October-2021-Exam

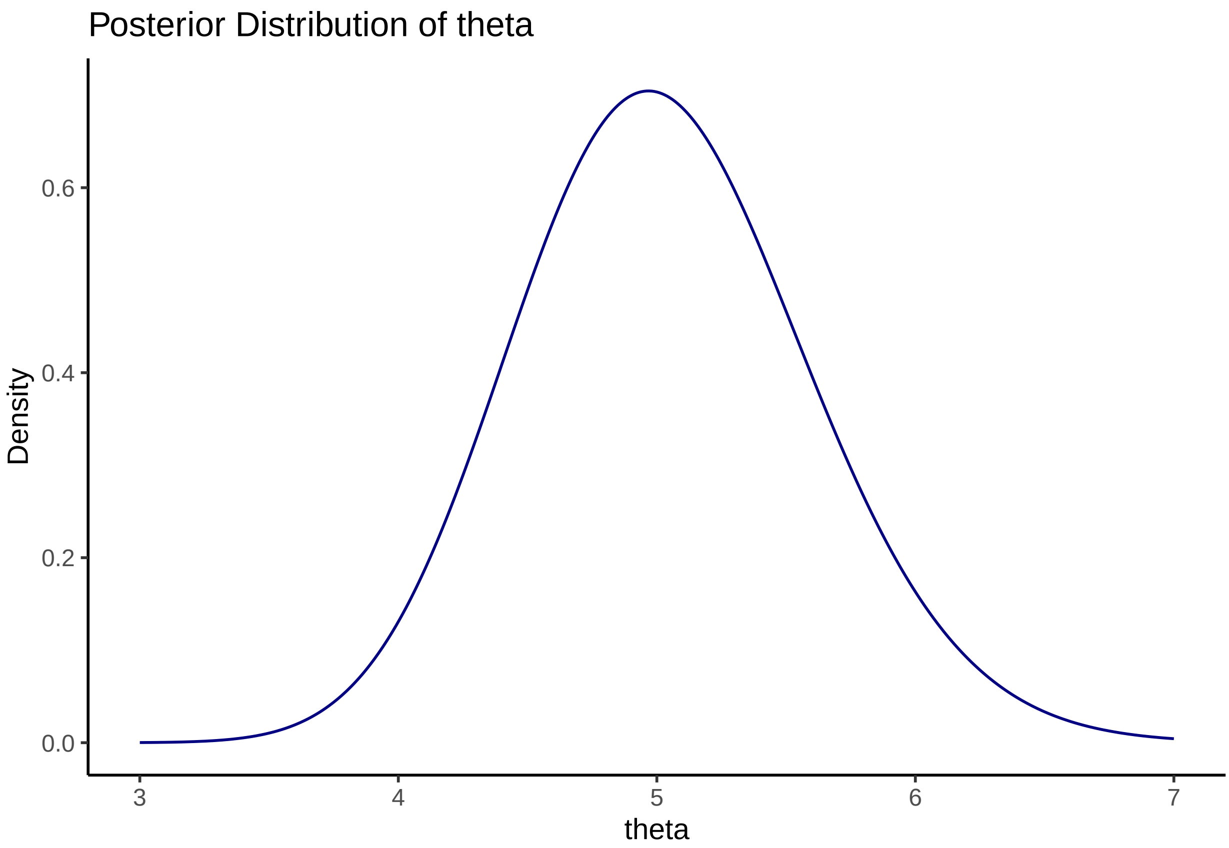
Christophoros Spyretos

Exercise 1 *Derivations and comparing posterior distributions*

Task a,b and c is a hand written solution

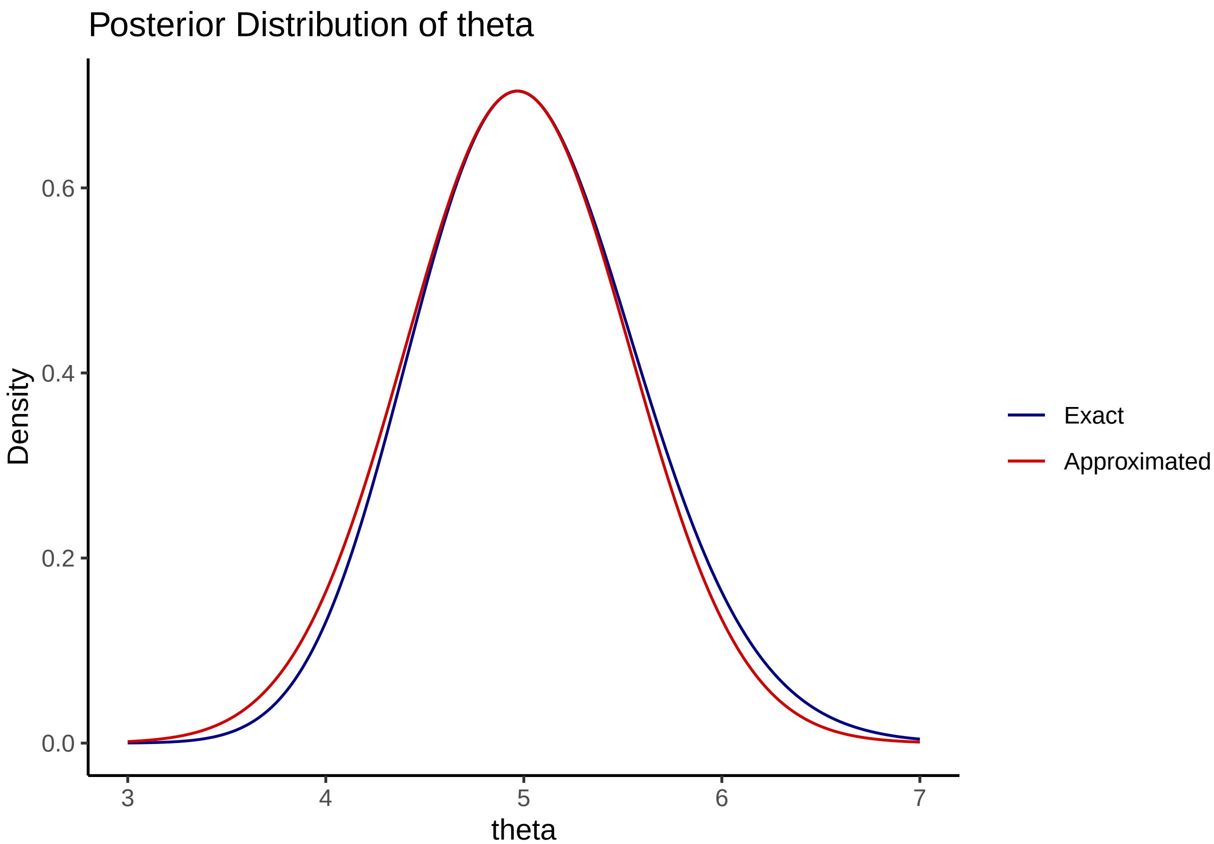
Task d

```
set.seed(1)

LogPost <- function(theta, n, sumx){

  LogLik <- sumx*log(theta) - n*theta
  LogPrior <- 2*log(theta) - 0.5*theta

  return(LogLik + LogPrior)
}

theta <- seq(3,7,0.01)
n <- 15
sumx <- 75

posterior <- LogPost(theta, n, sumx)
posterior_dens <- exp(posterior)
norm_posterior <- posterior_dens/(0.01*sum(posterior_dens))

df_posterior <- data.frame("theta" = theta, "dens" = norm_posterior)

ggplot(df_posterior) +
  geom_line(aes(x=theta,y=dens), color = "navy") +
  ggtitle("Posterior Distribution of theta") +
  ylab("Density") +
  theme_classic()
```



Task e

```
OptimRes <- optim(3,LogPost,gr=NULL,n,sumx,method=c("L-BFGS-B"),lower=3,control=list(fnscale=-1),hessian

approx_theta <- dnorm(theta, mean = OptimRes$par, sd = sqrt(diag(-solve(OptimRes$hessian))))

df_posterior$approx <- approx_theta

ggplot(df_posterior) +
  geom_line(aes(x=theta,y=dens, color = "navy")) +
   geom_line(aes(x=theta,y=approx, color = "red3")) +
  theme(legend.position="right") +
  scale_color_identity(guide = "legend",
                       name = "",
                       breaks=c("navy", "red3"),
                       labels = c("Exact",
                                  "Approximated")) +
  ggtitle("Posterior Distribution of theta") +
  ylab("Density") +
  theme_classic()
```



The posterior approximation is very accurate, however the exact posterior distribution is slightly skewed to the right.

Task f

```
set.seed(12345)

N <- 1000
T_x_rep <- matrix(NA, nrow = N,ncol = 1)

for (i in 1:N) {
  theta <- rgamma(1, shape = 3 + sumx, rate = n + 0.5)
  x_rep <- rpois(n,theta)
  T_x_rep[i,1] <- max(x_rep)
}

prob <- mean(T_x_rep > 14)
```

The posterior predictive p-value is 0.002. It is not reasonable to think that the maximum value of 14 from Gunnar originates from the Poisson distribution in this problem, because the probability is low.

Exercise 2

```
source("ExamData.R")
```

Task a

```
set.seed(12345)

mu_0 <- rep(0,3)
sigma_0 <- 16*diag(3)
nIter <- 10000

#BayesLogitReg <- function(y, X, mu_0, Sigma_0, nIter)
PosteriorDraws <- BayesLogitReg(y,X,mu_0,sigma_0,nIter)

betas <- PosteriorDraws$betaSample

interval <- quantile(betas[,2], probs =c(0.05,0.95))

df_interval <- data.frame("lower_bound" = interval[1], "upper_bound" = interval[2])
colnames(df_interval) <- c("lower bound", "upper bound")
rownames(df_interval) <- c("90% Equal Tail Credible Interval")
knitr::kable(df_interval)
```

	lower bound	upper bound
90% Equal Tail Credible Interval	0.1905025	1.891257

It is 90 % posterior probability that beta_1 is on the interval (0.19,1.89).

Task b

```
prob <- mean(betas[,3] > 0)
```

The posterior probability that $\beta_2 > 0$ approximately equals 0.88, the probability shows that x_2 has a positive effect on p_i when x_2 changes from 0 to 1.

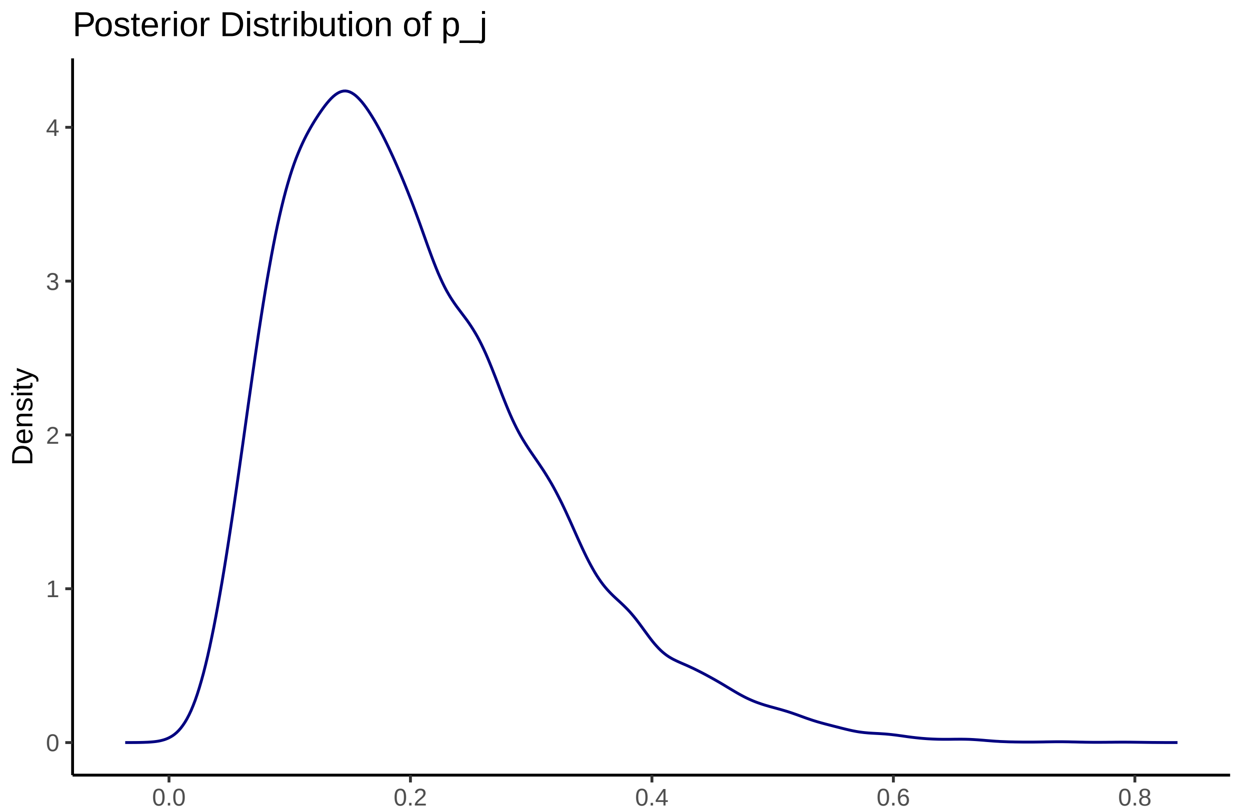
Task c

```
prob_joint <- mean(betas[,2] + betas[,3]> 0)
```

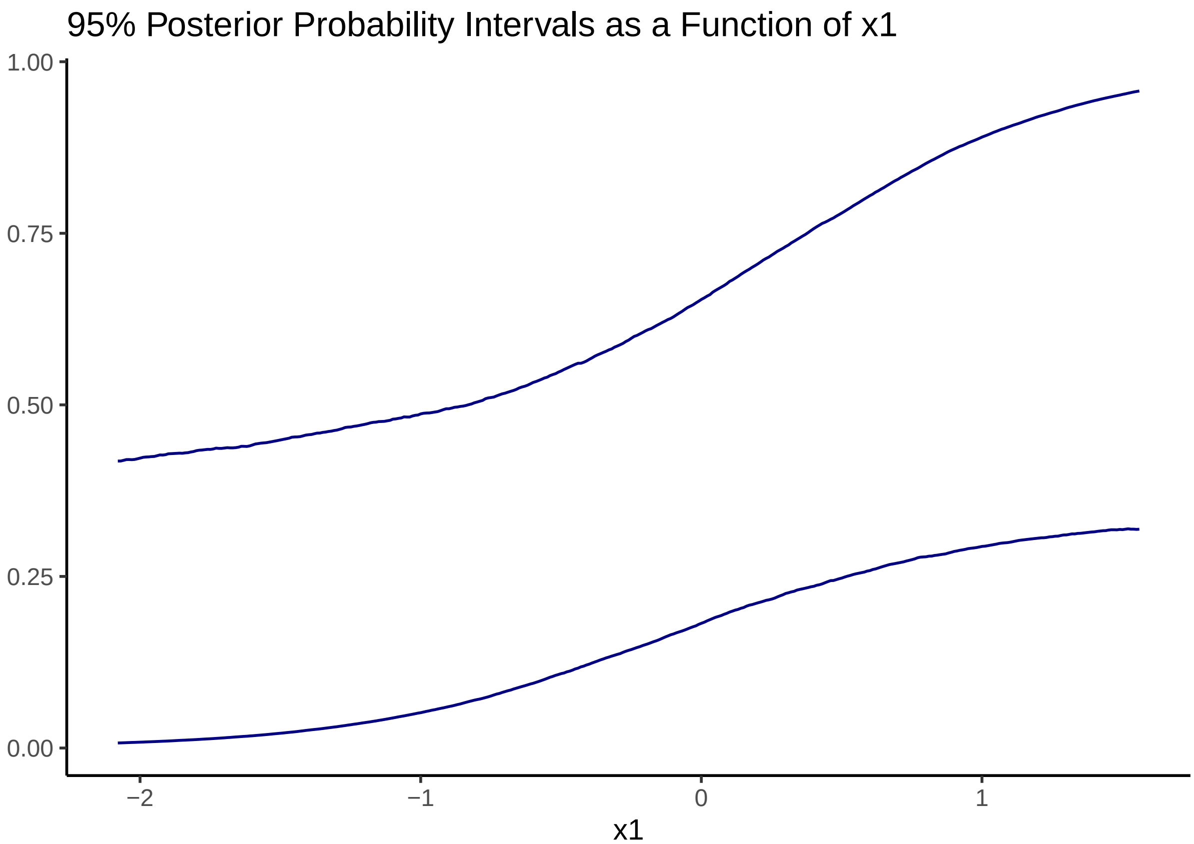
The joint posterior probability that both $\beta_1 > 0$ and $\beta_2 > 0$ is approximately 0.96.

Task d

```
p_j <- exp(betas[,1])/(1 + exp(betas[,1]))

p_j_dens <- density(p_j)

df_p_j <- data.frame("x" = p_j_dens$x, "y" = p_j_dens$y)

ggplot(df_p_j) +
  geom_line(aes(x=x, y=y), color = "navy") +
  ggtitle("Posterior Distribution of p_j") +
  xlab("") +
  ylab("Density") +
  theme_classic()
```




```
x1_grid <- seq(min(X[,2]), max(X[,2]), 0.01)
p_k <- matrix(NA, nrow = length(x1_grid), ncol = 1)
intervals <- matrix(NA, nrow = length(x1_grid), ncol = 2)
for (i in 1:length(x1_grid)){
  pred <- betas %*% c(1,x1_grid[i],1)
  p_k <- exp(pred)/(1 + exp(pred))
  intervals[i,] <- quantile(p_k, probs = c(0.025,0.975))
}

df_intervals <- as.data.frame(intervals)
colnames(df_intervals) <- c("low", "upper")
df_intervals$x1 <- x1_grid

ggplot(df_intervals) +
  geom_line(aes(x=x1, y=low), color = "navy") +
  geom_line(aes(x=x1, y=upper), color = "navy") +
  ggtitle("95% Posterior Probability Intervals as a Function of x1") +
  ylab("") +
  theme_classic()
```

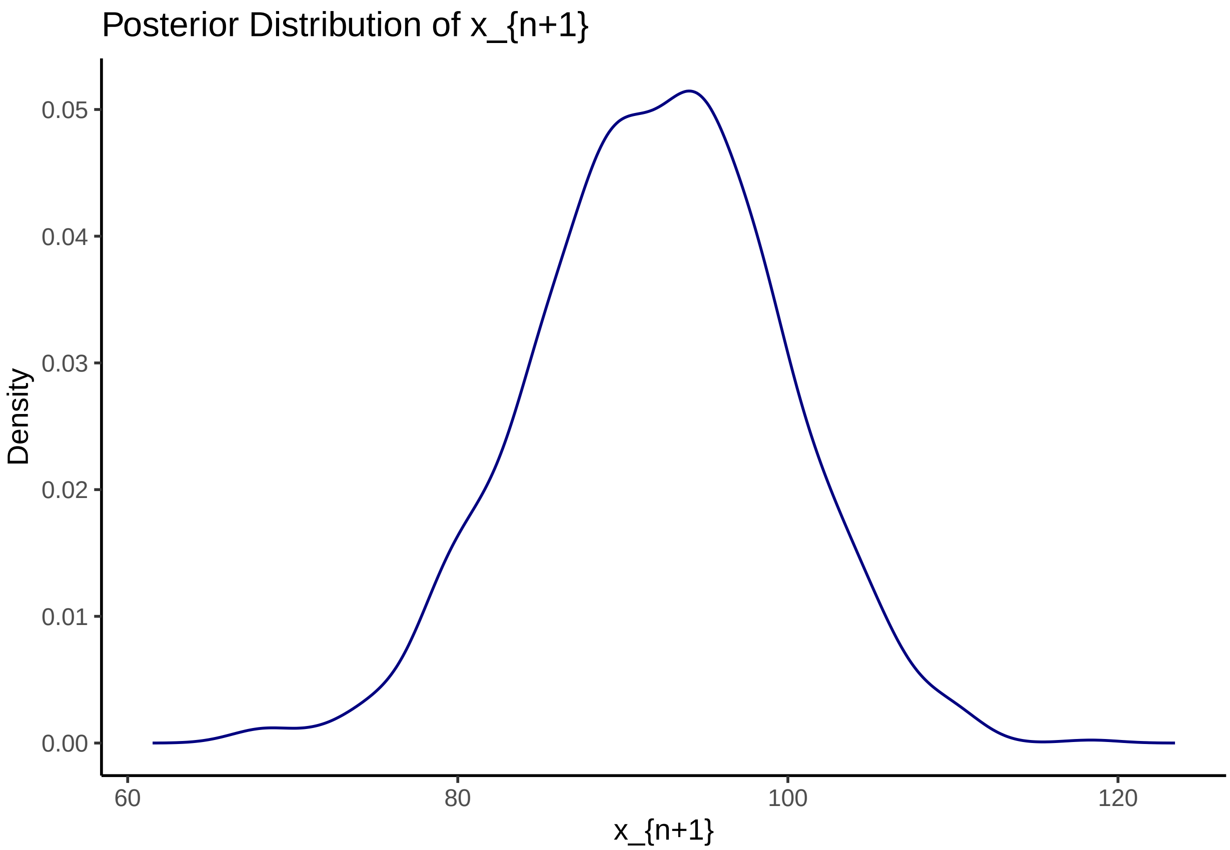


Exercise 3

Task a is a hand written solution

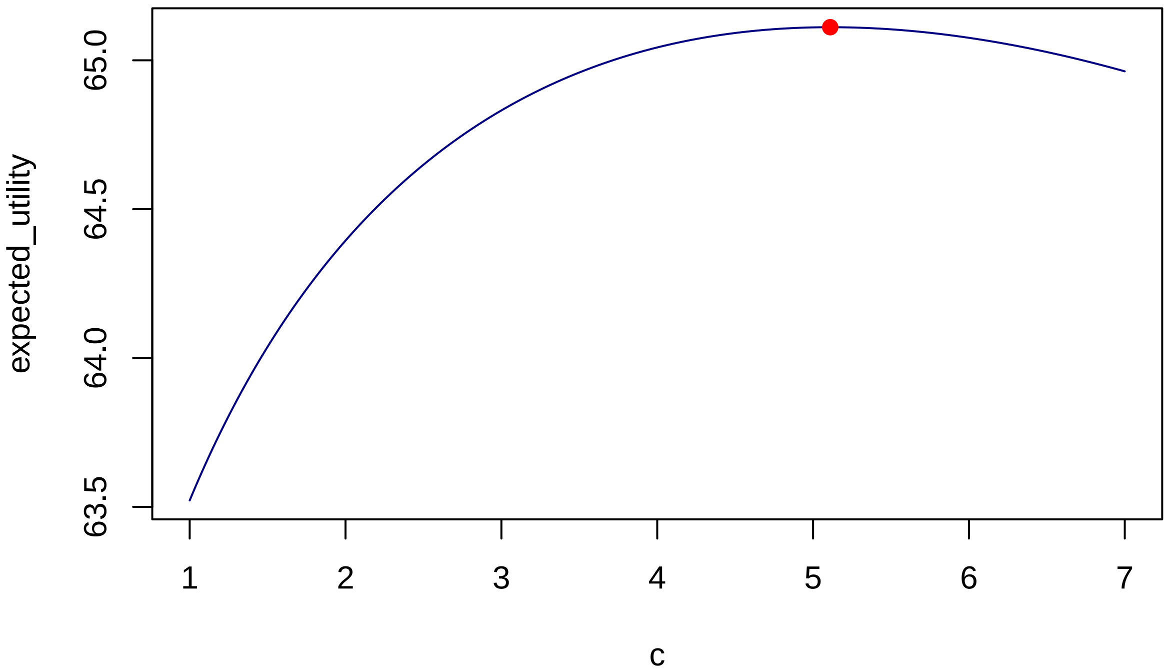
Task b

```
set.seed(12345)

N <- 1000
x_pred <- matrix(NA, nrow = N, ncol = 1)

for (i in 1:N){
  mu <- rnorm(1, mean = 92, sd = 2)
  x_pred[i,1] <- rnorm(1, mean = mu, sd = sqrt(50))
}

x_pred_dens <- density(x_pred)

df_x_pred_dens <- data.frame("x" = x_pred_dens$x, "y" = x_pred_dens$y)

ggplot(df_x_pred_dens) +
  geom_line(aes(x=x,y=y), color = "navy") +
  ggtitle("Posterior Distribution of x_{n+1}") +
  xlab("x_{n+1}") +
  ylab("Density") +
  theme_classic()
```



Task c

```
set.seed(12345)

utility_functiom <- function(c, mu){
  res <- 60 + sqrt(c) * mean(log(mu)) - c
  return(res)
}

c <- seq(1,7,0.01)
N <- 10000
expected_utility <- matrix(NA,nrow = length(c), ncol = 1)
count <- 0
mu <- rnorm(10000, mean = 92, sd = 2)

for (i in c){
  count <- count + 1
  expected_utility[count] <- utility_functiom(i,mu)
}

optimal_c <- c[which.max(expected_utility)]

plot(c,expected_utility, col = "navy", type = "l")
points(optimal_c,utility_function(c=optimal_c,mu), col = "red",pch=19)
```



A company should spend 5.11 MSEK on advertisements.