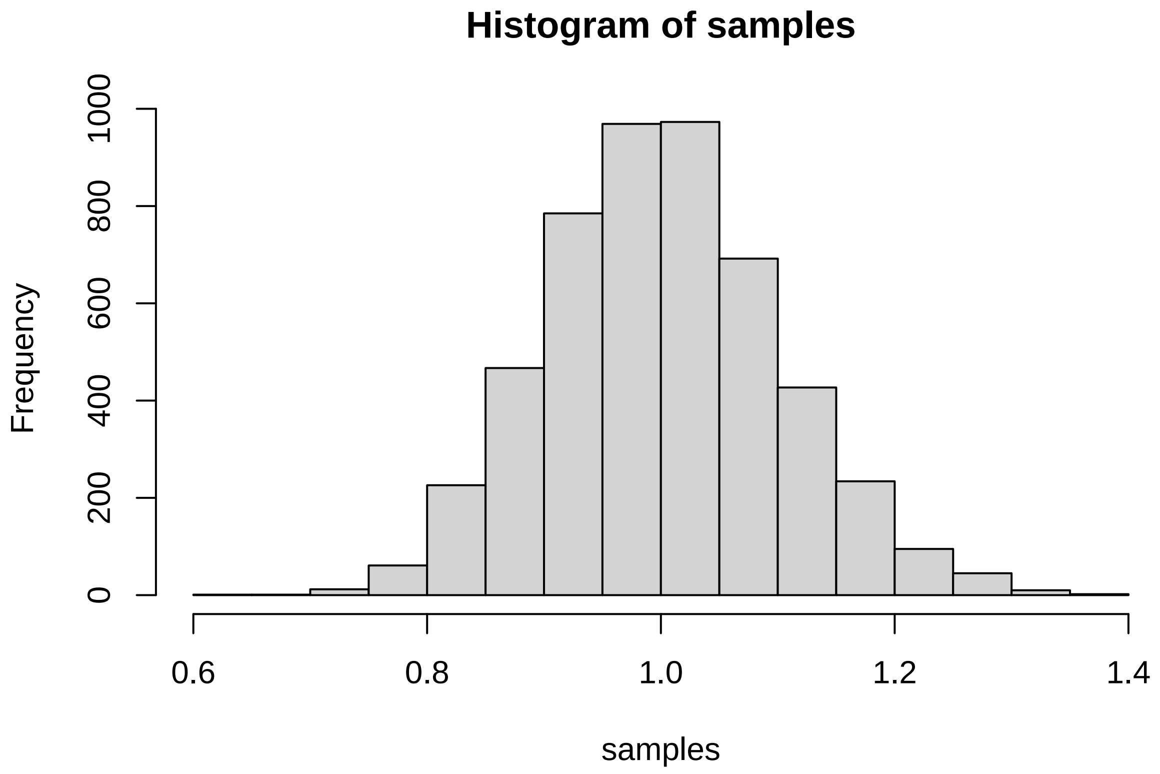
```
##   [1] 0.9603416 1.1378968 1.1185357 1.1166004 1.0886954 1.1150306 0.9028928
##   [8] 1.0551185 0.8122892 1.0442601 1.0662496 1.0728351 0.9557201 1.0790821
##  [15] 1.0095423 1.1547999 0.9202079 0.9569813 1.1278200 0.9517238 0.8927711
##  [22] 1.0795240 0.9680785 1.0136063 1.0429682 0.9548684 0.9497296 0.9148600
##  [29] 1.0337999 1.1521946 1.1998531 0.9588066 0.9068841 1.0142575 0.9454523
##  [36] 0.9523233 1.0031265 0.9964700 1.2510790 0.9594864 1.2621443 0.9391042
##  [43] 0.8991163 0.9720940 1.0023764 0.9220364 0.9016103 0.9958548 1.1356525
##  [50] 0.9643305 0.9930138 1.1190708 0.9850899 0.9330061 0.8914593 0.9029132
##  [57] 1.0077065 1.0034773 1.0568841 1.0936087 0.9722506 0.8825592 1.0169481
##  [64] 1.1217710 0.9459382 0.9653890 1.0777676 0.9070901 1.1228233 1.0525360
##  [71] 0.9663314 1.0229554 0.8361950 0.9640093 1.0283022 1.0335396 1.0113509
##  [78] 0.9153720 1.1491498 1.0131792 0.8340246 0.9204756 1.1264835 0.8449194
##  [85] 1.0022170 1.0796258 1.0657139 1.0537668 1.0648282 1.0309908 1.0133978
##  [92] 1.1704988 1.1528653 0.8247455 0.9877440 0.7989949 1.0108020 0.7977562
##  [99] 1.0874089 1.0847119
```

```
hist(samples)
```



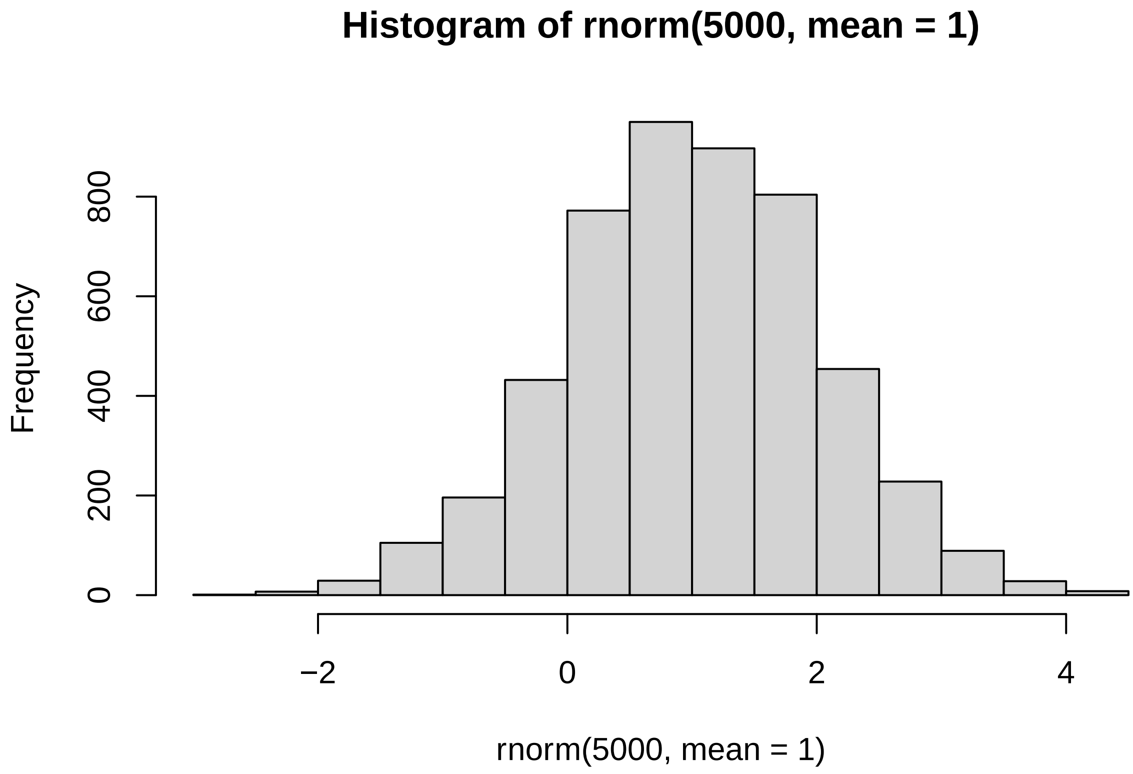
This looks even moreso normal, with tails being much smaller and the center of the distribution having more definition. Let's try 100,000:

```
samples <- array()

for(i in 1:5000){
  s <- sample(exponential_population, 100)
  mu <- mean(s)
  samples[i] <- mu
}

hist(samples)
```



For comparison, let's plot samples from a normal distribution:

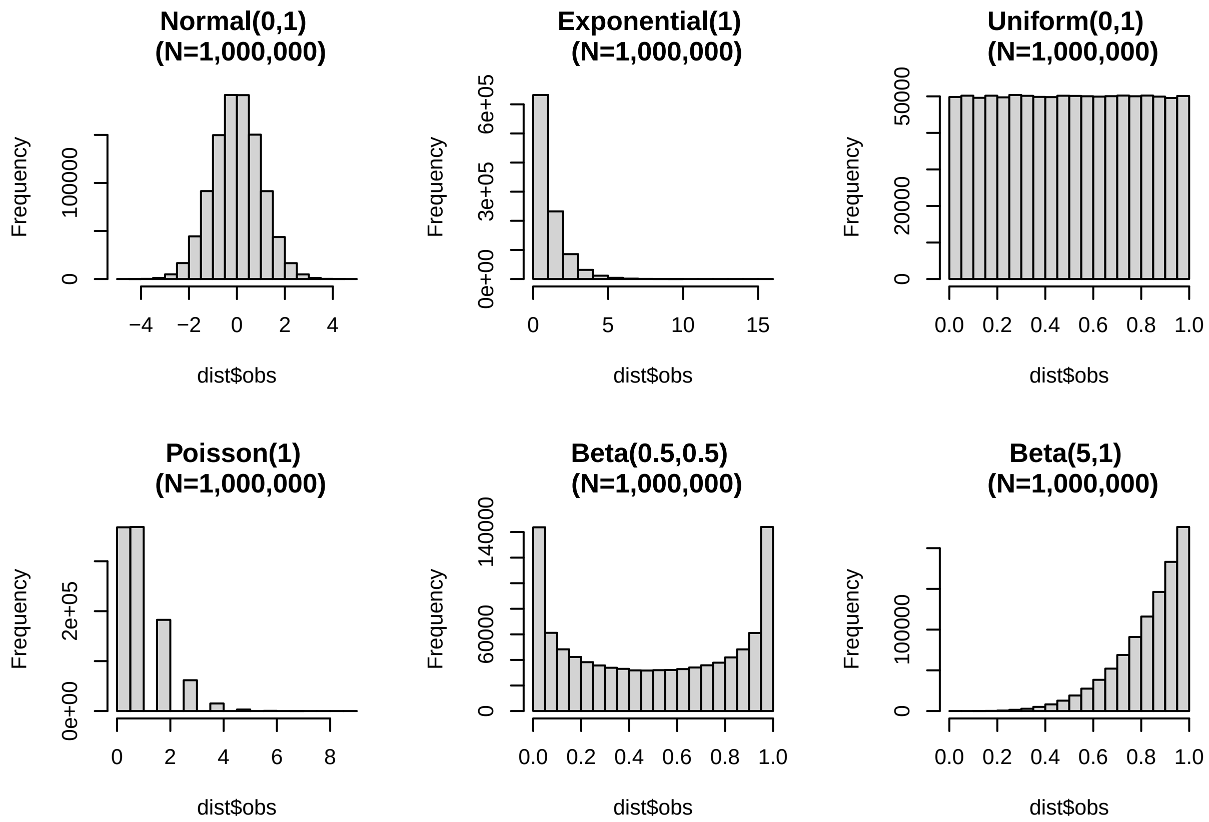
```
hist(rnorm(5000, mean=1))
```



As mentioned, this works for any distribution. To demonstrate, let's simulate data from other distributions and visualize.

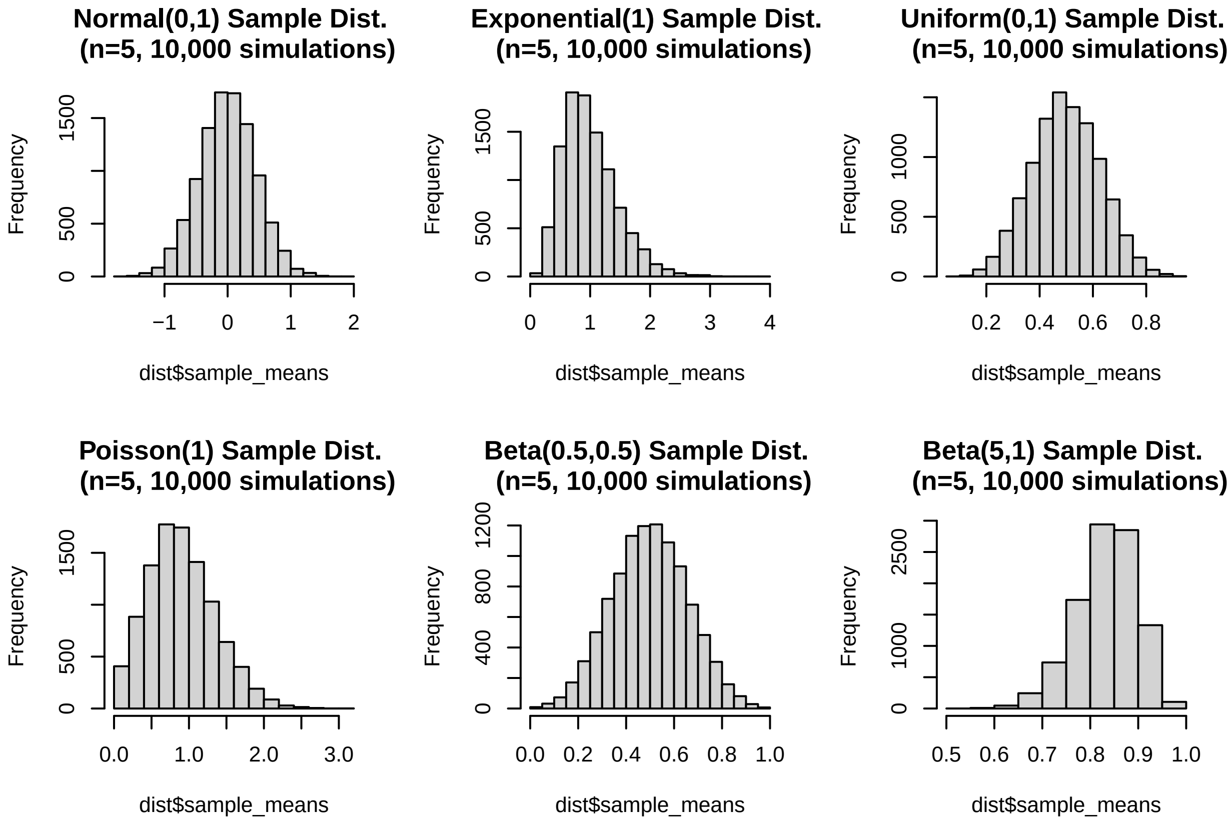
```
# Function for simulating data from a given distribution
generate_dist <- function(name, sample_func, n, ss=0.1*n, sims){
  dist <- {}
  dist$name <- name
  dist$obs <- sample_func(n)
  dist$ss <- ss
  dist$sample_means <- array()

  for(i in seq(1:sims)){
    s <- sample(dist$obs, ss, replace=TRUE)
    mu <- mean(s)
    dist$sample_means[i] <- mu
  }

  return(dist)
}

# Modified sample functions to set default parameters
rpois_mod <- function(n){return(rpois(n=n, lambda=1))}
beta_05_05 <- function(n){return(rbeta(n=n, shape1=0.5, shape2=0.5))}
beta_5_1 <- function(n){return(rbeta(n=n, shape1=5, shape2=1))}

# Function for generating a list containing data from several distributions
```

```
generate_dist_list <- function(n, ss, sims){
  list_out <- list(
    generate_dist("Normal(0,1)", rnorm, n, ss, sims),
    generate_dist("Exponential(1)", rexp, n, ss, sims),
    generate_dist("Uniform(0,1)", runif, n, ss, sims),
    generate_dist("Poisson(1)", rpois_mod, n, ss, sims),
    generate_dist("Beta(0.5,0.5)", beta_05_05, n, ss, sims),
    generate_dist("Beta(5,1)", beta_5_1, n, ss, sims)
  )
  return(list_out)
}

# Create distribution list
dist_list <- generate_dist_list(1000000, ss=50, sims=10000)

# Set up and iterate through data list to plot
par(mfrow=c(2,3))
for(dist in dist_list){
  n_size <- format(length(dist$obs), big.mark=",")
  hist(dist$obs, main=paste0(dist$name, " \n (N=", n_size, ")"))
}
```



What happens to sample mean distributions if we use a small sample size (n=5)?

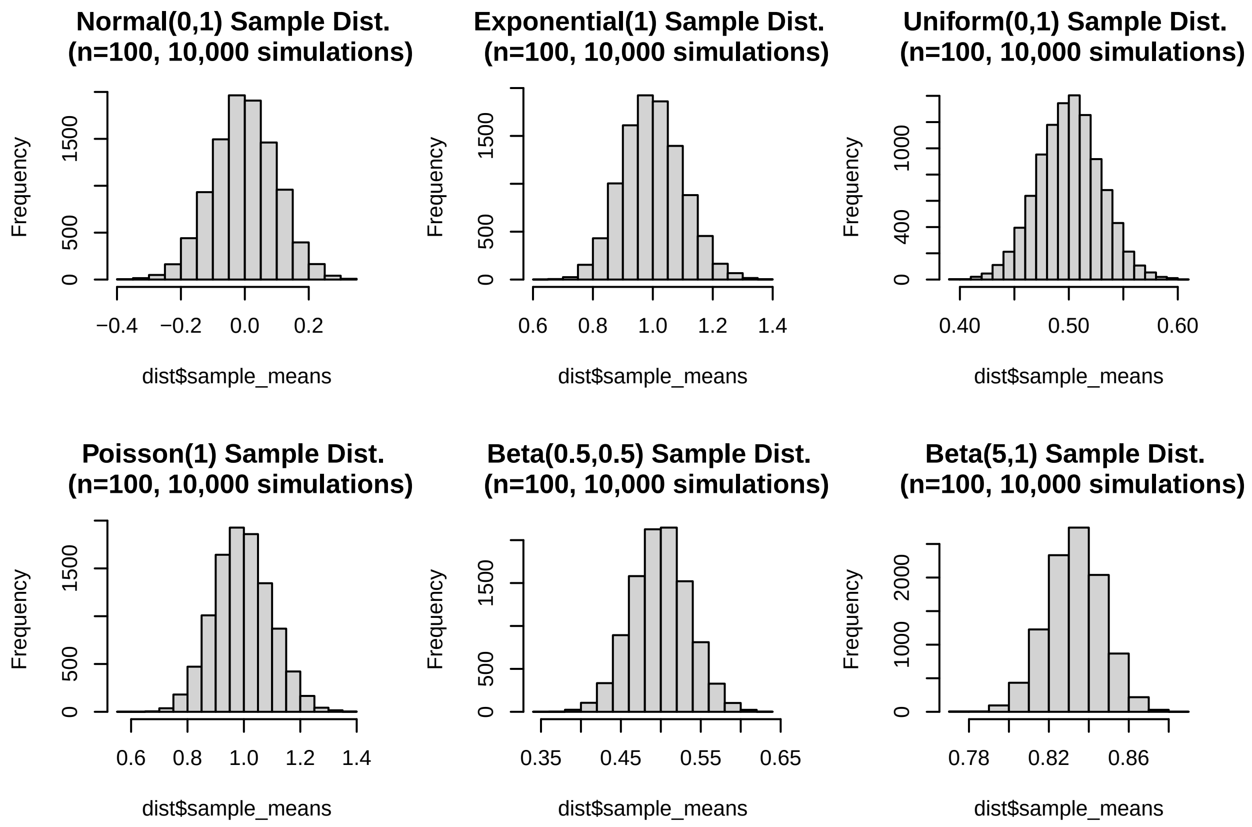
```r
dist_list <- generate_dist_list(1000000, ss=5, sims=10000)

par(mfrow=c(2,3))
for(dist in dist_list){
  ss <- format(dist$ss)
  sims <- format(length(dist$sample_means), big.mark=",")
  hist(dist$sample_means,
       main=paste0(dist$name, " Sample Dist. \n (n=", dist$ss, ", ", sims, " simulations)"))
}
```



A large number of sample means from small samples still appears normal when the population is normally distributed, but you can see that samples from other distributions have non-normal traits. Exponential, Poisson, and Beta($\alpha$=5, $\beta$=1) all show skew, while Beta($\alpha$=0.5, $\beta$=0.5) and Uniform(0,1) sample mean distributions have heavier tails.

Repeated again, this time with a larger sample size (n=100):

```r
dist_list <- generate_dist_list(1000000, ss=100, sims=10000)

par(mfrow=c(2,3))
for(dist in dist_list){
  ss <- format(dist$ss)
  sims <- format(length(dist$sample_means), big.mark=",")
  hist(dist$sample_means,
       main=paste0(dist$name, " Sample Dist. \n (n=", dist$ss, ", ", sims, " simulations)"))
}
```



Back to normality!