hw 1

Nathan Hawkins

1/14/2022

1)

All models are wrong, some models are useful.

2)

0's are circles, denotes censoring 1's are x's, denote death - 3 patients died - 18 person-years - 1/6 deaths per person-year

```
df1 = data.frame("patient" = c(1:5), "start"= c(0,0,1,1,2), "end" = c(5,5,5,4,3) , "t" = c(5,5,4,3,1),



sum(df1$delta)
```

```
## [1] 3
```

```
# 3 patients died
sum(df1$t)
```

```
## [1] 18
```

```
# 18 person years
```

3

- 32 patients had the event.
- 2866 weeks of follow-up time
- 0.01117 events per person-week

```
library(asaur)
gx = asaur::gastricXelox

sum(gx$delta)
```

```
## [1] 32
```

```r
sum(gx$timeWeeks)
```

```
## [1] 2866
```

```r
32/2866
```

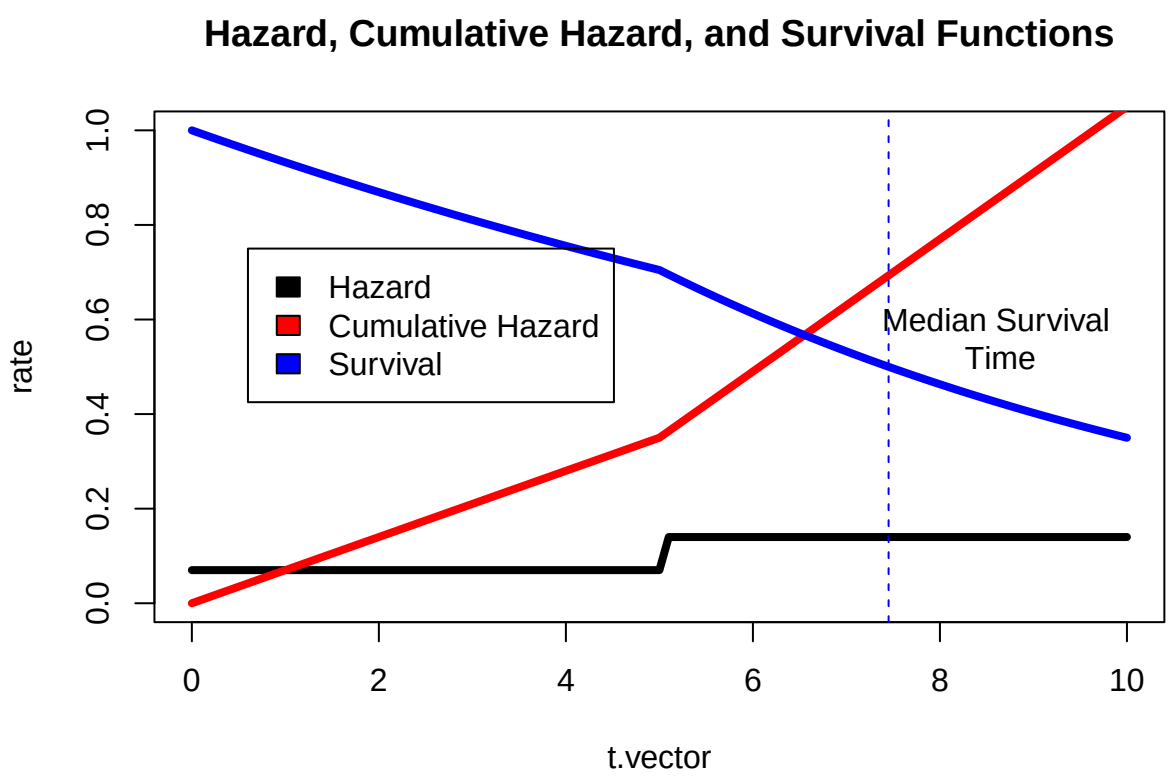
```
## [1] 0.01116539
```

4)

Median survival time is 7.45

Survival function = eˆ-H(t) H(t) = integral(h(t))

```r
t.vector = seq(0,10, by = 0.1)

hazard4 = function(t){
  if(t <= 5){
    return(0.07)
  }
  else{
    return(0.14)
  }
}

plot(x= t.vector, y = sapply(seq(0,10, by = 0.1), hazard4), ylim = c(0,1), type = "l", main = "Hazard, C


cum.hazard4 = function(t) {
  if (t <= 5) {
    return(hazard4(t) * t)
  }
  else{
    return(hazard4(t) * (t-5) + 0.35)
  }
}

lines(x= t.vector,sapply(t.vector, cum.hazard4), type = "l", col = "red", lwd = 4)


survival.function = function(t){
  return(exp(-cum.hazard4(t)))
}

median = uniroot(f = function(t){survival.function(t)-0.5}, interval = c(5,10))$root
lines(x= t.vector, sapply(t.vector, survival.function), type = "l", col = "blue", lwd = 4)
abline(v = median, col = "blue", lty = 2)
legend(0.6, 0.75,legend = c("Hazard", "Cumulative Hazard", "Survival"), fill = c("black", "red", "blue")
text(8.6, 0.56, "Median Survival\n Time")
```



5)

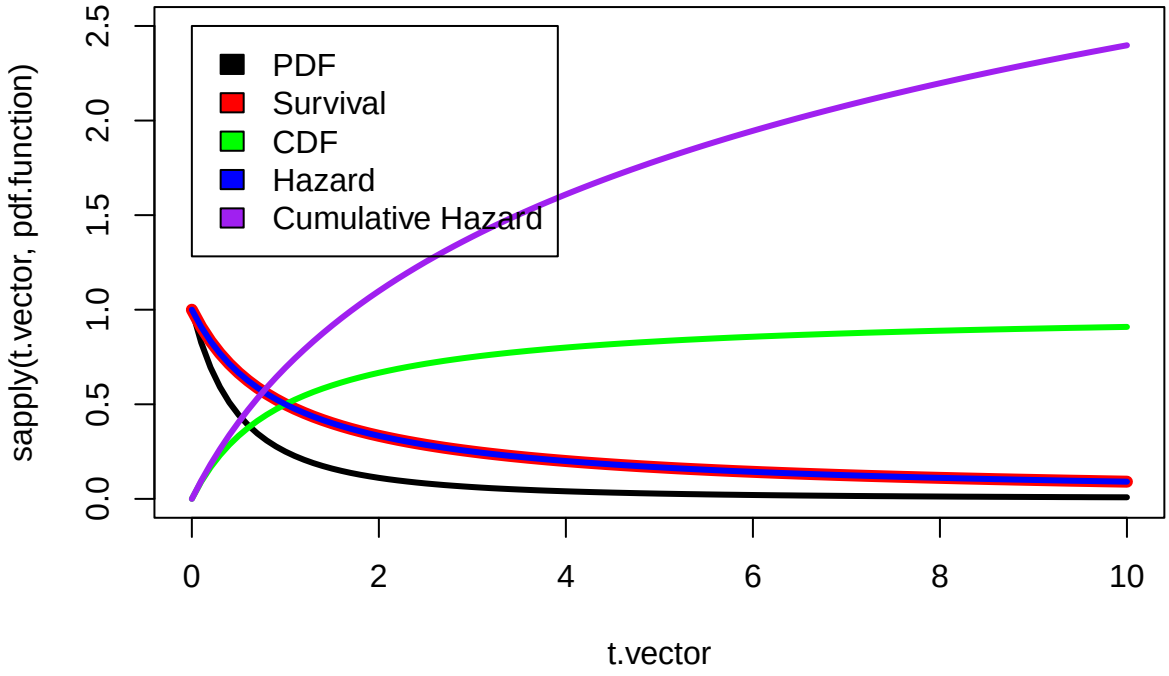
$f(x) = (x+1)^{-2}$ $F(x) = 1-(x+1)^{-1}$ $S(x) = 1-(1-(x+1)^{-1}) = (x+1)^{-1}$ $h(x) = \frac{(x+1)^{-2}}{(x+1)^{-1}} = (x+1)^{-1}$
$H(x) = -ln((x+1)^{-1})$

```
pdf.function = function(t){
return((t+1)^(-2))
}
cdf.function = function(t){
return(1-(t+1)^(-1))
}
survival.function = function(t){
return(1 - cdf.function(t))
}
hazard = function(t){
return(pdf.function(t)/survival.function(t))
}
cum.hazard = function(t){
return(-log(survival.function(t)))
}
t.vector
```

```
##   [1]  0.0  0.1  0.2  0.3  0.4  0.5  0.6  0.7  0.8  0.9  1.0  1.1  1.2  1.3  1.4
```

```
##  [16]  1.5  1.6  1.7  1.8  1.9  2.0  2.1  2.2  2.3  2.4  2.5  2.6  2.7  2.8  2.9
##  [31]  3.0  3.1  3.2  3.3  3.4  3.5  3.6  3.7  3.8  3.9  4.0  4.1  4.2  4.3  4.4
##  [46]  4.5  4.6  4.7  4.8  4.9  5.0  5.1  5.2  5.3  5.4  5.5  5.6  5.7  5.8  5.9
##  [61]  6.0  6.1  6.2  6.3  6.4  6.5  6.6  6.7  6.8  6.9  7.0  7.1  7.2  7.3  7.4
##  [76]  7.5  7.6  7.7  7.8  7.9  8.0  8.1  8.2  8.3  8.4  8.5  8.6  8.7  8.8  8.9
##  [91]  9.0  9.1  9.2  9.3  9.4  9.5  9.6  9.7  9.8  9.9 10.0
```

```
plot(x = t.vector, y = sapply(t.vector, pdf.function), col = "black", type = "l", lwd = 3, ylim = c(0,2
lines(x = t.vector, y = sapply(t.vector, survival.function), col = "red", lwd = 6)
lines(x = t.vector, y = sapply(t.vector, cdf.function), col = "green", lwd = 3)
lines(x = t.vector, y = sapply(t.vector, hazard), col = "blue", lwd = 3)
lines(x = t.vector, y = sapply(t.vector, cum.hazard), col = "purple", lwd = 3)
legend(0,2.5,legend = c("PDF",  "Survival", "CDF", "Hazard", "Cumulative Hazard"),
fill = c("black", "red", "green", "blue", "purple"))
```



6)

```
theta = 4
ll = function(theta){
  sum(
    dexp(3, rate = theta, log = TRUE),
    pexp(2, rate = theta, lower.tail = TRUE, log.p = TRUE),
    pexp(5, rate = theta, lower.tail = FALSE, log.p = TRUE),
```

```
    pexp(4, rate = theta, lower.tail = TRUE, log.p = TRUE) -
      pexp(2.5, rate = theta, lower.tail = TRUE, log.p = TRUE)
  )

}


opt1 = optimize(ll, c(0,5), maximum=TRUE)

curve(dexp(x, rate = opt1$maximum), from = 0, to = 5, ylim = c(0,1))
```