Report of Lab 3: Reliability and Survival

Alberto Nieto Sandino

November 2, 2018

Corrections for re-submission

All the parts that have been improved based on the comments given or new sections that have been added are written in blue. From Assignment 2.2 onwards, the sections are newly written.

1 Introduction

As the title suggests, in this lab, we analyze how different systems survive and their expected behavior when it comes to failure and life expectancy. We start by looking at the expected life length of a system and which component will cause the death of the system (Assignment 1). From there, we can then analyze how to extend the life of the system by adding redundant components and how that affects its reliability (Assignment 2). To finish, we will find the optimum components that maximize the life length of the system (Assignment 3).

2 Assignment 1.1

2.1 Problem

The system to analyze, seen in Figure 1, has four components:

- The first three are in parallel and have life lengths T_1, T_2, T_3 that are Weibull$(1, \frac{1}{2})$ distributed,
- while the last one has a life length T_4 that is $\exp(\frac{1}{2})$ distributed.

Our task is to find the expected life length $\mathbf{E}\{T\}$ of the system and to observe the death rate $r_T(t)$ over a time $t \in (0, 10)$ to determine its trend.

2.2 Theory and implementation

Based on *Theorem 3.3*, we know the life length of the system can be found by

$$\mathbf{E}\{T\} = \int_0^\infty R_T(t)dt, \tag{1}$$

where $R_T(t)$ is the survival function of the system, meaning the probability that the system is healthy at a time t (i.e. $R_T(t) = \mathbf{P}\{T > t\}$). This function is opposite to the distribution function of the life lengths $F_T(t)$ which is defined as $F_T(t) = \mathbf{P}\{T \leq t\} = 1 - R_T(t)$. Thus, we need

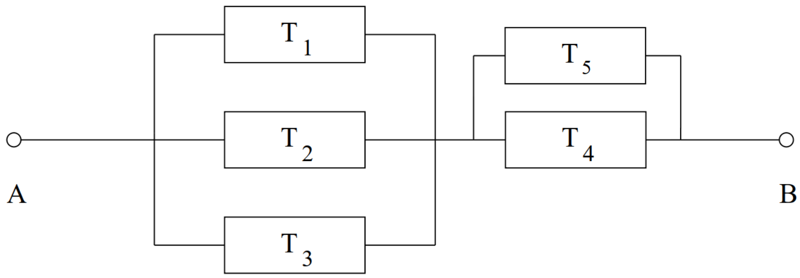

Figure 1: System to analyze

to find first the survival function of the system,$R_T(t)$, and then compute the integral (command `integrate`) in equation (1). All life lengths are assumed to be mutually independent onwards, unless it is stated otherwise.

$$\begin{aligned} R_T(t) &= \mathbf{P}\{T > t\} \\ &= [1-(1-\mathbf{P}\{T_1 > t\}])(1-\mathbf{P}\{T_2 > t\})(1-\mathbf{P}\{T_3 > t\}))]\mathbf{P}\{T_4 > t\} \\ &= [1-(1-R_{T_1}(t))(1-R_{T_w}(t))(1-R_{T_3}(t))]R_{T_4}(t) \end{aligned} \tag{2}$$

In equation (2), $R_{T_1} = R_{T_2} = R_{T_3} = 1 - F_W(t)$ where $F_W(t)$ is the cumulative distribution function (cdf) of a Weibull distribution with $F_W(t; a, b) = 1 - e^{-(at)^b}$ for $t \geq 0$. And $R_{T_4} = 1 - F_{exp}(t)$ where $F_{exp}(t)$ is the cdf of an exponential distribution where $F_{exp}(t; \lambda) = 1 - e^{-\lambda t}$. Knowing that the cdf of Weibull is defined by $a = 1$ and $b = 1/2$, and that the cdf of the exponential is defined by $\lambda = 1/2$. We can get that the expected life length of the system is

$$\begin{aligned} \mathbf{E}\{T\} &= \int_0^\infty R_T(t)dt = \int_0^\infty (1 - F_W(t)^3)(1 - F_{exp}(t)) \\ &= \int_0^\infty (1 - (1 - e^{-\sqrt{t}})^3)(e^{-t/2})dt \end{aligned} \tag{3}$$

The death or failure rate is the infinitesimal intensity when unhealth occurs. It is calculated as

$$r_T(t) = -\frac{d}{dt} \ln\left(R_T(t)\right) \text{ for } t > 0, \tag{4}$$

where the derivative is calculated numerically using `grad` from the library `numDeriv`. and the plot for the death rate of the system is evaluated for the given time $t \in [0, 10]$ to observe the trend of the failure rate.

2.3 Results and discussion

Applying equation (3), we obtain that the expected life length of the system is $\mathbf{E}\{T\} = 1.2948$. The plot for the death rate of the system can be seen in Figure 2. It shows a high increase in the death rate in the beginning with a slow decrease after $t \geq 1$, thus it can not be said that it has IFR (incresing failure rate) nor DFR (decreasing failure rate).

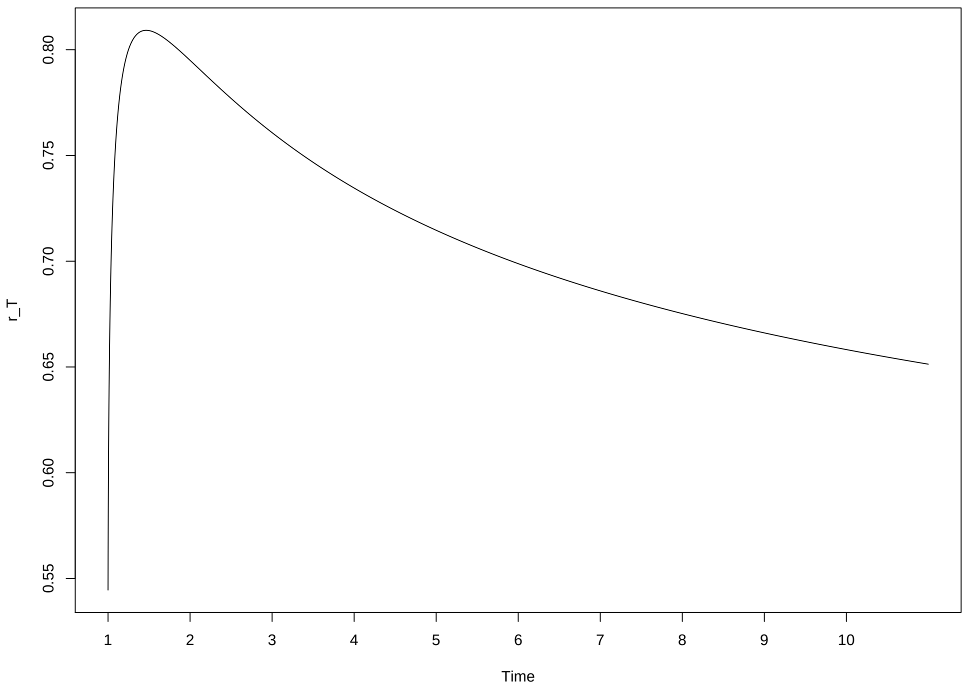

Figure 2: Evolution of the death rate, r_T, over a time $t \in (0, 10)$

3 Assignment 1.2

3.1 Problem

Looking at the system in Figure 1 and the derivation of its survival function in eq. (2), it can be noted that component four has a high probability of causing the death of the system. It is the task of this assignment to find the probability that this component will be responsible for the death of the system.

3.2 Theory and implementation

Observing the system in Figure 1, the life length of the system is given by $T = \min\{\max[T_1, T_2, T_3], T_4\}$. Thus, it can be derived that the probability of component four causing failure is

$$
\begin{aligned}
&\mathbf{P}\{\text{component with number 4 causes death}\} \\
&\quad = \mathbf{P}\{T_4 \leq \max[T_1, T_2, T_3]\} \\
&\quad = \int_0^\infty \mathbf{P}\{\max(T_1, T_2, T_3) \geq t\} f_{T_4} dt \\
&\quad = \int_0^\infty (1 - F_{T_1}(t) F_{T_2}(t) F_{T_3}(t)) f_{T_4}(t) dt \\
&\quad = \int_0^\infty (1 - F_W(t)^3) f_{exp}(t) dt \\
&\quad = \int_0^\infty (1 - (1 - e^{-\sqrt{t}})^3) \frac{e^{-t/2}}{2} dt
\end{aligned} \tag{5}
$$

Equation (5) wrote into R through a function which contains the integrand and solved by using `integrate` with the appropriate limits to the integral (`lower = 0, upper = Inf`).

3.3 Results and discussion

The evaluation of the integral in equation (5) yields a probability that component 4 will cause the failure of the system of $\mathbf{P} = 0.6474 = 64.74\%$.

This is what we would expect since component 4 does not have any redundant components to keep the system working in the case of its failure, while the other part of the system (parallel coupling) has two redundant components that would keep the system working in the case that one of them would fail.

That is the reason why in the following assignment, we will redo the calculations for a given redundant component for component number 4.

4 Assignment 2.1

4.1 Problem

In this assignment, we will focus on how the expected life length and the death intensity will change when a redundant component (both warm and cold) is added to the system and we will compare it to how the original system performs.

4.2 Theory and implementation

The system will have a new (redundant) component (number five) that will be added in parallel with component number four as can be seen in Figure 3. This component is $\exp(\frac{1}{2})$ distributed (command `pexp`), as well as the component number four.

The warm redundant component is added at time $t = 0$, therefore the life length of the parallel coupling is $T_{warm} = \max(T_1, T_2)$. The survival function of the system ($R_{T_{warm}}$) is derived in a similar way as equation (2).

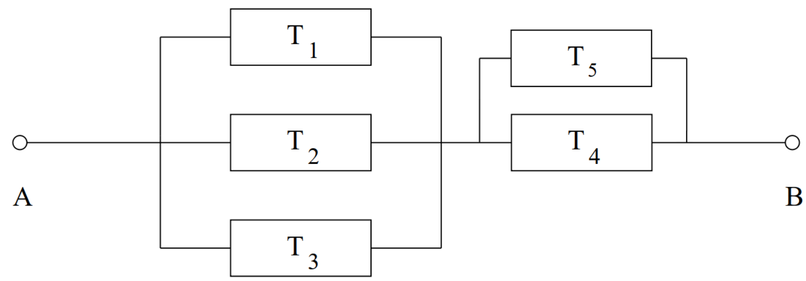

Figure 3: New layout of the system with a redundant component number five

$$
\begin{aligned}
R_{T_{warm}}(t) &= \mathbf{P}\{T > t\} \\
&= [1 - (1 - \mathbf{P}\{T_1 > t\}])(1 - \mathbf{P}\{T_2 > t\})(1 - \mathbf{P}\{T_3 > t\}))] \\
&[1 - (1 - \mathbf{P}\{T_4 > t\}])(1 - \mathbf{P}\{T_5 > t\}))] \\
&= [1 - (1 - R_{T_1}(t))(1 - R_{T_2}(t))(1 - R_{T_3}(t))][1 - (1 - R_{T_4}(t))(1 - R_{T_5}(t))] \\
&= [1 - F_{T_1}(t)F_{T_2}(t)F_{T_3}(t)][1 - F_{T_4}(t)F_{T_5}(t)] \\
&= [1 - F_W(t)^3][1 - F_{exp}(t)^2] \\
&= [1 - (1 - e^{-\sqrt{t}})^3][1 - (1 - e^{-t/2})^2].
\end{aligned} \tag{6}
$$

However, the cold redundant component is added at the time when component four is dead, thus the life length of the parallel coupling is $T_{cold} = T_4 + T_5$. The survival function of the system ($R_{T_{cold}}$) is derived more differently in this case. For two components in parallel (T_1 and T_2), we know that

$$
R_T(t) = 1 - \int_0^t (1 - R_{T_1}(t - x))R_{T_2}(x)r_{T_2}(x)dx. \tag{7}
$$

Based on this, the survival function ($R_{T_{cold}}$) of the system is calculated as follows.

$$
\begin{aligned}
R_{T_{cold}}(t) &= \mathbf{P}\{T > t\} \\
&= (1 - (1 - R_{T_1})(1 - R_{T_2})(1 - R_{T_3}))\left(1 - \int_0^t [(1 - R_{T_4}(t - x))R_{T_5}(x)r_{T_5}(x)]\,dx\right) \\
&= \left(1 - F_W(t)^3\right)\left(1 - \int_0^t F_{exp}(t - x)f_{exp}(x)dx\right) \\
&= (1 - (1 - e^{-\sqrt{t}})^3)\left(1 - \int_0^t \left(1 - e^{-(1/2)(t-x)}\right)\frac{1}{2}e^{-x/2}dx\right)
\end{aligned} \tag{8}
$$

The new expected life lengths are calculated by integrating from 0 to infinity the above mentioned survival functions.

The death rates are calculated in the same manner as those in assignment 1.1.

4.3 Results and discussion

Applying equations (6) and (1), we obtain an expected life length $E_{T_{warm}} = 1.8079$ for the warm redundancy. Whereas applying equation (8) and (1), the expected life length that results is

$E_{T_{cold}} = 2.1589$. Therefore it can be seen how the system will stay healthy more time when using a cold redundant component rather than a warm one. This makes sense since the cold redundant component only starts to act once the other component is broken, thus not "wasting" life length when it is not needed.

The death rates for the original system (blue line), the system with a warm redundancy (red line) and the system with a cold redundancy (black line) can be found in Figure 4. We can see that the probability of failure is smaller for both cases of redundant component compared to the original, as it would be expected. In addition, we can also see that the cold redundancy has lower risk of death when compared to the warm component, since it only acts if and only if the other component fails. In this scenario, we can appreciate, as well, that the systems with cold and warm redundancy are IFR since the derivative of the death rate is always positive.

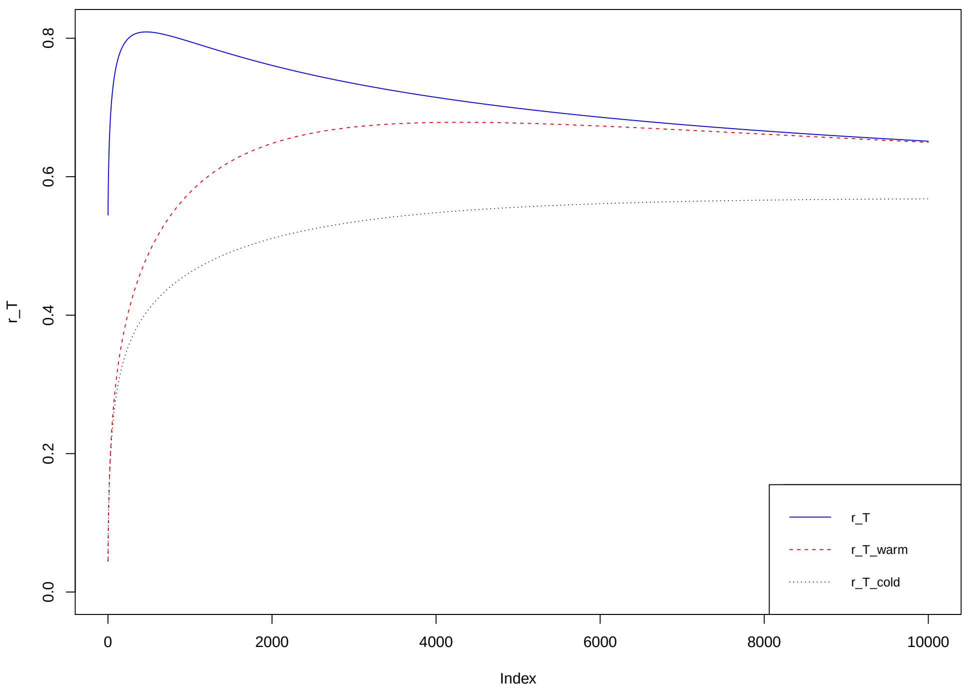

Figure 4: Evolution of the death rate for $t \in (0, 10)$

5 Assignment 2.2

5.1 Problem

In this task, we want to check what value ρ would need to take so that the expected life length of the system is the same for only component four $\exp(\rho)$ distributed as with a warm or cold component, respectively.

5.2 Theory and implementation

We find these values by solving these equations

$$\mathbf{E}\{T(t)\} - \mathbf{E}\{T_{warm}\} = 0 \quad \& \quad \mathbf{E}\{T(t)\} - \mathbf{E}\{T_{cold}\} = 0. \tag{9}$$

These can be summarized as

$$\int_0^\infty e^{-\rho t}(1-(1-e^{-\sqrt{t}})^3)dt - \int_0^\infty R_{(warm/cold)}(t)dt = 0, \tag{10}$$

where ρ is the unknown. These equations are solved using `uniroot` which searches for a root of the function a given interval, thus finding ρ.

5.3 Results and discussion

After computing the code, we find that it is required to have at least a $\rho \leq 0.2905$ to have a component four that gives an expected life length to the system equal to the system with a warm redundancy. For the system with a cold redundancy, we need a $\rho \leq 0.2075$ so that the system with no redundancy in component four has as much expected life length as the one with a cold redundancy. As it was expected, ρ must be smaller in the case of having the same expected life length of the cold redundancy since it has a more demanding (i.e. larger) expected life length.

6 Assignment 3

6.1 Problem

In this task, we have a system such as the on in Figure 5. T_1 is Weibull($\mu, \frac{1}{3}$)) distributed and the second and third components (T_2 and T_3) are Weibull($\lambda, \frac{1}{3}$). The cost for each of these components is $1/5 + 1/\gamma$ monetary units. Our aim is to optimize μ and λ so as to maximize the expected life length $\mathbb{E}\{T\}$ at a total cost $c = 1:10$.

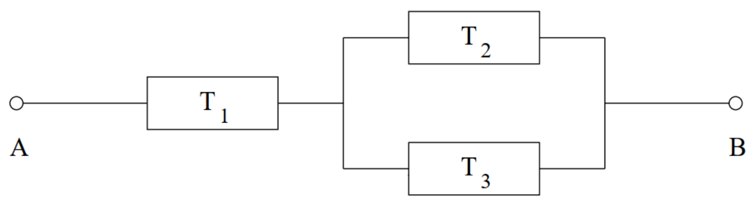

Figure 5: Three component system

6.2 Theory and implementation

In order to solve this problem, we are gonna perform nonlinear optimization with constraints using the function `constrOptim.nl` from the `Alabama` library. Thus, we first need to define the function to optimize. We have to consider that the algorithm perform minimization, so we will multiply our function times negative one. Our function to optimize is the expected life length with can be seen in equation (1). Thus, we need to define clearly our survival function

$$R(t) = e^{-(\mu t)^{1/3}}(1-(1-e^{-(\lambda t)^{1/3}})^2). \tag{11}$$

The implementation of the expected life length is as follows.

```
E_T_var <- function(x){
  integrate(
    function(t){
      (-1*(exp(-(x[1]*t)^1/3)*(1-(1-exp(-(x[2]*t)^(1/3)))^2)))
    }
    ,lower = 0, upper = Inf, rel.tol = 1e-10)$value
}
```

After that we define the inequality constraints for μ and λ.

```
hin <- function(x) {
    h <- rep(NA, 1)
    h[1] <- x[1]
    h[2] <- x[2]
    h
  }
```

And the equality constraints for the total cost c.

```
heq <- function(x) {
    h <- rep(NA, 1)
    h[1] <- ((1/5) + (1/x[1]) + 2*((1/5) + (1/x[2]))) - c
    h
  }
```

6.3 Results and discussion

In Figure 6, we can see the values of the parameters λ and μ, that maximizes the expected life length $\mathbb{E}\{T\}$ of the system, at the total costs $c = 1, 2, ...10$ monetary units, respectively, of the system. In Figure 7, the expected life length $\mathbb{E}\{T\}$ is plotted as a function of the total costs.

It can be seen that both parameters decrease as the total costs increase, which is in accordance with the equality constraint implement in the optimization algorithm. The expected life length $\mathbb{E}\{T\}$ increases linearly as a function of the total costs. This seems to point that the more monetary units we pay, the more the systems lasts healthy. Something that makes sense, otherwise, it would be a scam.

Appendix - R code

```
# Lab 03 - Reliability and survival
# Author: Alberto Nieto Sandino
# Date: 2018-11-02

library(numDeriv)
library(ggplot2)
library(alabama)
```

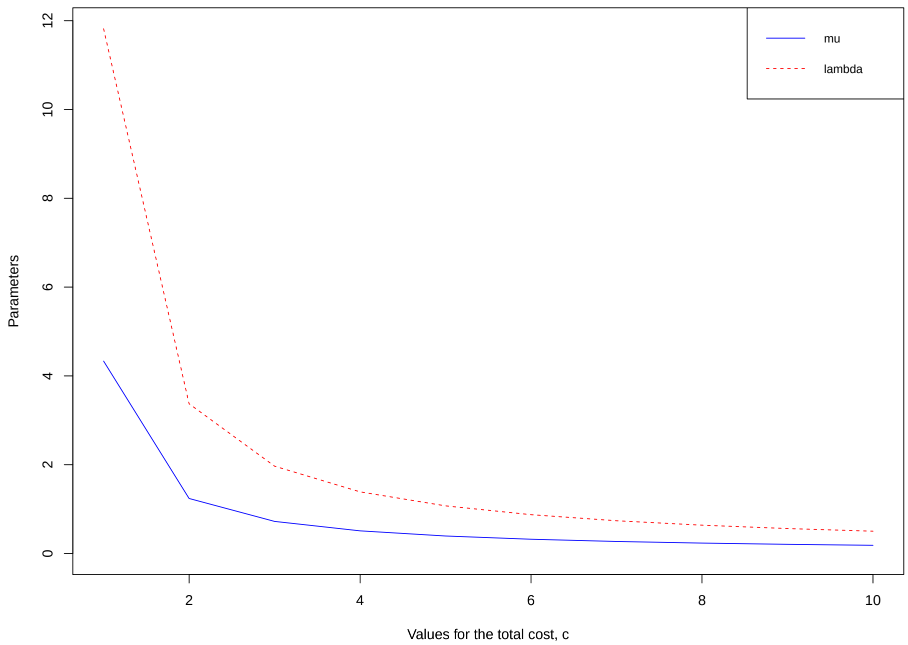

Figure 6: Evolution of the parameters μ and λ as the total costs increase

```

# Clean environment
closeAllConnections()
rm(list=ls())
# Clear plots
graphics.off()
# Clear history
clearhistory <- function() {
  write("", file=".blank")
  loadhistory(".blank")
  unlink(".blank")
}
clearhistory()

# Assigment 1 (4p)

R_T <- function(t){
  f1 <- ((1-(1-exp(-sqrt(t)))^3)*(exp(-t/2)))
  return (f1)
}

# Expected life length
```

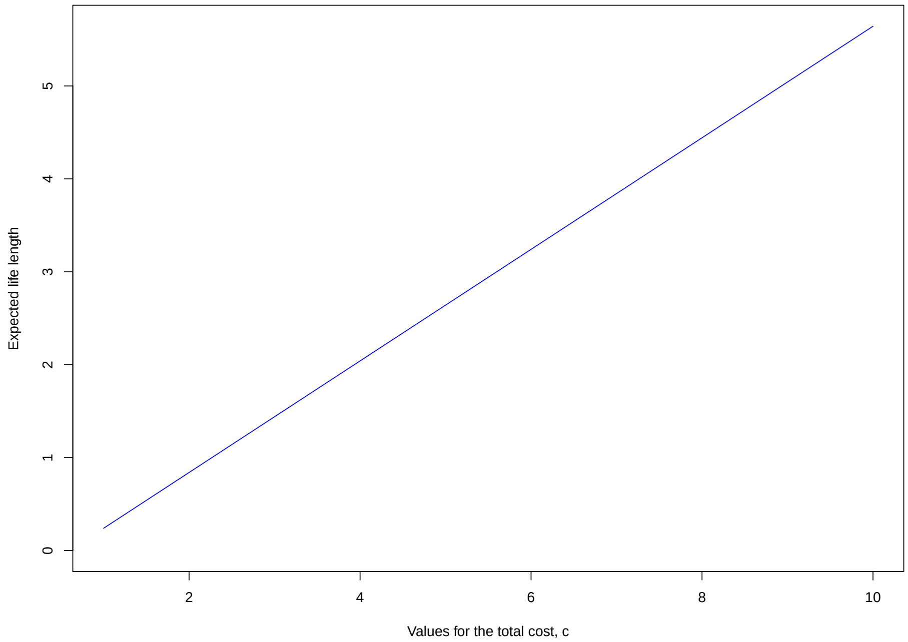

Figure 7: Evolution of the expected life length $\mathbb{E}\{T\}$ as the total costs increase

```
E_T = integrate(R_T, lower = 0, upper = Inf)$value
E_T

loga <- function(t){
  f1 <- log((1-(1-exp(-sqrt(t)))^3)*(exp(-t/2)))
  return (f1)
}

# Death rate, r_T
x <- seq(0.001,10,0.001)
r_T <- -grad(loga,x)
plot(r_T, type="l", xaxt = "n", xlab='Time')
axis(1, at=seq(1,10000,1000), labels=1:10)


# Assigment 1.2

integrand <- function(t){
  f1 <- (1-(1-exp(-sqrt(t)))^3)*(exp(-t/2)/2)
  return (f1)
}
P_4 <- integrate(integrand, lower = 0, upper = Inf)$value
```

```
P_4


# Assignment 2 (3p)

# Assignment 2.1

##########################################################
## WARM COMPONENT ########################################
##########################################################

# Survival function
R_T_warm <- function(t){
  f1 <- ((1-(1-exp(-sqrt(t)))^3)*(1-(1-(exp(-t/2)))^2))
  return (f1)
}

# Expected life length with warm component
E_T_warm = integrate(R_T_warm, lower = 0, upper = Inf)$value
E_T_warm

# Death rate, r_T_warm
loga_warm <- function(t){
  f1 <- log((1-(1-exp(-sqrt(t)))^3)*(1-(1-(exp(-t/2)))^2))
  return (f1)
}

x <- seq(0.001,10,0.001)
r_T_warm <- -grad(loga_warm,x)
plot(r_T_warm, type="l", xaxt = "n", xlab='Time')
axis(1, at=seq(1,10000,1000), labels=1:10)

##########################################################
## COLD COMPONENT ########################################
##########################################################

# Calculate the doble integral inside R_T_cold
g <- function(t){
  f <- function(x) {(1-exp(-(1/2)*(t-x)))*(1/2)*exp(-(1/2)*x)}
  integrate(f,0,t)$value
}
int <- Vectorize(g)

# Survival function
R_T_cold<- function(t){
```

```
  f1 <- ((1-(1-exp(-sqrt(t)))^3)*(1-int(t)))
  return (f1)
}

# Expected life length with cold component
E_T_cold = integrate(R_T_cold, lower = 0, upper = Inf)$value
E_T_cold

# Death rate, r_T_cold
loga_cold <- function(t){
  f1 <- log((1-(1-exp(-sqrt(t)))^3)*(1-int(t)))
  return (f1)
}

x <- seq(0.001,10,0.001)
r_T_cold <- -grad(loga_cold,x)
plot(r_T_cold, type="l", xaxt = "n", xlab='Time')
axis(1, at=seq(1,10000,1000), labels=1:10)


# Plot of the death rates for both cases and the one from task 1
lim_range <- range(0, r_T, r_T_warm, r_T_cold) # calculate range of values
plot(r_T, type="l", col ="blue", ylim = lim_range)
lines(r_T_warm, type = "l", pch = 22, lty = 2, col = "red")
lines(r_T_cold, type = "l", pch = 23, lty = 3, col = "black")
legend("bottomright", legend = c("r_T", "r_T_warm", "r_T_cold"), cex=0.8,
       col = c("blue","red","black"), lty = 1:3)



## Assignment 2.2

######################
## WARM COMPONENT ####
######################

# x is rho
E_T_rho <- function(x){
  integrate(
    function(t){
      ((1-(1-exp(-sqrt(t)))^3)*(exp(-t*x)))
    }
    ,lower = 0, upper = Inf)$value
```

```
}
E_T_rho_v <- Vectorize(E_T_rho)

# Find the root for the warm redundancy
equation <- function(x){
  E_T_rho_v(x)-E_T_warm
}

rho_warm <- uniroot(equation,interval = c(0.0,0.5))$root
rho_warm

#####################
## COLD COMPONENT ####
#####################

# x is rho
E_T_rho <- function(x){
  integrate(
    function(t){
      ((1-(1-exp(-sqrt(t)))^3)*(exp(-t*x)))
    }
    ,lower = 0, upper = Inf)$value
}
E_T_rho_v <- Vectorize(E_T_rho)

# Find the root for the warm redundancy
equation <- function(x){
  E_T_rho_v(x)-E_T_cold
}

rho_cold <- uniroot(equation,interval = c(0.0,0.5))$root
rho_cold

#####################################################
## Assignment 3 #####################################
#####################################################

# Remember that we do minimization -> times -1

# Function for the expected life length
E_T_var <- function(x){
  integrate(
    function(t){
      (-1*(exp(-(x[1]*t)^1/3)*(1-(1-exp(-(x[2]*t)^(1/3)))^2)))
    }
```

```
    ,lower = 0, upper = Inf, rel.tol = 1e-10)$value
}

# Perform non-linear optimization with mu,lambda > 0 as constraints
solution <- rep(NA,10)
fval <- rep(NA,10)
mu <- rep(NA,10)
lambda <- rep(NA,10)

for (c in 1:10){
  p0 <- c(1,1)
  # Inequalities: mu, lambda
  hin <- function(x) {
    h <- rep(NA, 1)
    h[1] <- x[1]
    h[2] <- x[2]
    h
  }

  # Equalities: cost
  heq <- function(x) {
    h <- rep(NA, 1)
    h[1] <- ((1/5) + (1/x[1]) + 2*((1/5) + (1/x[2]))) - c
    h
  }

  solution <- constrOptim.nl(par=p0, fn=E_T_var, heq=heq, hin=hin)
  fval[c] <- -1*solution$value # reverse the change for minimization
  mu[c] <- solution$par[1] # x[1] = mu
  lambda[c] <- solution$par[2] # x[2] = lambda
}


# Plot of mu and lambda
lim_range <- range(0, mu, lambda) # calculate range of values
plot(mu, type="l", col ="blue", ylim = lim_range, ylab ="Parameters"
  ,
     xlab = "Values for the total cost, c")
lines(lambda, type = "l", pch = 22, lty = 2, col = "red")
legend("topright", legend = c("mu", "lambda"), cex=0.8,
       col = c("blue","red"), lty = 1:2)


# Plot of the function value
lim_range <- range(0, fval) # calculate range of values
```

```
plot(fval, type="l", col ="blue", ylim = lim_range, ylab ="Expected
   life length",
      xlab = "Values for the total cost, c")
```