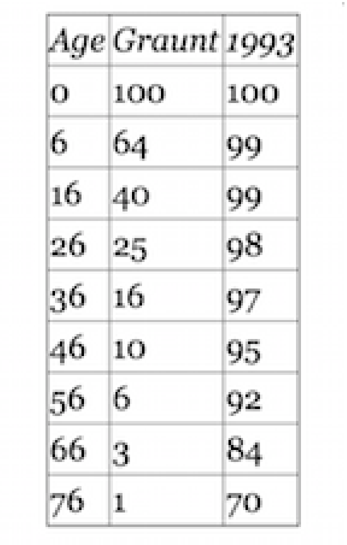
Graunt and US 1993 Life Table

MYUNG BIN KWAK

2020-04-22

Source of Data

```
knitr::include_graphics("../pics/graunt_table.png")
```



Age	Graunt	1993
0	100	100
6	64	99
16	40	99
26	25	98
36	16	97
46	10	95
56	6	92
66	3	84
76	1	70

Data Input

- Graunt's Life Table

```
graunt <- data.frame(x = c(0, seq(6, 76, by = 10)),
                     xPo_g = c(100, 64, 40, 25, 16, 10, 6, 3, 1))
```

More data

- US 1993 life table for the same age group

```
us93 <- data.frame(x = graunt$x,
                   xPo_us = c(100, 99, 99, 98, 97, 95, 92, 84, 70))
```

Data Extraction

There are many ways to extract part of `us93` data frame.

```
us93["xPo_us"]
```

```
##   xPo_us
## 1    100
## 2     99
## 3     99
## 4     98
## 5     97
## 6     95
## 7     92
## 8     84
## 9     70
```

```
us93["xPo_us"][[1]]
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93["xPo_us"]$xPo_us
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93["xPo_us"]$xPo
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93[2]
```

```
##   xPo_us
## 1    100
## 2     99
## 3     99
## 4     98
## 5     97
## 6     95
## 7     92
## 8     84
## 9     70
```

```
us93[2][[1]]
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93[2]$xPo_us
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93[ , "xPo_us"]
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93[ , 2]
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93$xPo_us
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

```
us93$xPo
```

```
## [1] 100  99  99  98  97  95  92  84  70
```

Into one single data frame

Combine two data frames into one single data frame, compare the results.

```
(graunt_us <- data.frame(graunt, xPo_us = us93$xPo))
```

```
##    x xPo_g xPo_us
## 1  0   100    100
## 2  6    64     99
## 3 16    40     99
## 4 26    25     98
## 5 36    16     97
## 6 46    10     95
## 7 56     6     92
## 8 66     3     84
## 9 76     1     70
```

```
(graunt_us_2 <- data.frame(graunt, us93[2]))
```

```
##    x xPo_g xPo_us
## 1  0   100    100
## 2  6    64     99
## 3 16    40     99
## 4 26    25     98
## 5 36    16     97
## 6 46    10     95
## 7 56     6     92
## 8 66     3     84
## 9 76     1     70
```

```
(graunt_us_3 <- data.frame(graunt, us93[, 2]))
```

```
##    x xPo_g us93...2.
## 1  0   100       100
## 2  6    64        99
## 3 16    40        99
## 4 26    25        98
## 5 36    16        97
## 6 46    10        95
## 7 56     6        92
## 8 66     3        84
## 9 76     1        70
```

Life Expectancy

The basic principle is that the area under the survival function is the life expectancy.

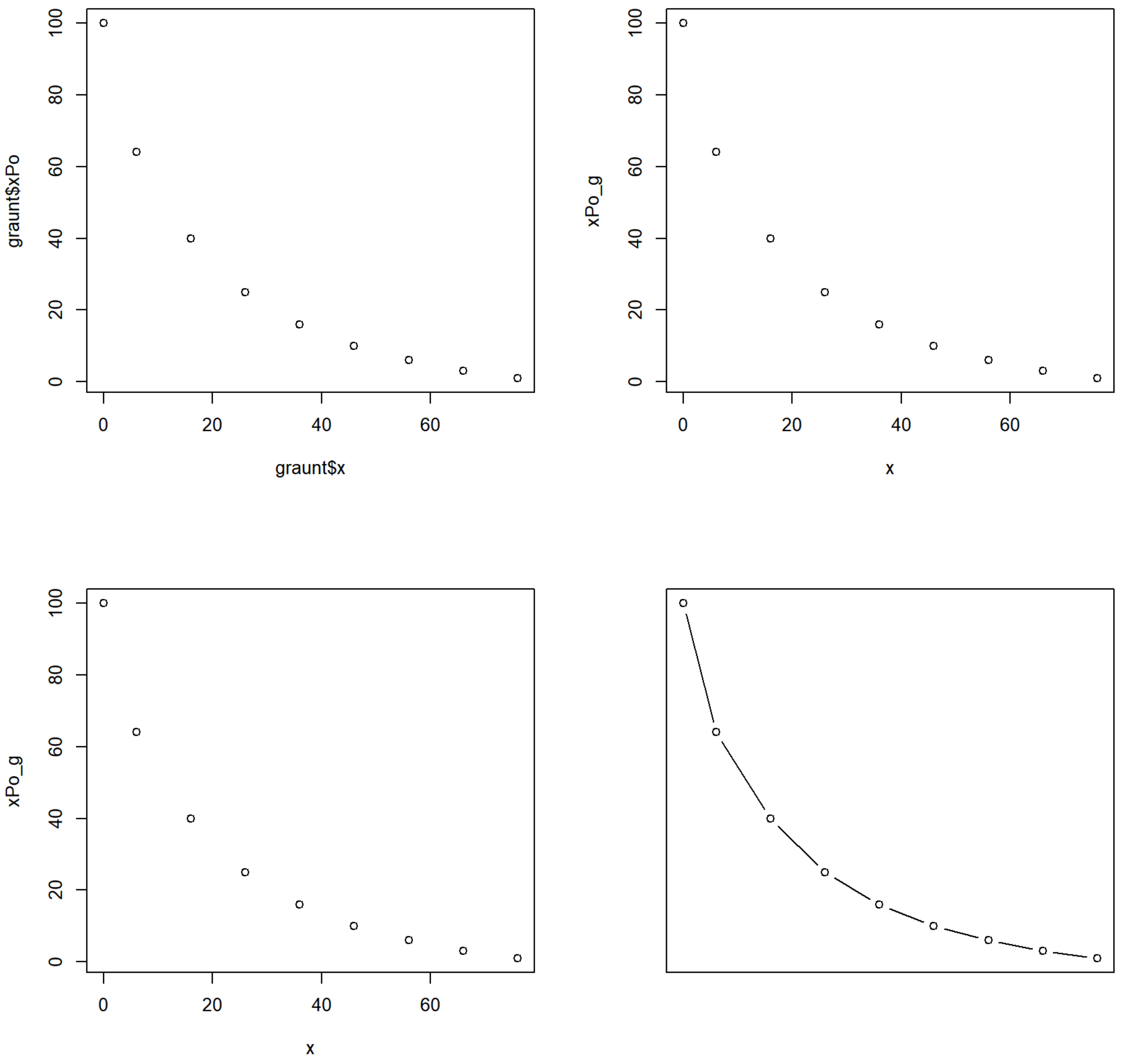
$X \geq 0, X \sim F(x)$ => $X \equiv F^{-1}(U), U \sim U(0,1)$, therefore,

$E(X) = E\{F^{-1}(U)\} = \int_0^1 F^{-1}(u)du = \int_0^\infty 1 - F(x)dx = \int_0^\infty S(x)dx$

Step by step approach to draw survival function plot

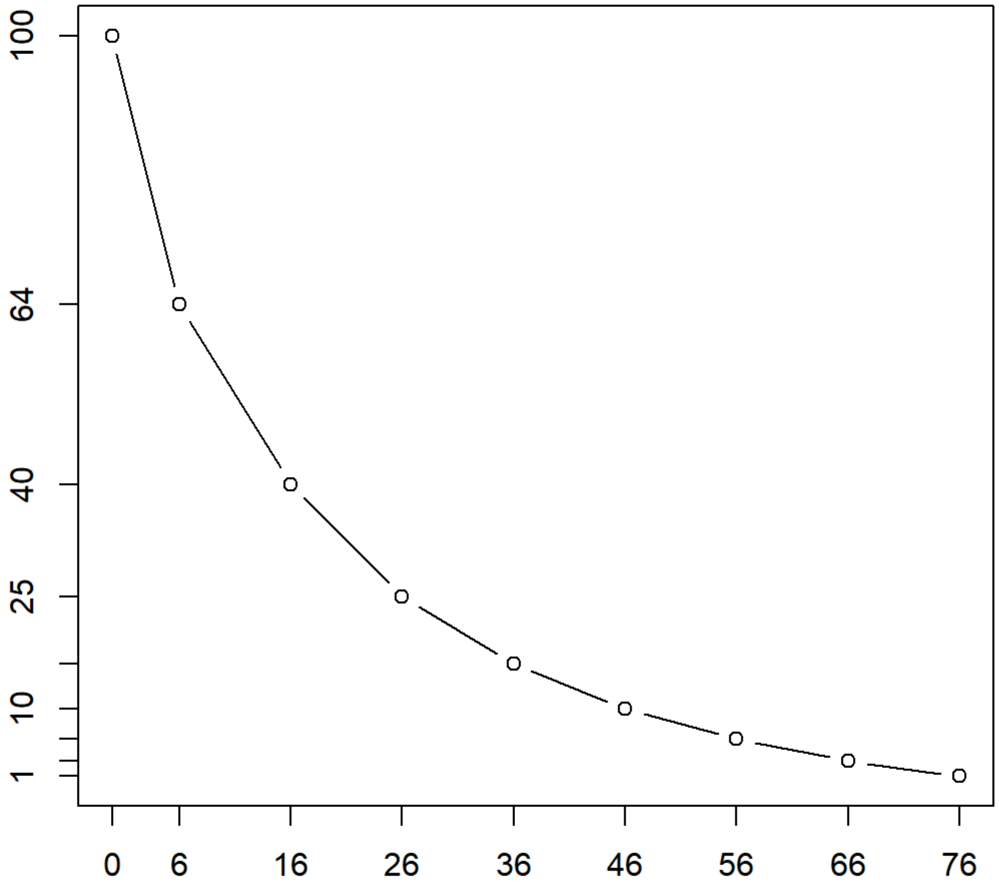
1. Basic plot with points and lines, compare the following threes methods

```
par(mfrow = c(2, 2))
plot(x = graunt$x, y = graunt$xPo)
plot(xPo_g ~ x, data = graunt)
plot(graunt)
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
```



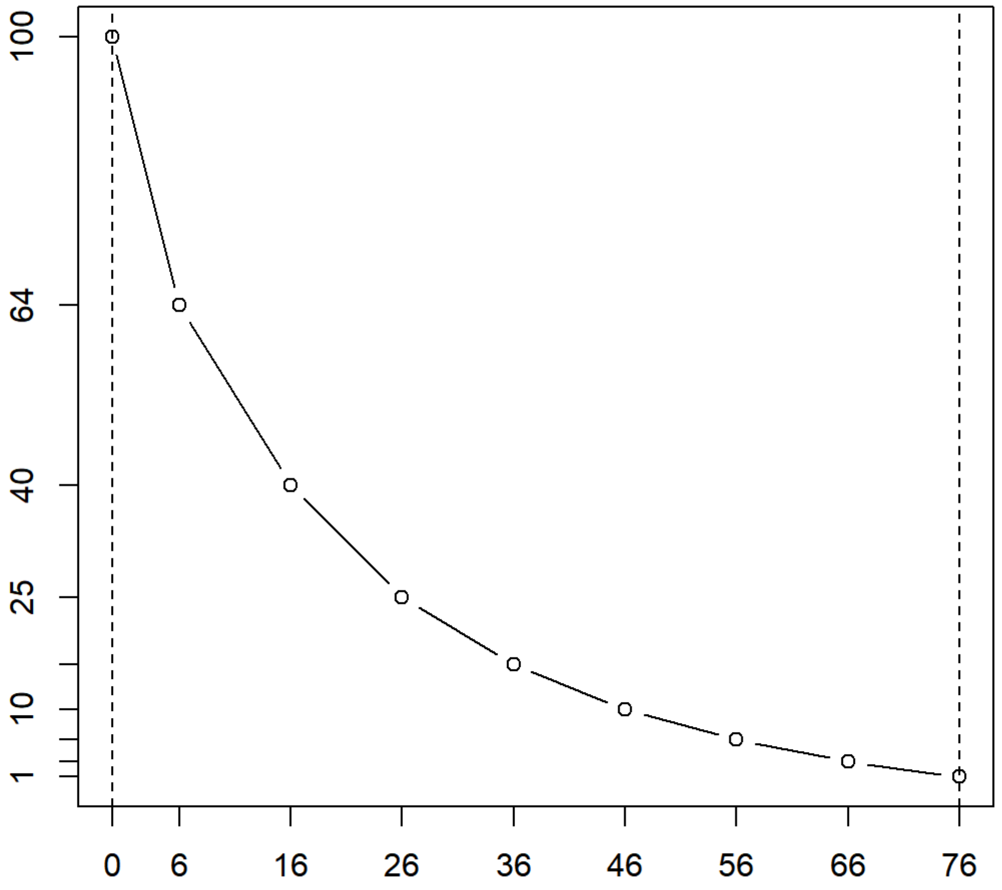
2. Denote the ages and observed survival rates on the axes

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo_g, labels = graunt$xPo_g)
```



3. Denote the age 0 and 76 by dotted lines

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo_g, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
```



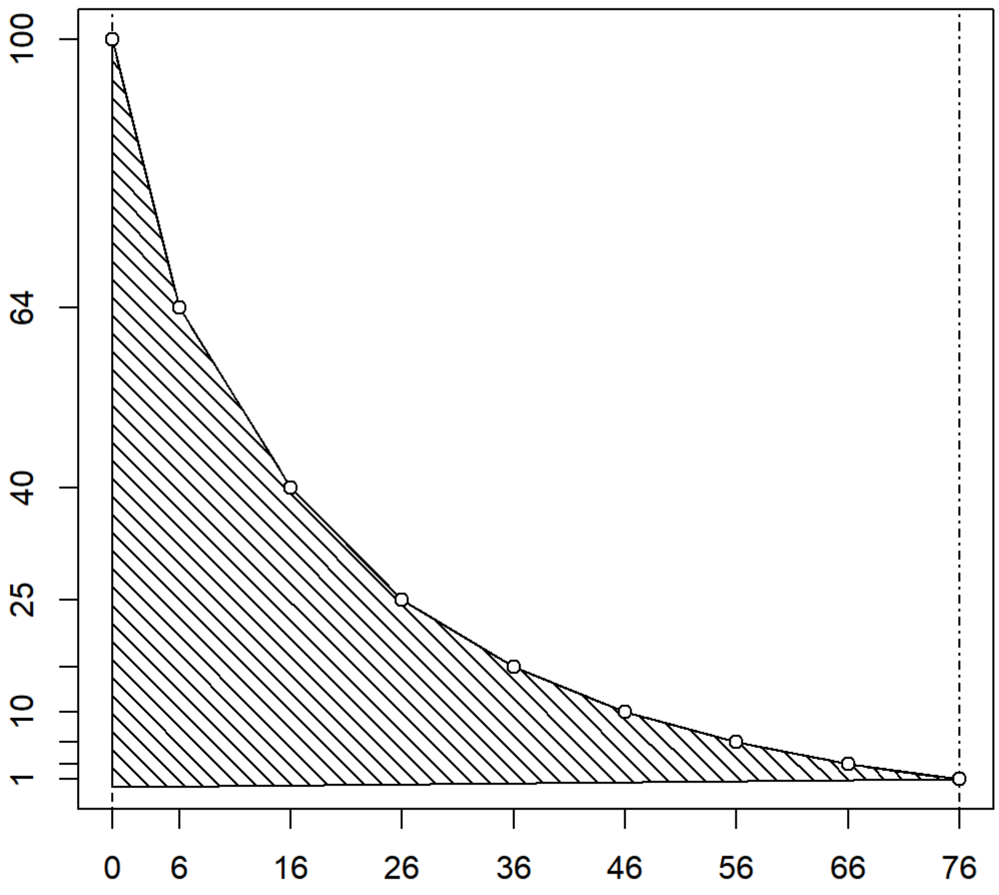
Setting up coordinates for `polygon()` (Clockwise)

```
graunt_x <- c(graunt$x, 0)
graunt_y <- c(graunt$xPo_g, 0)
graunt_poly <- data.frame(x = graunt_x, y = graunt_y)
```

4. Shading

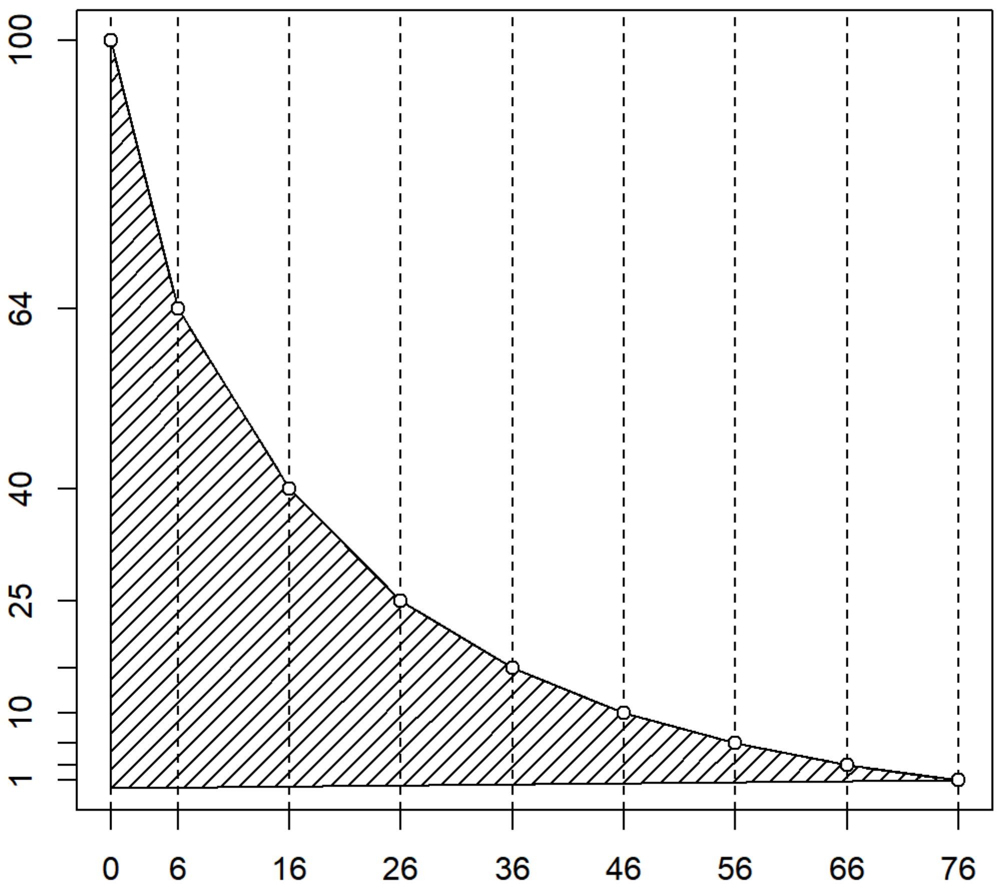
Note the effect of the last line of code.

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo_g, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 4)
polygon(graunt_poly, density = 15, angle = 135)
points(graunt, pch = 21, col = "black", bg = "white")
```



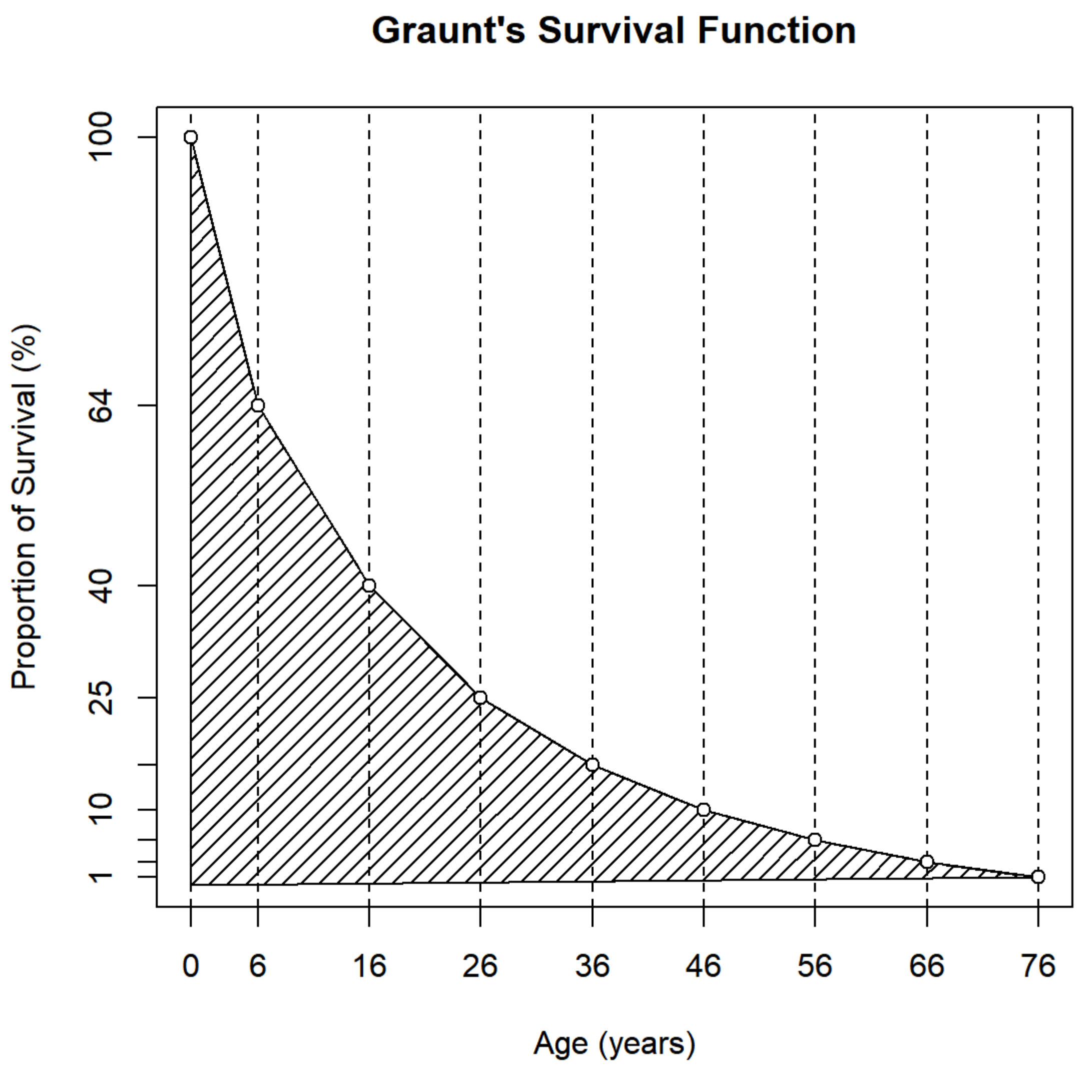
5. Grids

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo_g, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
polygon(graunt_poly, density = 15)
abline(v = graunt$x, lty = 2)
points(graunt, pch = 21, col = "black", bg = "white")
```



6. Title, x-axis label, and y-axis label

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo_g, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
polygon(graunt_poly, density = 15)
abline(v = graunt$x, lty = 2)
points(graunt, pch = 21, col = "black", bg = "white")
main_title <- "Graunt's Survival Function"
x_lab <- "Age (years)"
y_lab <- "Proportion of Survival (%)"
title(main = main_title, xlab = x_lab, ylab = y_lab)
```



Area under the curve

The area under the curve can be approximated by the sum of the areas of trapezoids, therefore the area is $\sum_{i=1}^{n-1}(x_{i+1} - x_i) \times \frac{1}{2}(y_i + y_{i+1})$.

- `diff()`, `head()`, and `tail()` can be used to write a function to compute the area easily.

```
area.R <- function(x, y) {
  sum(diff(x) * (head(y, -1) + tail(y, -1))/2)
  }
area.R(graunt$x, graunt$xPo_g)/100
```

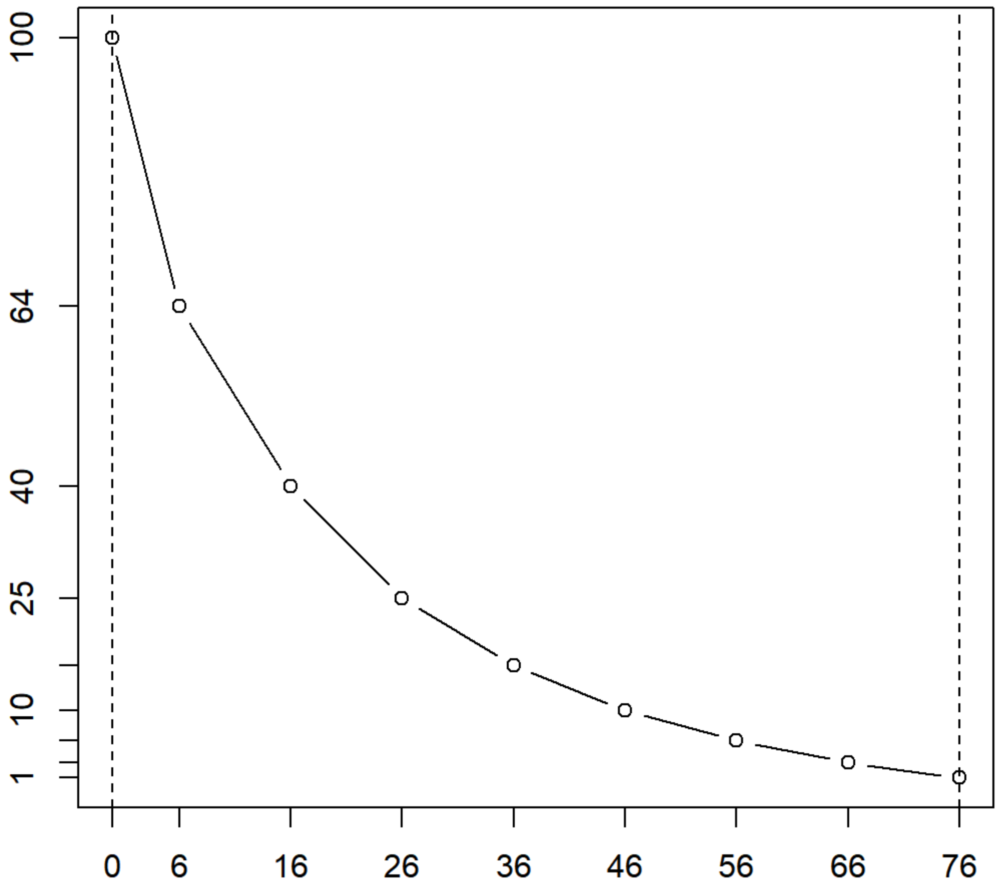
```
## [1] 18.17
```

Comparison with US 1993 life table

The shaded area between the survival function of Graunt and that of US 1993 represents the difference of life expectancies.

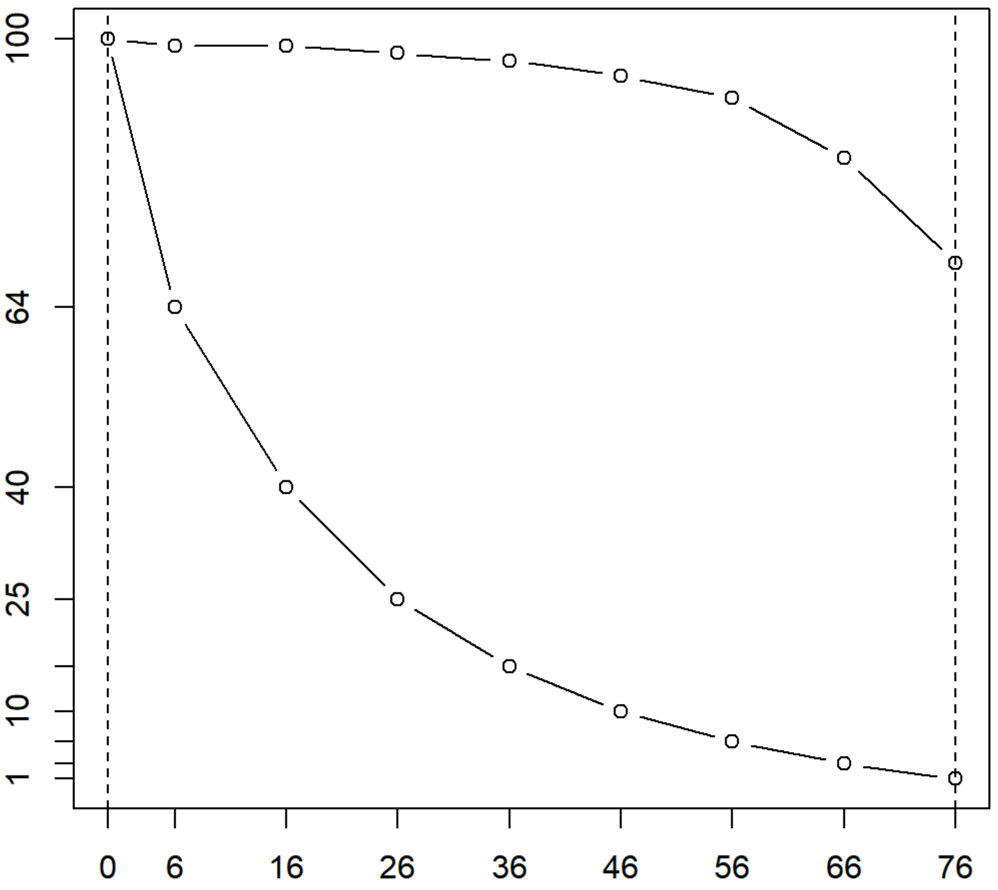
1. Draw Graunt's first with axes, lower and upper limits. Check what happens if you place `abline(...)` right after `plot(...)`.

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
```



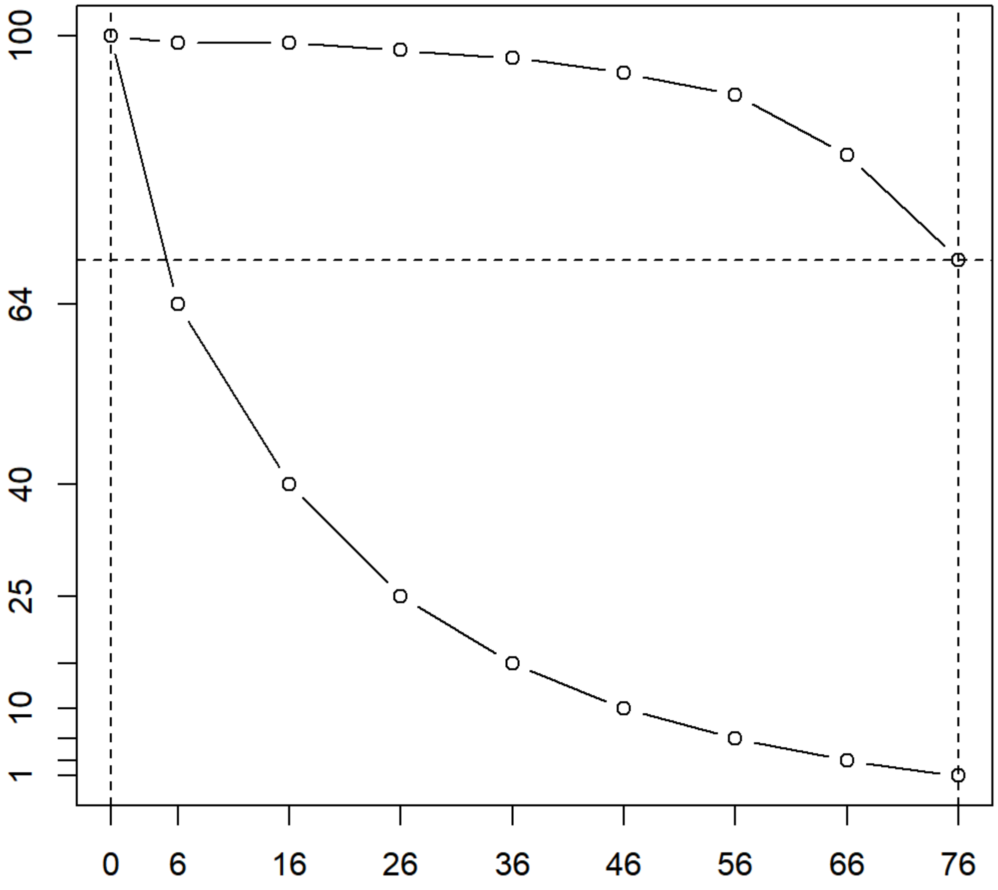
2. Add US 1993 survival function

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
lines(us93, type = "b")
```



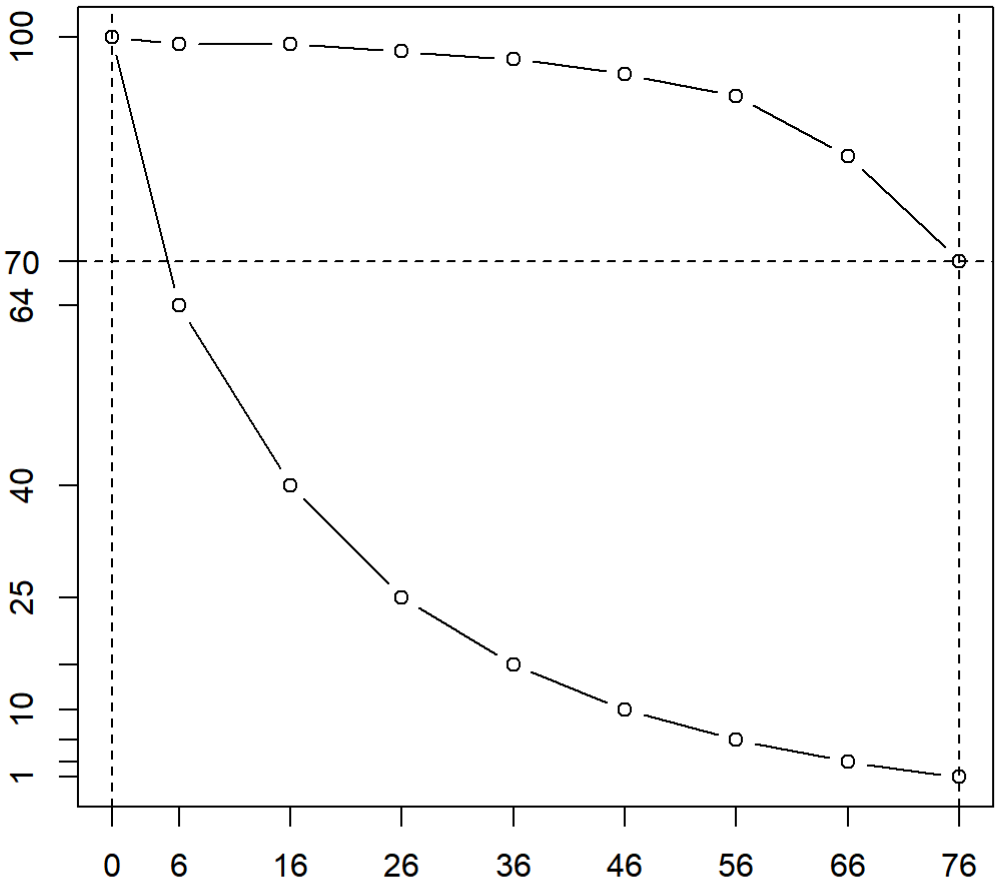
3. Actually, US 1993 life table is truncated at the age 76. Specify that point.

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
lines(us93, type = "b")
abline(h = 70, lty = 2)
```



4. Using `las = 1` to specify 70%.

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
lines(us93, type = "b")
abline(h = 70, lty = 2)
axis(side = 2, at = 70, labels = 70, las = 1)
```



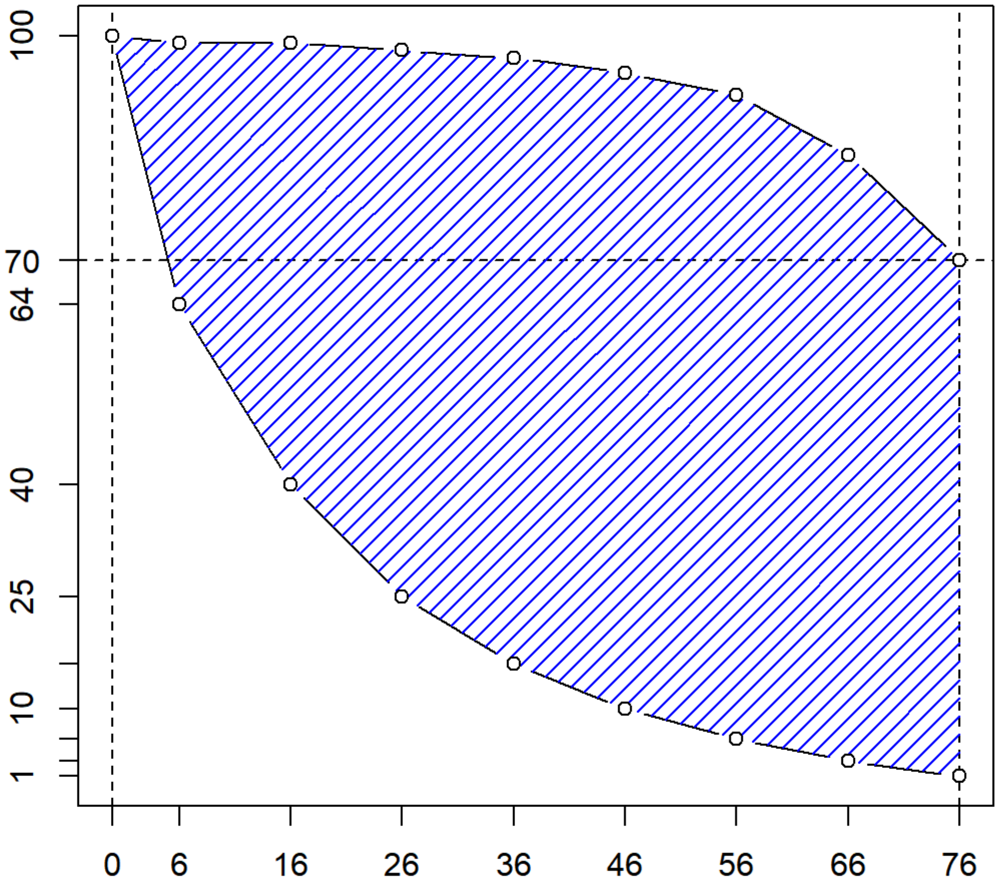
Setting coordinates for `polygon()`

```
us_graunt_x <- c(us93$x, rev(graunt$x))
us_graunt_y <- c(us93$xPo_us, rev(graunt$xPo_g))
us_graunt <- data.frame(x = us_graunt_x, y = us_graunt_y)
```

5. Shading

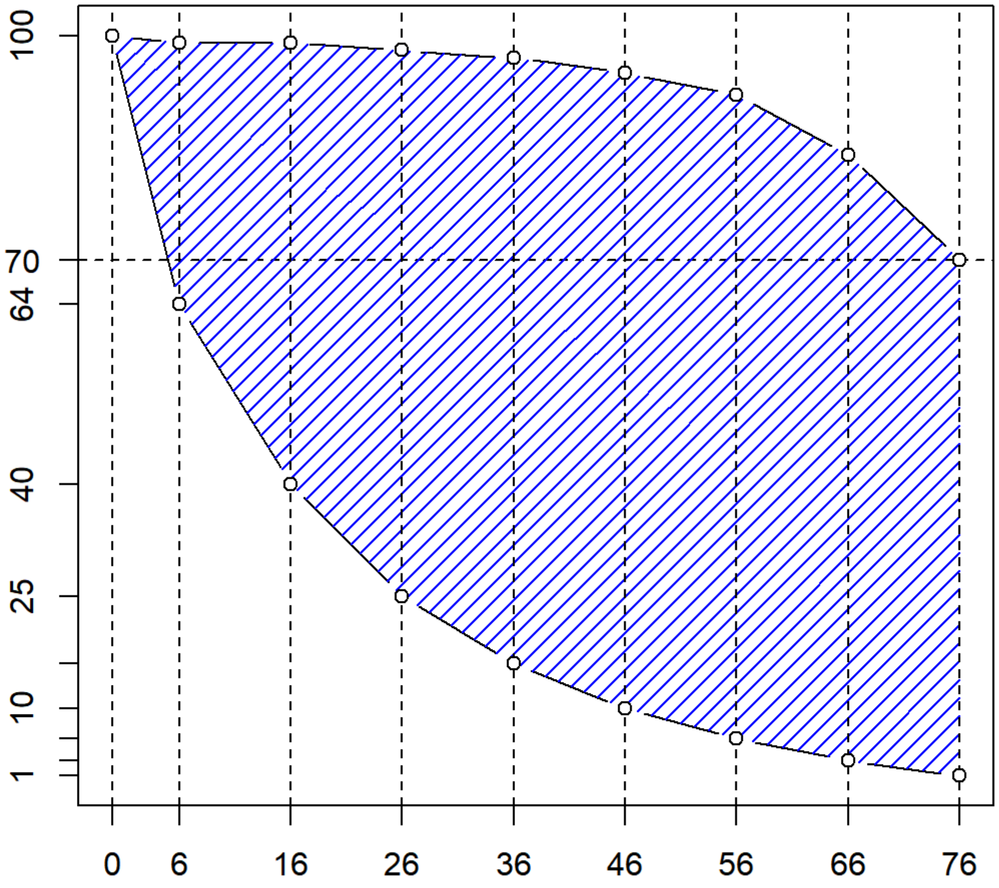
What is the effect of `border = NA` , the last line of code?

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
lines(us93, type = "b")
abline(h = 70, lty = 2)
axis(side = 2, at = 70, labels = 70, las = 1)
polygon(us_graunt, density = 15, col = "blue", border = NA)
points(us_graunt, pch = 21, col = "black", bg = "white")
```



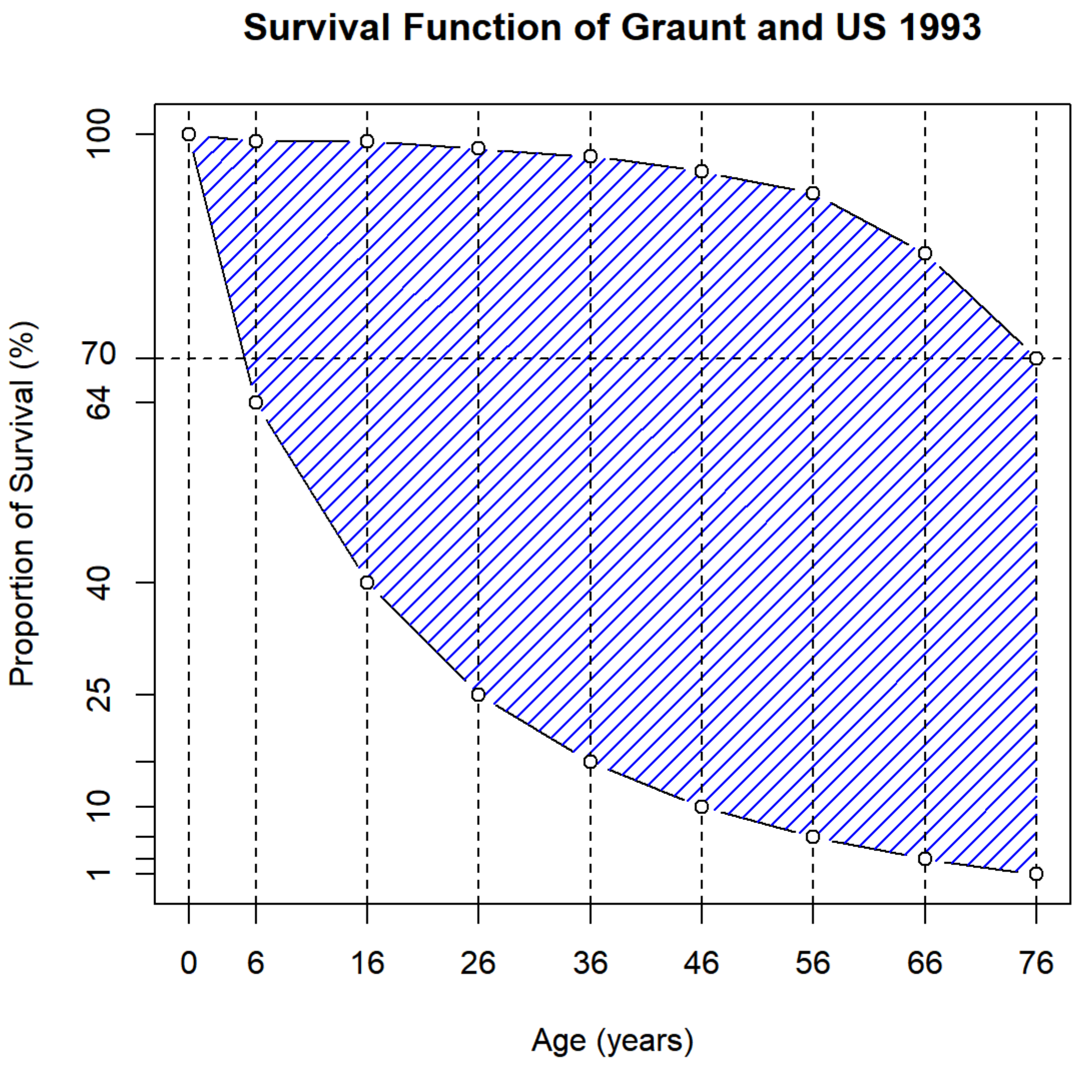
6. Grids

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
lines(us93, type = "b")
abline(h = 70, lty = 2)
axis(side = 2, at = 70, labels = 70, las = 1)
polygon(us_graunt, density = 15, col = "blue", border = NA)
abline(v = graunt$x, lty = 2)
points(us_graunt, pch = 21, col = "black", bg = "white")
```



7. Title, x-axis and y-axis labels

```
plot(graunt, ann = FALSE, xaxt = "n", yaxt = "n", type = "b")
axis(side = 1, at = graunt$x, labels = graunt$x)
axis(side = 2, at = graunt$xPo, labels = graunt$xPo_g)
abline(v = c(0, 76), lty = 2)
lines(us93, type = "b")
abline(h = 70, lty = 2)
axis(side = 2, at = 70, labels = 70, las = 1)
polygon(us_graunt, density = 15, col = "blue", border = NA)
abline(v = graunt$x, lty = 2)
points(us_graunt, pch = 21, col = "black", bg = "white")
main_title_g_us <- "Survival Function of Graunt and US 1993"
title(main = main_title_g_us, xlab = x_lab, ylab = y_lab)
```



```
dev.copy(device = png, file = "../pics/graunt_us93.png")
```

```
## png
##   3
```

```
dev.off()
```

```
## png
##   2
```

Life expectancy

The area under the US 1993 survival function is

```
area.R(us93$x, us93$xPo_us)/100
```

```
## [1] 70.92
```

The area of shaded region is

```
area.R(us93$x, us93$xPo_us)/100 - area.R(graunt$x, graunt$xPo_g)/100
```

```
## [1] 52.75
```

ggplot

```
library(ggplot2)
```

Data Reshape

Attach `reshape2` package to change wide format to long format

```
library(reshape2)
```

How `melt()` works

```
graunt_us_melt <- melt(graunt_us,
                       id.vars = "x",
                       measure.vars = c("xPo_g", "xPo_us"),
                       value.name = "xPo",
                       variable.name = "times")
graunt_us_melt
```

```
##     x  times xPo
## 1   0  xPo_g 100
## 2   6  xPo_g  64
## 3  16  xPo_g  40
## 4  26  xPo_g  25
## 5  36  xPo_g  16
## 6  46  xPo_g  10
## 7  56  xPo_g   6
## 8  66  xPo_g   3
## 9  76  xPo_g   1
## 10  0 xPo_us 100
## 11  6 xPo_us  99
## 12 16 xPo_us  99
## 13 26 xPo_us  98
## 14 36 xPo_us  97
## 15 46 xPo_us  95
## 16 56 xPo_us  92
## 17 66 xPo_us  84
## 18 76 xPo_us  70
```

```
str(graunt_us_melt)
```

```
## 'data.frame':    18 obs. of  3 variables:
##  $ x    : num  0 6 16 26 36 46 56 66 76 0 ...
##  $ times: Factor w/ 2 levels "xPo_g","xPo_us": 1 1 1 1 1 1 1 1 1 2 ...
##  $ xPo  : num  100 64 40 25 16 10 6 3 1 100 ...
```

- Change factor levels of `times`

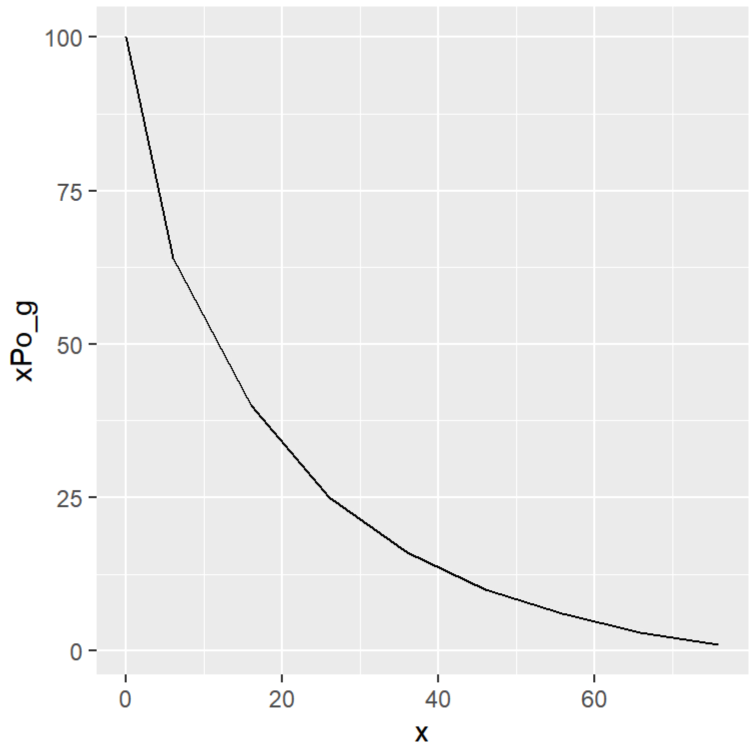
```
levels(graunt_us_melt$times) <- c("Graunt", "US1993")
graunt_us_melt
```

```
##     x  times xPo
## 1   0 Graunt 100
## 2   6 Graunt  64
## 3  16 Graunt  40
## 4  26 Graunt  25
## 5  36 Graunt  16
## 6  46 Graunt  10
## 7  56 Graunt   6
## 8  66 Graunt   3
## 9  76 Graunt   1
## 10  0 US1993 100
## 11  6 US1993  99
## 12 16 US1993  99
## 13 26 US1993  98
## 14 36 US1993  97
## 15 46 US1993  95
## 16 56 US1993  92
## 17 66 US1993  84
## 18 76 US1993  70
```

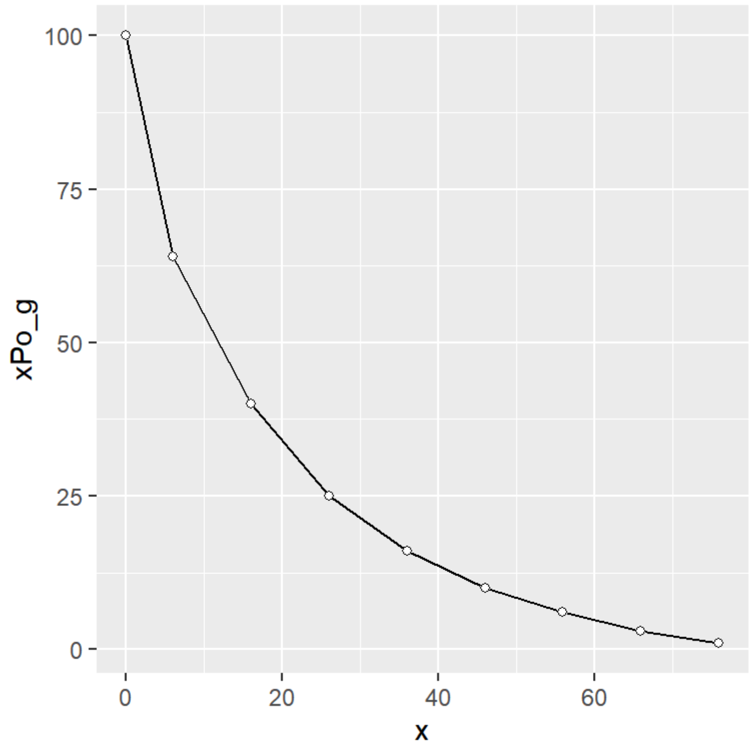
Graunt

Structure of ggplot

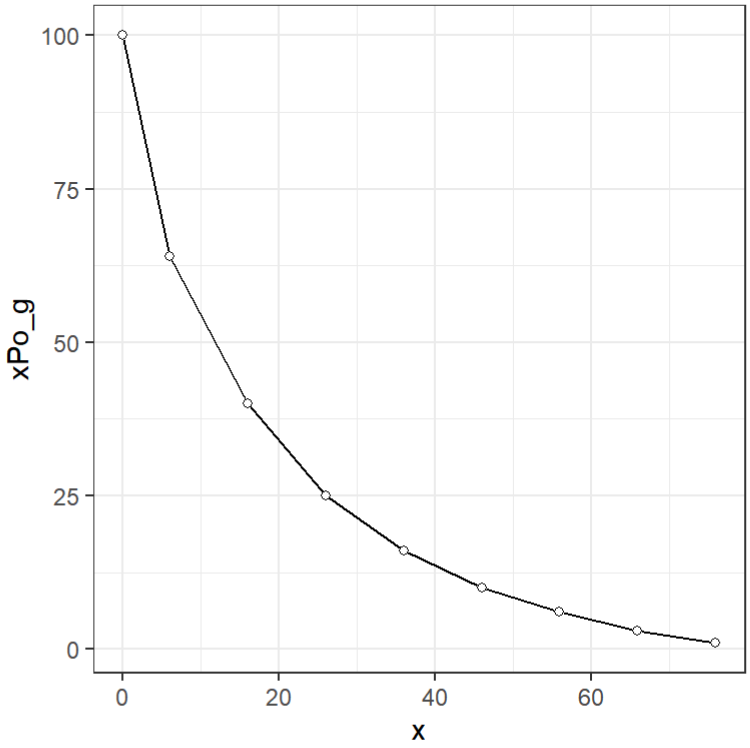
```
(g1 <- ggplot(data = graunt,
              mapping = aes(x = x, y = xPo_g)) +
   geom_line())
```



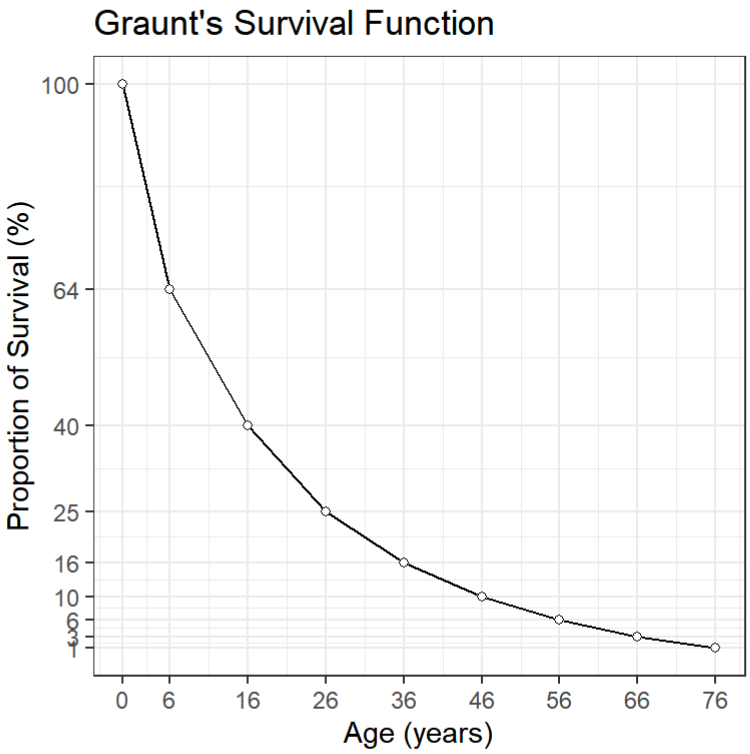
```
(g2 <- g1 +
  geom_point(shape = 21, fill = "white"))
```



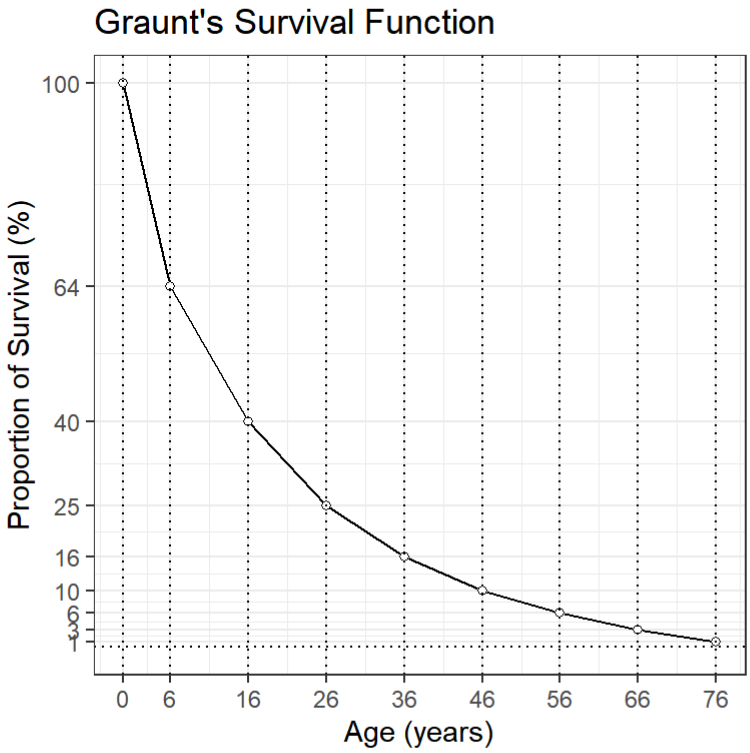
```
(g3 <- g2 +
  theme_bw())
```



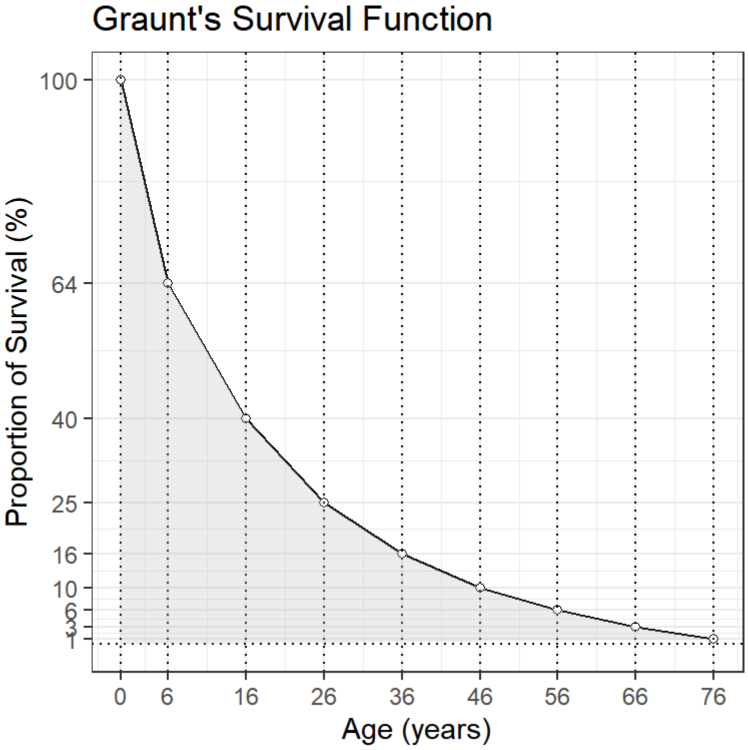
```
(g4 <- g3 +
   xlab(x_lab) +
   ylab(y_lab) +
   ggtitle(main_title) +
   scale_x_continuous(breaks = graunt$x) +
   scale_y_continuous(breaks = graunt$xPo_g))
```



```
(g5 <- g4 +
  geom_vline(xintercept = graunt$x, linetype = "dotted") +
  geom_hline(yintercept = 0, linetype = "dotted"))
```

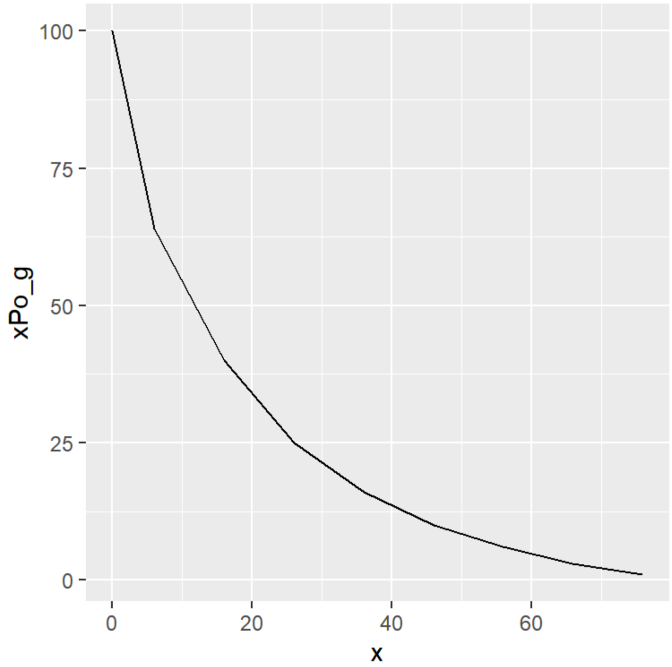


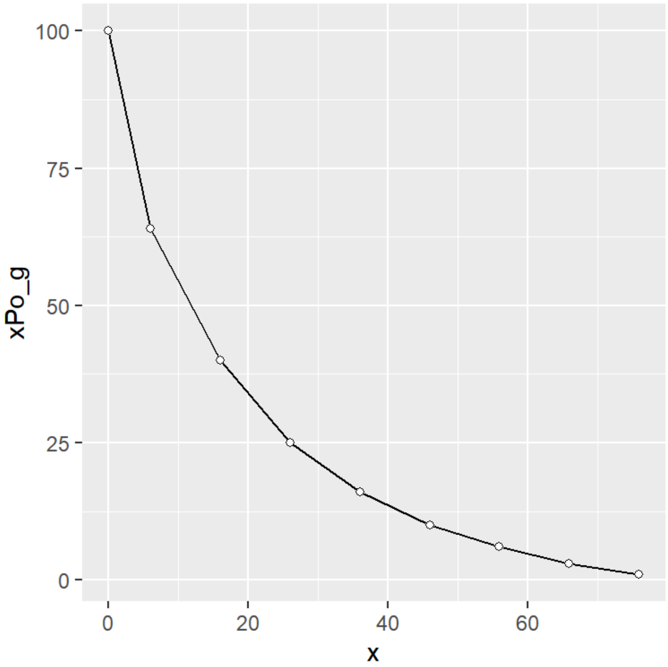
```
(pg5 <- g5 +
  geom_polygon(data = graunt_poly,
               mapping = aes(x = x, y = y),
               alpha = 0.3, fill = "grey"))
```

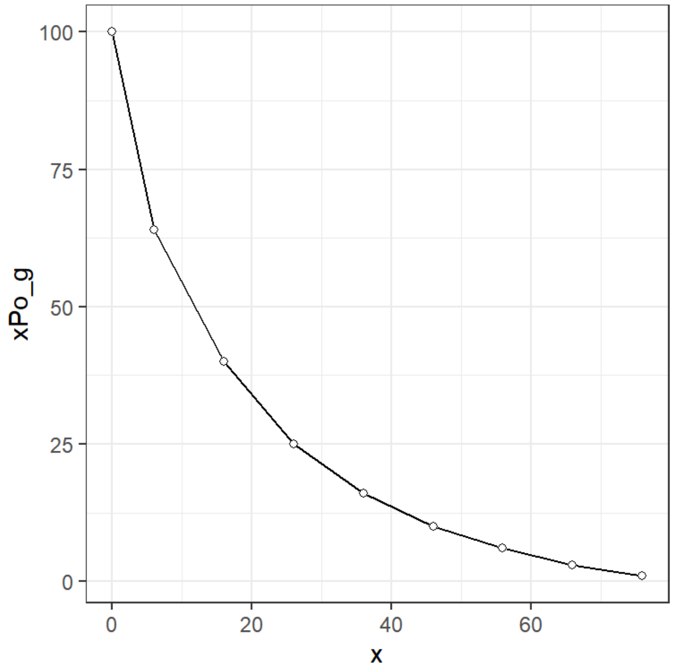


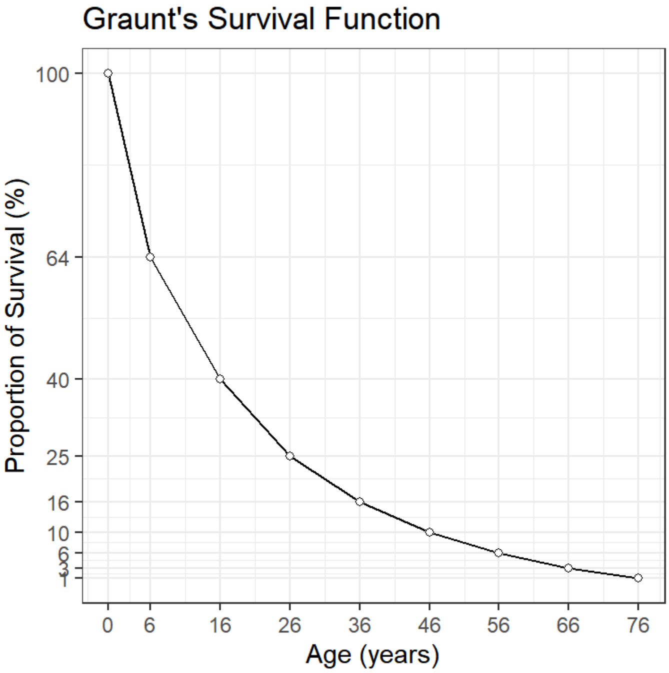
```
# ggsave("../pics/graunt_poly_ggplot.png", pg5)
```

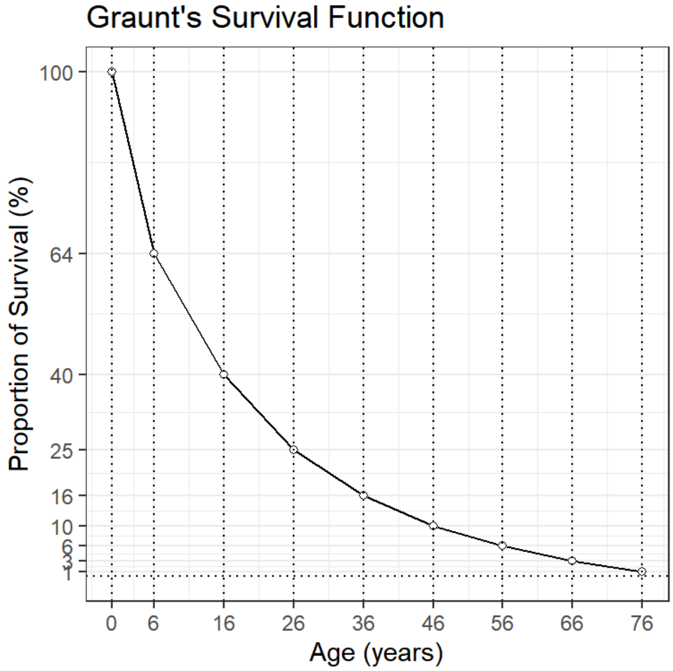
```
library(gridExtra)
g_graunt <- grid.arrange(g1, g2, g3, g4, g5, pg5, nrow = 3)
```

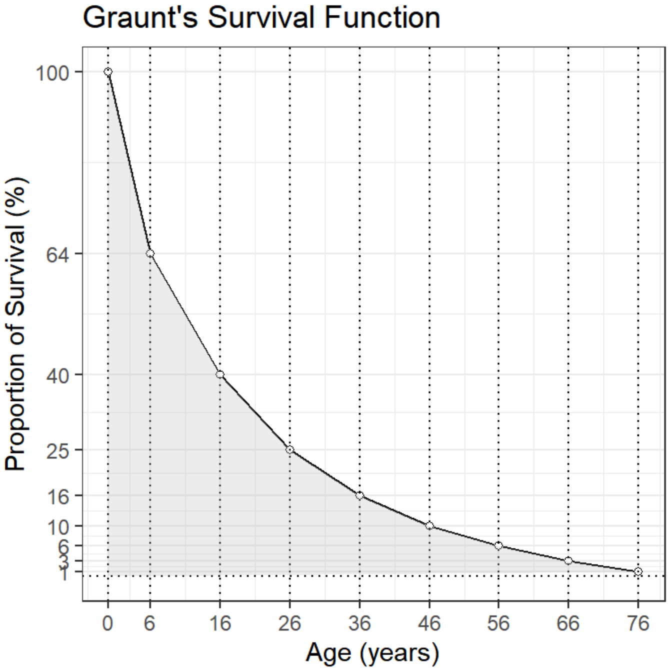

xPo_g
100
75
50
25
0
0
20
40
60
x


xPo_g
100
75
50
25
0
0
20
40
60
x


xPo_g
100
75
50
25
0
0
20
40
60
x


Graunt's Survival Function
Proportion of Survival (%)
100
64
40
25
16
10
6
3
1
0
6
16
26
36
46
56
66
76
Age (years)


Graunt's Survival Function
Proportion of Survival (%)
100
64
40
25
16
10
6
3
1
0
6
16
26
36
46
56
66
76
Age (years)


Graunt's Survival Function
Proportion of Survival (%)
100
64
40
25
16
10
6
3
1
0
6
16
26
36
46
56
66
76
Age (years)

```
# ggsave(g_graunt, file = "../pics/graunt_ggplots.png", width = 8, height = 12)
```

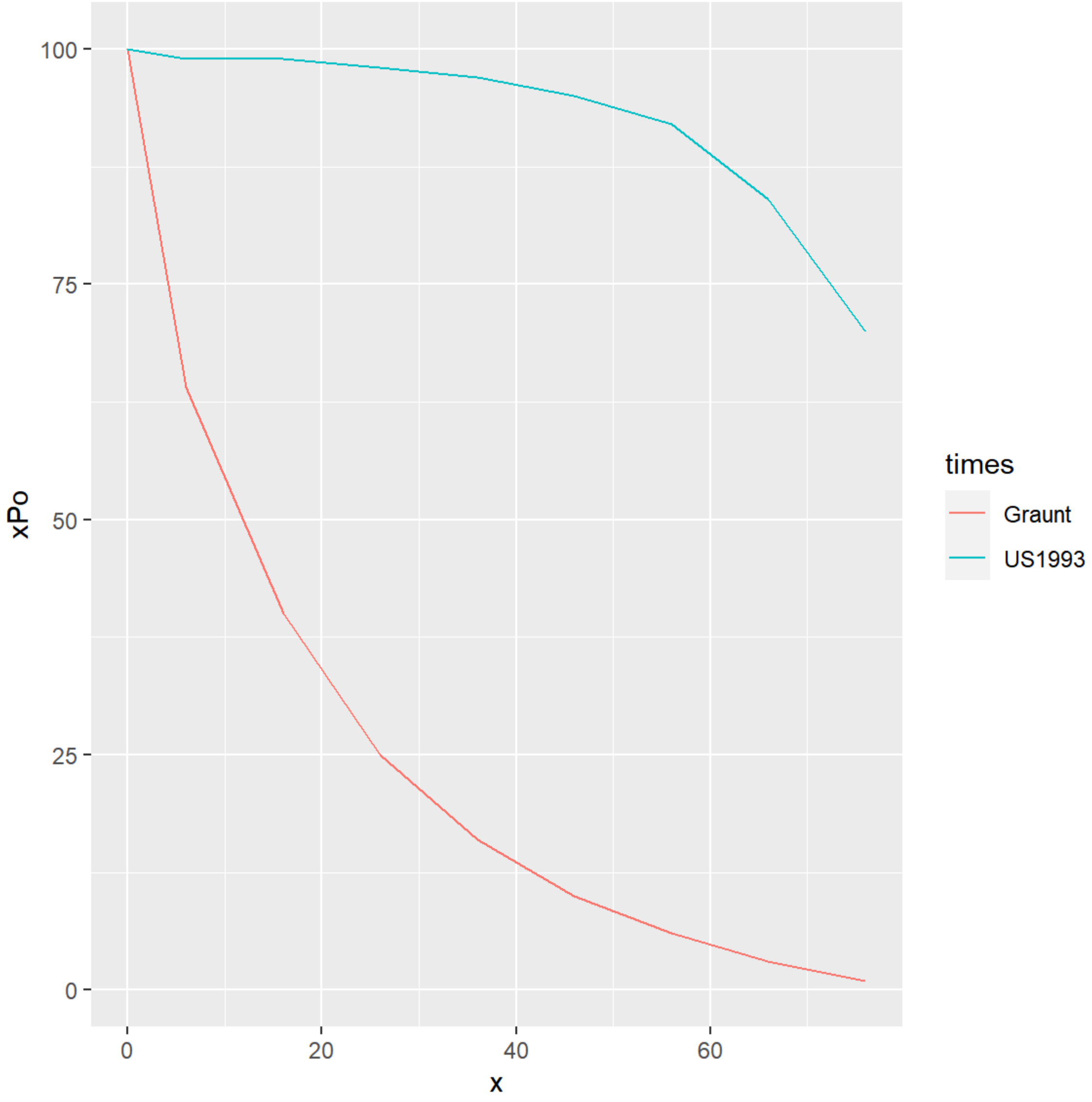
Graunt and US 1993

Points and Lines

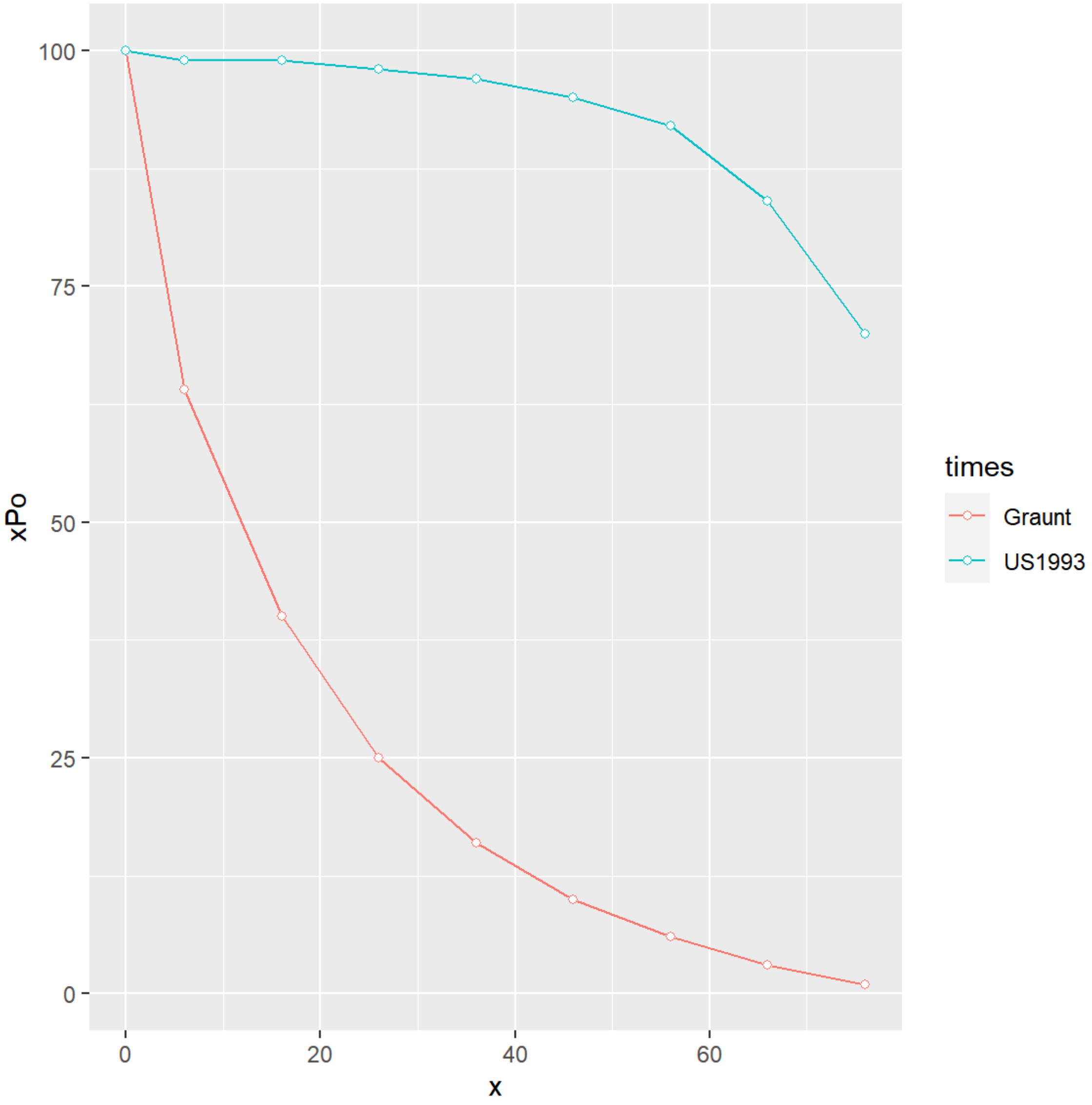
Step by step approach to understand the grammar of ggplot

- We set `ggplot()` to accept varying `data.frame()` and `aes()` in `geom_polygon`

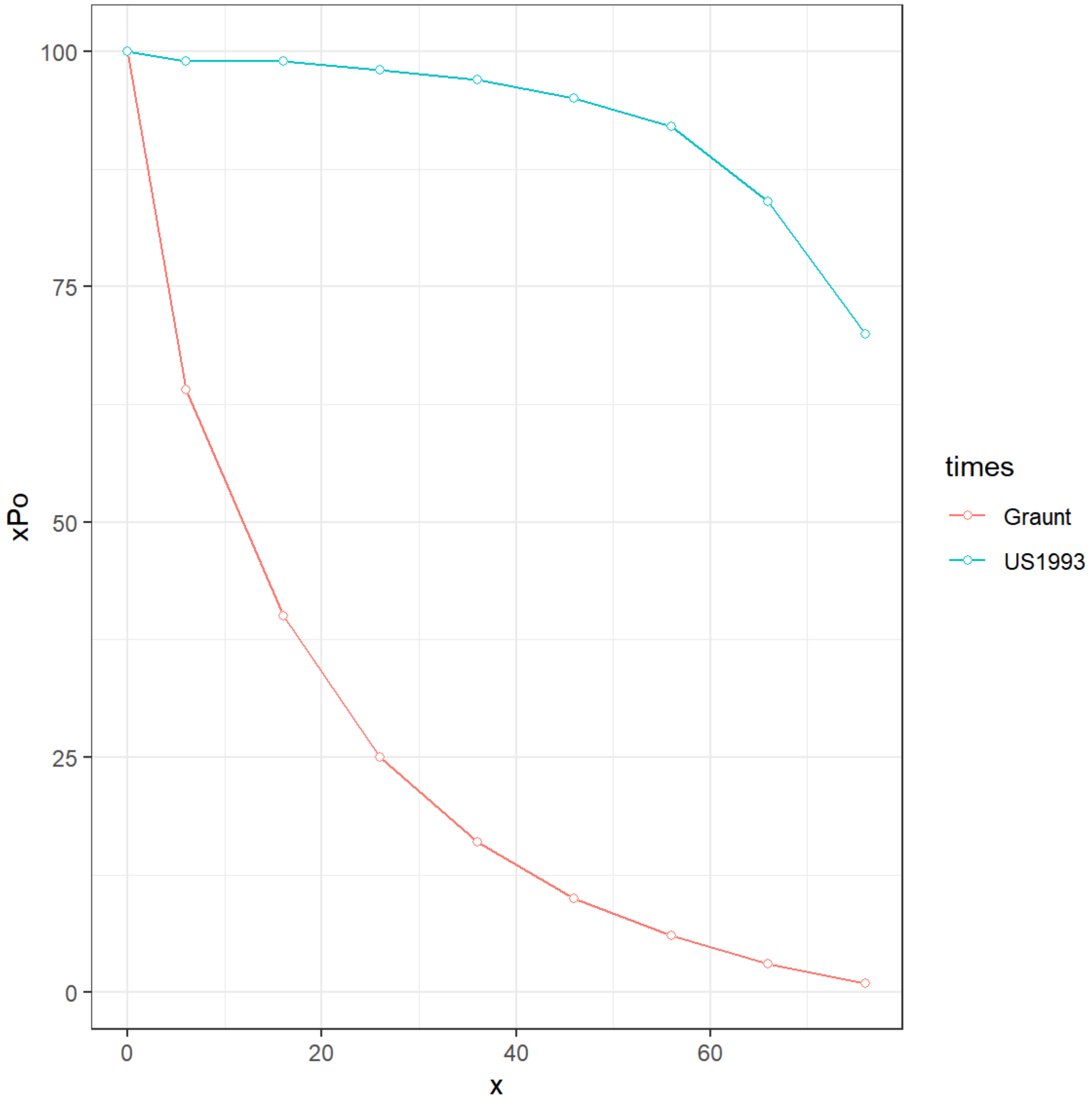
```
(gu1 <- ggplot() +
  geom_line(data = graunt_us_melt,
            mapping = aes(x = x, y = xPo, colour = times)))
```



```
(gu2 <- gu1 +
  geom_point(data = graunt_us_melt,
             mapping = aes(x = x, y = xPo, colour = times),
             shape = 21, fill = "white"))
```



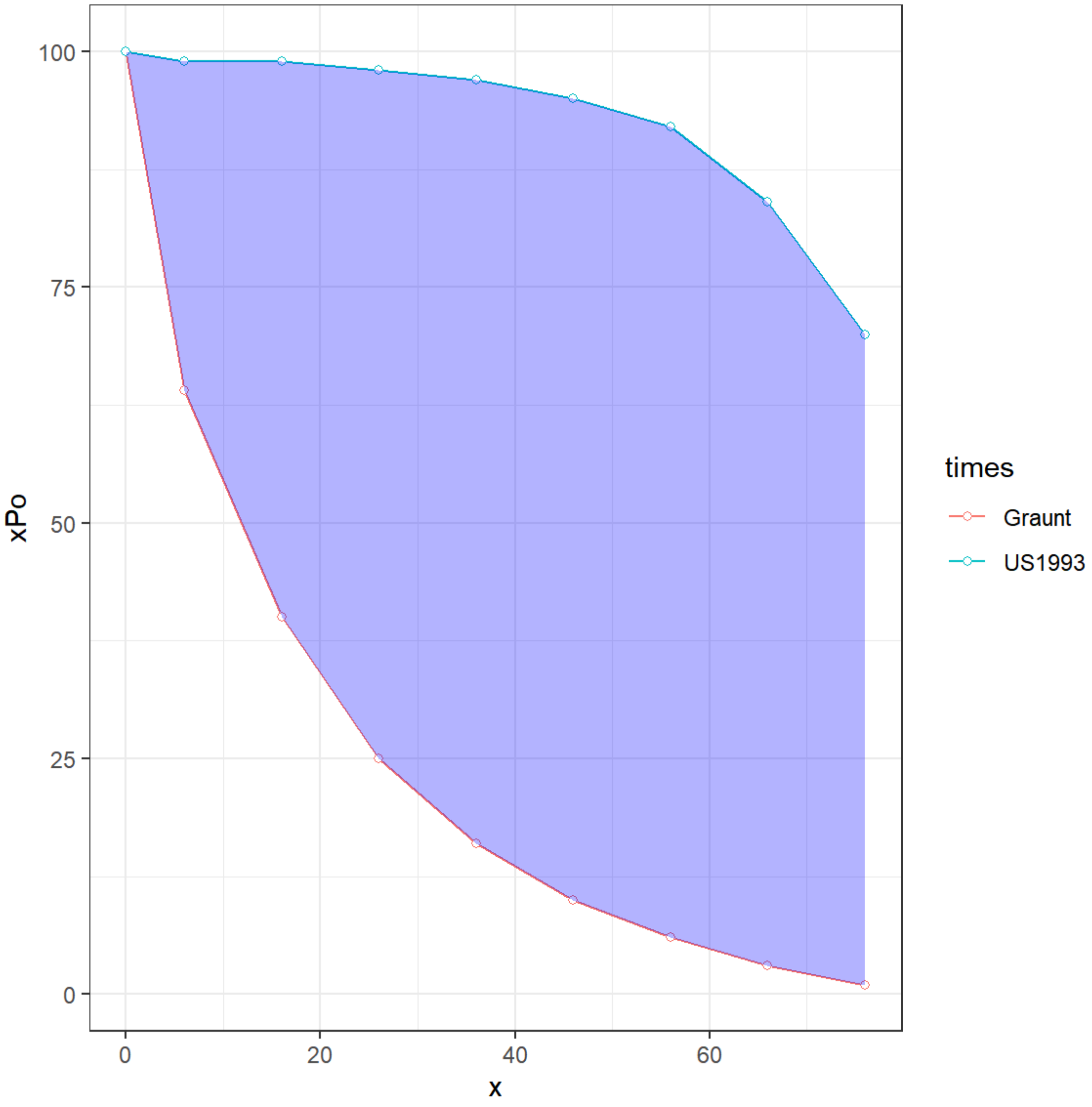
```
(gu3 <- gu2 +
  theme_bw())
```



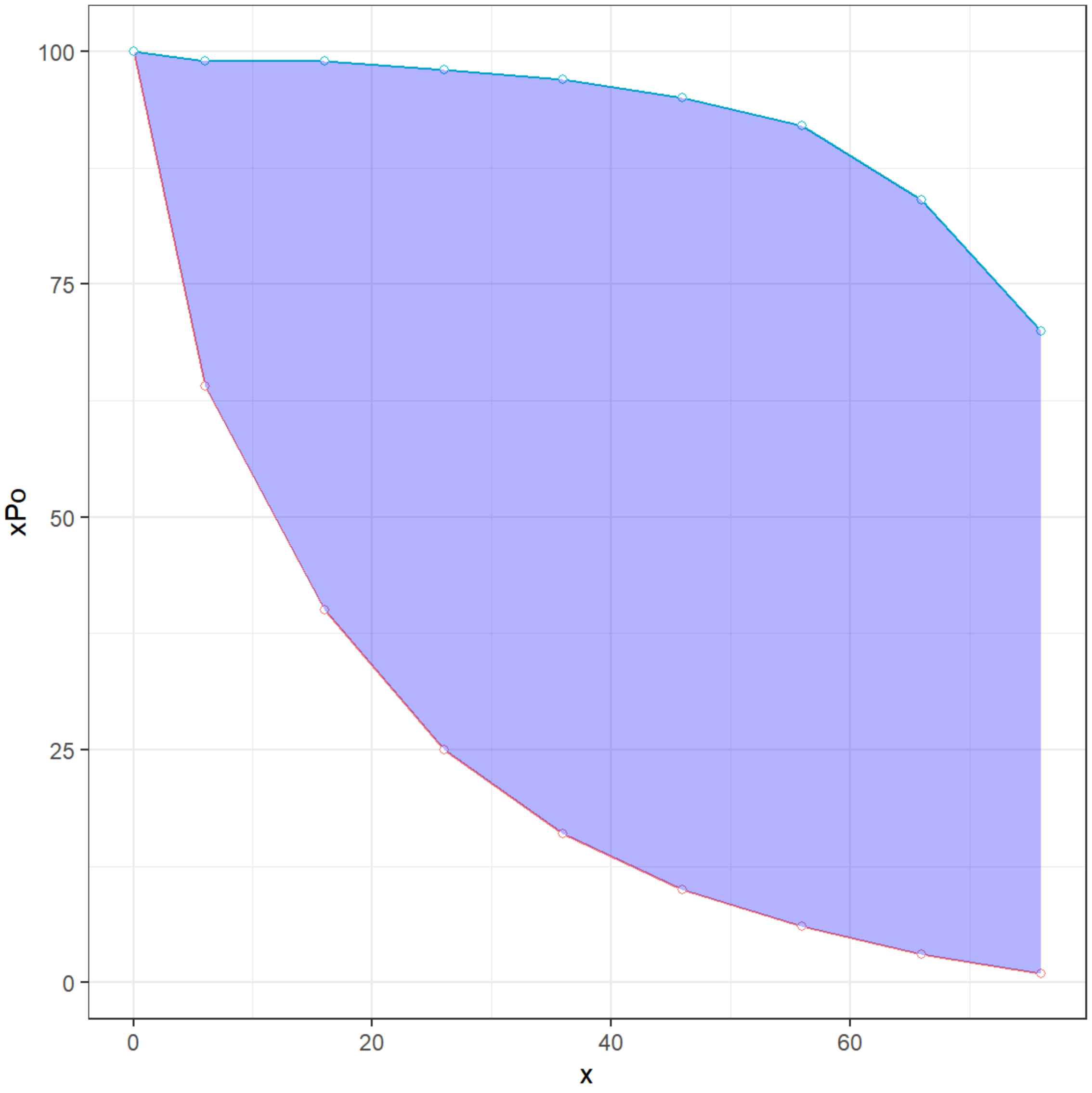
Polygon

Reuse `us_graunt` which contains `x = us_graunt_x` and `y = us_graunt_y` for `polygon()` . Note that we start with `gu3` , and also note how to remove default legends.

```
(gup3 <- gu3 +
  geom_polygon(data = us_graunt,
               mapping = aes(x = x, y = y),
               alpha = 0.3, fill = "blue"))
```



```
(gup4 <- gup3 +
  guides(colour = "none"))
```

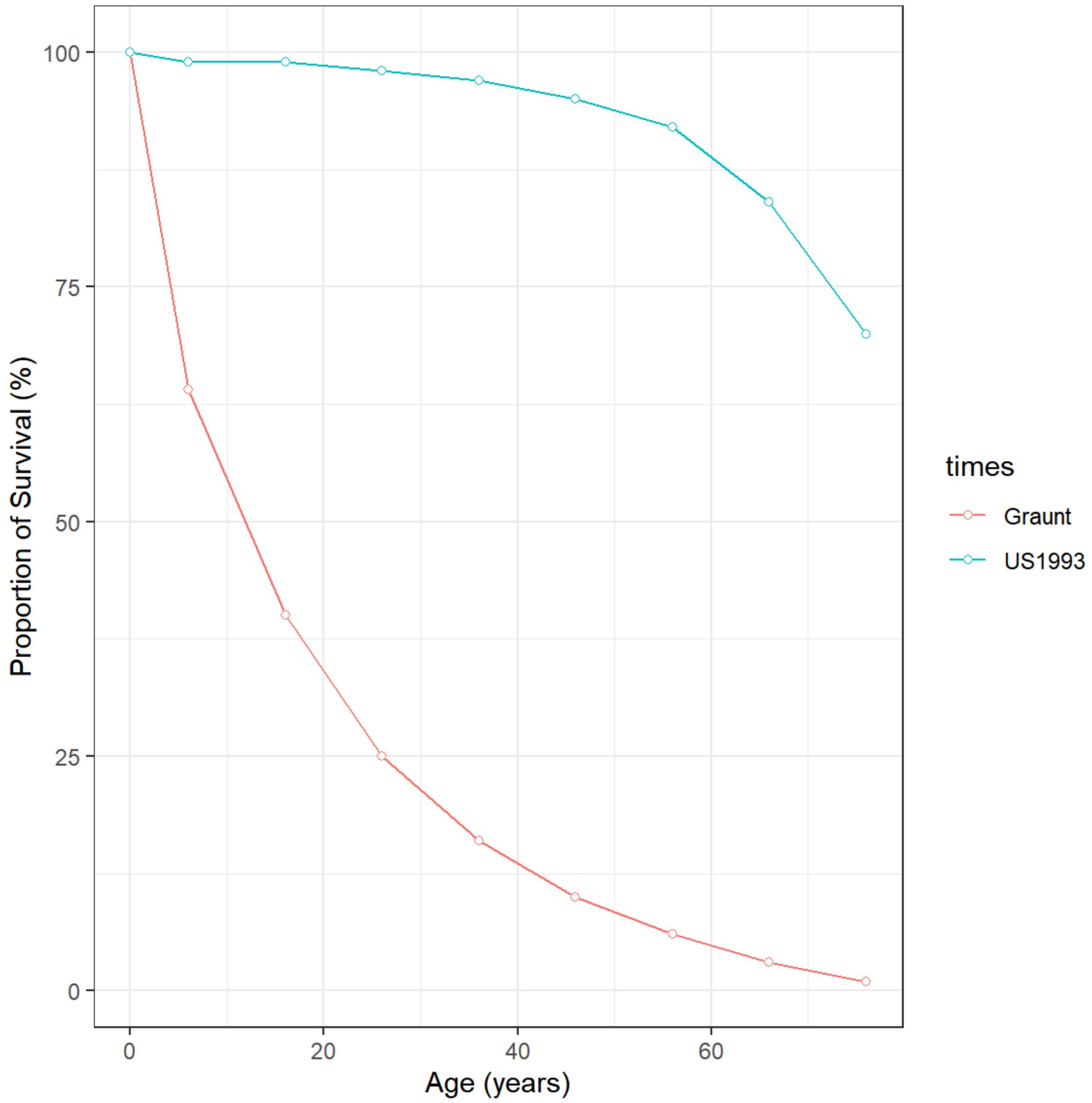


Change default annotations

Points and Lines

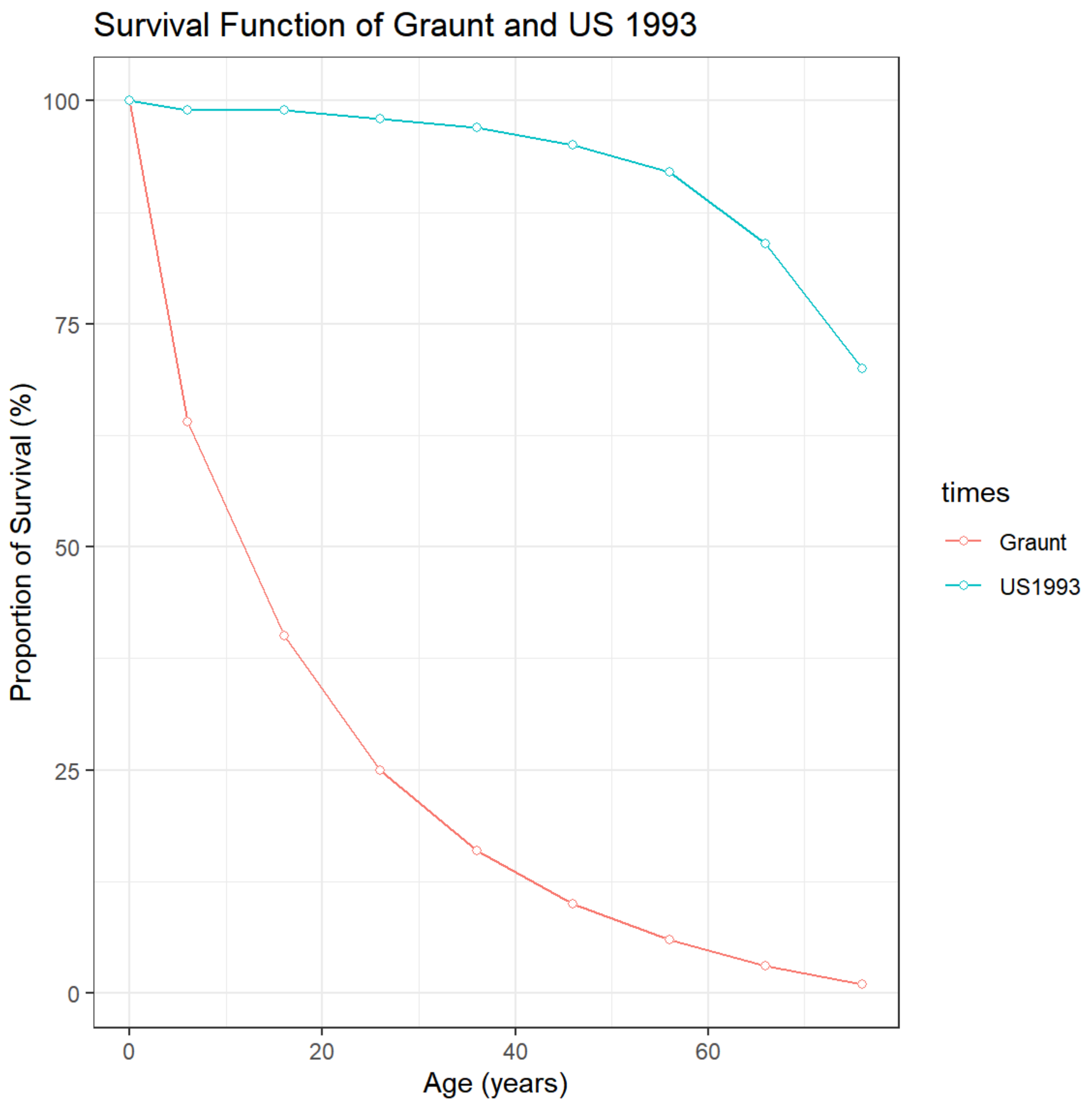
1. Change the x-axis and y-axis labels

```
(gu4 <- gu3 +
   xlab(x_lab) +
   ylab(y_lab))
```



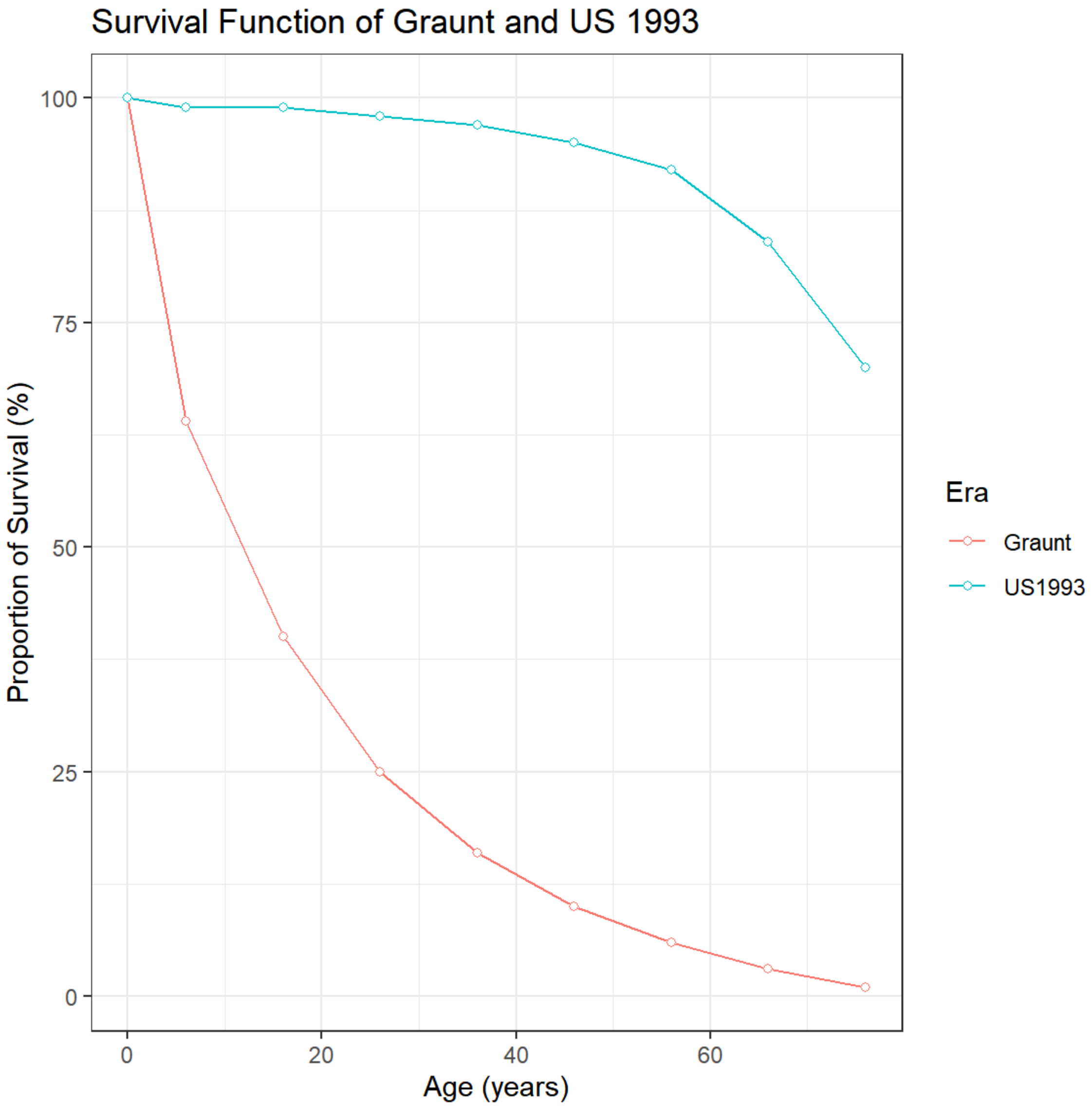
2. Add main title

```
(gu4 <- gu3 +
   xlab(x_lab) +
   ylab(y_lab) +
   ggtitle(main_title_g_us))
```



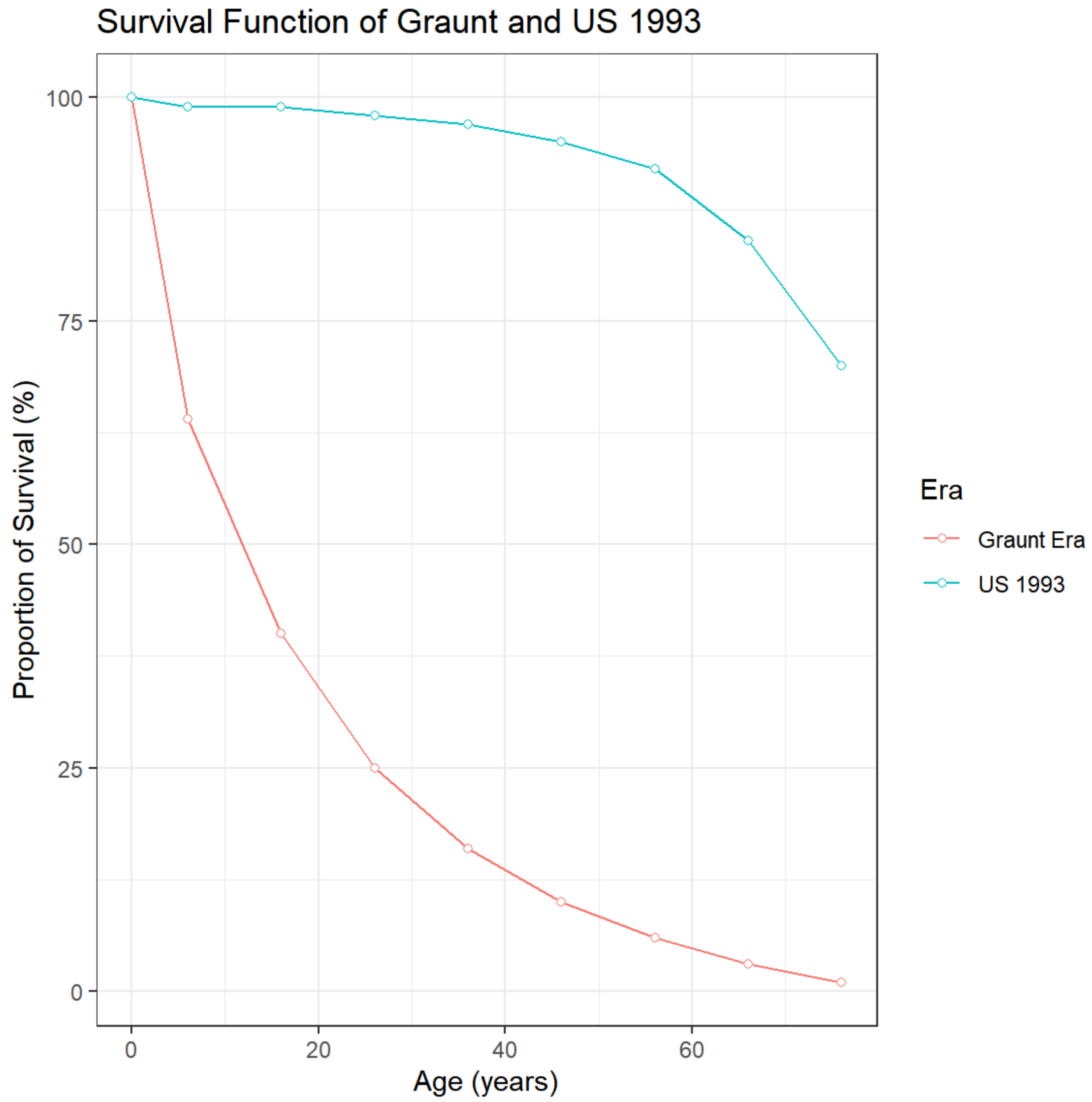
3. Change legend title

```
(gu4 <- gu3 +
   xlab(x_lab) +
   ylab(y_lab) +
   ggtitle(main_title_g_us) +
   labs(colour = "Era"))
```



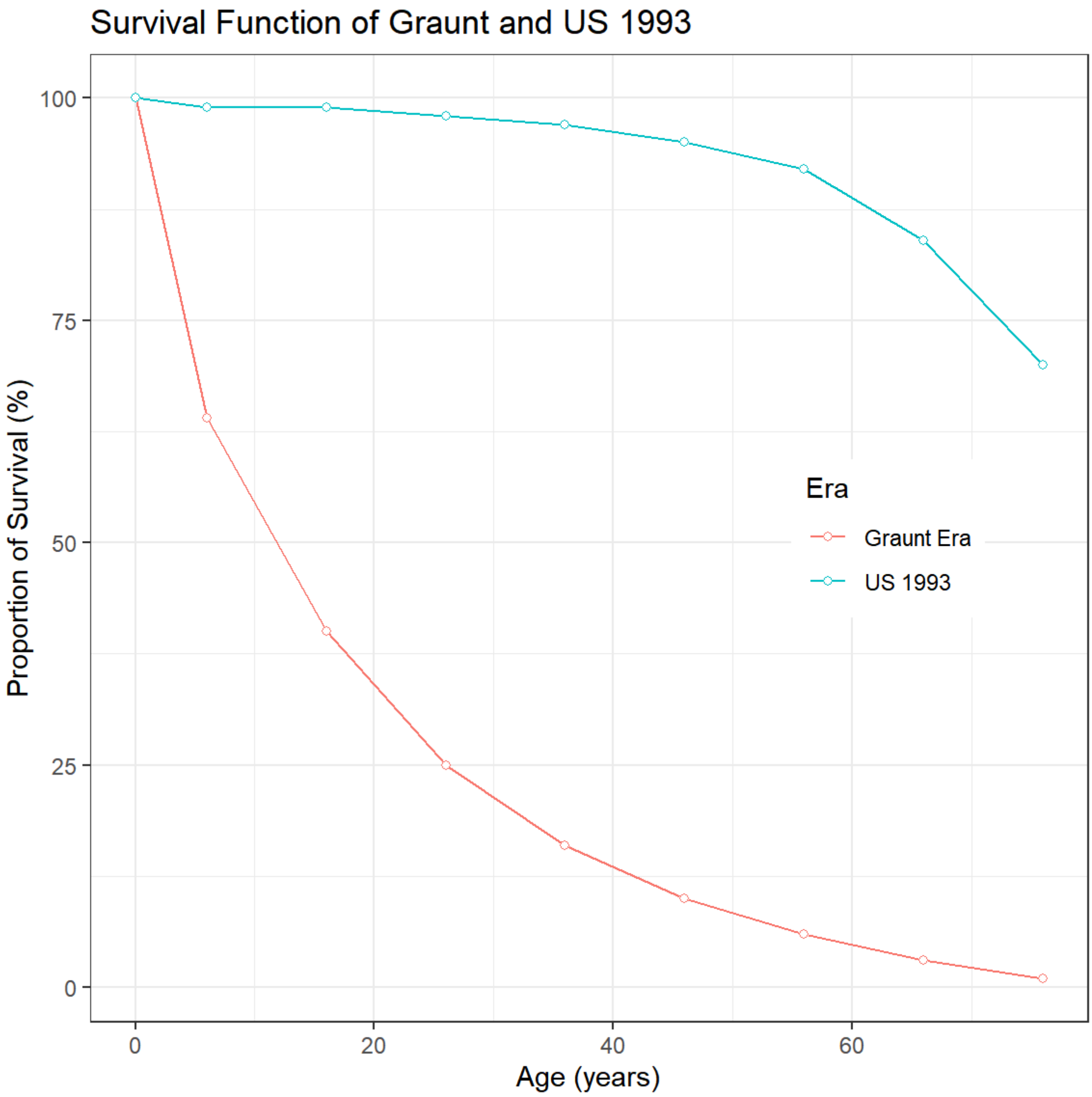
4. Change legends.

```
(gu4 <- gu3 +
   xlab(x_lab) +
   ylab(y_lab) +
   ggtitle(main_title_g_us) +
   labs(colour = "Era") +
   scale_colour_discrete(labels = c("Graunt Era", "US 1993")))
```



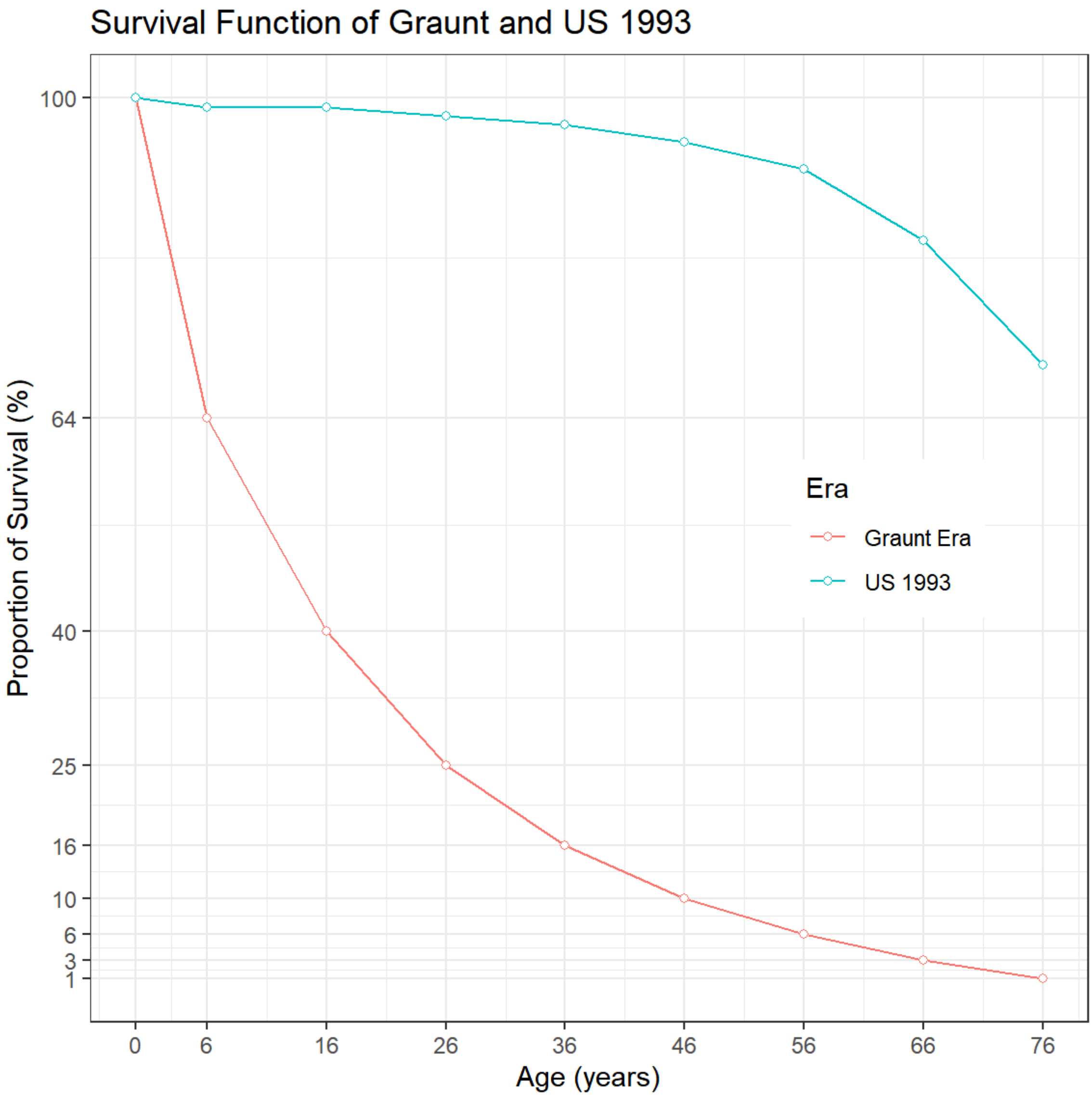
5. Place legends inside the plot

```
(gu5 <- gu4 +
   theme(legend.position = c(0.8, 0.5)))
```



6. Change x-axis and y-axis tick marks

```
(gu6 <- gu5 +
   scale_x_continuous(breaks = graunt$x) +
   scale_y_continuous(breaks = graunt$xPo_g))
```



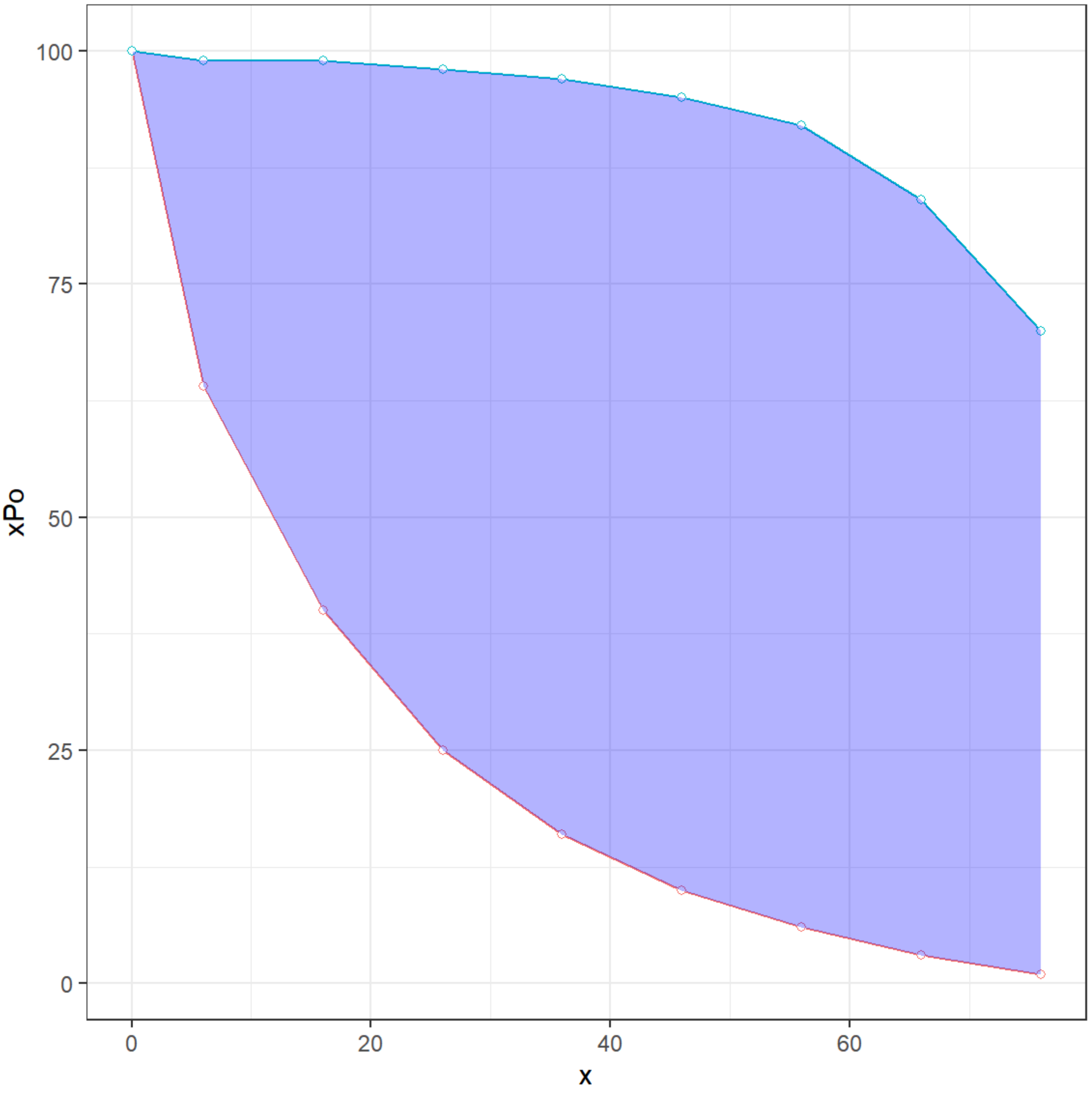
```
# ggsave("../pics/graunt_us_ggplot.png", gu6)
```

Polygon

Add information to the plot drawn with `polygon()`

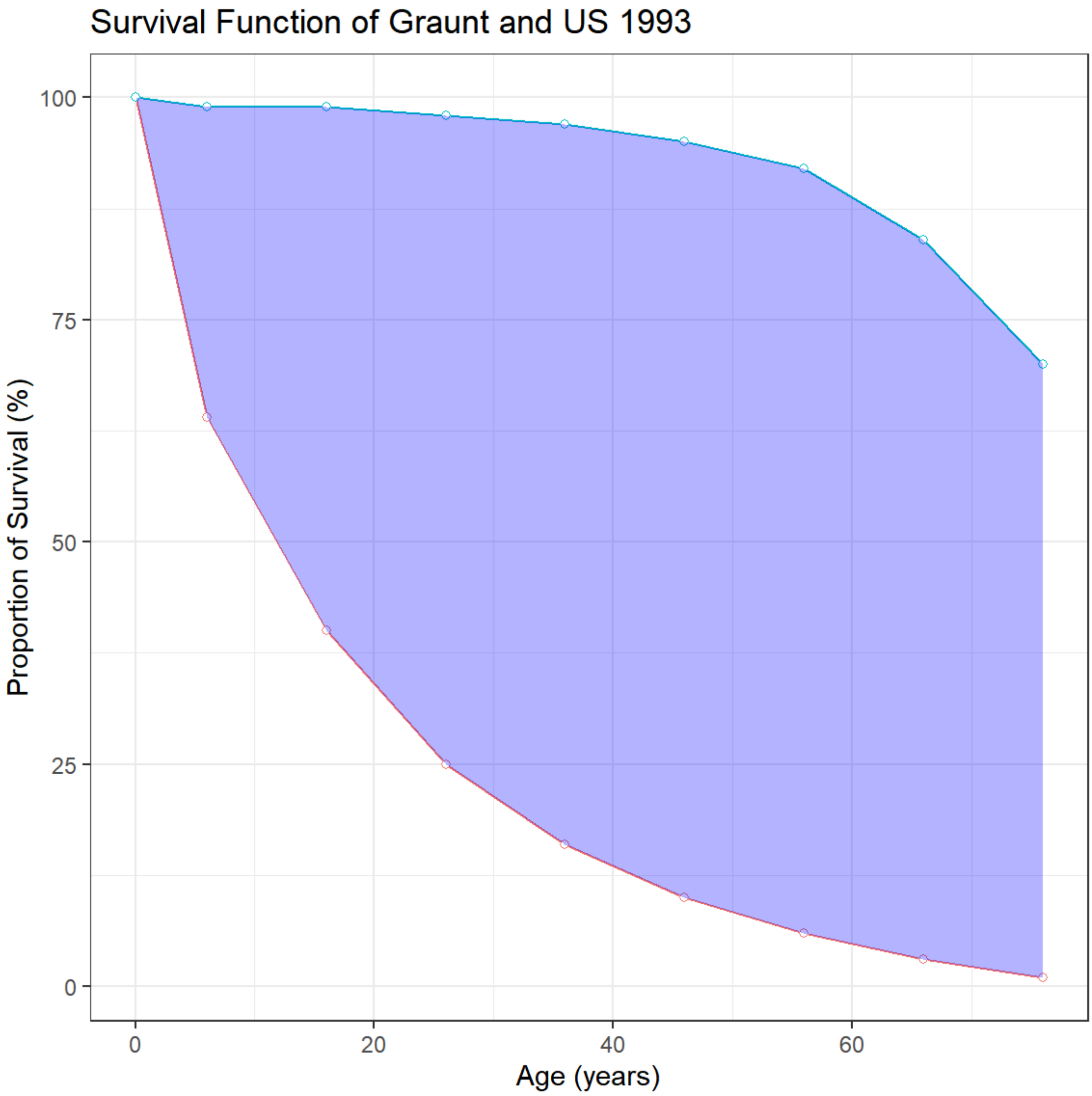
1. Start with `gup4`

```
gup4
```



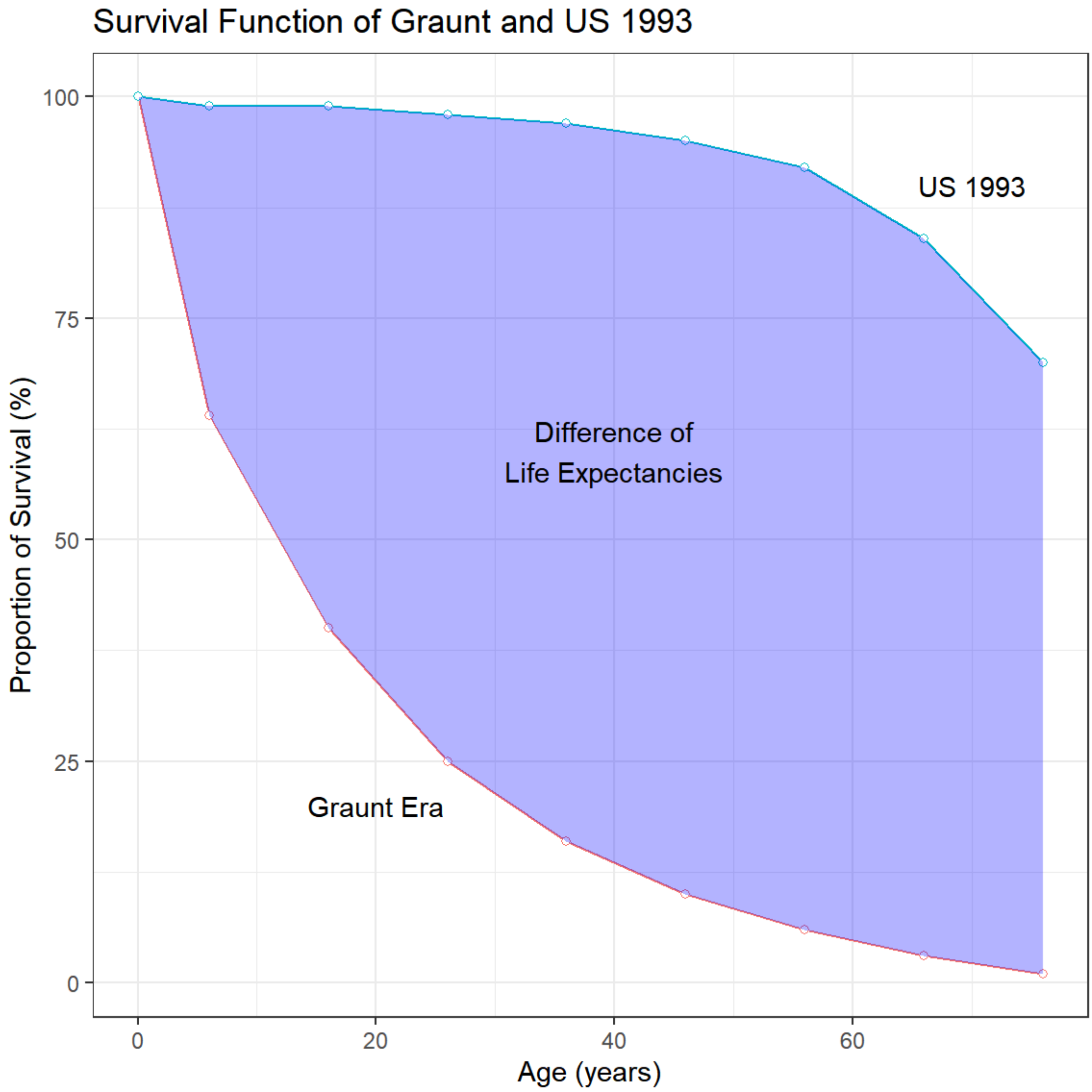
2. Main title, x-axis and y-axis labels

```
(gup5 <- gup4 +
   xlab(x_lab) +
   ylab(y_lab) +
   ggtitle(main_title_g_us))
```



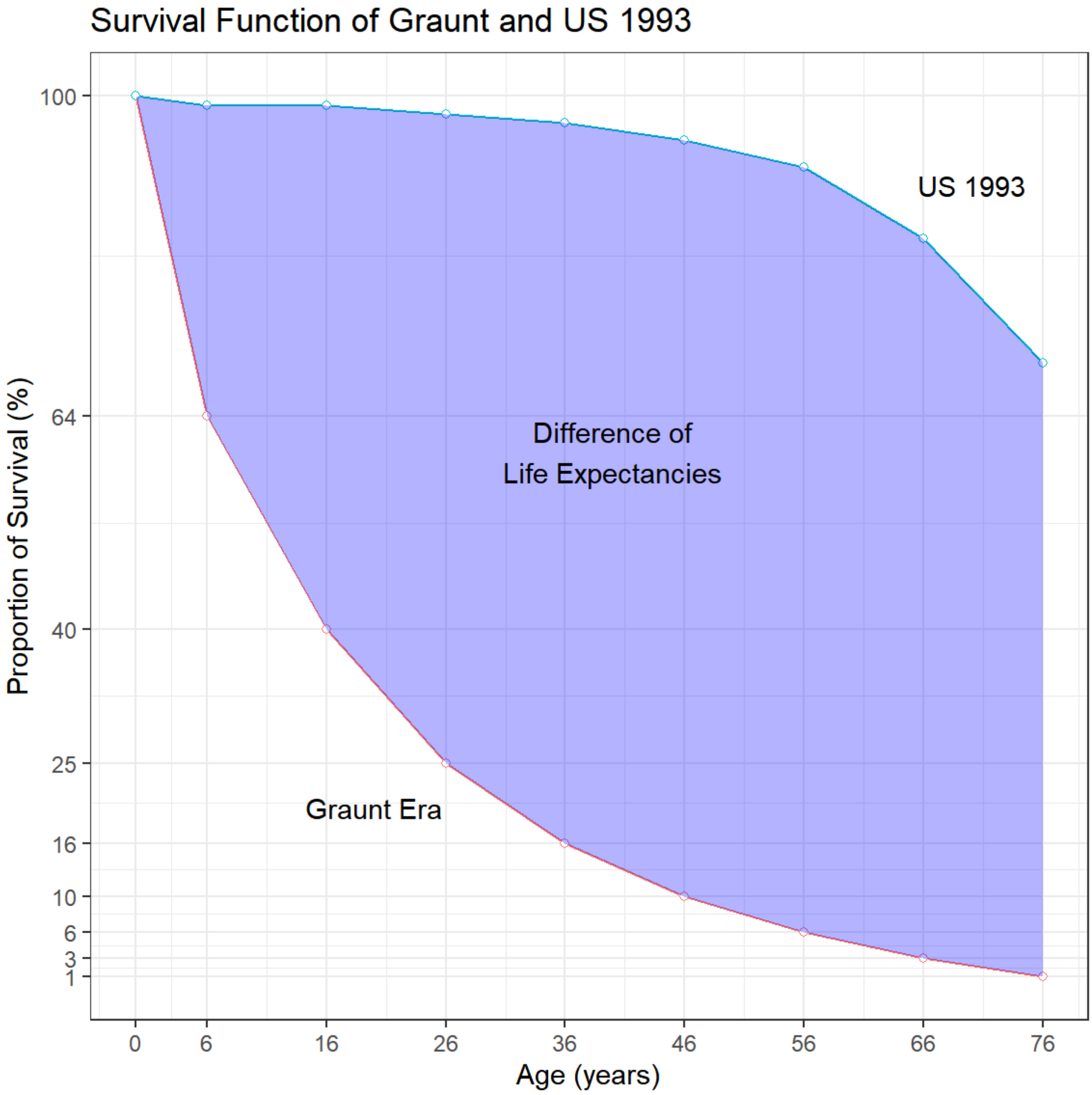
3. `"Graunt Era"`, `"US 1993"`, `"Difference of Life Expectancies"` at proper positions

```
(gup6 <- gup5 +
   annotate("text",
            x = c(20, 40, 70), y = c(20, 60, 90),
            label = c("Graunt Era", "Difference of₩nLife Expectancies", "US 1993"),
            family = ""))
```



4. x-axis and y-axis tick marks

```
(gup7 <- gup6 +
   scale_x_continuous(breaks = graunt$x) +
   scale_y_continuous(breaks = graunt$xPo_g))
```



```
# ggsave("../pics/graunt_us_poly.png", gup7)
```

`dump()` and `source()`

- Check out how to save and retrieve. Use `source()` and `load()` for retrieval.

```
dump("area.R", file = "area.R")
save.image("./graunt_halley.RData")
```

Comments

이번 시간에는 생명표를 통해 그래프를 그리는 법을 배울수 있었습니다. 이를 통해 1600년대와 1900년대의 기대수명을 알수 있었는데, 의학기술의 발전에 따라 기대수명의 증가량이 높다는 사실을 알게 되었습니다. 또한 그래프를 통해 기대수명의 차이가 더 직관적여서 쉽게 알아볼수 있었습니다. 이를 통해 보험계리의 기본을 배울수 있었다고 생각합니다.