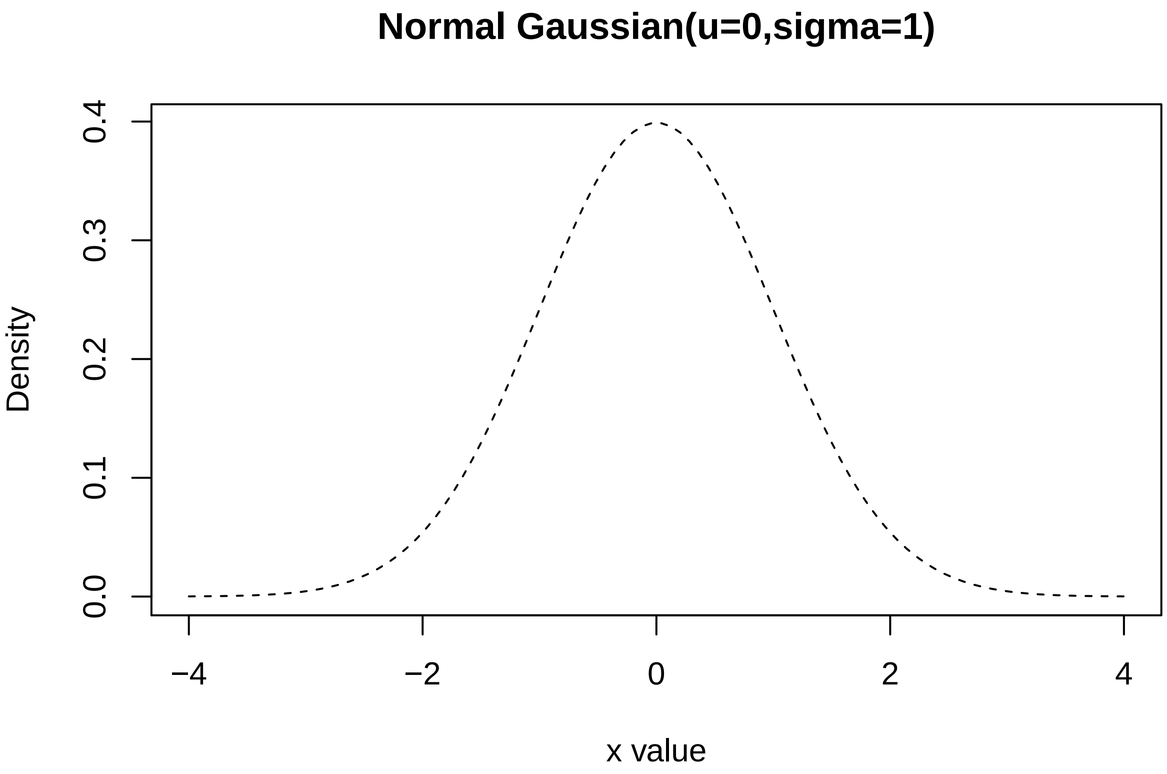
Maximum Likelihood

Daniel Frederico Lins Leite

19 March 2017

Gaussian

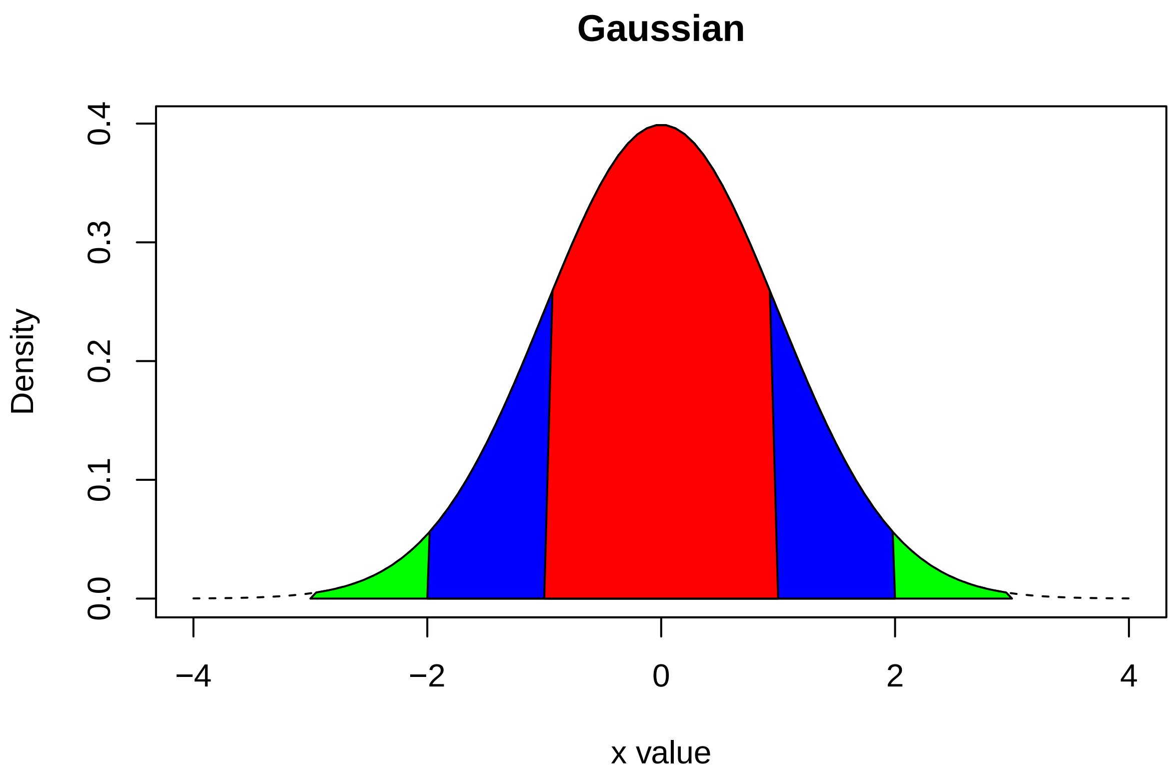
```r
x <- seq(-4, 4, length=100)
hx <- dnorm(x)
plot(x, hx, type="l", lty=2, xlab="x value", ylab="Density", main="Normal Gaussian(u=0,sigma=1)")
```



Percentage

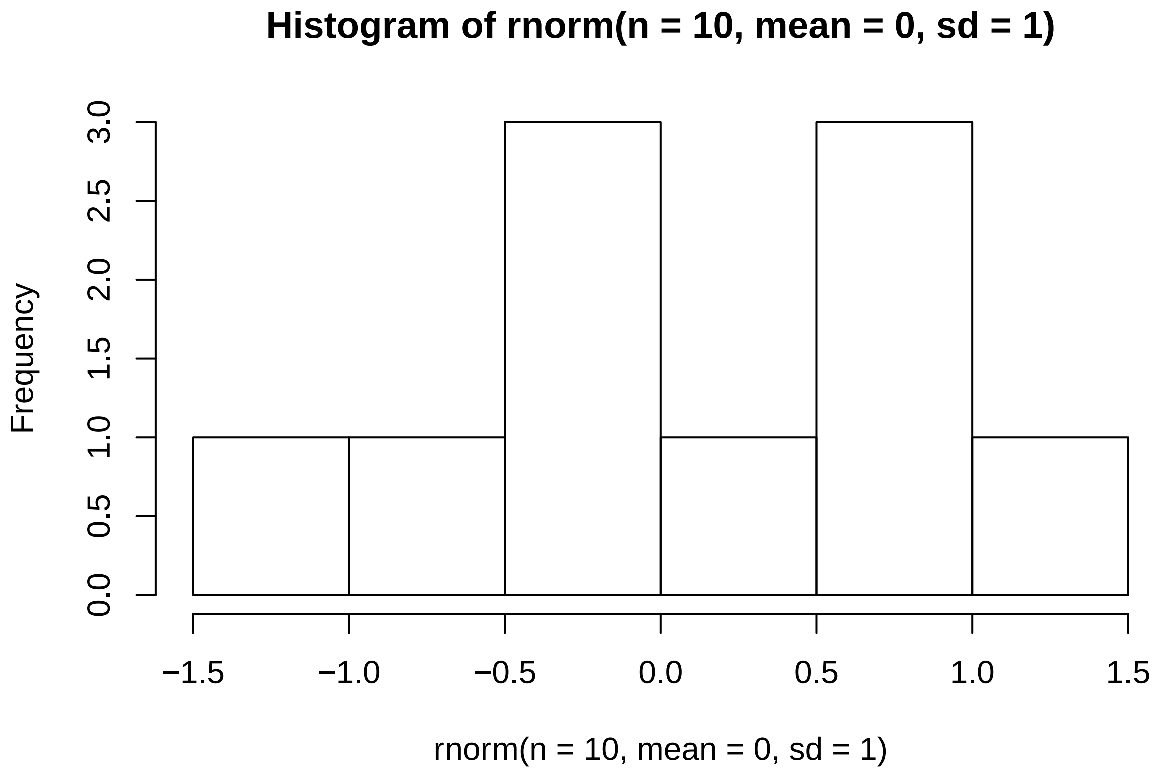
```r
plotGaussian <- function(u, sigma, color = "black"){
  x <- seq(-4, 4, length=100)
  hx <- dnorm(x, mean = u, sd = sigma)
  plot(x, hx, type="l", lty=2, xlab="x value", ylab="Density", main="Gaussian", col = color)
}
plotArea <- function(u, sigma, sigmaSize, color){
  x <- seq(-4, 4, length=100)
  hx <- dnorm(x, mean = u, sd = sigma)
  l <- -(sigma*sigmaSize)
  r <- (sigma*sigmaSize)
  i <- x >= l & x <= r
  polygon(c(l,x[i],r), c(0,hx[i],0), col=color)
}

plot.new()
plotGaussian(0,1)
plotArea(0,1,3,"green")
plotArea(0,1,2,"blue")
plotArea(0,1,1,"red")
```

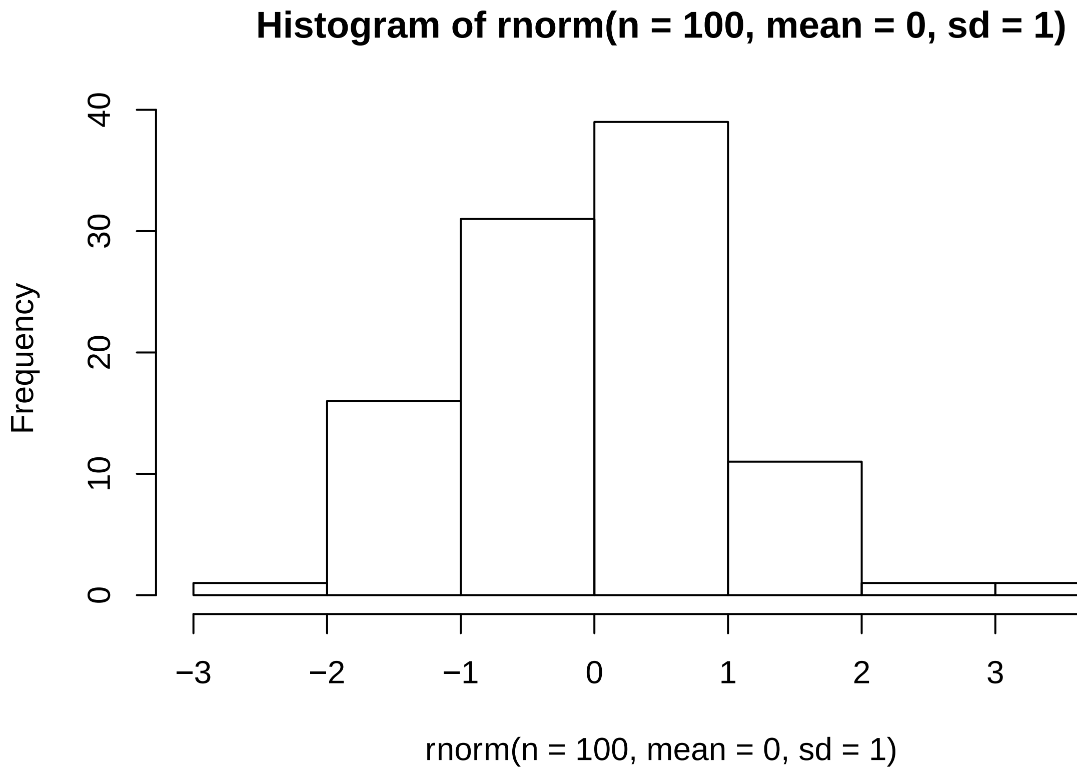


Density Simulation

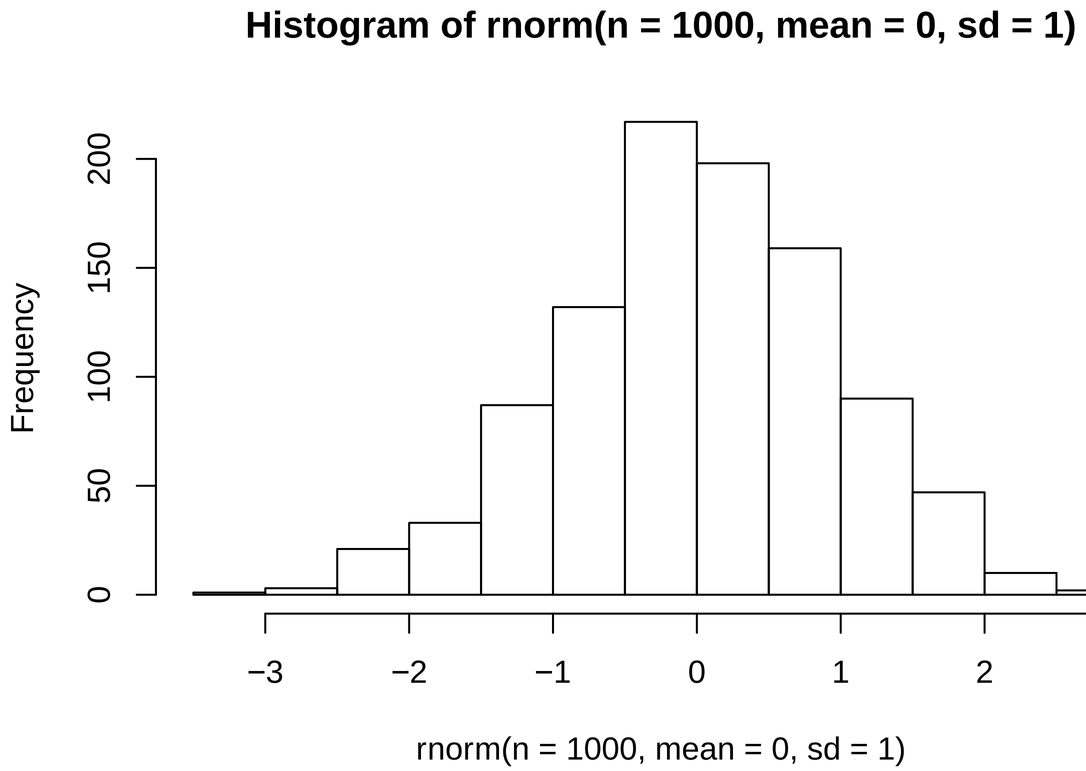
```
hist(rnorm(n = 10, mean = 0, sd = 1))
```



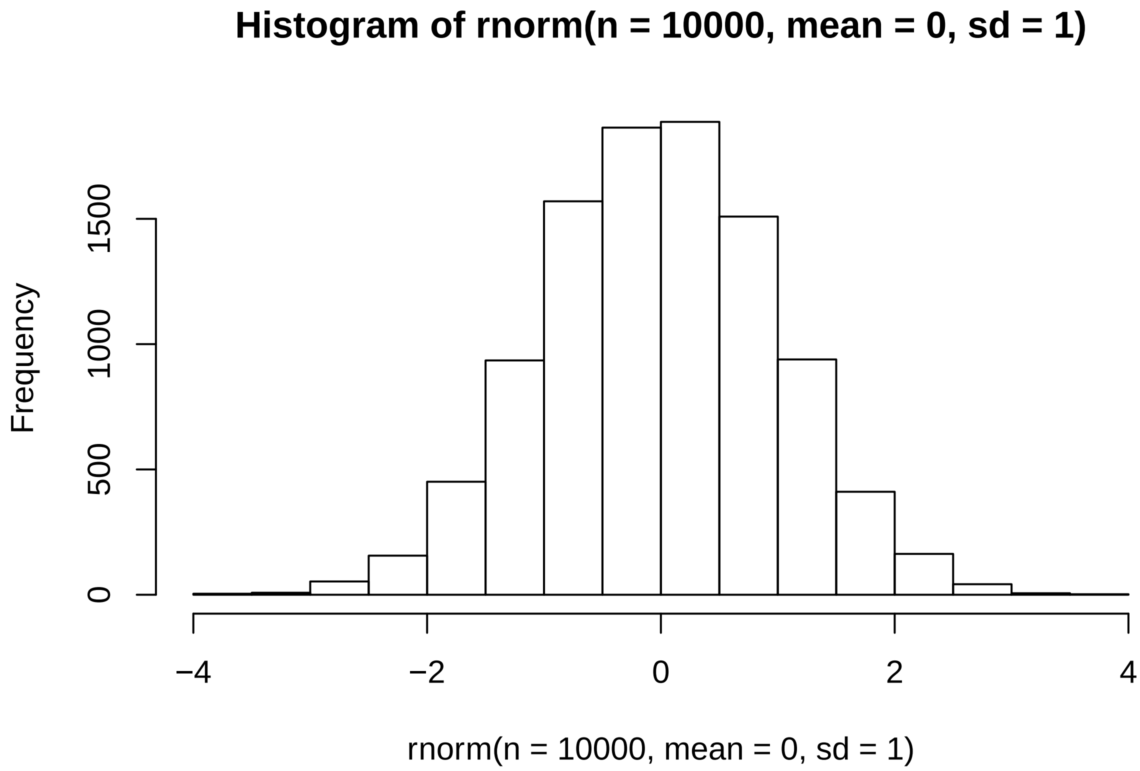
```
hist(rnorm(n = 100, mean = 0, sd = 1))
```



```r
hist(rnorm(n = 1000, mean = 0, sd = 1))
```



```r
hist(rnorm(n = 10000, mean = 0, sd = 1))
```



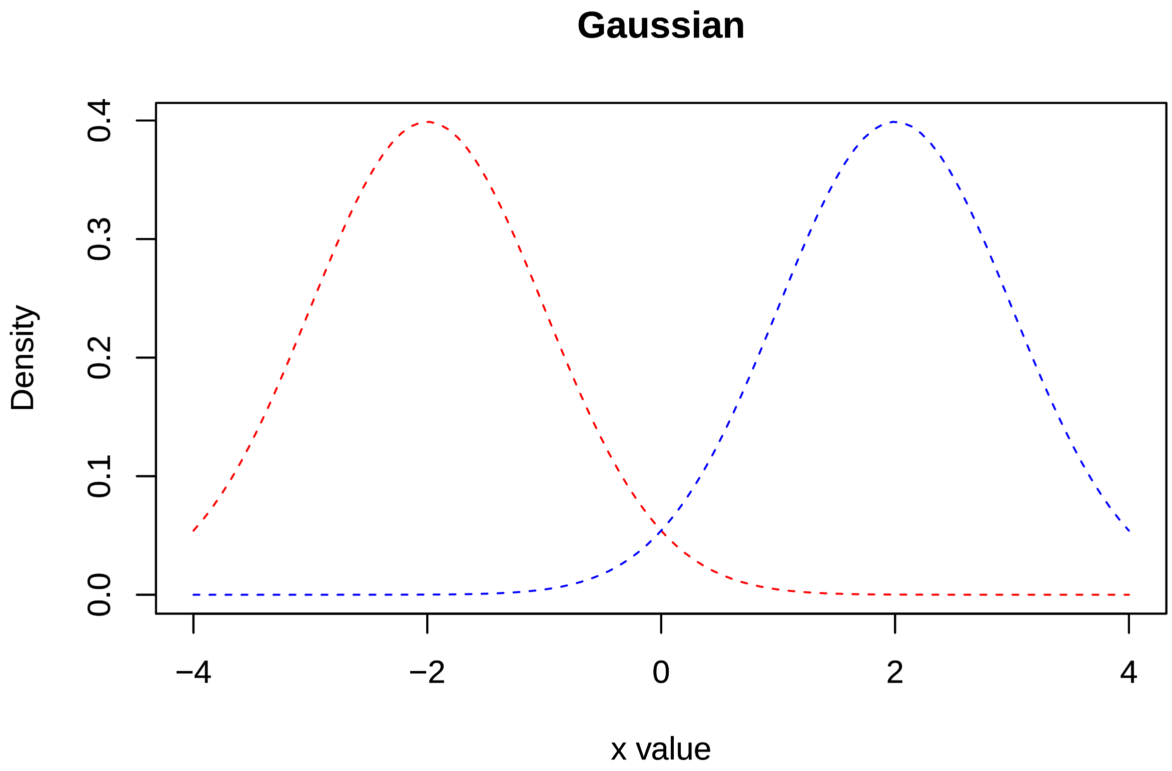
Likelihood

Suppose we have an observation and two possible distributions that can be considered as the source distributions of this observations. We want to choose the best option: in this case the most probable source distribution.

Options:
Gaussian #1
u = -2
sd = 1

Gaussian #2
u = 2
sd = 1

```
plotGaussian(-2,1,"red")
par(new=TRUE)
plotGaussian(2,1, "blue")
```



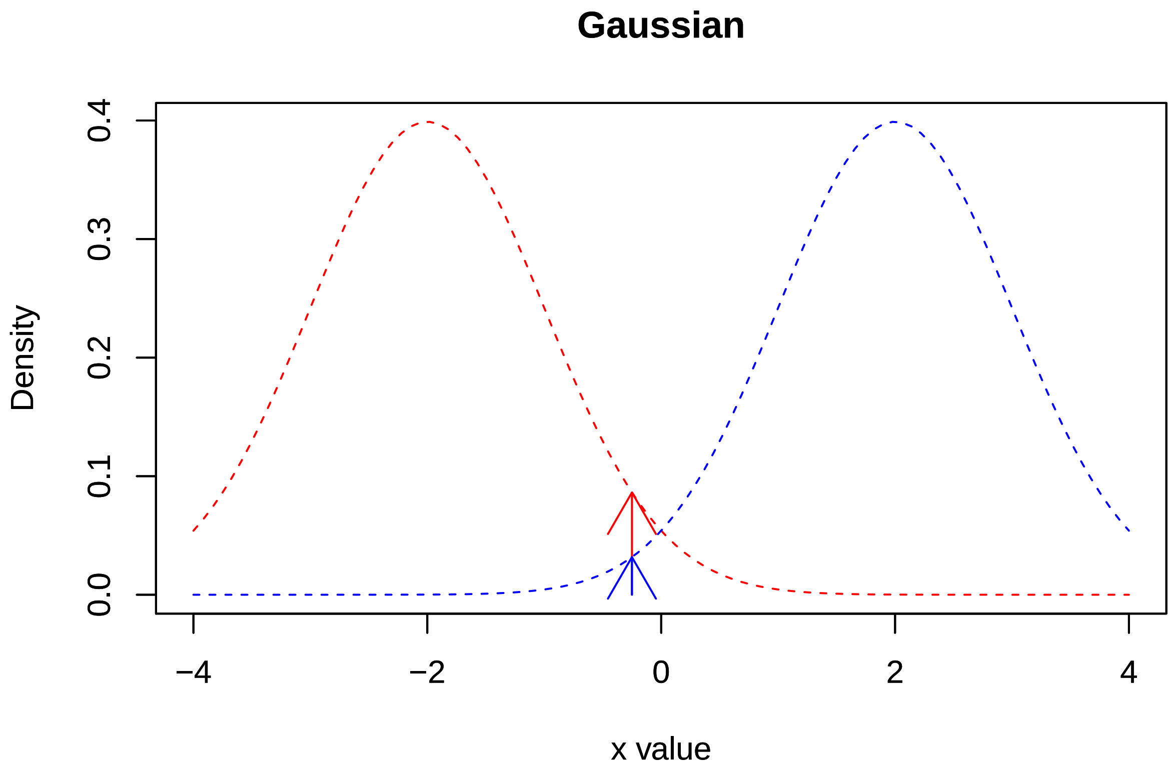
Observations
A = -0.25
B = -1
C = 0.45

To choose the best distribution we will choose the distribution whose density is bigger in that particular point. For example, for observation A:

Probability of observing A from Gaussian #1 (u-2,sd=1) = 0.0862773
Probability of observing A from Gaussian #2 (u=2,sd=1) = 0.0317397

In this case the best option is Gaussian #1. It is more likely/probable that the observation #A comes from Gaussian #1 than Gaussian #2.

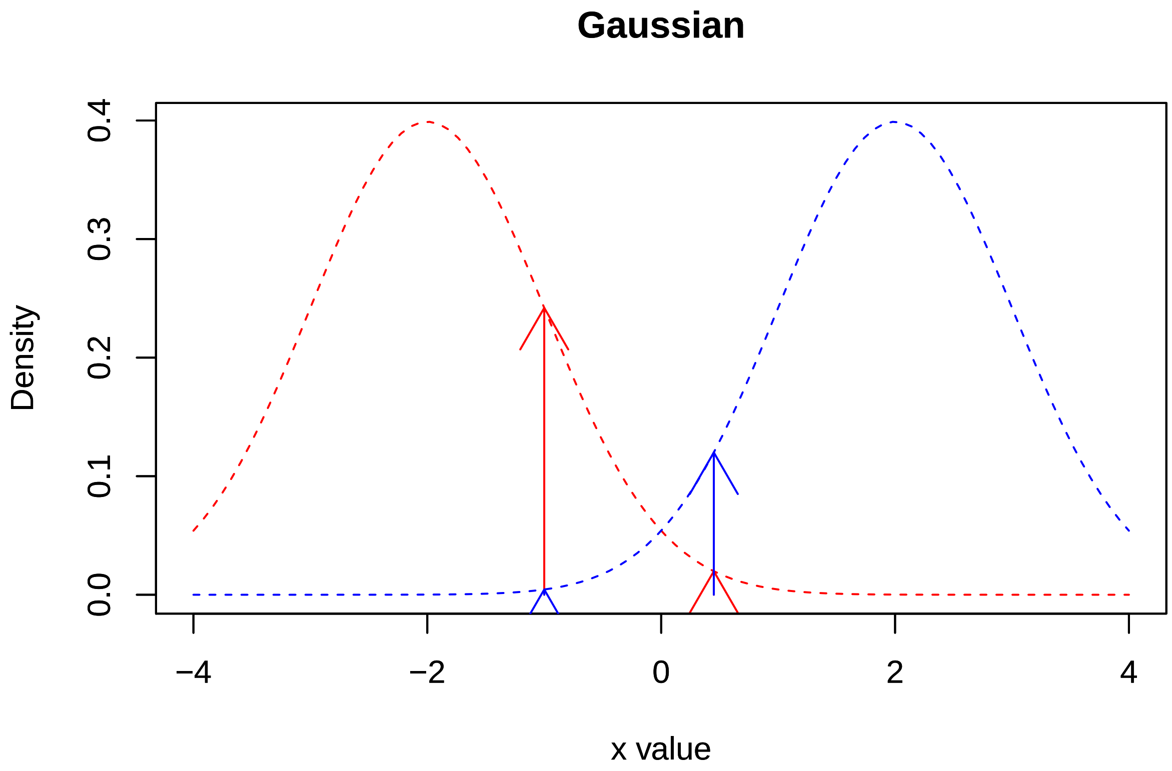
```r
plotGaussian(-2,1,"red")
arrows(-0.25,0,-0.25,dnorm(-0.25,-2,1), col = "red")
par(new=TRUE)
plotGaussian(2,1, "blue")
arrows(-0.25,0,-0.25,dnorm(-0.25,2,1), col = "blue")
```



We can use the same calculations to the others observations.

Probability of observing B from Gaussian #1 (u-2,sd=1) = 0.2419707
Probability of observing B from Gaussian #2 (u=2,sd=1) = 0.0044318
Probability of observing C from Gaussian #1 (u-2,sd=1) = 0.0198374
Probability of observing C from Gaussian #2 (u=2,sd=1) = 0.120009

```
plotGaussian(-2,1,"red")
arrows(-1,0,-1,dnorm(-1,-2,1), col = "red")
arrows(0.45,0,0.45,dnorm(0.45,-2,1), col = "red")
par(new=TRUE)
plotGaussian(2,1, "blue")
arrows(-1,0,-1,dnorm(-1,2,1), col = "blue")
arrows(0.45,0,0.45,dnorm(0.45,2,1), col = "blue")
```



So we can sau that:
The likelihood of u = -2 and sd = 1 for a observation -1 is 0.2419707 because the probability of seeing a value -1 from a Gaussian with u = -2 and sd = 1 is 0.2419707.
In others words:
The likelihood of theta, the gaussian parameters, given x is 0.2419707 because the probability of x given theta is 0.2419707.
Or:
L(theta|x) = p(x|theta)

Maximum Likelihood

Maximum Likelihood of just one point

Imagine now that we do not have two options predertimined. We have all possible Gaussians and we want to find the best gaussian to each observation, first one-by-one, and them as a set. The best gaussian for observation A is the Gaussian with the maximum likelihod. So our problem can be described as:

maximize theta in L(theta|x)

We saw that this is the same as

maximize theta in p(x | theta)

Se given x, which is a known value, we must find the theta that maximize the function. theta in this particular case is the set {u,sd}. So given X, we must find u and sd that will maximize p(x|u,sd).

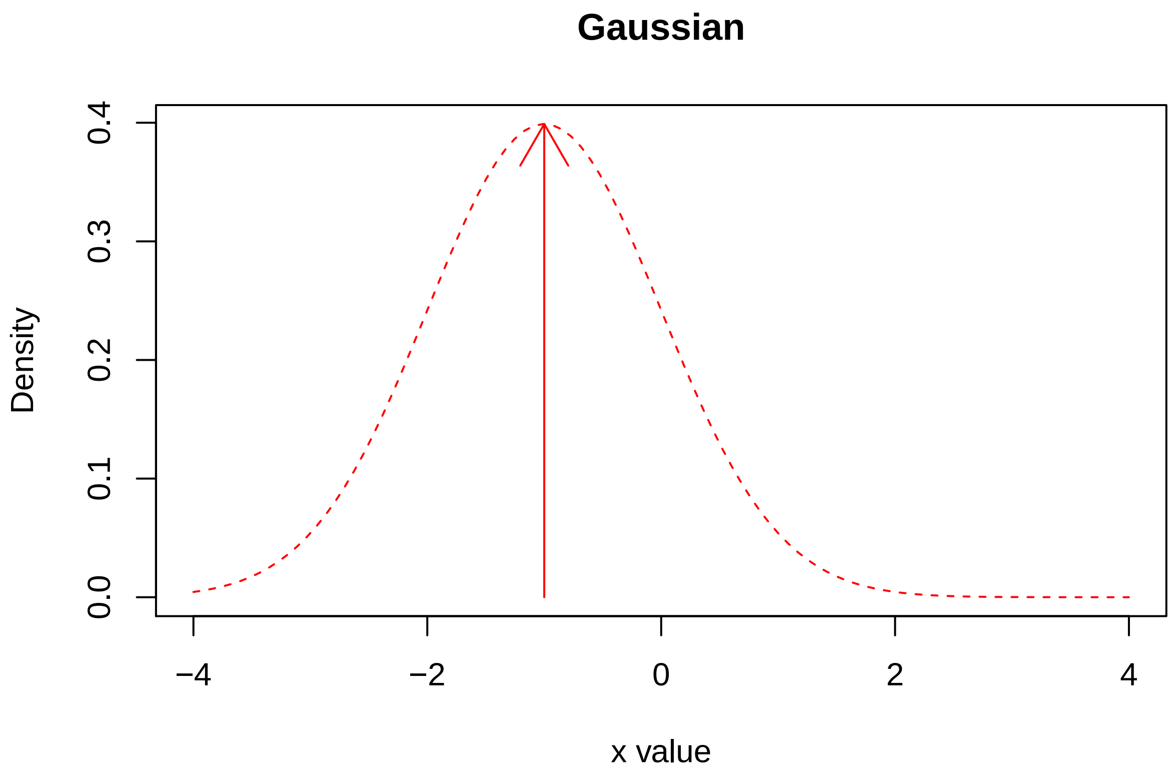
For this trivial case we will have a problem, that will be shown later. For now let fix sd =1 and try to maximize just for u. So we have:

maximize u in p(x | u, sd = 1)

Well… looking again to the Gaussian distribution we see that the maximum value is at u. So if we want to maximize p(x | u, sd = 1), we just have to make u = x.

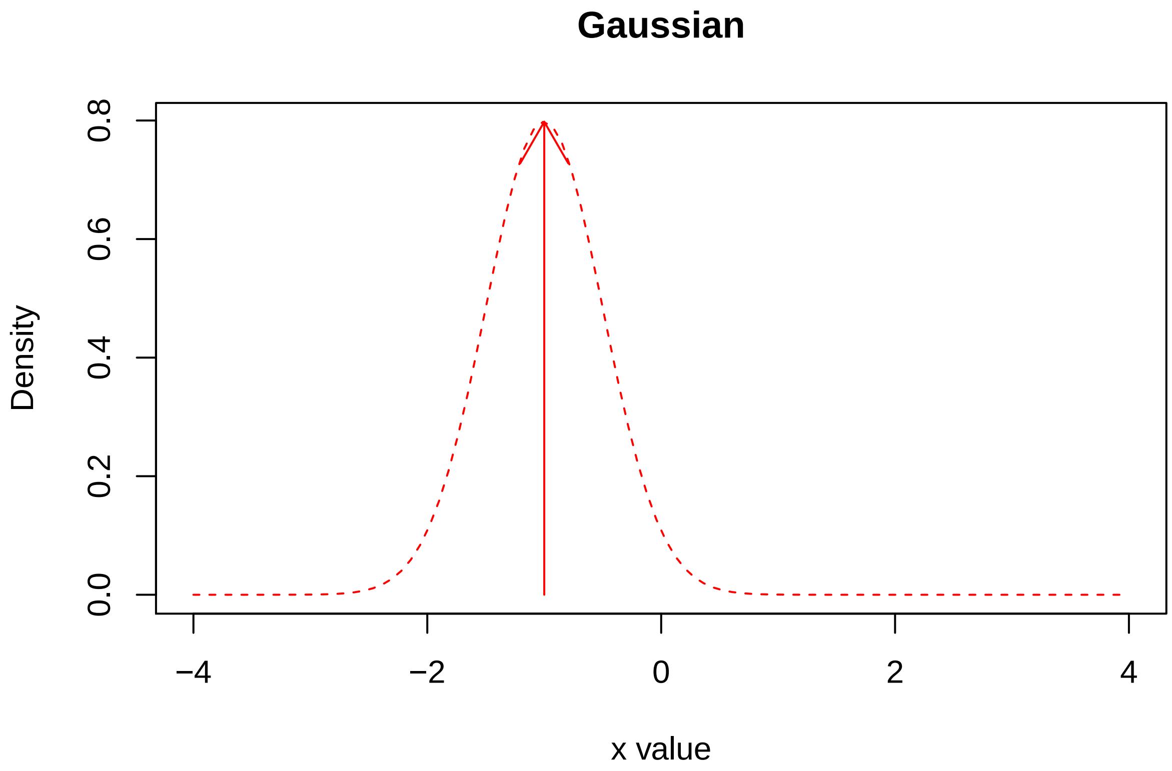
So in this case we have:

```r
plotGaussian(-1,1,"red")
arrows(-1,0,-1,dnorm(-1,-1,1), col = "red")
```



But we still have one problem. We can improve the likelihood in this case decreasing the sd. If we chose a sd = 0.5, for example, we will have:

```r
plotGaussian(-1,0.5,"red")
arrows(-1,0,-1,dnorm(-1,-1,0.5), col = "red")
```



Actually we can always improve the likelihood by decreasing the sd in this particular case, because we just have one point.

Likelihood of two points

For the likelihood of two points we are going to use the same method to find the best Gaussian distribution to explain both observations together.

This means:

L(theta|x) = L(theta|x1,x2) = p(x1,x2|theta)

To simplify let start optimizing this function for u first. So we have:

p(x1,x2|u) = p(x1|x2)p(x2) = p(x2|x1)p(x1)

We generraly assume that both observations are independent because in this way we have that

p(x2|x1) = p(x2)
p(x1|x2) = p(x1)

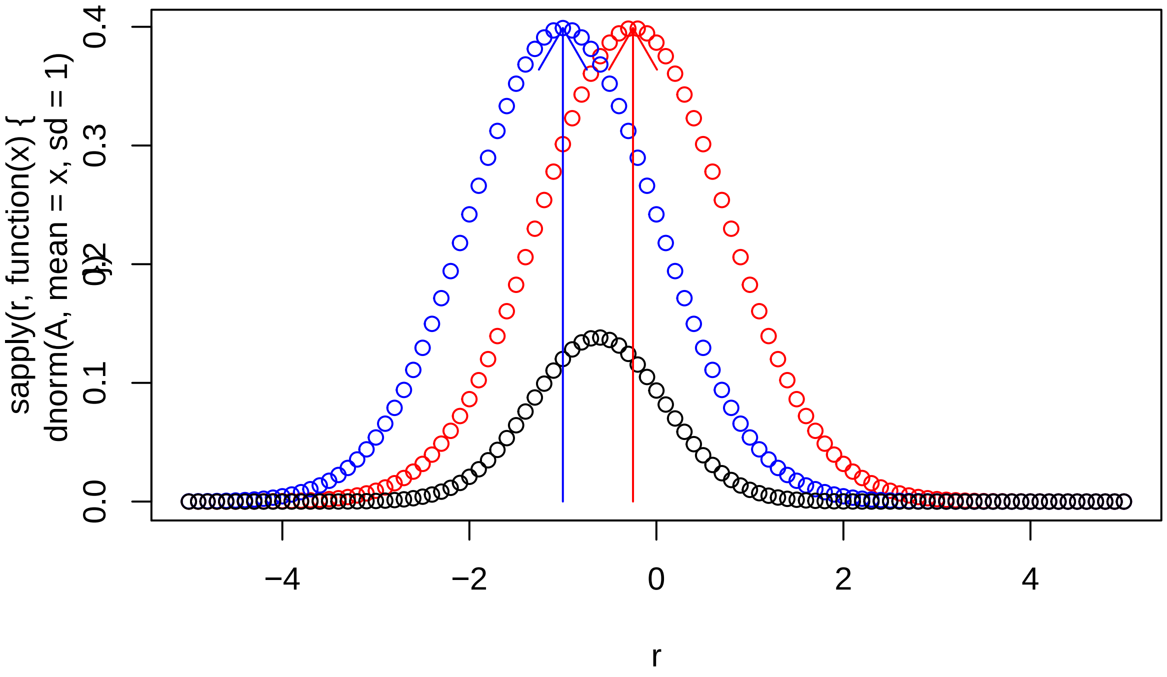
so

p(x1,x2|u) = p(x1)p(x2)

This means that the we must optimize the product of the probability density function.

First let plot this product value for various u values. With this plot we can clearly see that the function is optimized when the product is maximized.

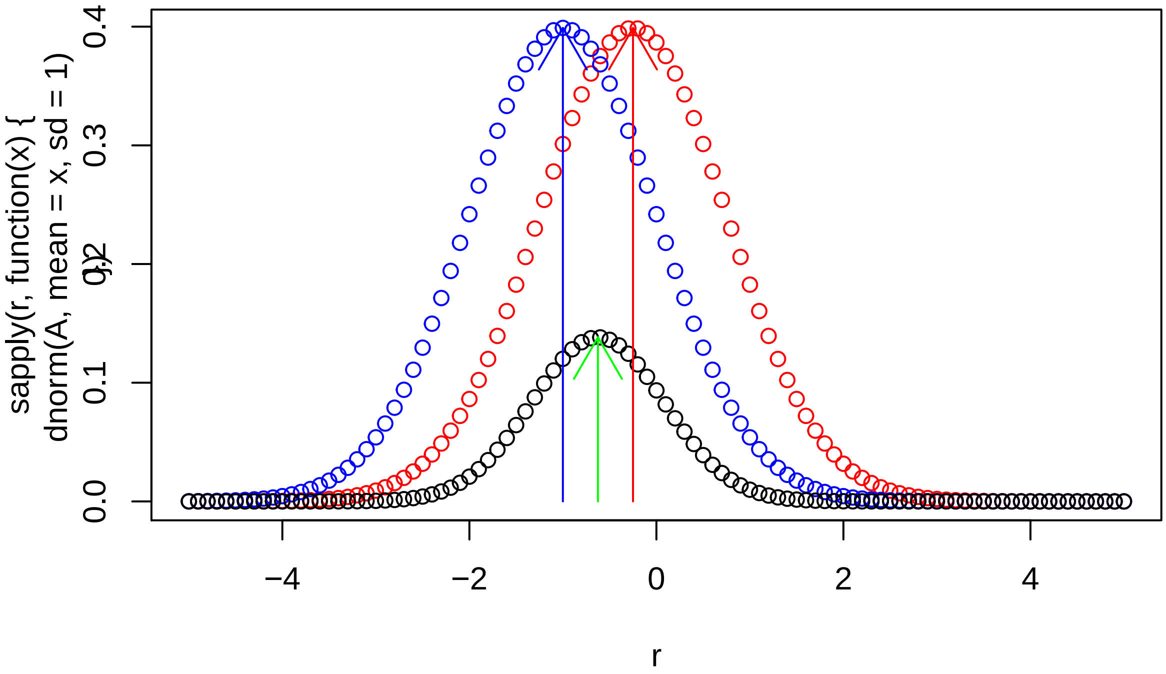
```
A <- -0.25
B <- -1
obs <- c(A,B)
r <- seq(-5,5, by = 0.1)
plot(r,sapply(r, function(x) {dnorm(A, mean = x, sd = 1)}), col = "red")
points(r,sapply(r, function(x) {dnorm(B, mean = x, sd = 1)}), col = "blue")
points(r,sapply(r, function(x) {dnorm(A, mean = x, sd = 1)*dnorm(B, mean = x, sd = 1)}))
arrows(A,0,A,dnorm(A,A,1), col = "red")
arrows(B,0,B,dnorm(B,B,1), col = "blue")
```



So, first lets try to optimize the function numeracally:

```
A <- -0.25
B <- -1
f <- function (x) {dnorm(A, mean = x, sd = 1)*dnorm(B, mean = x, sd = 1)}
xmax <- optimize(f, c(-5, 5), tol = 0.0001, maximum = TRUE)

A <- -0.25
B <- -1
obs <- c(A,B)
r <- seq(-5,5, by = 0.1)
plot(r,sapply(r, function(x) {dnorm(A, mean = x, sd = 1)}), col = "red")
points(r,sapply(r, function(x) {dnorm(B, mean = x, sd = 1)}), col = "blue")
points(r,sapply(r, function(x) {dnorm(A, mean = x, sd = 1)*dnorm(B, mean = x, sd = 1)}))
arrows(A,0,A,dnorm(A,A,1), col = "red")
arrows(B,0,B,dnorm(B,B,1), col = "blue")
arrows(xmax$maximum,0,xmax$maximum,xmax$objective, col = "green")
```



So now we now that: L(theta|x)
= L(theta|x1,x2)
= p(x1,x2|theta) (for simplicity theta={u})
= p(x1|x2)p(x2) for Gaussian(u,sd = 1)
= p(x1)p(x2) for Gaussian(u,sd = 1)

argmax(u) in p(x1)p(x2) for Gaussian(u,sd = 1)
maximum at -0.6249842

L(u = -0.6249842, sd = 1) = 0.1382762

so the best distribution is: Gaussian(u = -0.6249842, sd = 1)

But we have more than one parameter to optimize. Maybe it is possible to optimize even futher the function by using other values of sd.

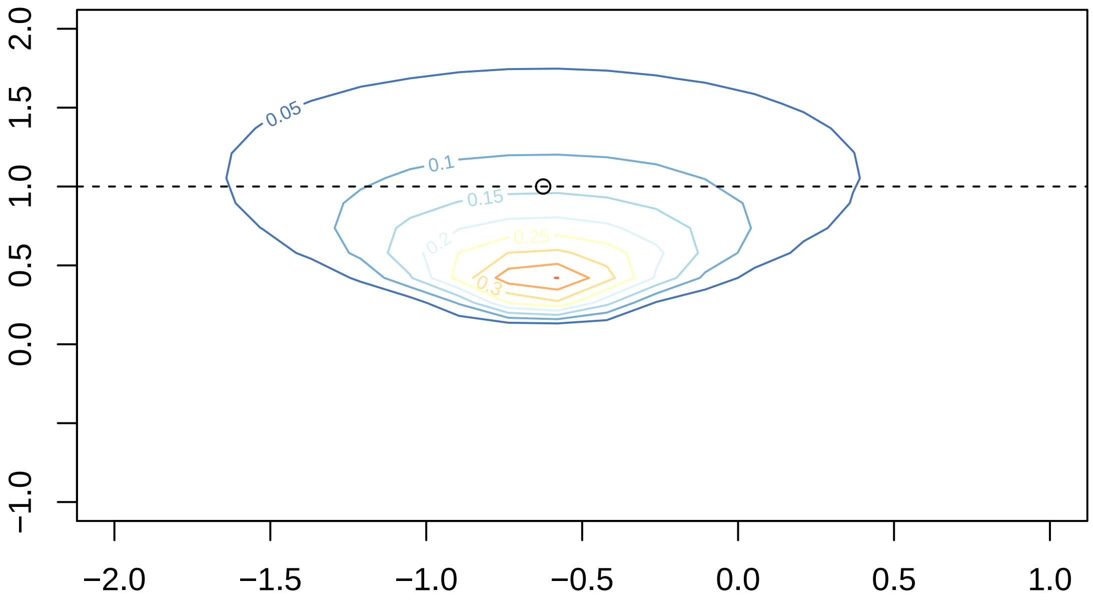
Let try some values:

```r
A <- -0.25
B <- -1
x = seq(-2, 1, length= 20)
y = seq(-1, 2, length= 20)
f = function(x, y) { dnorm(A, mean = x, sd = y)*dnorm(B, mean = x, sd = y) }
z = outer(x, y, f)
```

```
## Warning in dnorm(A, mean = x, sd = y): NaNs produced
```

```
## Warning in dnorm(B, mean = x, sd = y): NaNs produced
```

```r
z[is.na(z)] = 0
library(RColorBrewer)
contour(x,y,z,col=rev(brewer.pal(11, "RdYlBu")), nlevels = 11)
abline(h=1, lty=2)
points(xmax$maximum, 1)
```

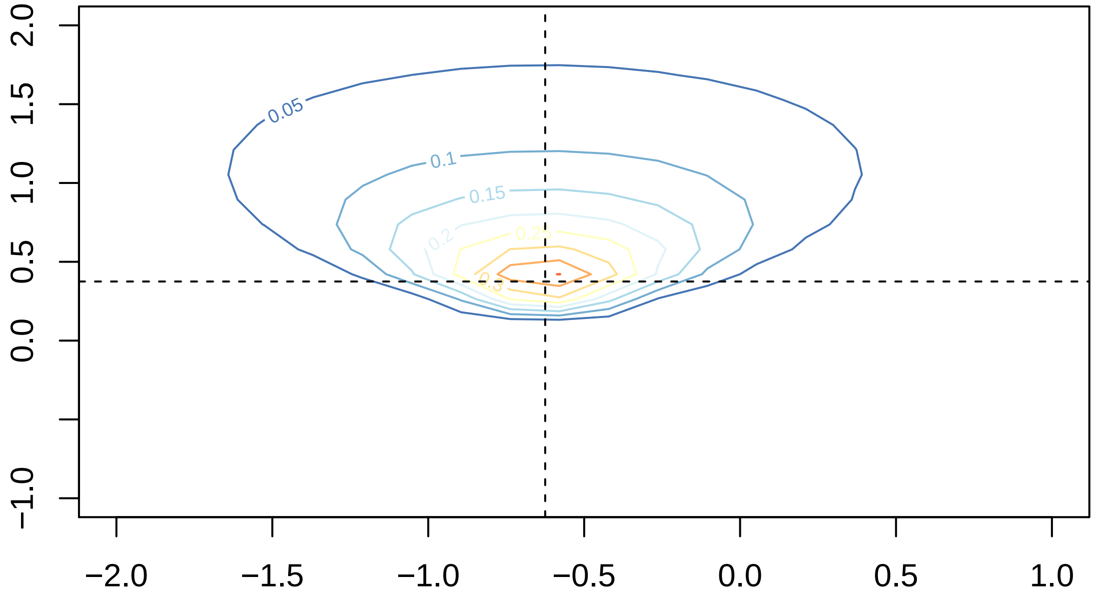


We have analyzed the problem with sd fixed as 1, and searched for the maximum likelihood in this scenario. If we analyze the above plot we will see that we searched the maximum following the dashed line and found it on the circled point.

Now it is clear that we hace now actually found the best solution to the problem.

```r
A <- -0.25
B <- -1
f = function(x) { dnorm(A, mean = x[1], sd = x[2])*dnorm(B, mean = x[1], sd = x[2]) }
max = optim( c(xmax$maximum,1), f ,control=list(fnscale=-1))

contour(x,y,z,col=rev(brewer.pal(11, "RdYlBu")), nlevels = 11)
abline(h=max$par[2], v = max$par[1], lty=2)
```



OK! So now we have the best Gaussian distribution for our two observations.

Gaussian

L(theta|x)
= L(theta|x1,x2)
= p(x1,x2|theta) (theta={u,sd})
= p(x1|x2)p(x2) for Gaussian(u,sd)
= p(x1)p(x2) for Gaussian(u,sd)

argmax(u,sd) in p(x1)p(x2) for Gaussian(u,sd)
maximum at (u = -0.6250109, sd = 0.375003)
L({u = -0.6250109, sd = 0.375003}|x1,x2) = 0.4163544

So taking sd into consideration we improved out likelihood from 0.1382762 to 0.4163544

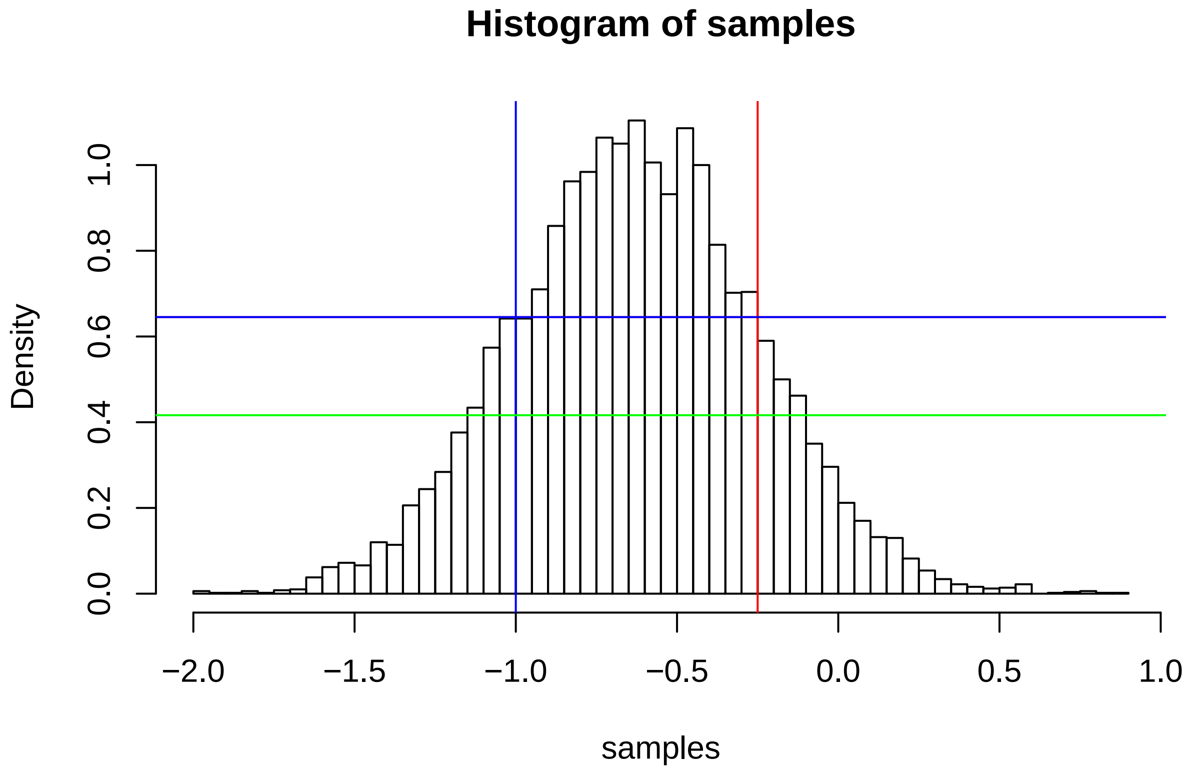
Likelihood Interpretation

OK. We have a likelihood of 0.4163544, but what this means? We know that L(theta|x) = p(x|theta). So this means that we have calculated the probability of the product of two variables sampled from a distribution with the found configuration to be A*B.

Let us see if this is true.

```r
A <- -0.25
B <- -1
samples <- rnorm(10000, mean = max$par[1], sd = max$par[2])
hist(samples, breaks = 48, freq = FALSE)
```

```r
abline(v=A, col = "red")
abline(v=B, col = "blue")
abline(h=dnorm(A, mean = max$par[1], sd = max$par[2])*dnorm(B, mean = max$par[1], sd = max$par[2]), col
abline(h=dnorm(A, mean = max$par[1], sd = max$par[2]), col = "red")
abline(h=dnorm(B, mean = max$par[1], sd = max$par[2]), col = "blue")
```


Histogram of samples
Density
samples