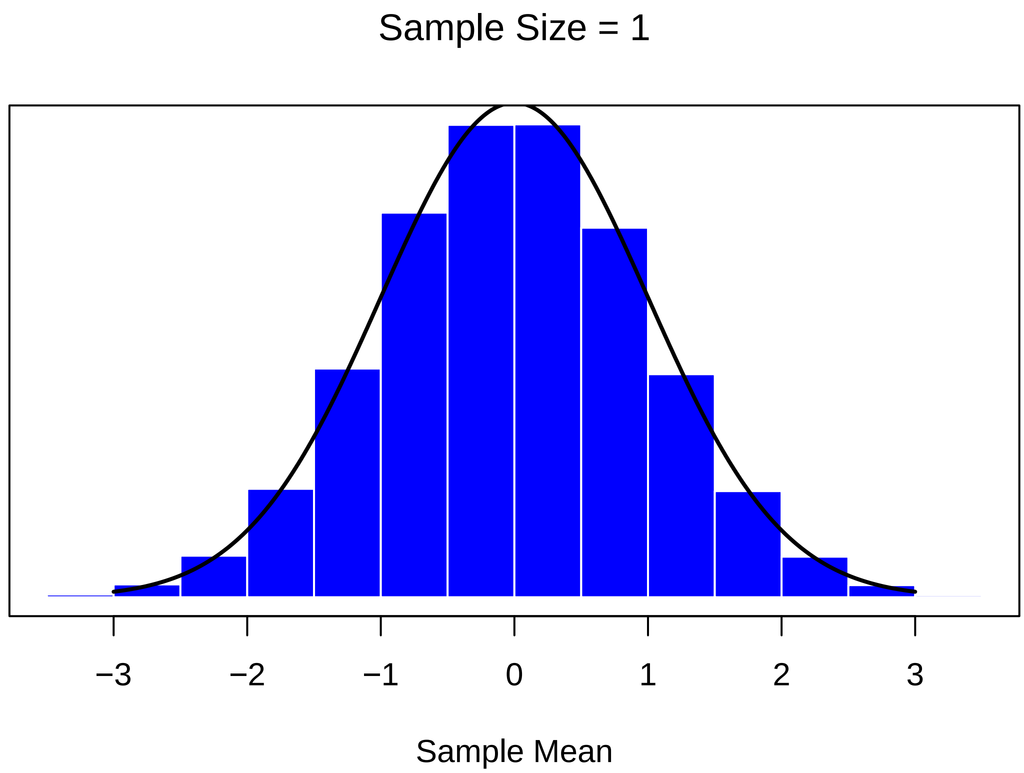
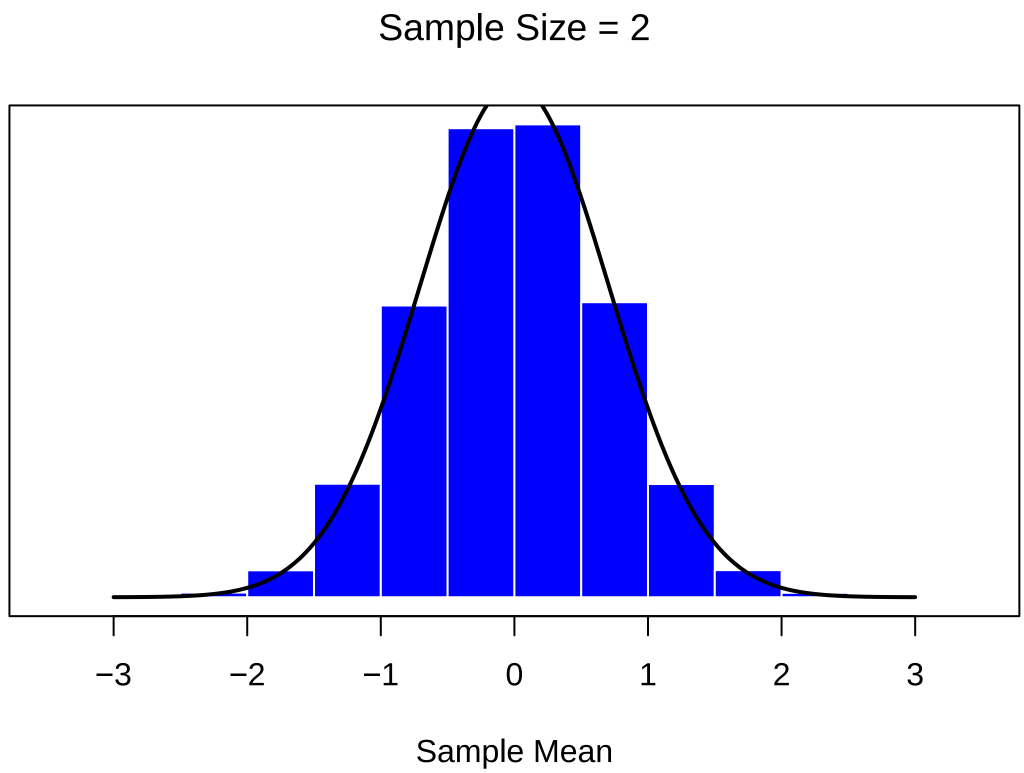
Lab 13: Central Limit Theorem, Sampling Distribution, parameter estimation

The central limit theorem

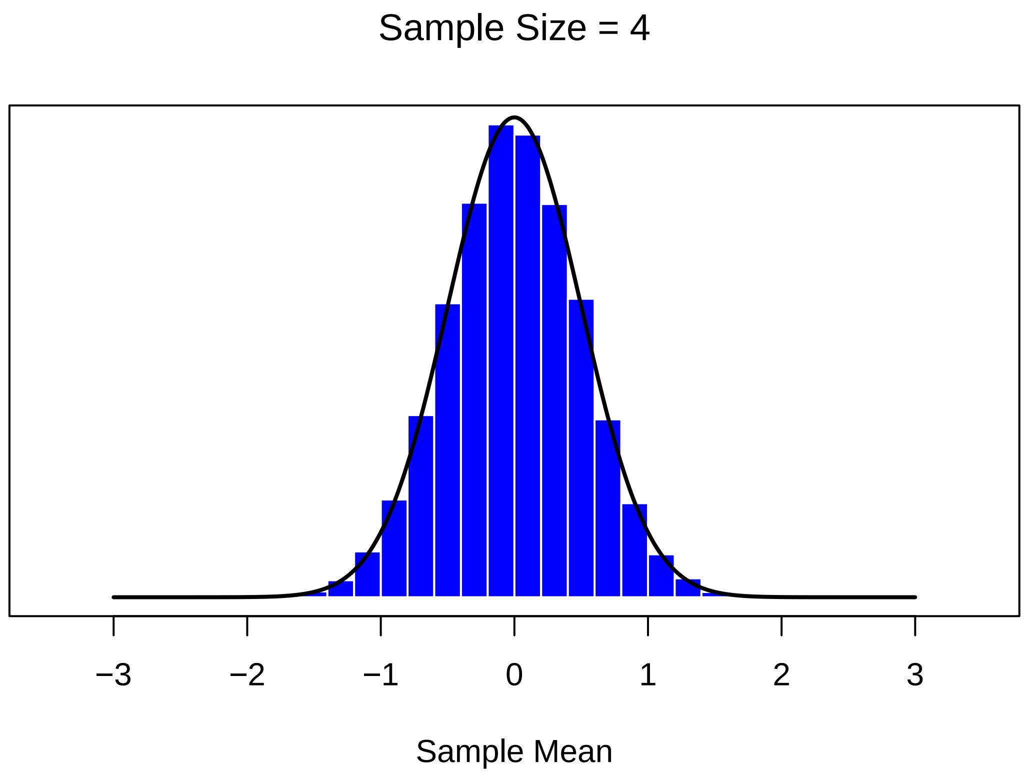
```
# mean and standard deviation of the beta
m <- 0
s <- 1

# define function to draw a plot
plotOne <- function(n,N=50000) {
  # generate N random sample means of size n
    X <- matrix(rnorm(n*N,m,s),n,N)
    X <- colMeans(X)
    # plot the data
    hist( X, breaks="Sturges", border="white", freq=FALSE,
            col="blue",
            xlab="Sample Mean", ylab="", xlim=c(-3.5,3.5),
            main=paste("Sample Size =",n), axes=FALSE,
            font.main=1
        )
    box()
    axis(1)
        # plot the theoretical distribution
    lines( x <- seq(-3.0,3.0,.01), dnorm(x,m,s/sqrt(n)),
    lwd=2, col="black", type="l"
    )
}

plotOne(1)
```



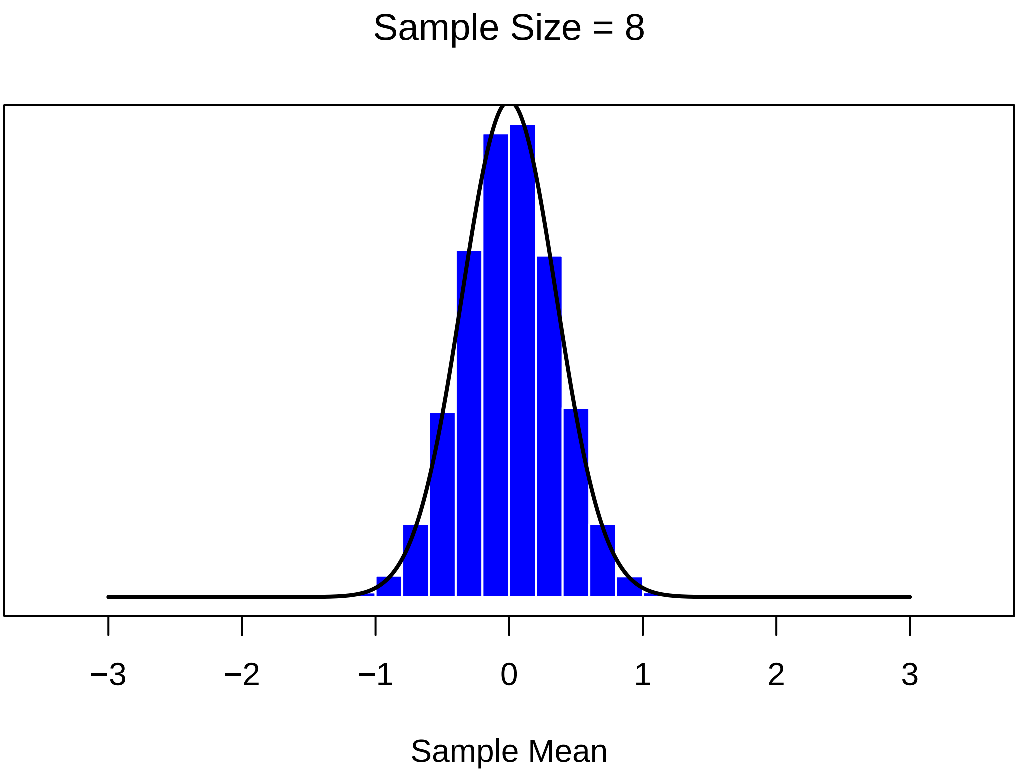

```
plotOne(2)
```




```
plotOne(4)
```



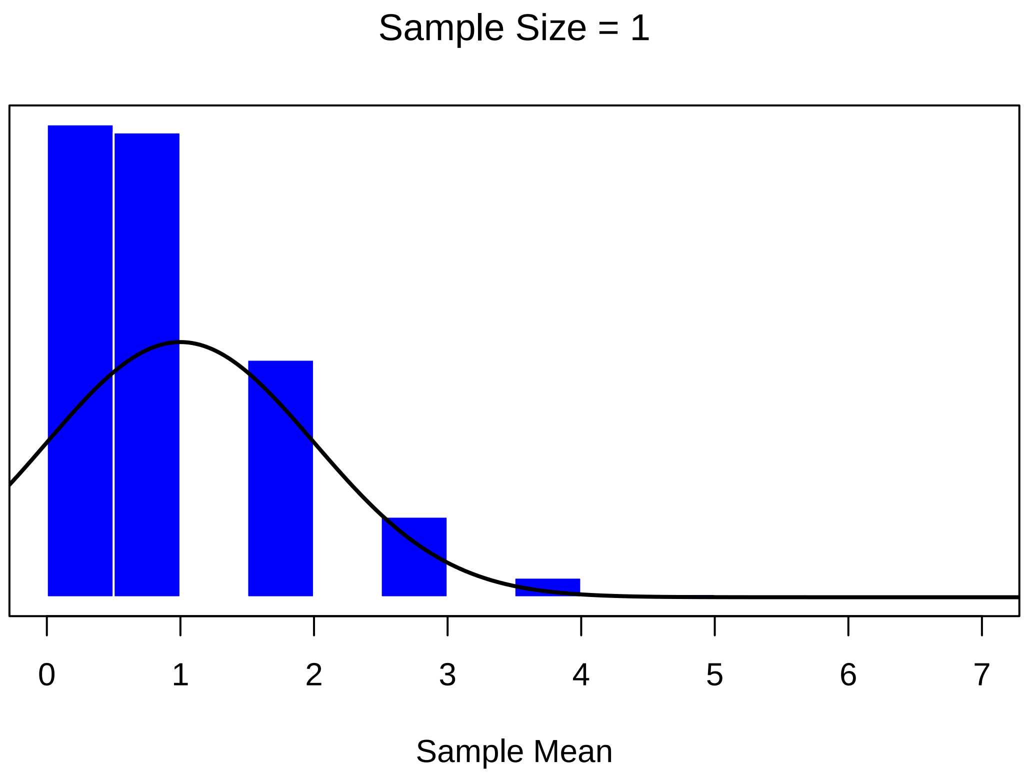

```
plotOne(8)
```



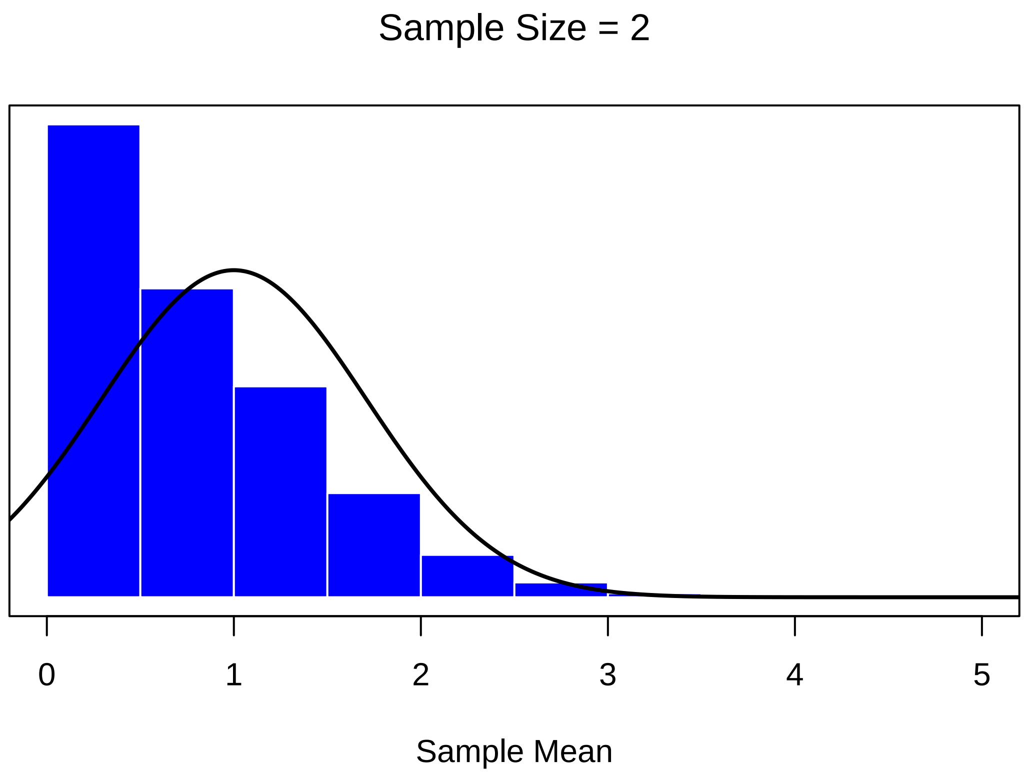

```
# mean and standard deviation of the beta
L<-1
m <- L
s <- L

# define function to draw a plot
plotOne <- function(n,N=50000) {
  # generate N random sample means of size n
    X <- matrix(rpois(n*N,L),n,N)
    X <- colMeans(X)
    # plot the data
    hist( X, breaks="Sturges", border="white", freq=FALSE,
            col="blue",
            xlab="Sample Mean", ylab="",
            main=paste("Sample Size =",n), axes=FALSE,
            font.main=1
        )
    box()
    axis(1)
        # plot the theoretical distribution
    lines( x <- seq(-30.0,30.0,.01), dnorm(x,m,s/sqrt(n)),
    lwd=2, col="black", type="l"
    )
}

plotOne(1)
```

rexp

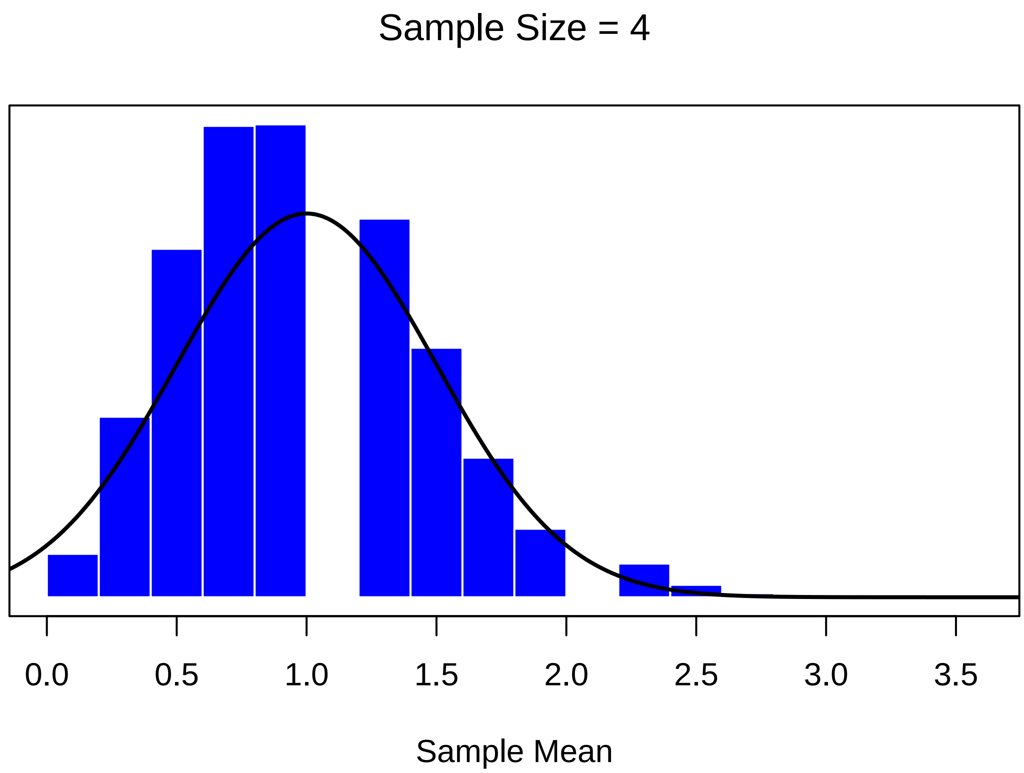
exponential



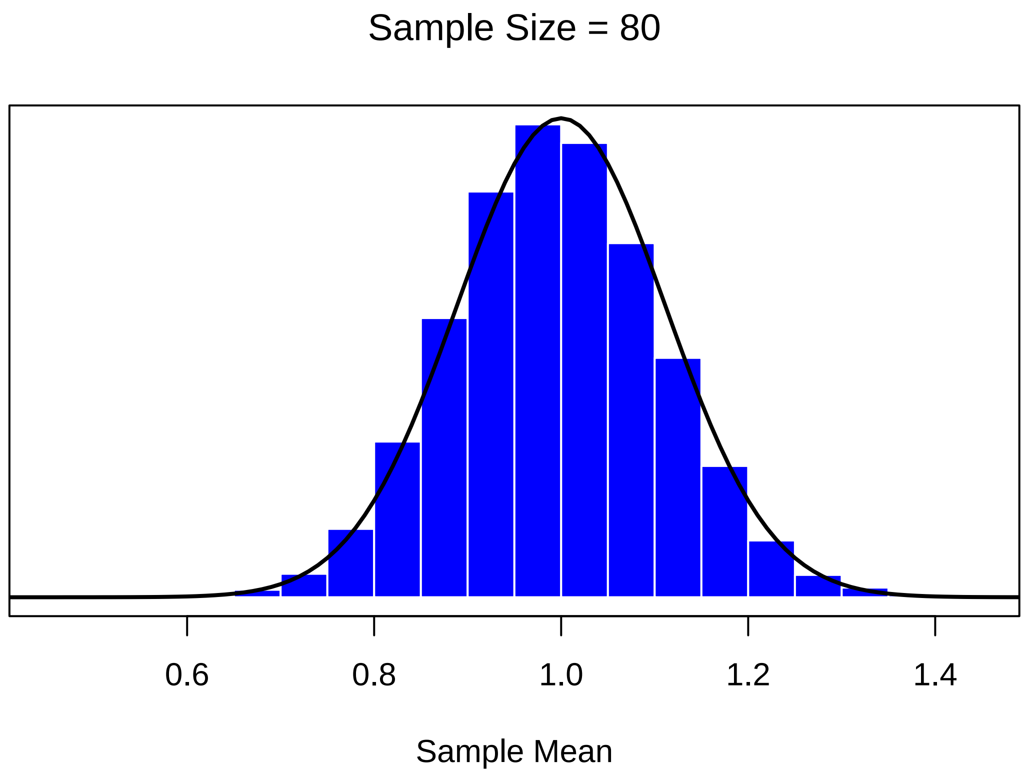

```
plotOne(2)
```




```
plotOne(4)
```

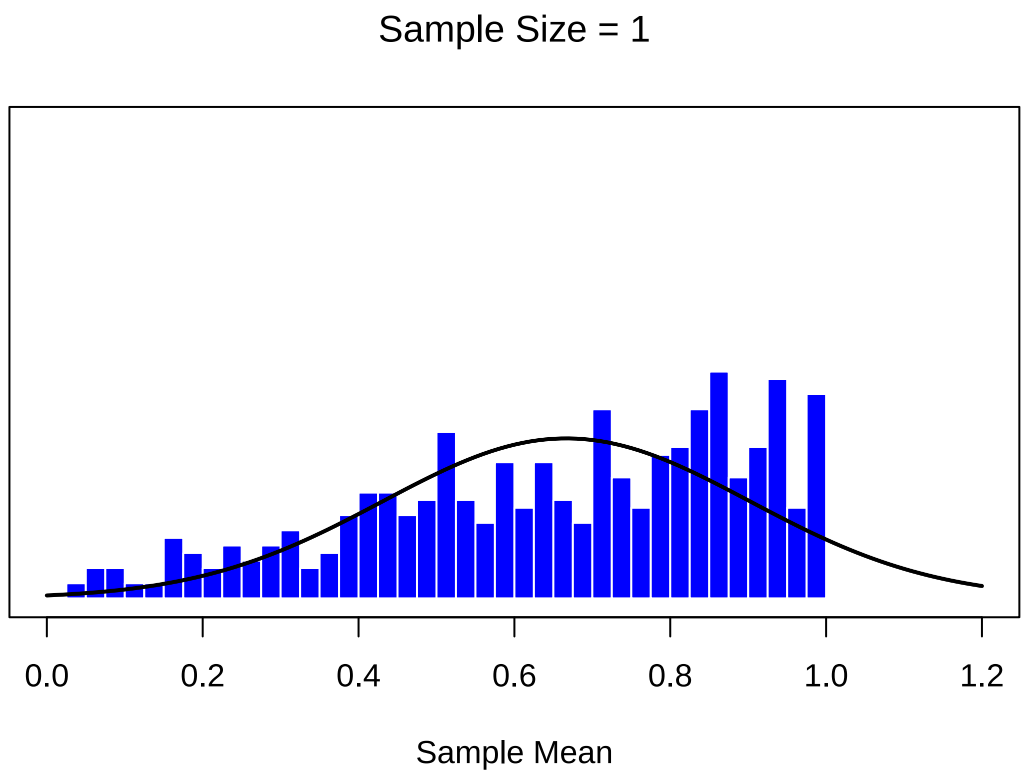



```
plotOne(80)
```

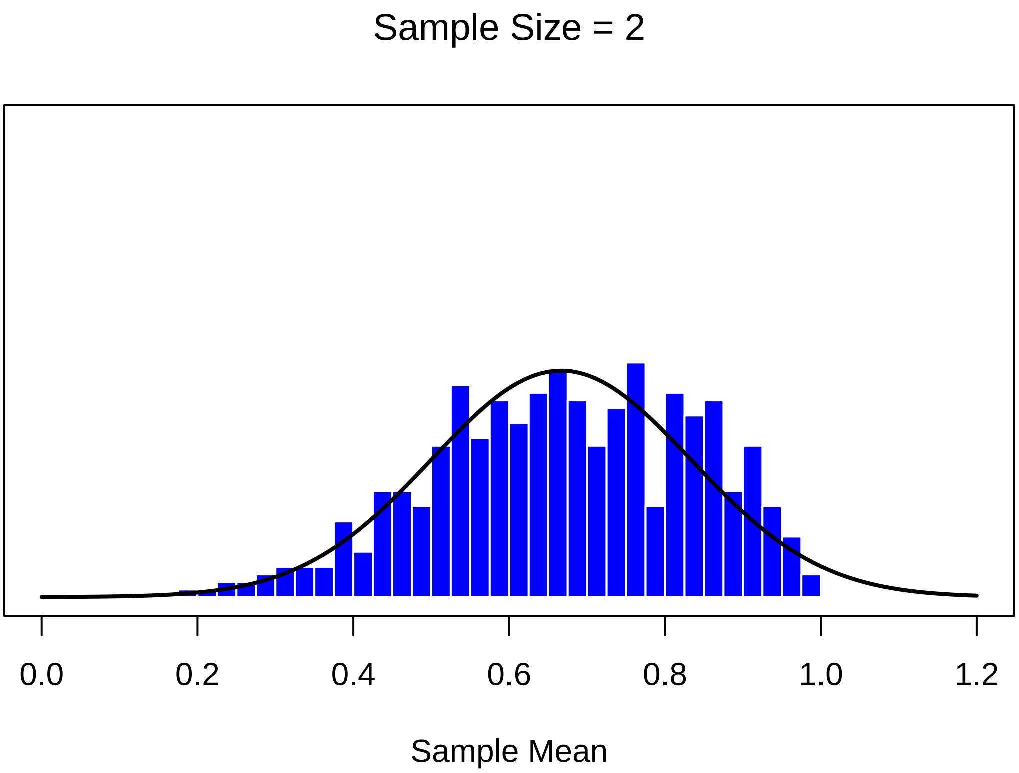



```
# needed for printing
width <- 6
height <- 6

# parameters of the beta
a <- 2
b <- 1

# mean and standard deviation of the beta
s <- sqrt( a*b / (a+b)^2 / (a+b+1) )
m <- a / (a+b)

# define function to draw a plot
plotOne <- function(n,N=500) {
  # generate N random sample means of size n
    X <- matrix(rbeta(n*N,a,b),n,N)
    X <- colMeans(X)
    # plot the data
    hist( X, breaks=seq(0,1,.025), border="white", freq=FALSE,
            col="blue",
            xlab="Sample Mean", ylab="", xlim=c(0,1.2),
            main=paste("Sample Size =",n), axes=FALSE,
            font.main=1, ylim=c(0,5)
        )
    box()
    axis(1)
```

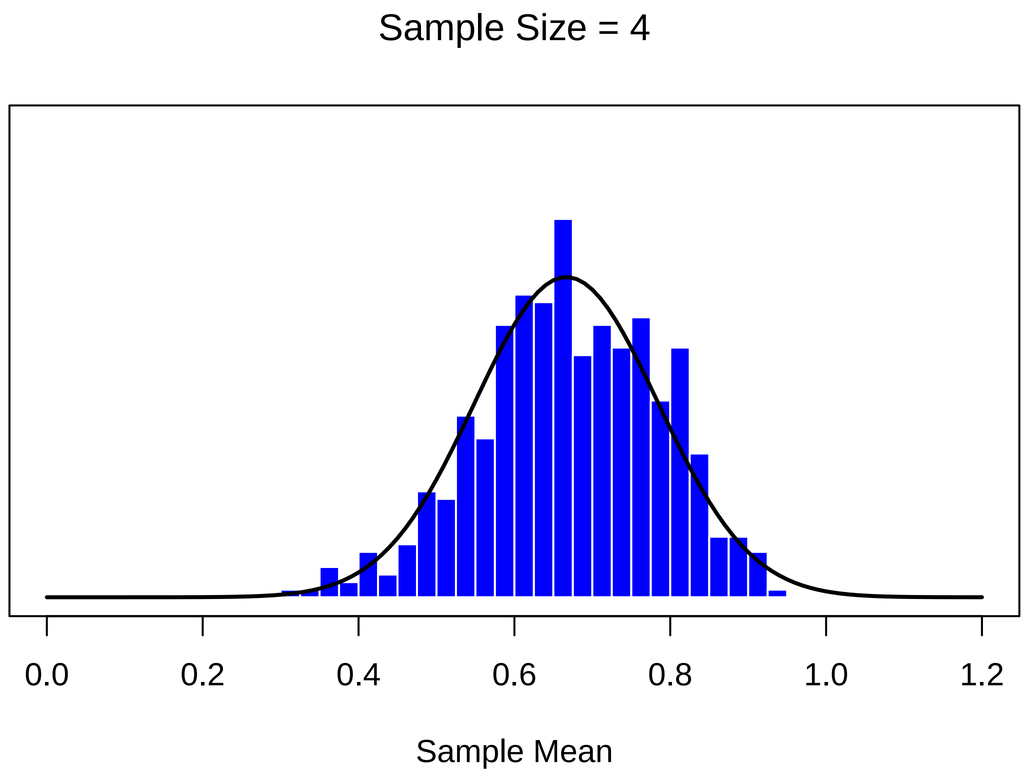
```r
        # plot the theoretical distribution
    lines( x <- seq(0,1.2,.01), dnorm(x,m,s/sqrt(n)),
    lwd=2, col="black", type="l"
    )
}

plotOne(1)
```



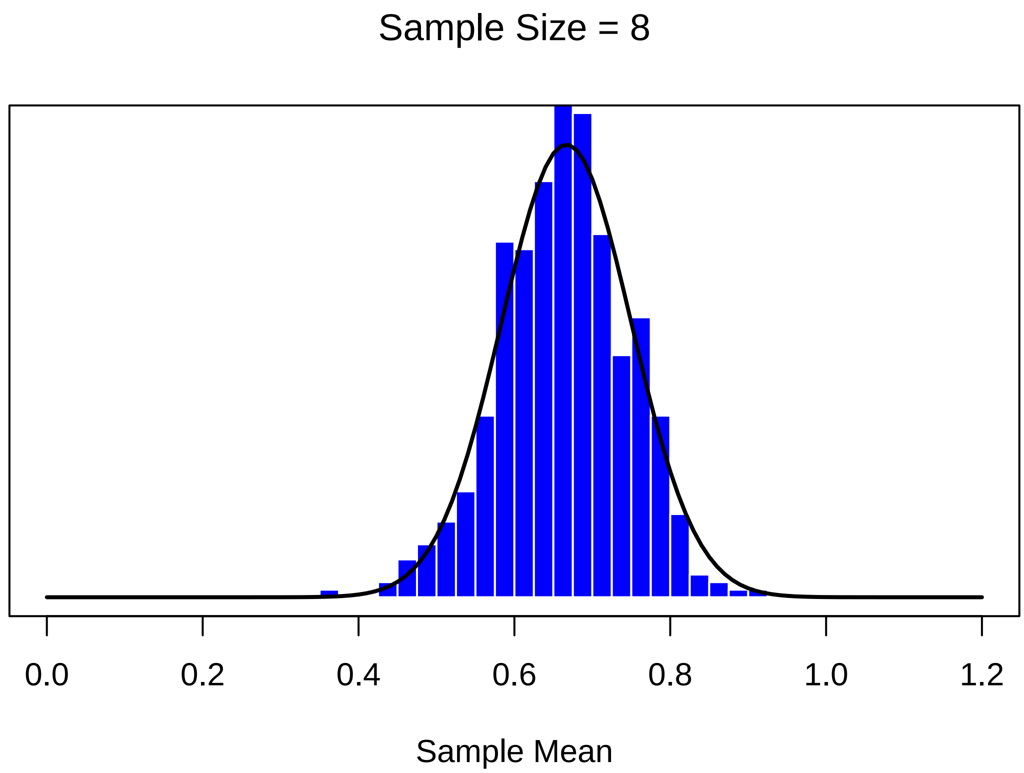

```r
plotOne(2)
```




```
plotOne(4)
```




```
plotOne(8)
```



Central limit theorem. The central limit theorem tells us that if the population distribution has mean μ and standard deviation σ, then the sampling distribution of the mean also has mean μ, and the standard error of the mean is

$$\mathrm{S} = \frac{\sigma}{\sqrt{N}}$$

Parameter estimation

```
set.seed(22)
heads <- rbinom(1,100,0.5)
heads
```

```
## [1] 52
```

```
sprob <- 0.65
choose(100,heads)*(sprob**heads)*(1-sprob)**(100-heads)
```

```
## [1] 0.002270948
```

```
dbinom(heads,100,sprob)
```

```
## [1] 0.002270948
```

```r
likelihood <- function(p){
  dbinom(heads, 100, p)
}
likelihood(sprob)
```

```
## [1] 0.002270948
```

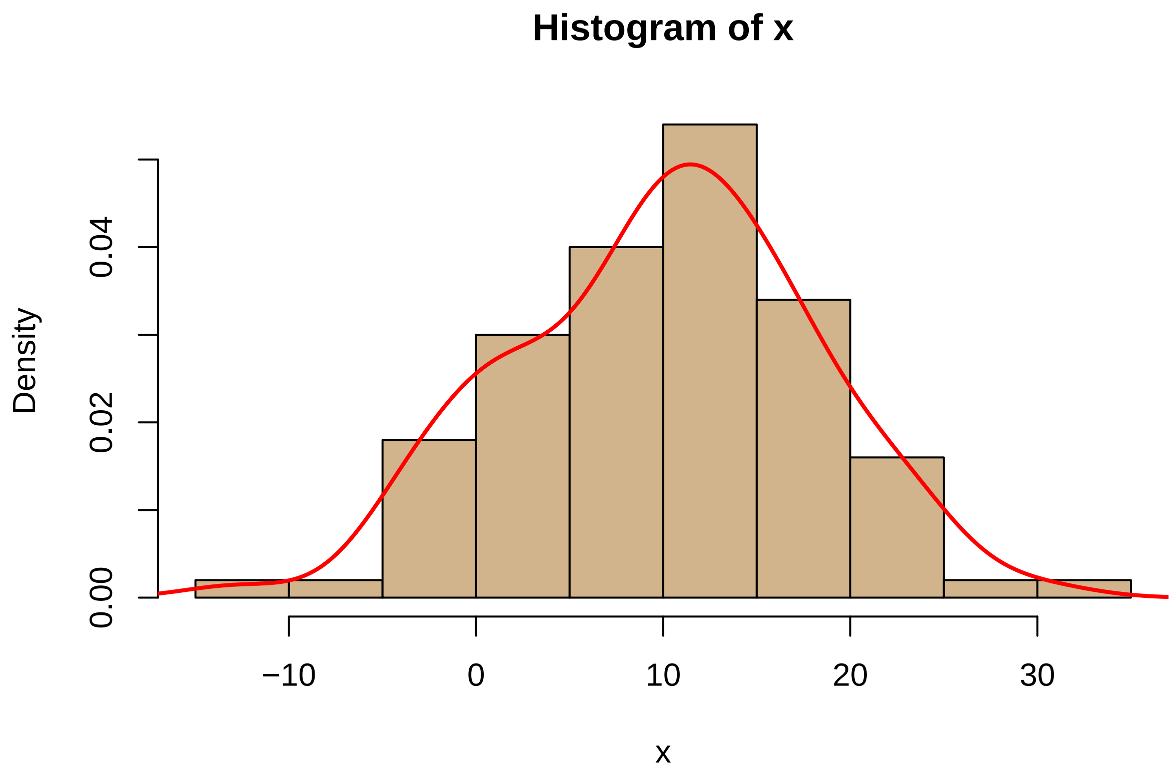
```r
neg_likelihood <- function(p){
  (-1)*dbinom(heads, 100, p)
}

nlm(neg_likelihood,0.5,stepmax=0.5)
```

```
## $minimum
## [1] -0.07965256
##
## $estimate
## [1] 0.5199995
##
## $gradient
## [1] -2.775558e-11
##
## $code
## [1] 1
##
## $iterations
## [1] 4
```

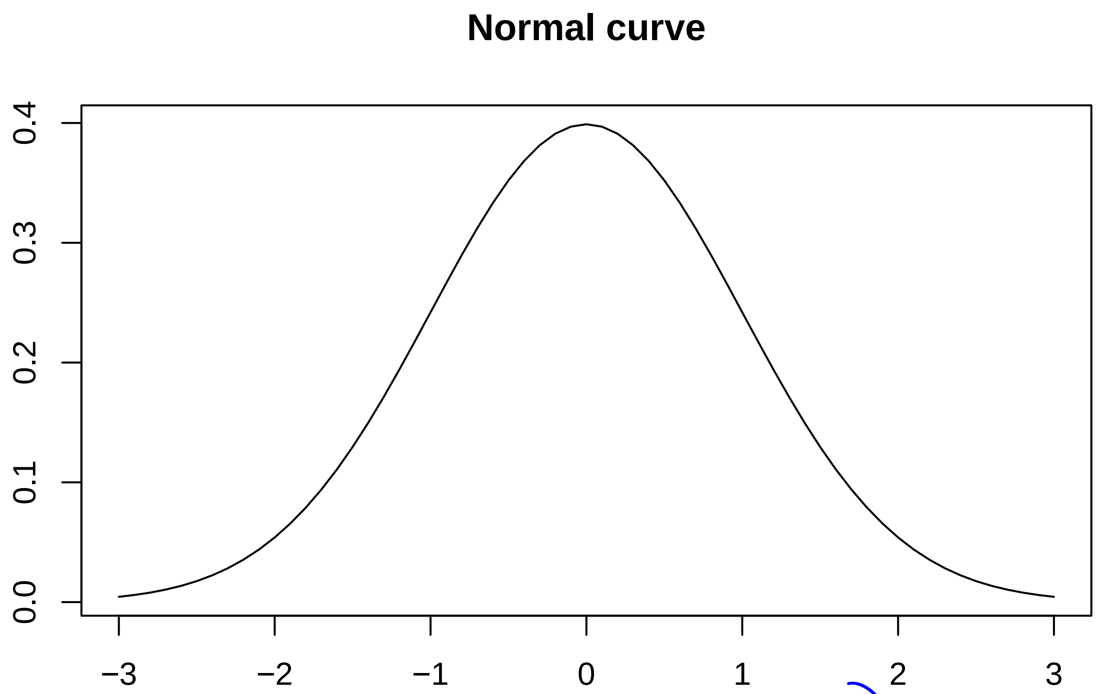
```r
set.seed(1123)
x = rnorm(100)
x = x/sd(x) * 8
x = x-mean(x) + 10
c('mean'=mean(x),'sd'=sd(x)) # double check
```

```
## mean   sd
##   10    8
```

```r
# histogram
hist(x, freq=FALSE,col='tan')
lines(density(x),col='red',lwd=2)
```




```
norm_lik = function(x, m, s){
y = 1/sqrt(2*pi*s^2)*exp((-1/(2*s^2))*(x-m)^2)
}
plot(seq(-3,3,.1),sapply(seq(-3,3,.1),FUN=norm_lik,m=0,s=1),type='l',
ylab='',xlab='', main='Normal curve')
```



x = rpois(100, 2)

$p_k = \frac{\alpha^k \cdot e^{-\alpha}}{k!}$

$m = \alpha$

$\sigma = \sqrt{\alpha}$

```r
llik = function(x,par){
m=par[1]
s=par[2]
n=length(x)
# log of the normal likelihood
# -n/2 * log(2*pi*s^2) + (-1/(2*s^2)) * sum((x-m)^2)
ll = -(n/2)*(log(2*pi*s^2)) + (-1/(2*s^2)) * sum((x-m)^2)
# return the negative to maximize rather than minimize
return(-ll)
}

res = optim(par=c(.5,.5), llik, x=x)
print(knitr::kable(
cbind('direct'=c('mean'=mean(x),'sd'=sd(x)),
'optim'=res$par),digits=3))
```

0.5

```
##
##
## |     | direct|  optim|
## |:----|------:|------:|
## |mean |     10| 10.002|
## |sd   |      8|  7.976|
```

$f(x_1, x_2, \ldots, x_n | \alpha)$

$= f(x_1|\alpha) f(x_2|\alpha) \cdots f(x_n|\alpha)$

$= \frac{\alpha^{x_1} e^{-\alpha}}{x_1!} \cdot \frac{\alpha^{x_2} e^{-\alpha}}{x_2!} \cdots$

$\text{Log} f = \text{Log}\, \alpha^{x_1+x_2+\cdots+x_n} e^{-n\alpha}$

$= \sum x_i \log \alpha - n\alpha$