

Lab 2

Simon Jakobsson (simja649), Erik Halvarsson (eriha353) och Gustav Hanstorp(gusha433)

10/1/2020

Uppgift 3.1.1

Del 1

```
set.seed(4711)
x1 <- rgamma(n=10, shape=4, scale=1)

x2 <- rgamma(n=100, shape=4, scale=1)

llgamma <- function(x, alpha, beta){
  n <- length(x)
  return (n*(alpha*log(beta)-lgamma(alpha))+(alpha-1)*sum(log(x))-beta*sum(x))
}

llgamma(x1, alpha=2,beta=2)
```

```
## [1] -75.18981
```

Del 2

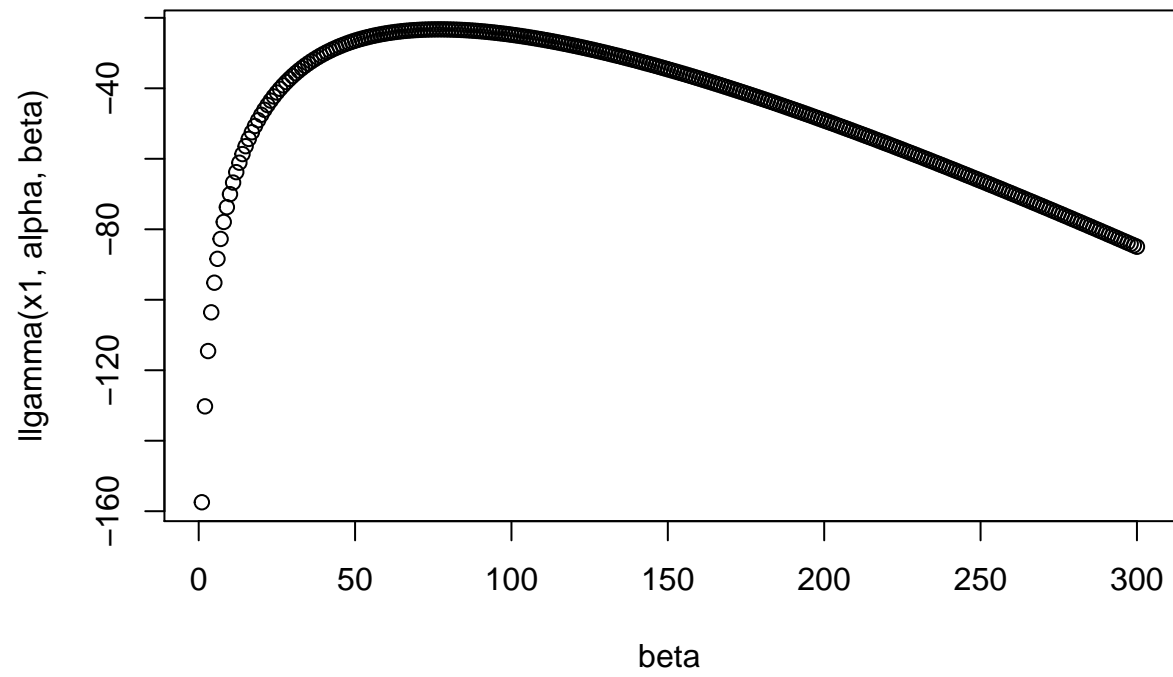
```
beta <- seq(0.01, 3, 0.01)
beta.1 = llgamma(x1, alpha=4, beta)
beta.2 = llgamma(x2, alpha=4, beta)
print(beta[which.max(beta.1)])
```

```
## [1] 0.77
```

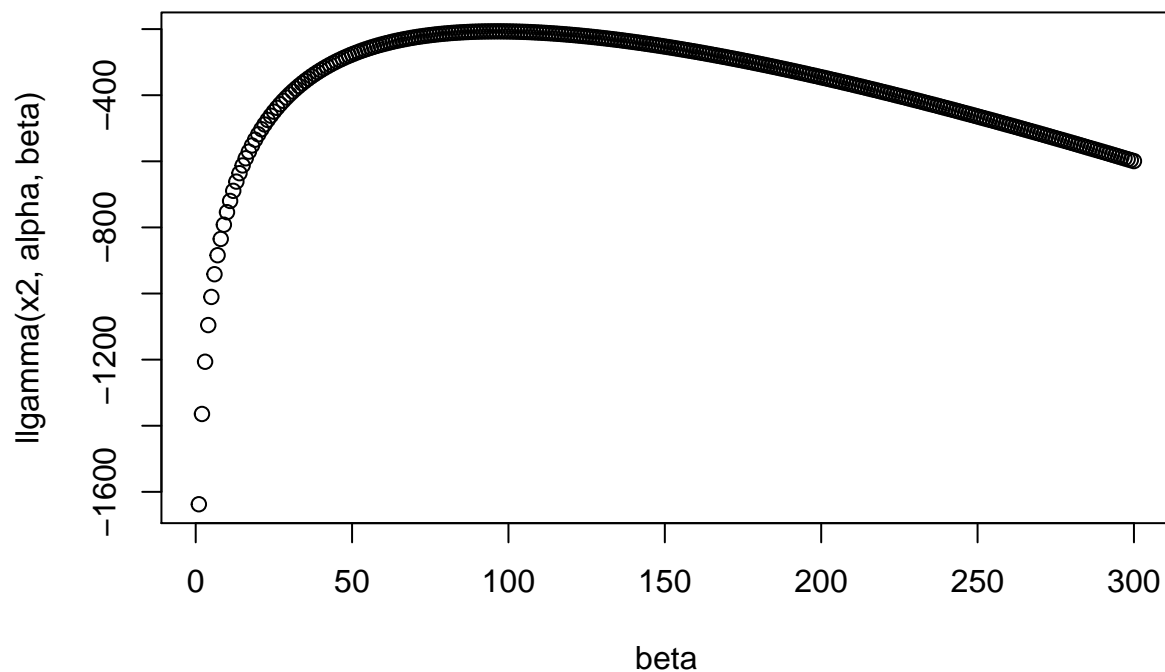
```
print(beta[which.max(beta.2)])
```

```
## [1] 0.96
```

```
plot(beta.1, xlab="beta", ylab="llgamma(x1, alpha, beta)")
```



```
plot(beta.2, xlab="beta", ylab="llgamma(x2, alpha, beta)")
```



Del 3

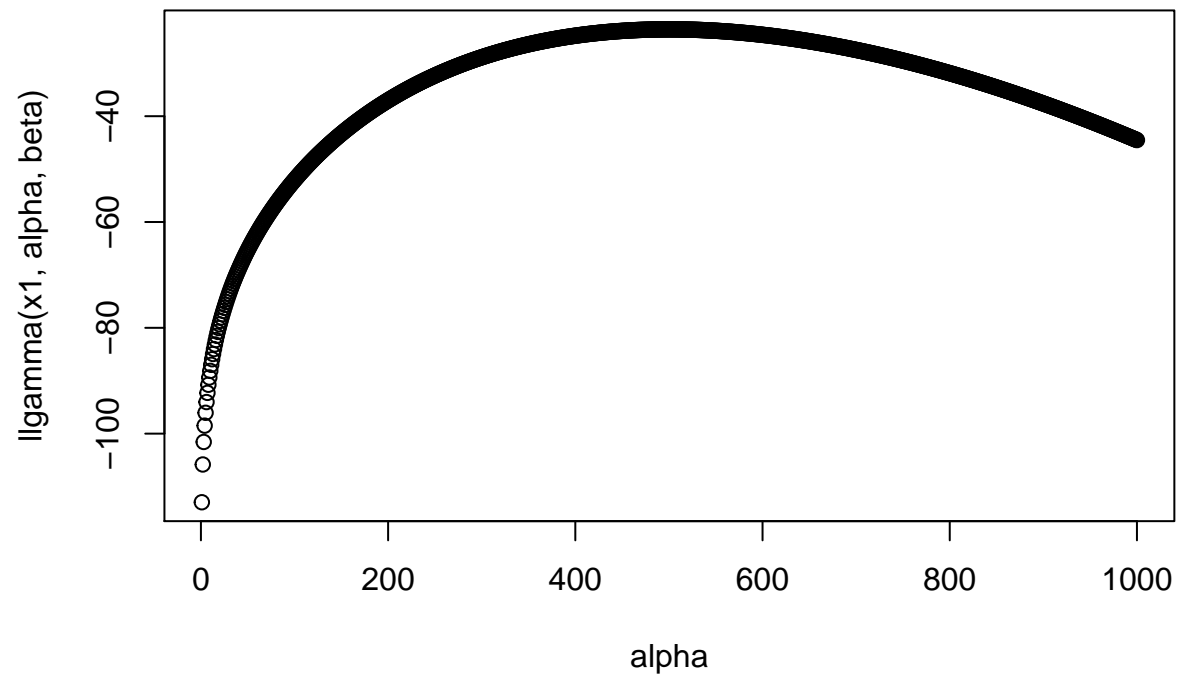
```
alpha <- seq(0.01, 10, 0.01)
alpha.1 = llgamma(x1, alpha, beta=1)
alpha.2 = llgamma(x2, alpha, beta=1)
print(alpha[which.max(alpha.1)])
```

```
## [1] 5
```

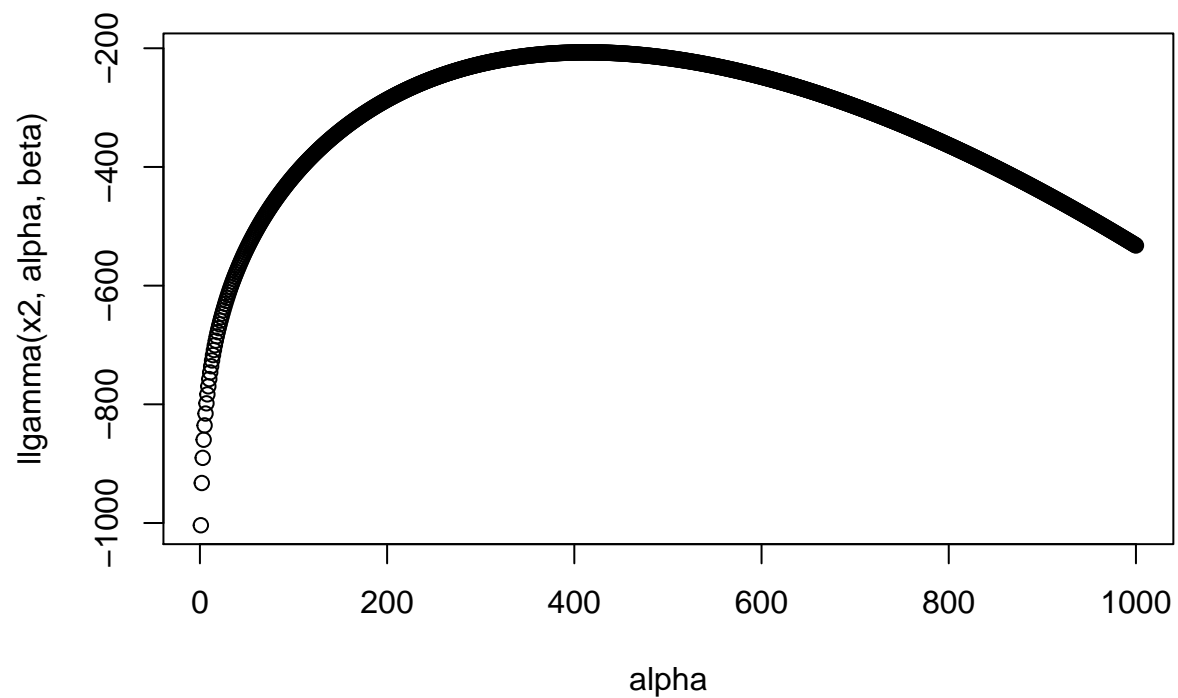
```
print(alpha[which.max(alpha.2)])
```

```
## [1] 4.13
```

```
plot(alpha.1, xlab="alpha", ylab="llgamma(x1, alpha, beta)")
```



```
plot(alpha.2, xlab="alpha", ylab="llgamma(x2, alpha, beta)")
```



Del 4

```
llnormal <- function(x, mu, sigma2){
  n <- length(x)
  return ((-n/2)*log(2*pi)- (n/2)*log(sigma2)- (1/2*sigma2)*sum((x-mu)^2))
}

llnormal(x = x1, mu = 2, sigma2 = 1)
```

```
## [1] -87.25743
```

Del 5

```
mu <- seq(0, 10, 0.01)
mu.1 <- c()
for (i in mu){
  mu.1 <- append(llnormal(x1, i, sigma2=1), mu.1)
}
mu.2 <- c()
for (i in mu){
  mu.2 <- append(llnormal(x2, i, sigma2=1), mu.2)
}
```

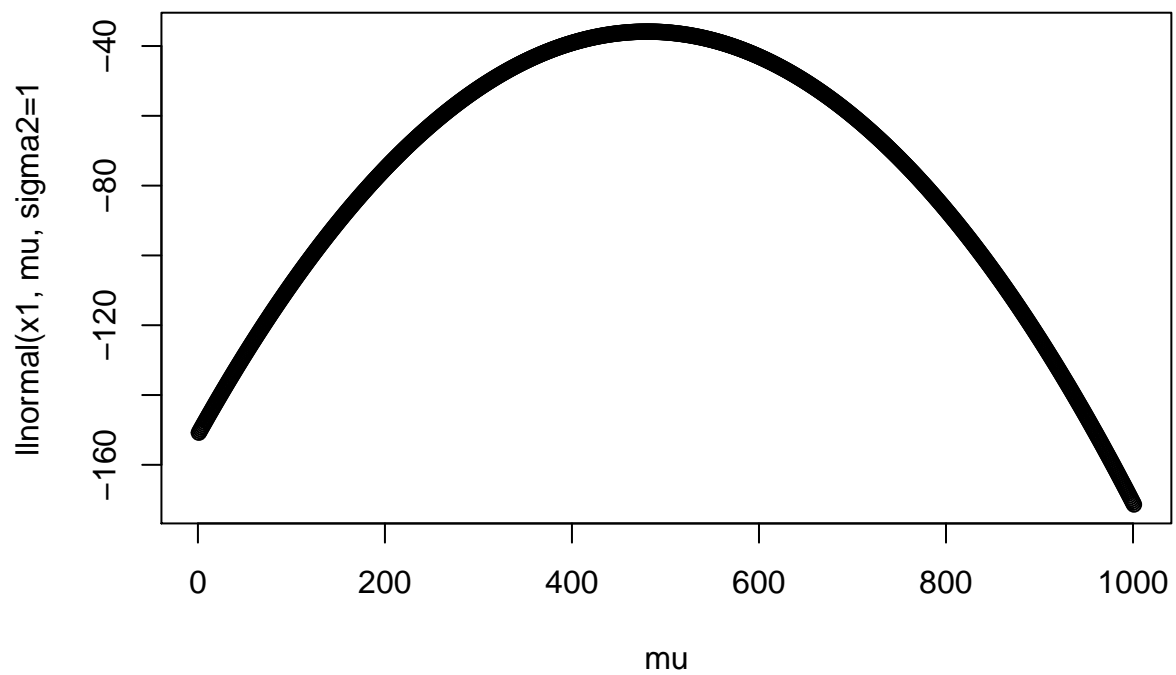
```
print(mu[which.max(mu.1)])
```

```
## [1] 4.79
```

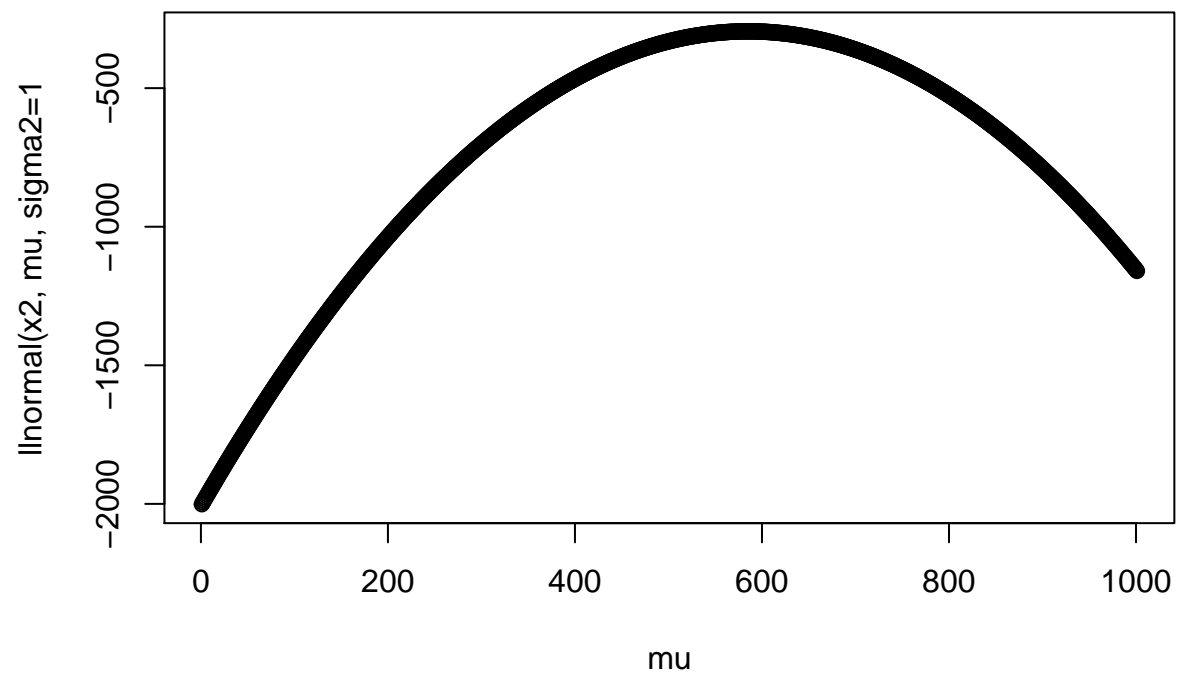
```
print(mu[which.max(mu.2)])
```

```
## [1] 5.84
```

```
plot(mu.1, xlab="mu", ylab="llnormal(x1, mu, sigma2=1)")
```

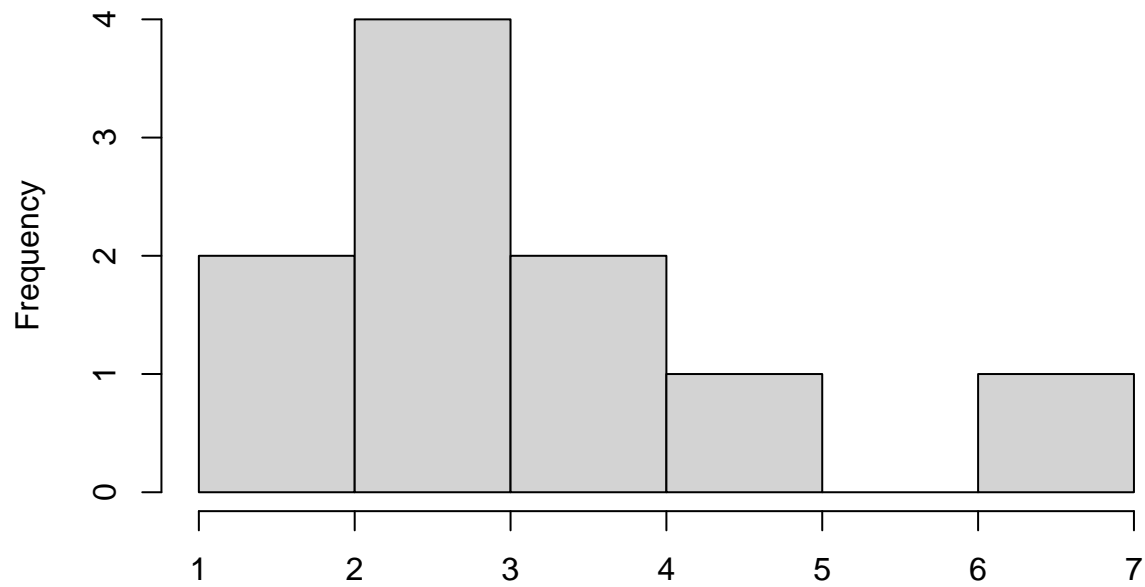


```
plot(mu.2, xlab="mu", ylab="llnormal(x2, mu, sigma2=1)")
```



```
hist(rgamma(n = 10, shape = alpha[which.max(alpha.1)], scale = beta[which.max(beta.1)]), main = "X1, ma
```

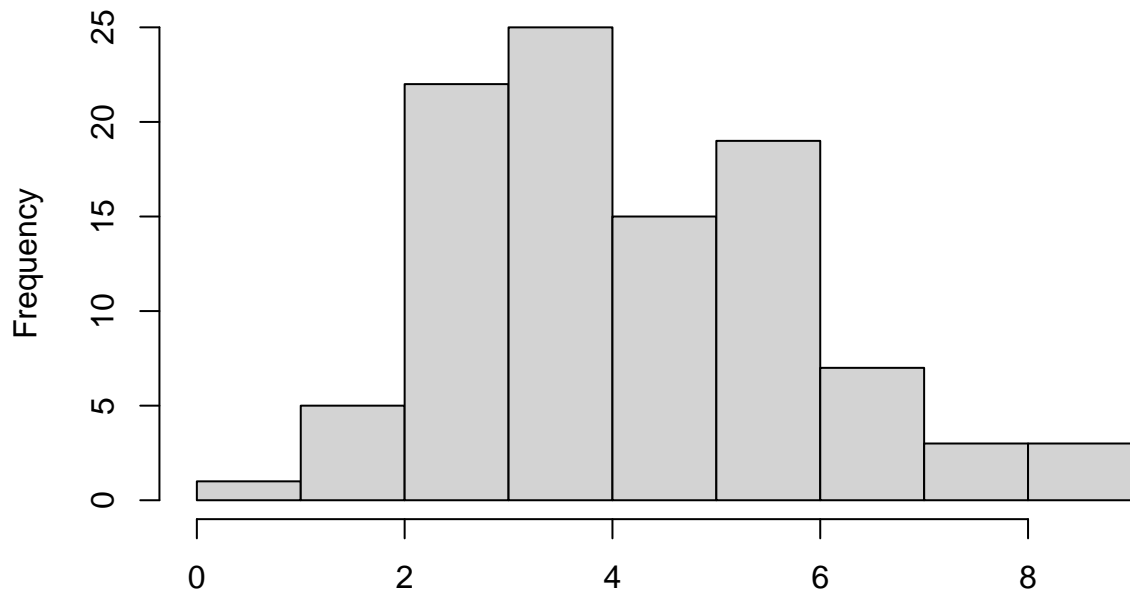
X1, max alpha and beta



`rgamma(n = 10, shape = alpha[which.max(alpha.1)], scale = beta[which.max(beta.1)])`

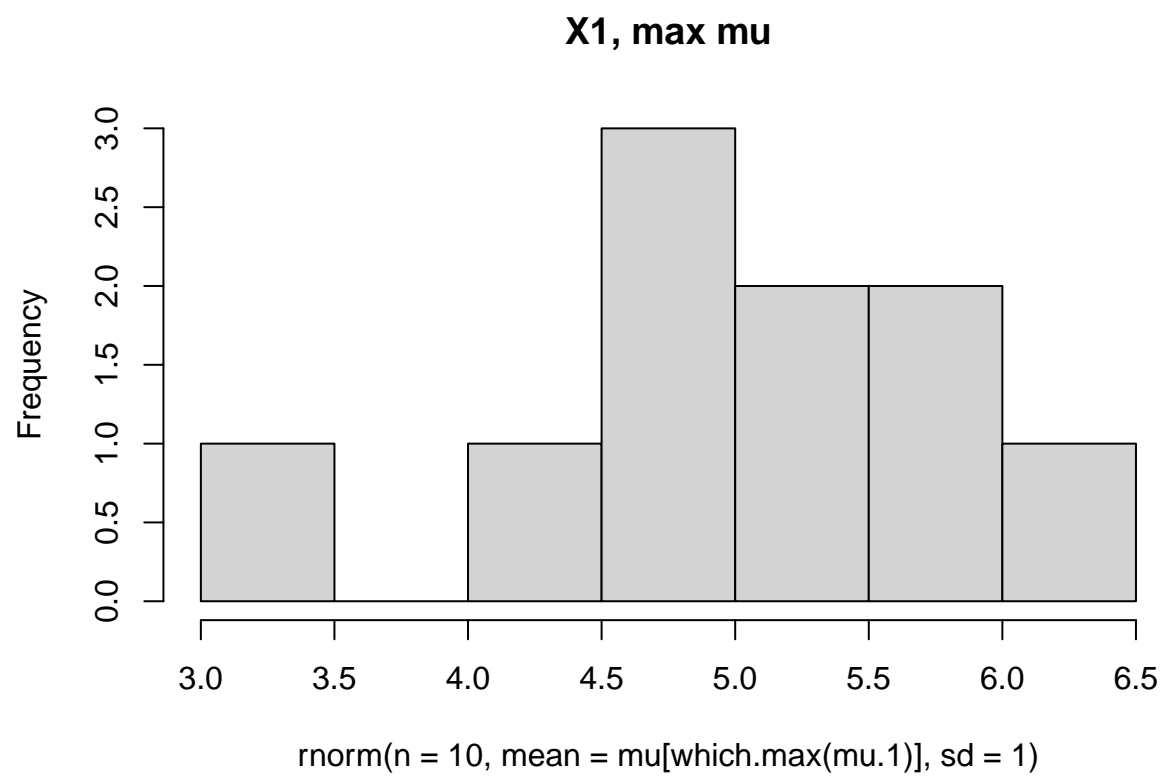
```
hist(rgamma(n = 100, shape = alpha[which.max(alpha.2)], scale = beta[which.max(beta.2)]), main = "X2, m
```


X2, max alpha and beta

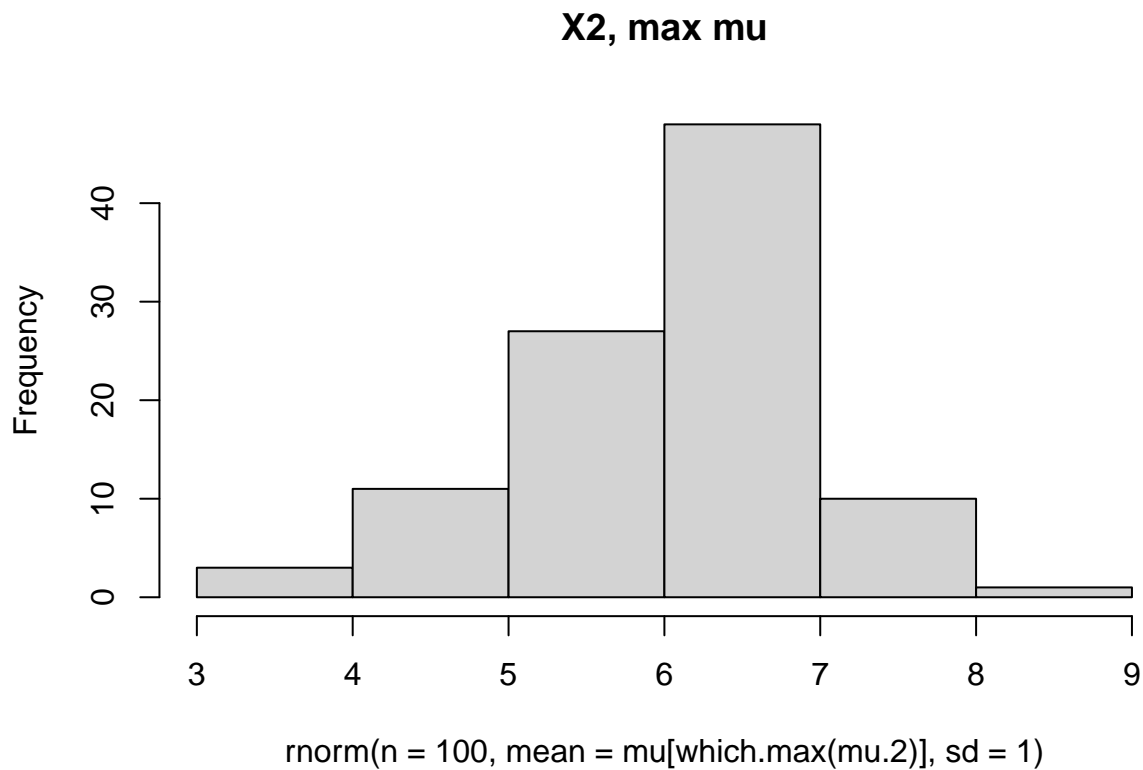


`rgamma(n = 100, shape = alpha[which.max(alpha.2)], scale = beta[which.max(beta.2`

```
hist(rnorm(n = 10, mean = mu[which.max(mu.1)], sd = 1), main = "X1, max mu")
```



```
hist(rnorm(n = 100, mean = mu[which.max(mu.2)], sd = 1), main = "X2, max mu")
```



```
# TODO: Visualisera saker
```

I histogrammen där $n = 10$ är båda resultaten oklara och inget är direkt bättre än det andra, däremot när vi höjer n till 100 så blir det en tydlig skillnad där normalfördelningen passar bättre till datamaterialet.

##Uppgift 3.2.1

```
gamma_beta_mle <- function(x, alpha){
  return((length(x)*alpha)/sum(x))
}
gamma_beta_mle(x = x1, alpha = 4)
```

```
## [1] 0.7683785
```

```
gamma_beta_mle(x = x2, alpha = 4)
```

```
## [1] 0.9619473
```

Vi ser att ju mer dragningar man gör ju närmare kommer man ordinarie värdet för beta (vilket då är 1).

##Uppgift 3.2.2 ### Del 1

```
norm_mu_le <-function(x){
  return((1/length(x)) * sum(x))
}
```

```
norm_sigma2_mle <- function(x){
  xbar <- norm_mu_le(x)
  return((1/length(x))*(sum((x - xbar)^2)))
}
```

```
test_x <- 1:10
norm_sigma2_mle(test_x)
```

```
## [1] 8.25
```

Del 2

```
set.seed(42)
draw10 <- rnorm(n = 10, mean = 10, sd = 2)
draw10000 <- rnorm(n = 10000, mean = 10, sd = 2)

norm_mu_le(draw10)
```

```
## [1] 11.09459
```

```
norm_mu_le(draw10000)
```

```
## [1] 9.9762
```

```
norm_sigma2_mle(draw10)
```

```
## [1] 2.512709
```

```
norm_sigma2_mle(draw10000)
```

```
## [1] 4.048198
```

Ju fler dragningar ju bättre kan vi representera de "riktiga" värdena

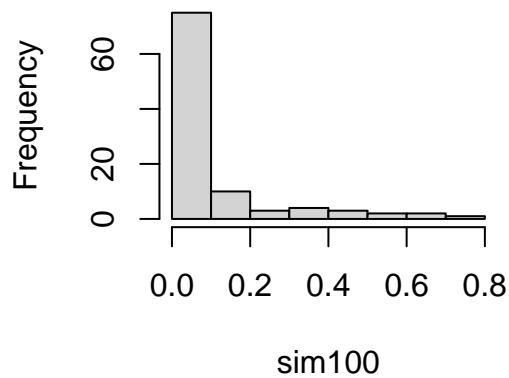
##Uppgift 3.3.1 ### Del 1

```
llbeta <- function(par,x){
  return (-sum(dbeta(x, shape1 = par[1], shape = par[2], log = TRUE)))
}
```

Del 2

```
sim100 <- rbeta(n = 100, shape1 = 0.2, shape2 = 2)
hist(sim100)
```

Histogram of sim100



Del 3

```
opt_res <- optim(par= c(1,2), fn= llbeta, x = sim100, method = 'L-BFGS-B', lower= c(.Machine$double.eps, 0))
opt_res$par
```

```
## [1] 0.2211386 2.1439233
```

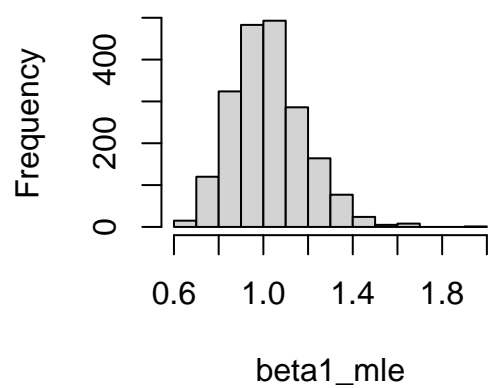
##Uppgift 3.4.1 ### Del 1

```
beta1_mle <- vector()
beta2_mle <- vector()
mu1 <- vector()
mu2 <- vector()
sigma1 <- vector()
sigma2 <- vector()

for (i in 1:2000){
  x1 <- rgamma(n= 10, shape = 4, scale = 1)
  x2 <- rgamma(n = 10000, shape = 4, scale = 1)
  beta1_mle[i] <- gamma_beta_mle(x = x1, alpha = 4)
  beta2_mle[i] <- gamma_beta_mle(x = x2, alpha = 4)
  y1 <- rnorm(n = 10, mean = 10, sd = 4)
  y2 <- rnorm(n = 10000, mean = 10, sd = 4)

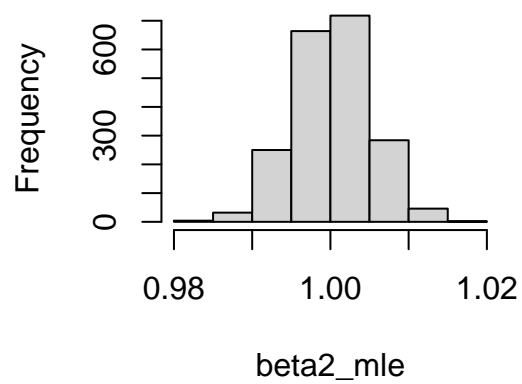
  mu1[i] <- norm_mu_le(x = y1)
  mu2[i] <- norm_mu_le(x = y2)
  sigma1[i] <- norm_sigma2_mle(x = y1)
  sigma2[i] <- norm_sigma2_mle(x = y2)
}
hist(beta1_mle)
```

Histogram of beta1_mle



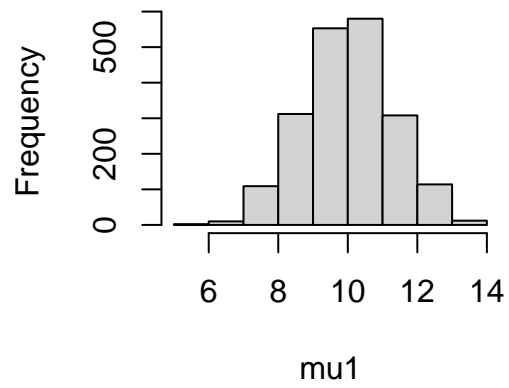
```
hist(beta2_mle)
```

Histogram of beta2_mle



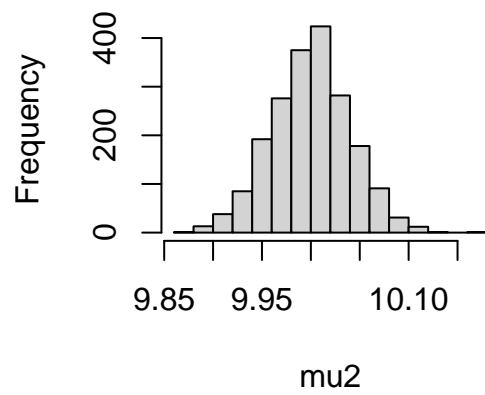
```
hist(mu1)
```

Histogram of mu1



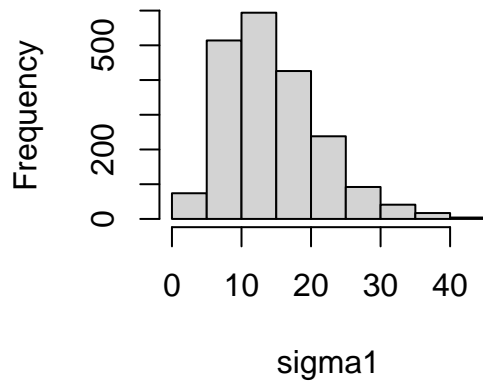
```
hist(mu2)
```

Histogram of mu2



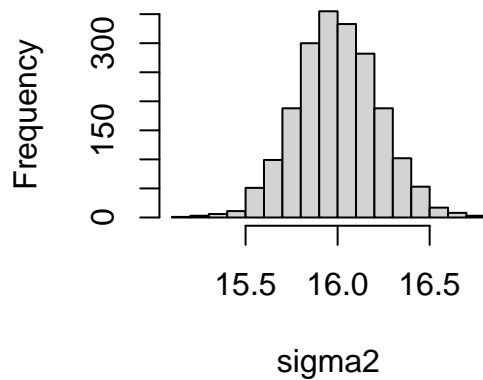
```
hist(signal)
```

Histogram of sigma1



```
hist(sigma2)
```

Histogram of sigma2



(spridningen) sant går mot normalfördelningen

Desto fler dragningar vi har, desto mindre blir varansen

###Del 2

```
beta1_mle <- vector()
beta2_mle <- vector()
mu1 <- vector()
mu2 <- vector()
sigma1 <- vector()
sigma2 <- vector()
x1 <- rgamma(n= 10, shape = 4, scale = 1)
x2 <- rgamma(n = 10000, shape = 4, scale = 1)
y1 <- rnorm(n = 10, mean = 10, sd = 4)
y2 <- rnorm(n = 10000, mean = 10, sd = 4)
for (i in 1:2000){
```



```

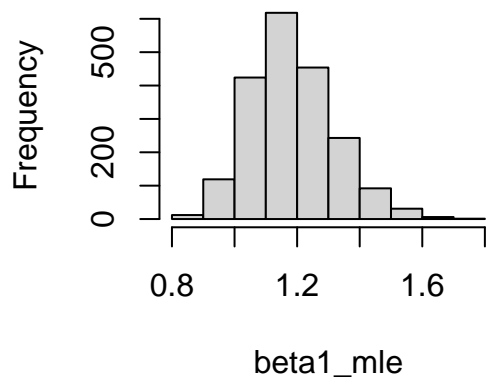
beta1_mle[i] <- gamma_beta_mle(x = sample(x1, 10, replace = TRUE), alpha = 4)
beta2_mle[i] <- gamma_beta_mle(x = sample(x2, 10000, replace = TRUE), alpha = 4)

mu1[i] <- norm_mu_le(x = sample(y1, 10, replace = TRUE))
mu2[i] <- norm_mu_le(x = sample(y2, 10000, replace = TRUE))
sigma1[i] <- norm_sigma2_mle(x = sample(y1, 10, replace = TRUE))
sigma2[i] <- norm_sigma2_mle(x = sample(y2, 10000, replace = TRUE))

}
hist(beta1_mle)

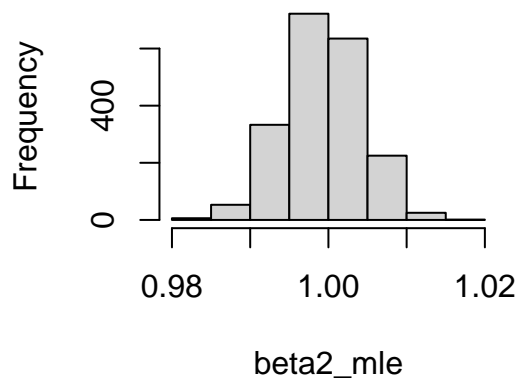
```

Histogram of beta1_mle

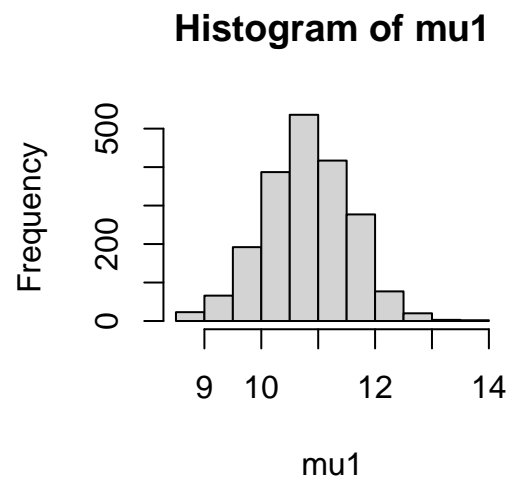


```
hist(beta2_mle)
```

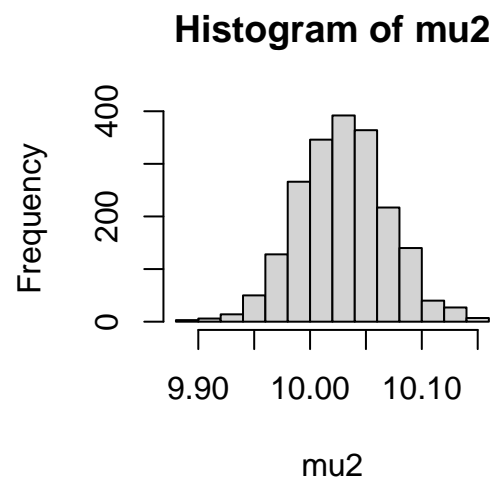
Histogram of beta2_mle



```
hist(mu1)
```

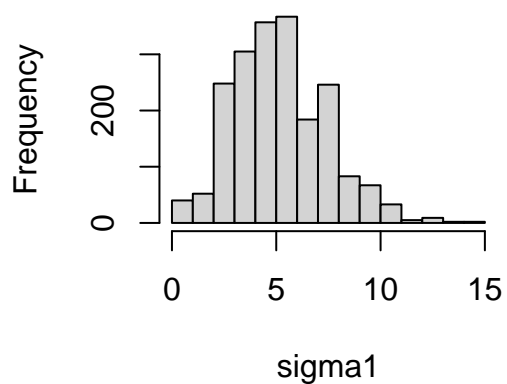


```
hist(mu2)
```



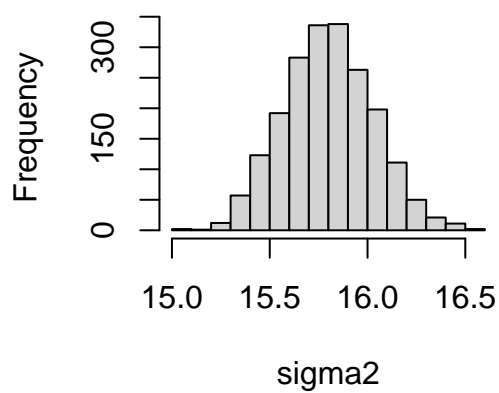
```
hist(sigma1)
```

Histogram of sigma1



```
hist(sigma2)
```

Histogram of sigma2



Vi ser att som i förra uppgiften så minskar variansen på alla samt går mot normalfördelningen. Vi ser även att väntevärdet skiftar på alla fördelningarna beroende på hur många samples vi har, men väldigt markant från sigma1 till sigma2.