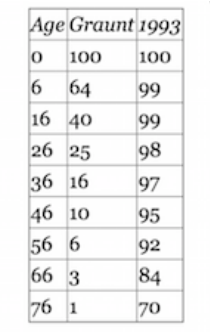
Graunt, Halley, and US 1993 Life Table with ggplot

coop711

2017-04-09

Source of Data



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

```
rm(list = ls())
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.2 <- 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.

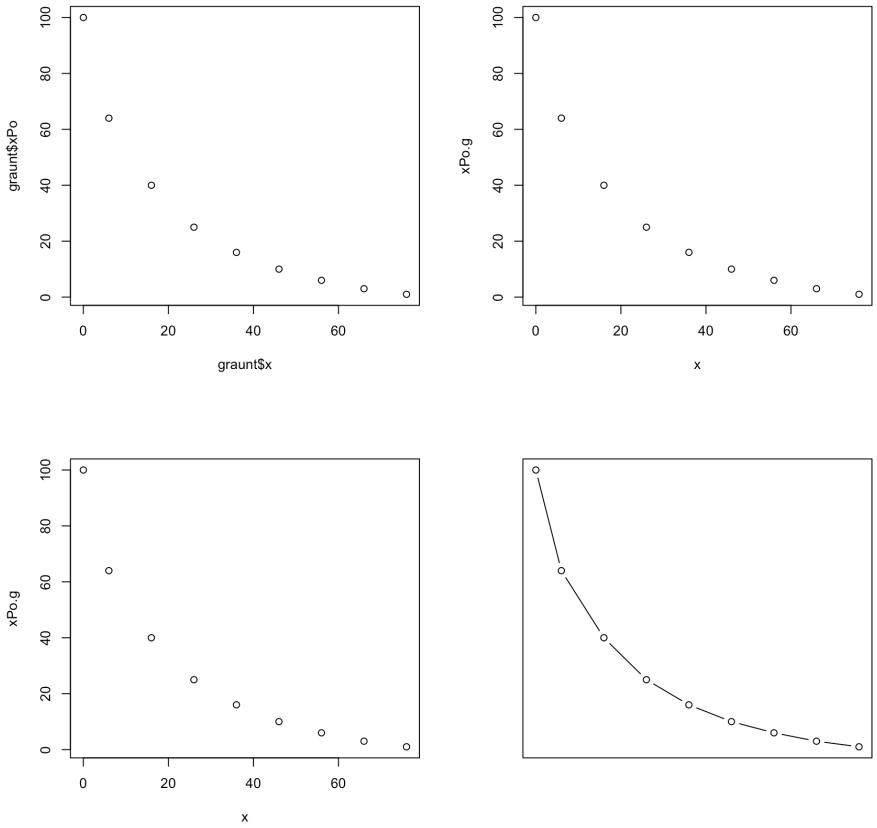
$X \geq 0, X \sim F(x) \Rightarrow X \equiv F^{-1}(U), U \sim U(0,1)$, therefore,

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

Step by step approach to draw survival function plot

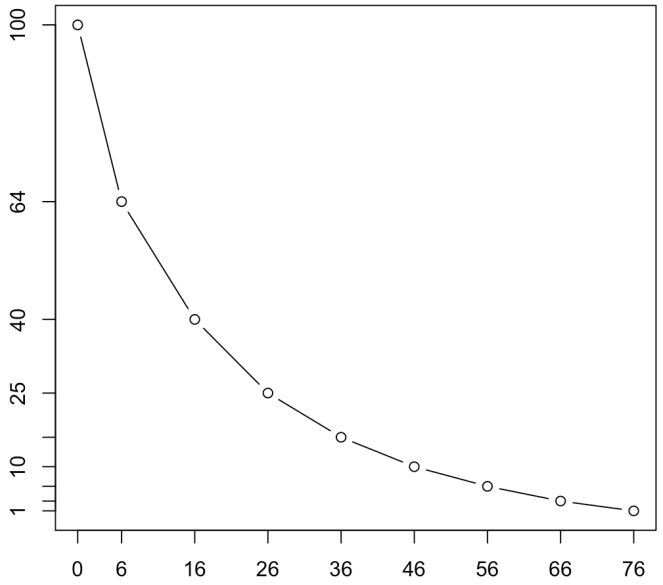
1. Basic plot with points and lines, compare the following threes methods

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



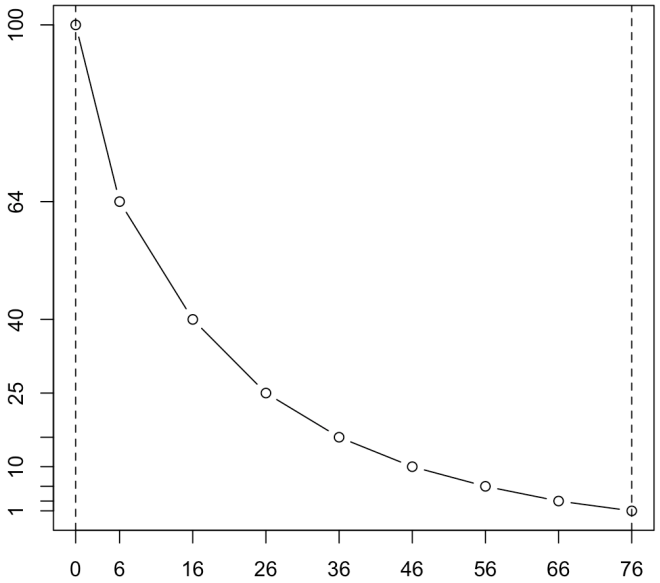
2. Denote the ages and observed survival rates on the axes

```
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)
```



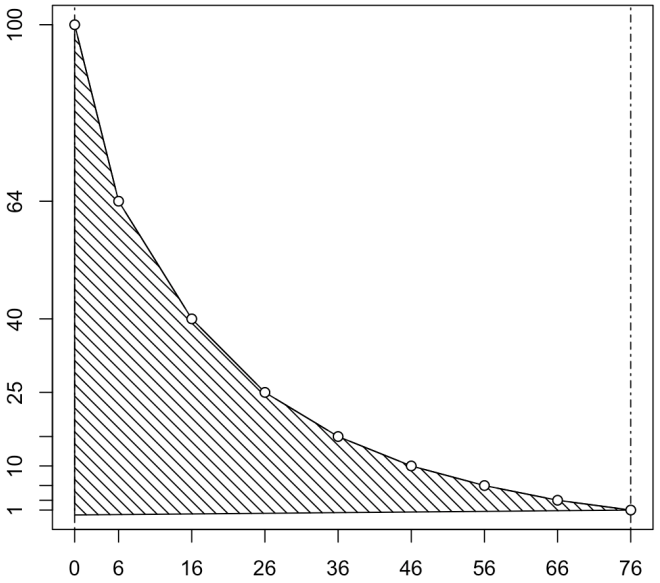
Setting up coordinates for `polygon()` (Clockwise)

```
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

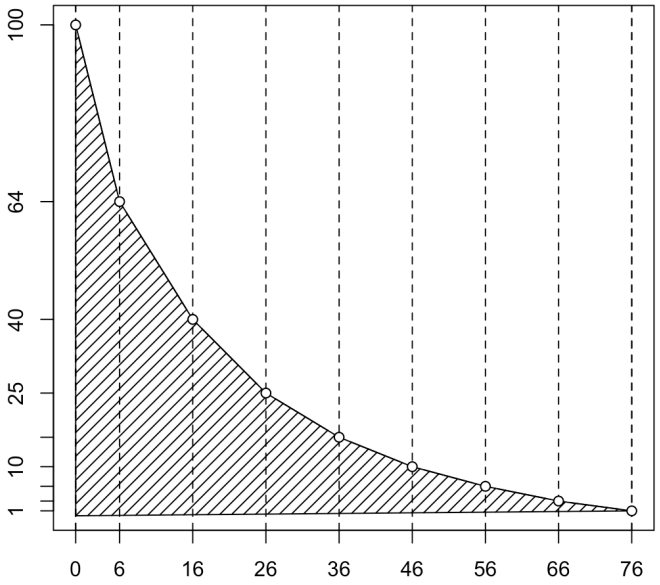
Note the effect of the last line of code.

```
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")
```



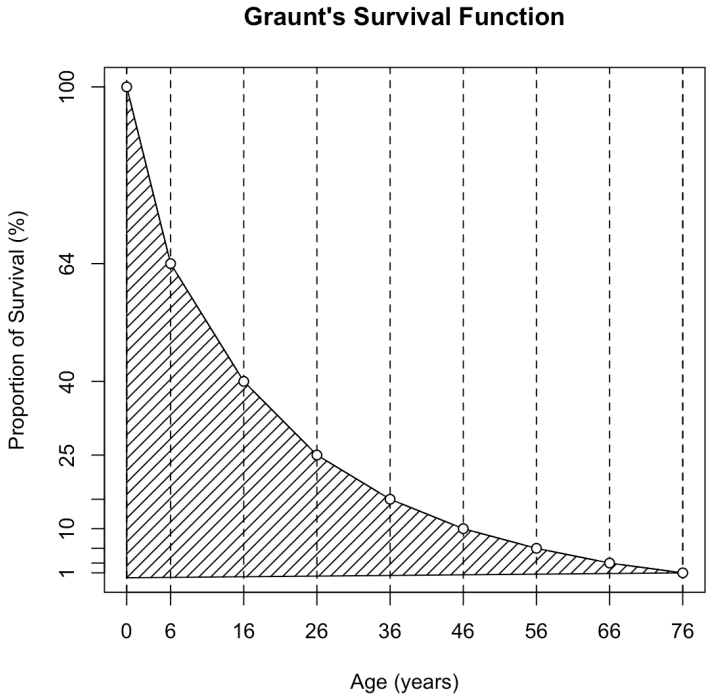
5. Grids

```
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
```

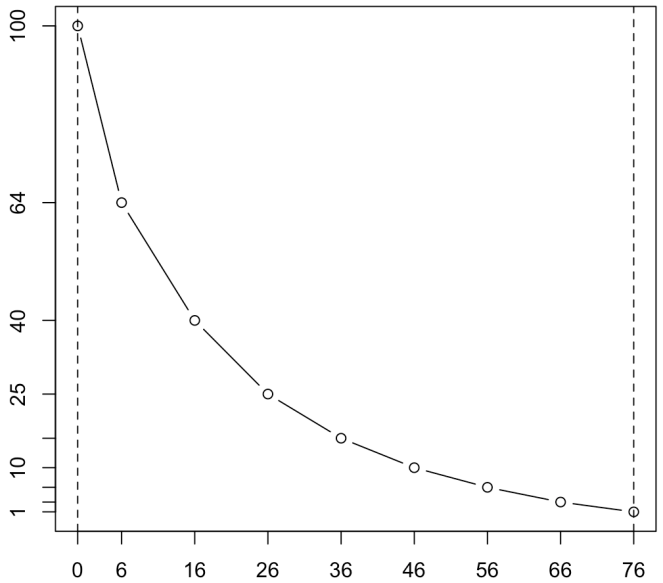
```
## [1] 18.17
```

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.

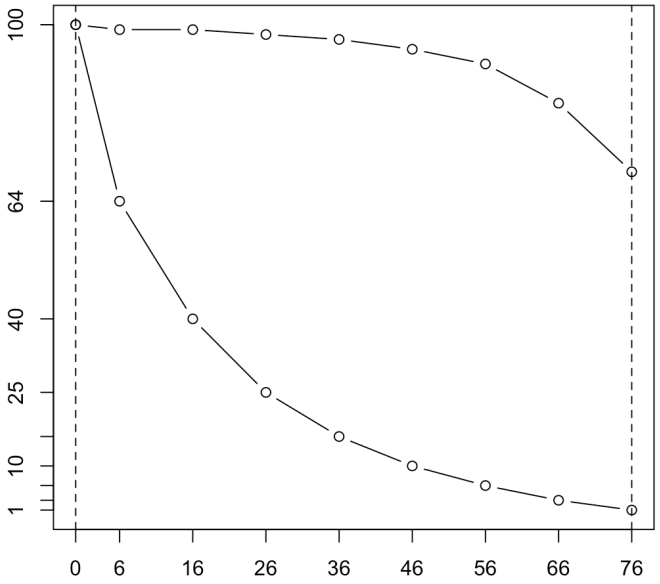
1. Draw Graunt's first with axes, lower and upper limits. Check what happens if you place `abline(...)` right after `plot(...)`.

```
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)
```



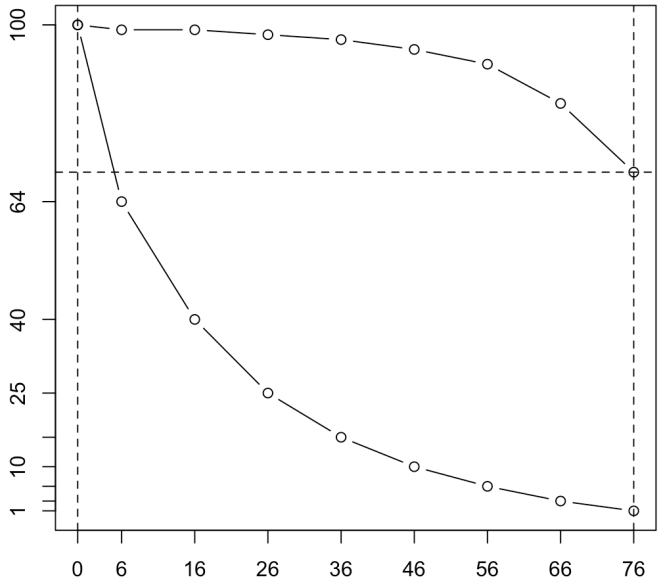
2. Add US 1993 survival function

```
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")
```



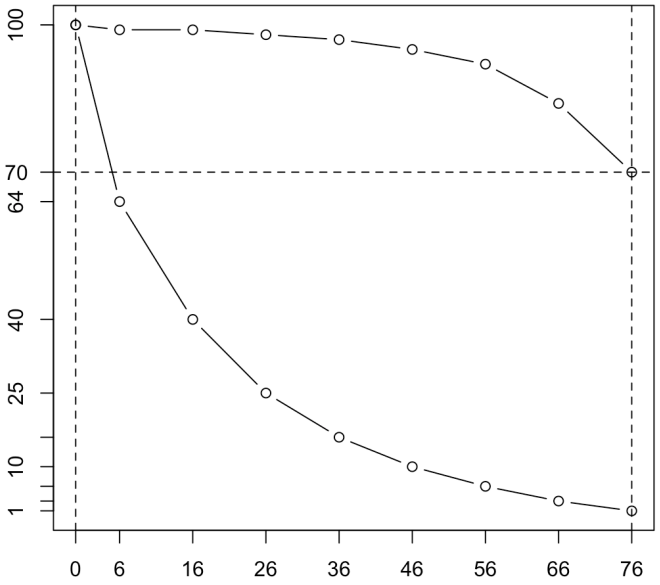
3. Actually, US 1993 life table is truncated at the age 76. Specify that point.

```
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)
```



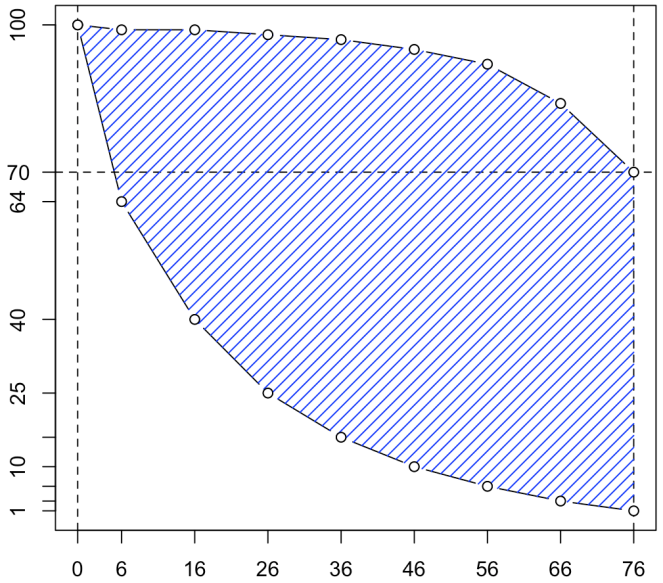
Setting coordinates for `polygon()`

```
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

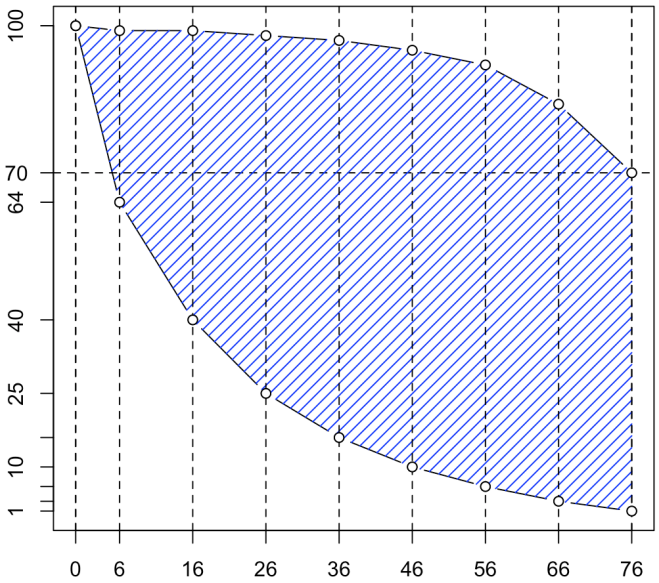
What is the effect of `border = NA`, the last line of code?

```
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")
```



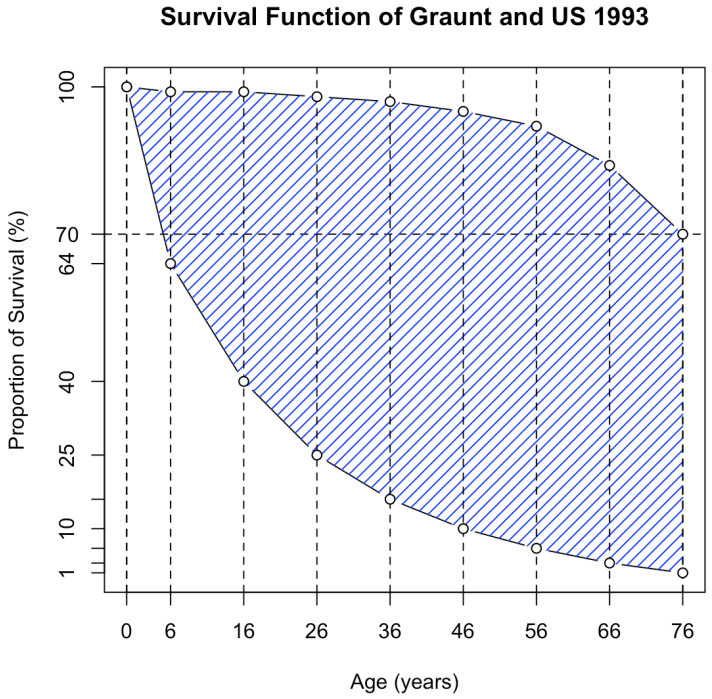
6. Grids

```
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")
```

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
```

Comparison with Halley's life table

Halley's life table

```
age <- 0:84
lx <- c(1238, 1000, 855, 798, 760, 732, 710, 692, 680, 670, 661, 653, 646, 640, 634,
628, 622, 616, 610, 604, 598, 592, 586, 579, 573, 567, 560, 553, 546, 539, 531, 523,
515, 507, 499, 490, 481, 472, 463, 454, 445, 436, 427, 417, 407, 397, 387, 377, 367,
357, 346, 335, 324, 313, 302, 292, 282, 272, 262, 252, 242, 232, 222, 212, 202, 192,
182, 172, 162, 152, 142, 131, 120, 109, 98, 88, 78, 68, 58, 50, 41, 34, 28, 23, 20)
length(lx)
```

```
## [1] 85
```

```
halley <- data.frame(age, lx)
halley$xPo <- round(halley$lx/lx[1]*100, digits = 1)
head(halley)
```

```
##   age   lx   xPo
## 1   0 1238 100.0
## 2   1 1000  80.8
## 3   2  855  69.1
## 4   3  798  64.5
## 5   4  760  61.4
## 6   5  732  59.1
```

```
tail(halley)
```

```
##    age lx xPo
## 80  79 50 4.0
## 81  80 41 3.3
## 82  81 34 2.7
## 83  82 28 2.3
## 84  83 23 1.9
## 85  84 20 1.6
```

```
halley.lx <- halley[-3]
halley <- halley[-2]
head(halley)
```

```
##   age   xPo
## 1   0 100.0
## 2   1  80.8
## 3   2  69.1
## 4   3  64.5
## 5   4  61.4
## 6   5  59.1
```

```
tail(halley)
```

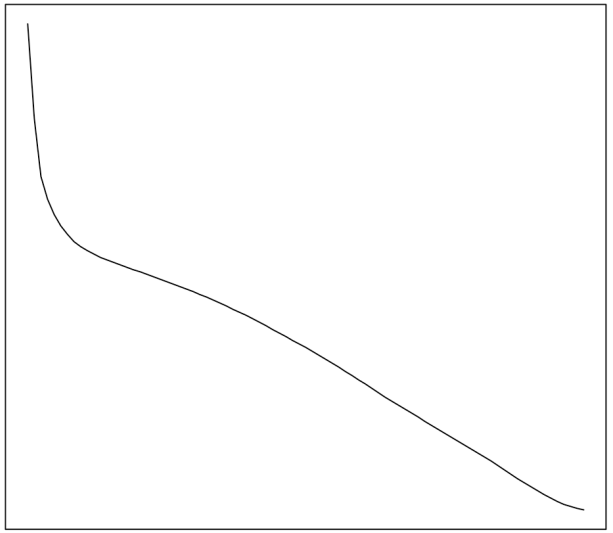
```
##    age xPo
## 80  79 4.0
## 81  80 3.3
## 82  81 2.7
## 83  82 2.3
## 84  83 1.9
## 85  84 1.6
```

R base graphics

To make the comparison easy, plot the points at the same age group of Graunt's, 0, 6, 16, 26, 36, 46, 56, 66, 76. Step by step approach

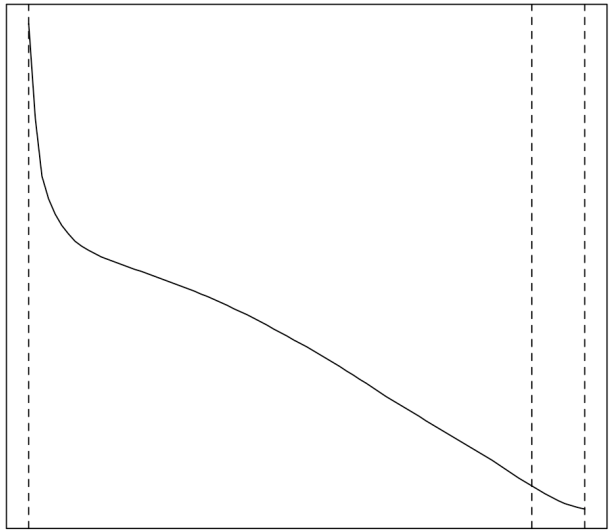
1. Halley's survival function first

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
```



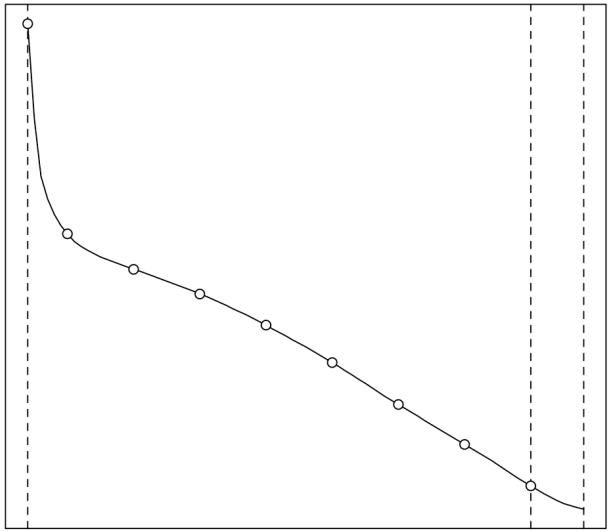
2. Denote the age at 0, 76, and 84 by vertical dotted lines

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
```



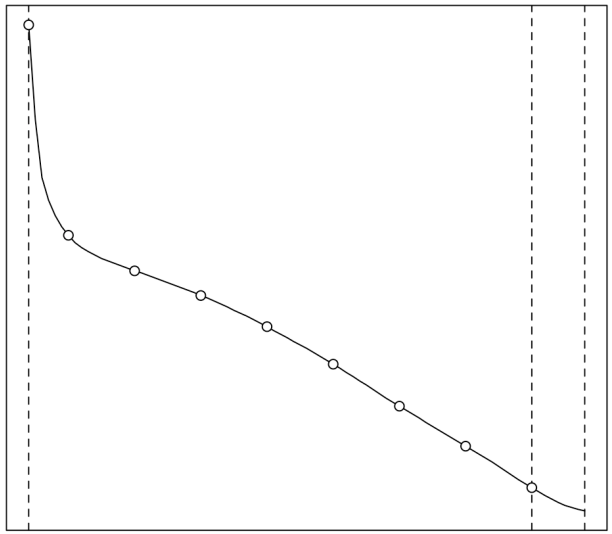
3. Mark the points at 0, 6, 16, 26, 36, 46, 56, 66, 76 on Halley's survival function.

```
age.graunt <- age %in% graunt$x
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(xPo[age.graunt] ~ age[age.graunt], data = halley, pch = 21, col = "black", bg
= "white")
```



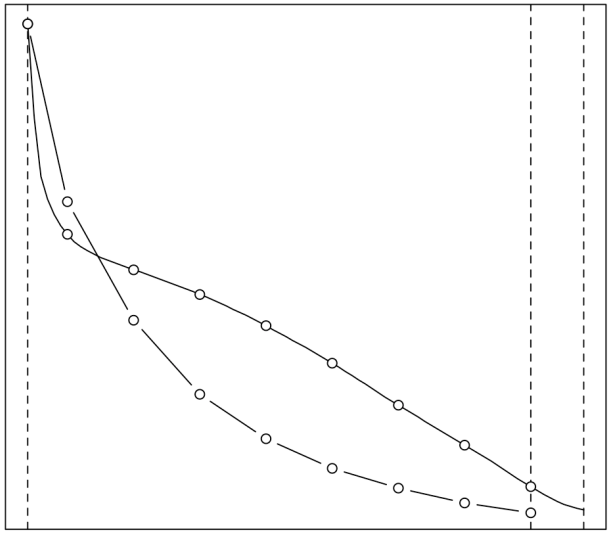
Using `subset()`

```
halley.graunt <- subset(halley, age.graunt)
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
```



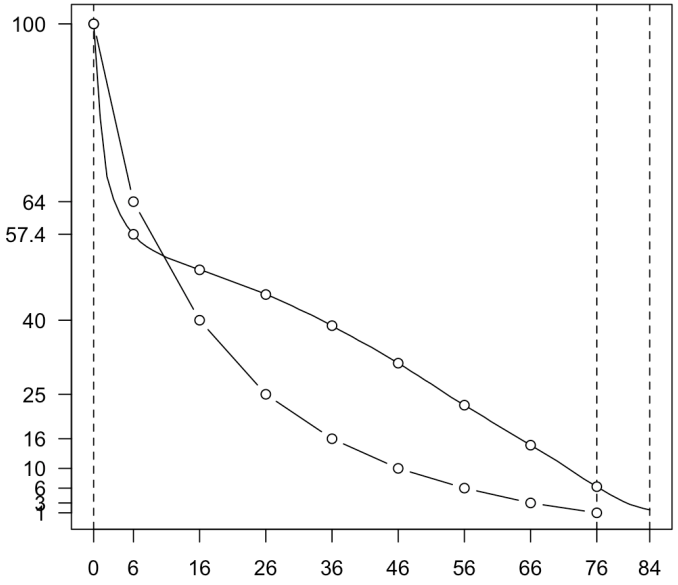
4. Add Graunt's survival function

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
```



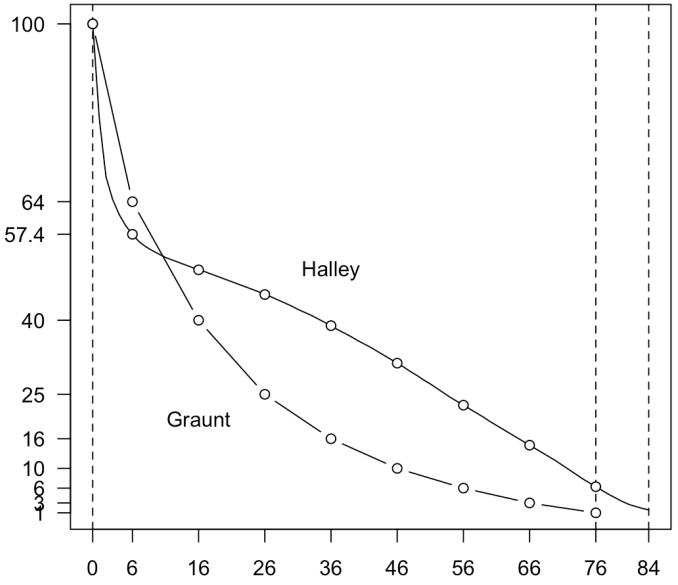
5. x-axis label and y-axis label with `las = 1`. Add Halley's proprtion of survival at `age = 6`.

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = graunt$xPo.g, labels = graunt$xPo.g, las = 1)
xPo.halley.age.6 <- halley$xPo[age == 6]
axis(side = 2, at = xPo.halley.age.6, labels = xPo.halley.age.6, las = 1)
```



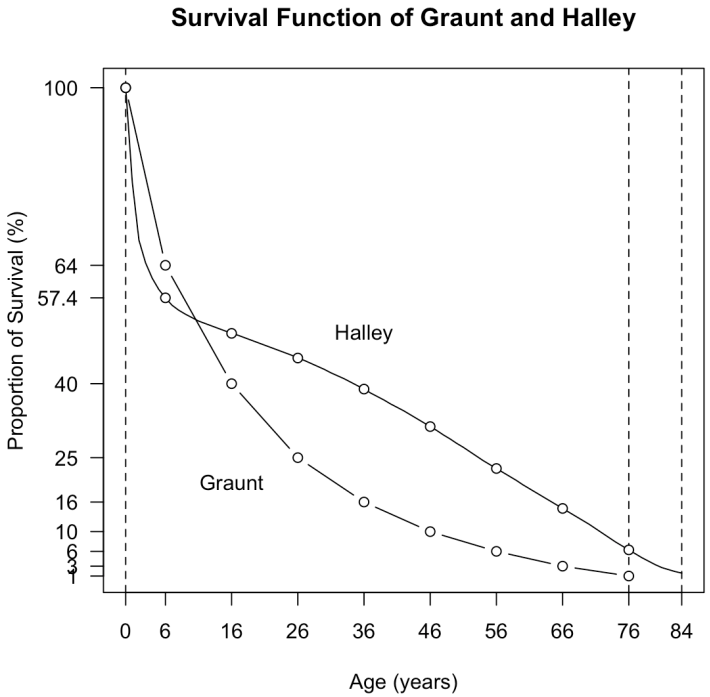
6. Specify the developers at proper coordinates with `text()`

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = graunt$xPo.g, labels = graunt$xPo.g, las = 1)
axis(side = 2, at = xPo.halley.age.6, labels = xPo.halley.age.6, las = 1)
text(x = c(16, 36), y = c(20, 50), label = c("Graunt", "Halley"))
```



7. Main title, x-axis label, and y-axis label

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = graunt$xPo.g, labels = graunt$xPo.g, las = 1)
axis(side = 2, at = xPo.halley.age.6, labels = xPo.halley.age.6, las = 1)
text(x = c(16, 36), y = c(20, 50), label = c("Graunt", "Halley"))
main.title.2 <- "Survival Function of Graunt and Halley"
title(main = main.title.2, xlab = x.lab, ylab = y.lab)
```



Polygon

Setting the coordinates for `polygon()`. The intersection is found at `x = 10.8, y = 52.8` with `locator(1)` and couple of trial and errors.

- Upper region

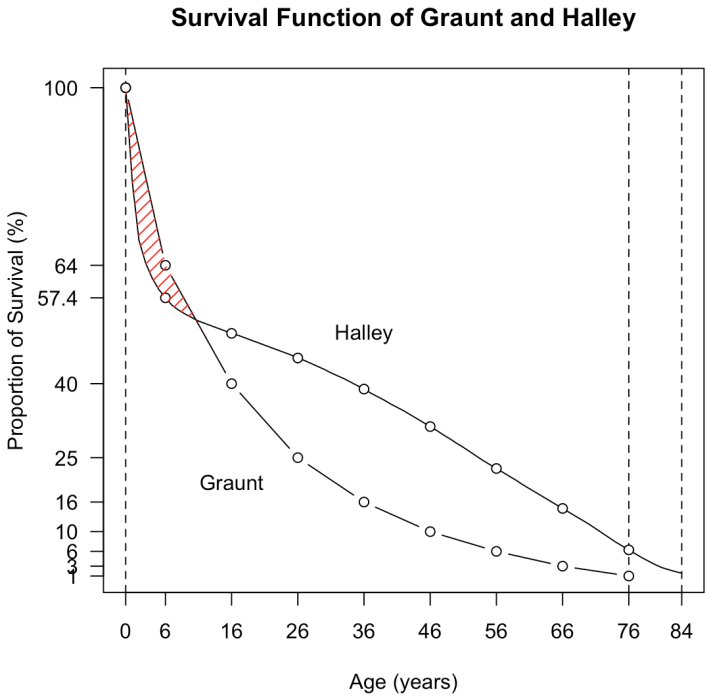
```
poly.1.x <- c(graunt$x[1:2], 10.8, halley$age[11:1])
poly.1.y <- c(graunt$xPo.g[1:2], 52.8, halley$xPo[11:1])
poly.upper <- data.frame(x = poly.1.x, y = poly.1.y)
```

- Lower region

```
poly.2.x <- c(10.8, halley$age[12:85], graunt$x[9:3])
poly.2.y <- c(52.8, halley$xPo[12:85], graunt$xPo.g[9:3])
poly.lower <- data.frame(x = poly.2.x, y = poly.2.y)
```

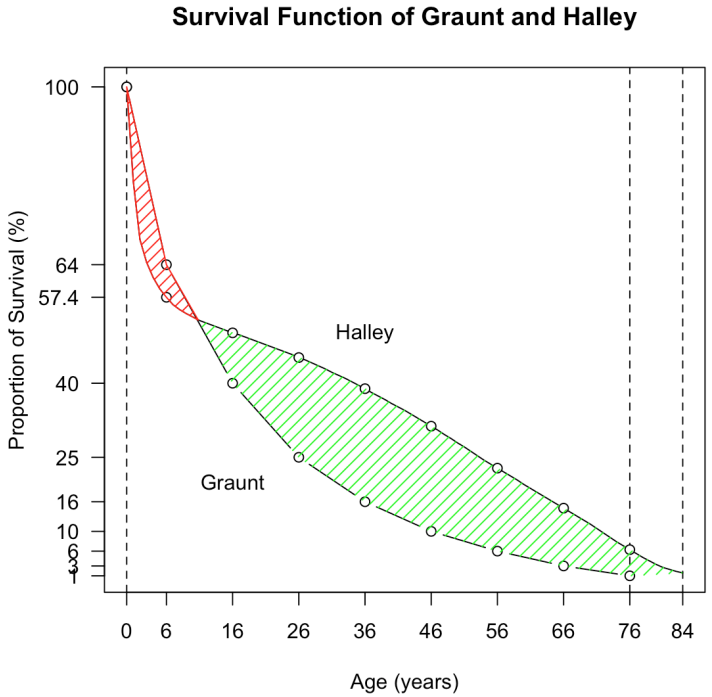
8. Shading upper region first

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = graunt$xPo.g, labels = graunt$xPo.g, las = 1)
axis(side = 2, at = xPo.halley.age.6, labels = xPo.halley.age.6, las = 1)
text(x = c(16, 36), y = c(20, 50), label = c("Graunt", "Halley"))
title(main = main.title.2, xlab = x.lab, ylab = y.lab)
polygon(poly.upper, angle = 45, density = 15, col = "red", border = NA)
```



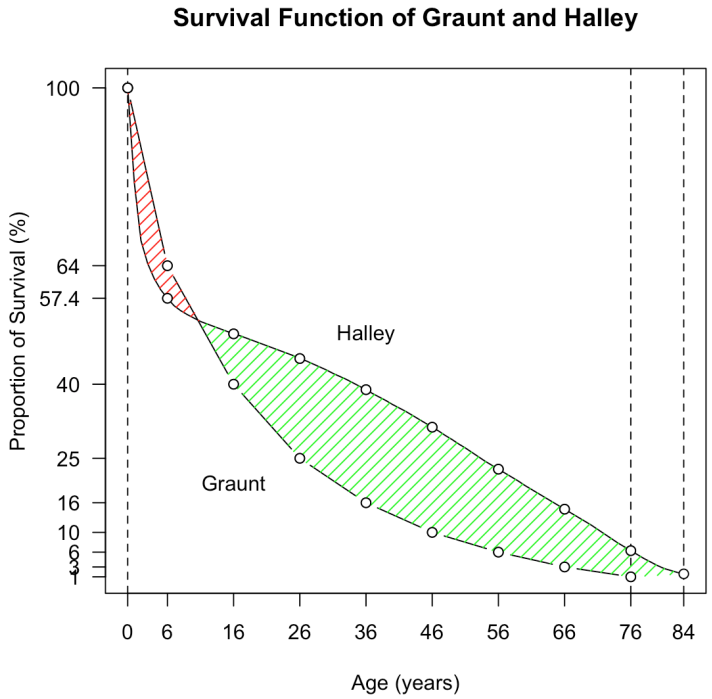
9. Shading lower region next

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = graunt$xPo.g, labels = graunt$xPo.g, las = 1)
axis(side = 2, at = xPo.halley.age.6, labels = xPo.halley.age.6, las = 1)
text(x = c(16, 36), y = c(20, 50), label = c("Graunt", "Halley"))
title(main = main.title.2, xlab = x.lab, ylab = y.lab)
polygon(poly.upper, angle = 45, density = 15, col = "red")
polygon(poly.lower, angle = 45, density = 15, col = "green", border = NA)
```



10. Fill the points. Extra points at the 84.

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = graunt$xPo.g, labels = graunt$xPo.g, las = 1)
axis(side = 2, at = xPo.halley.age.6, labels = xPo.halley.age.6, las = 1)
text(x = c(16, 36), y = c(20, 50), label = c("Graunt", "Halley"))
title(main = main.title.2, xlab = x.lab, ylab = y.lab)
polygon(poly.upper, angle = 45, density = 15, col = "red", border = NA)
polygon(poly.lower, angle = 45, density = 15, col = "green", border = NA)
points(graunt, pch = 21, col = "black", bg = "white")
points(halley.graunt, pch = 21, col = "black", bg = "white")
points(x = 84, y = halley$xPo[85], pch = 21, col = "black", bg = "white")
```




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

Life expectancy

Compute the difference of life expectancies

```
(life.exp.halley <- area.R(halley$age, halley$xPo)/100)
```

```
## [1] 27.872
```

```
(life.exp.graunt <- area.R(graunt$x, graunt$xPo.g)/100)
```

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

Graunt, Halley, and US 1993

Polygon with R Base Plot

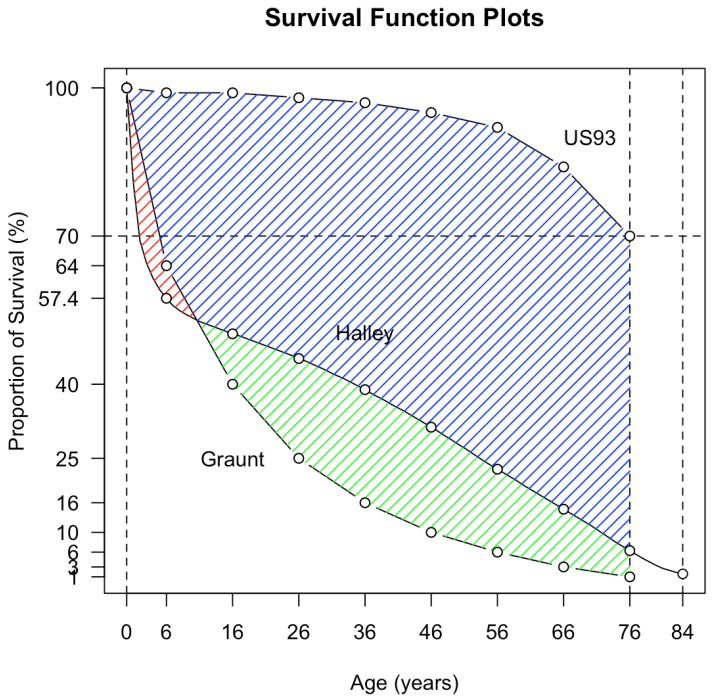
Coordinates

In order to make the graphs truncated at the age 76, restrict the age of Halley up to 76.

```
graunt.2 <- graunt
halley.2 <- halley
us93.2 <- us93
names(graunt.2) <- c("x", "Graunt")
names(halley.2) <- c("x", "Halley")
names(us93.2) <- c("x", "US93")
poly.lower.76 <- subset(poly.lower, poly.lower$x <= 76)
poly.3.x <- c(us93.2$x, halley.2$x[85:12], 10.8, graunt.2$x[2:1])
poly.3.y <- c(us93.2$US93, halley.2$Halley[85:12], 52.8, graunt.2$Graunt[2:1])
poly.us <- data.frame(x = poly.3.x, y = poly.3.y)
poly.us.76 <- subset(poly.us, poly.us$x <= 76)
```

Straight to Polygon

```
plot(halley, ann = FALSE, xaxt = "n", yaxt = "n", type = "l")
abline(v = c(0, 76, 84), lty = 2)
points(halley.graunt, pch = 21, col = "black", bg = "white")
lines(graunt, type = "b", pch = 21, col = "black", bg = "white")
lines(us93, type = "b", pch = 21, col = "black", bg = "white")
axis(side = 1, at = c(graunt$x, 84), labels = c(graunt$x, 84))
axis(side = 2, at = c(graunt$xPo.g, xPo.halley.age.6), labels = c(graunt$xPo.g, xPo.h
alley.age.6), las = 1)
abline(h = 70, lty = 2)
axis(side = 2, at = 70, labels = 70, las = 1)
main.title.3 <- "Survival Function Plots"
title(main = main.title.3, xlab = x.lab, ylab = y.lab)
polygon(poly.upper, angle = 45, density = 15, col = "red", border = NA)
polygon(poly.lower.76, angle = 45, density = 15, col = "green", border = NA)
polygon(poly.us.76, angle = 45, density = 15, col = "blue", border = NA)
points(graunt, pch = 21, col = "black", bg = "white")
points(halley.graunt, pch = 21, col = "black", bg = "white")
points(us93.2, pch = 21, col = "black", bg = "white")
points(x = 84, y = halley$xPo[85], pch = 21, col = "black", bg = "white")
text(x = c(16, 36, 70), y = c(25, 50, 90), label = c("Graunt", "Halley", "US93"))
```




```
# dev.copy(device = png, file = "../pics/graunt_halley_us93_poly.png")
```

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`

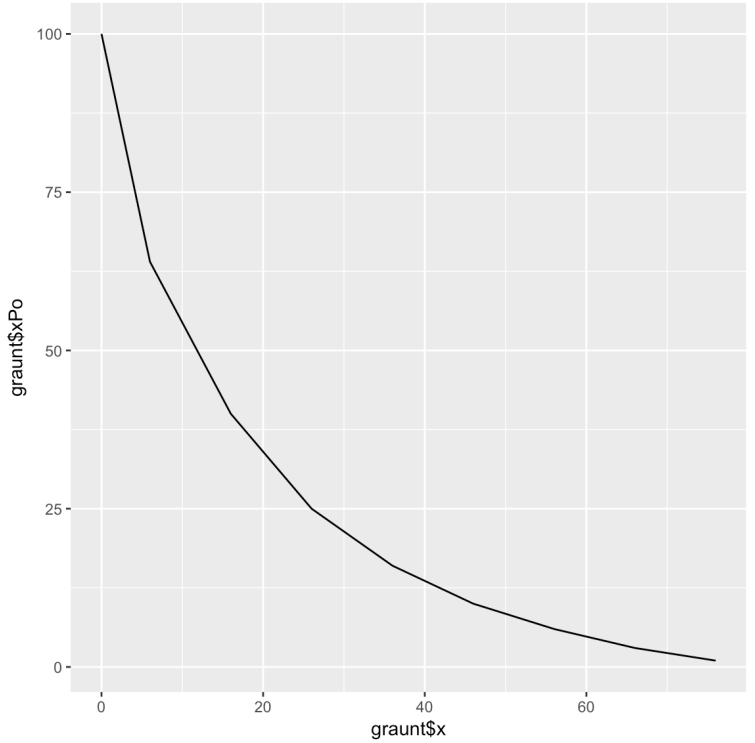
```
levels(graunt.us.melt$times) <- c("Graunt", "US1993")
str(graunt.us.melt)
```

```
## 'data.frame':    18 obs. of  3 variables:
##  $ x    : num  0 6 16 26 36 46 56 66 76 0 ...
##  $ times: Factor w/ 2 levels "Graunt","US1993": 1 1 1 1 1 1 1 1 1 2 ...
##  $ xPo  : num  100 64 40 25 16 10 6 3 1 100 ...
```

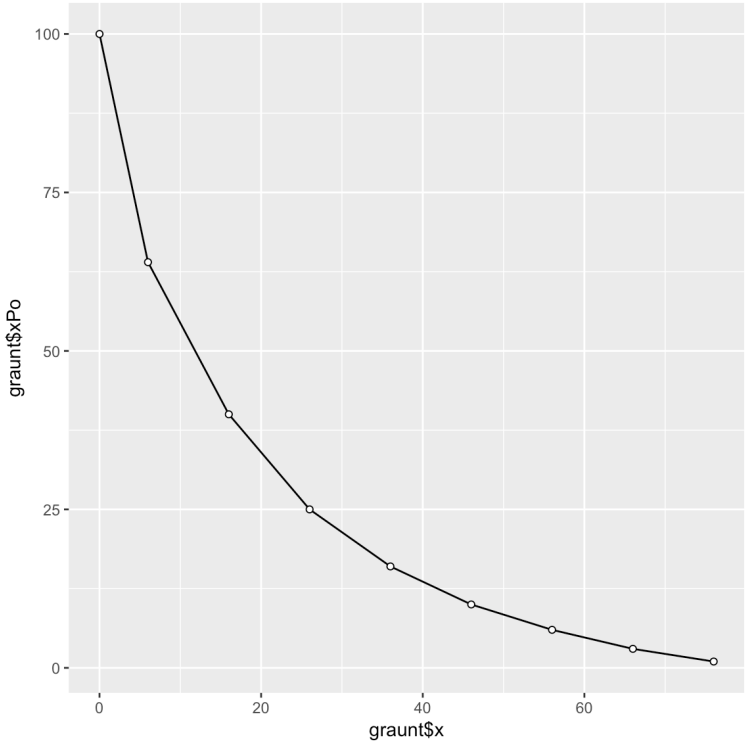
Graunt

Structure of ggplot

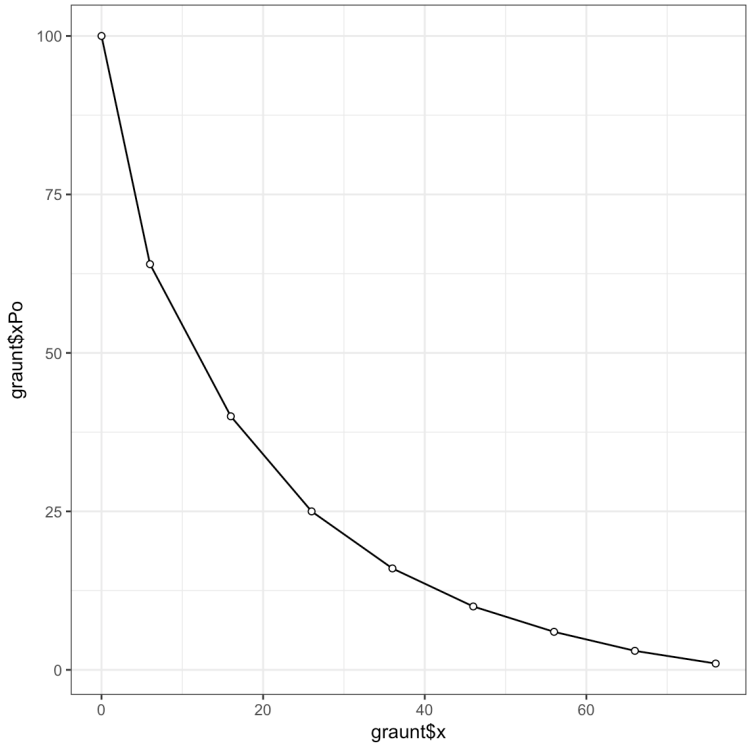
```
(g1 <- ggplot() +
  geom_line(data = graunt, aes(x = graunt$x, y = graunt$xPo)))
```



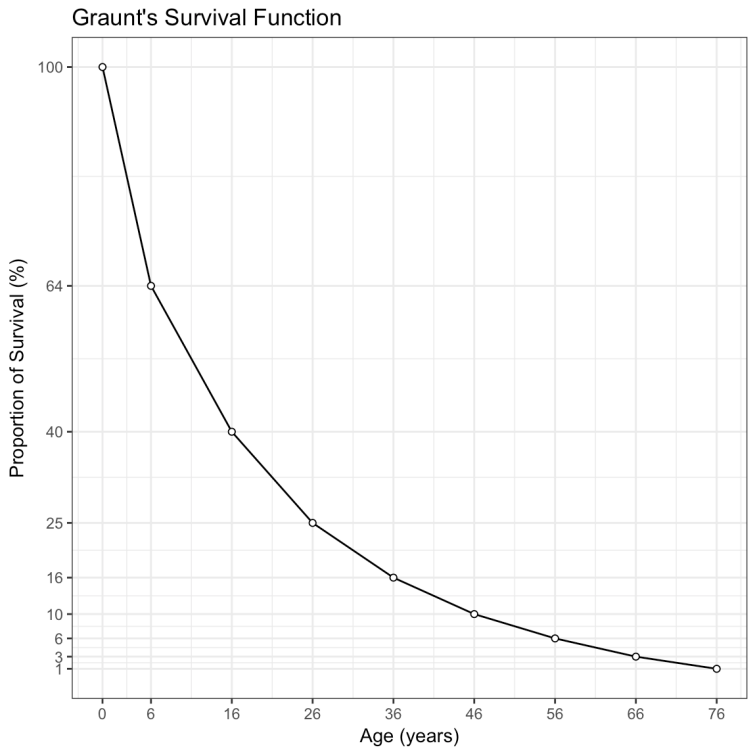
```
(g2 <- g1 +
  geom_point(data = graunt, aes(x = graunt$x, y = graunt$xPo), shape = 21, fill = "wh
ite"))
```



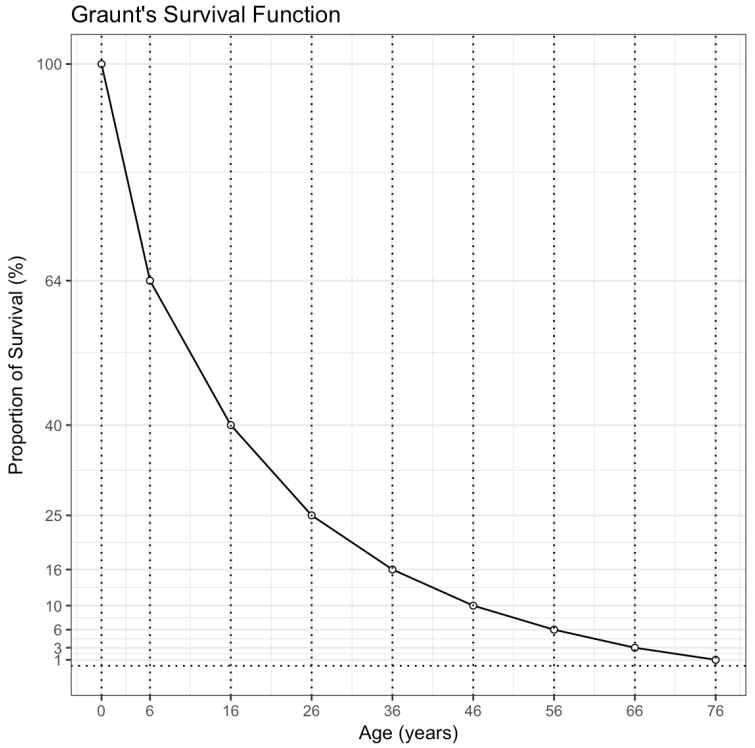

```
(g3 <- g2 +
  theme_bw())
```



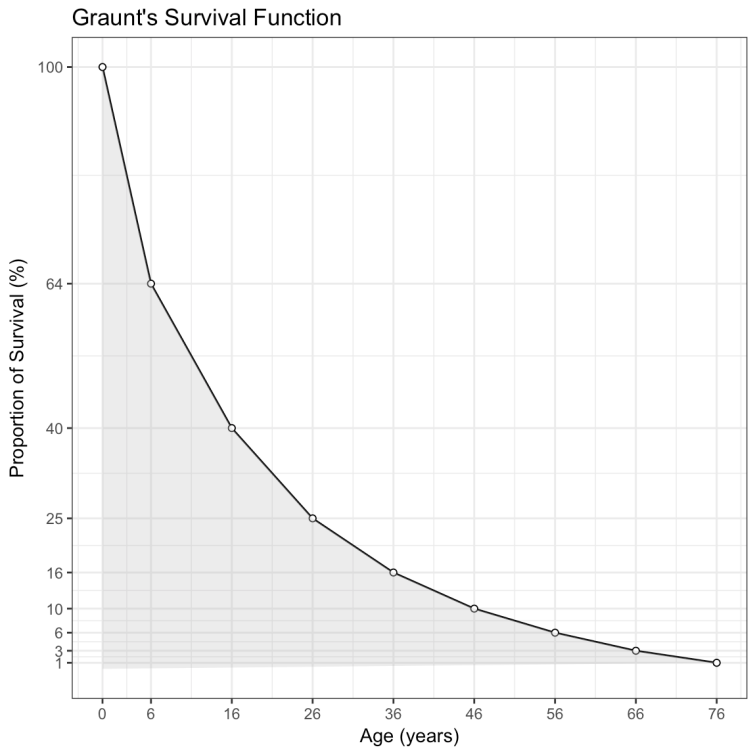
```
(g4 <- g3 +
   xlab(x.lab) +
   ylab(y.lab) +
   ggtitle(main.title) +
   scale_x_continuous(breaks = graunt$x) +
   scale_y_continuous(breaks = graunt$xPo.g))
```




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



```
(pg4 <- g4 +
  geom_polygon(data = graunt.poly, aes(x = x, y = y), alpha = 0.3, fill = "grey"))
```




```
# ggsave("../pics/graunt_poly_ggplot.png", pg4)
```

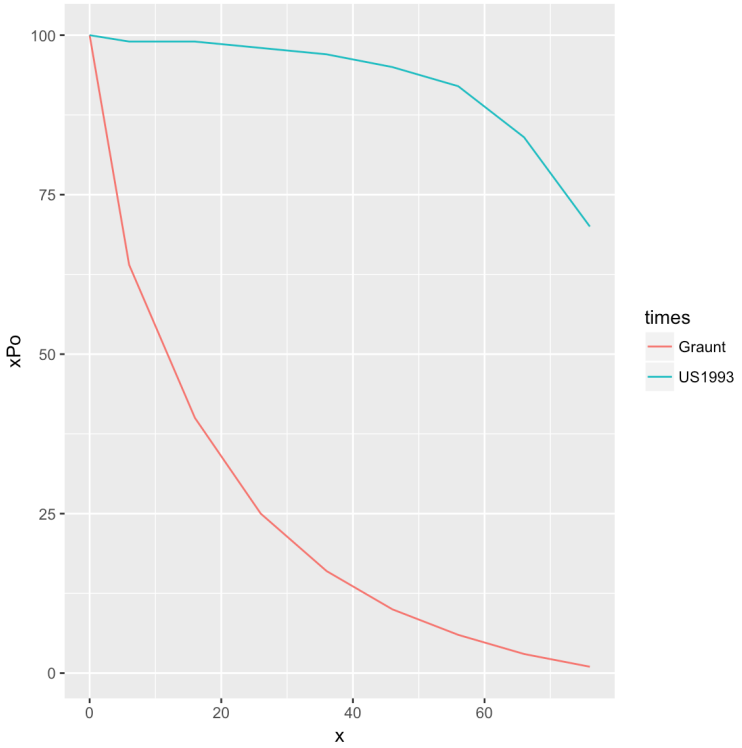
Graunt and US 1993

Points and Lines

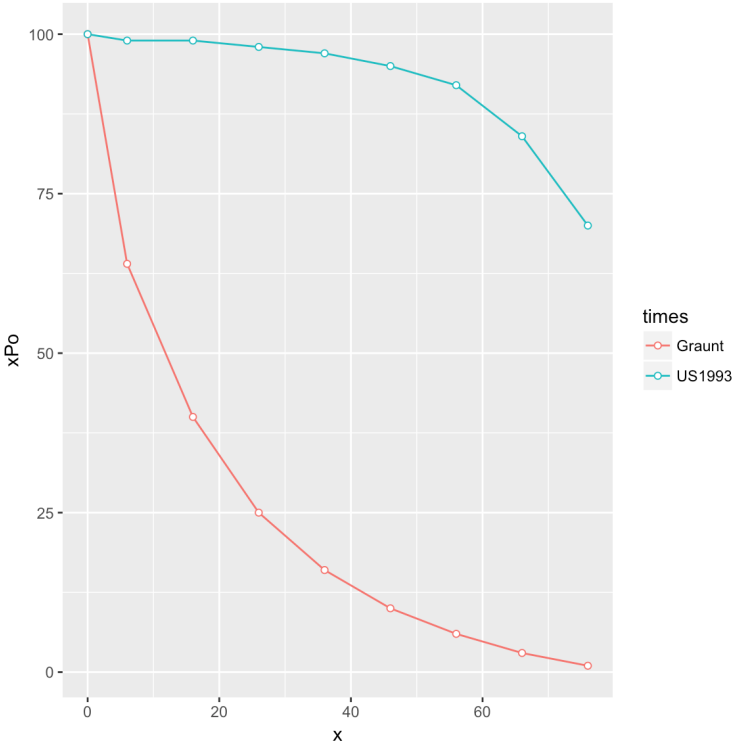
Step by step approach to understand the grammar of ggplot

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

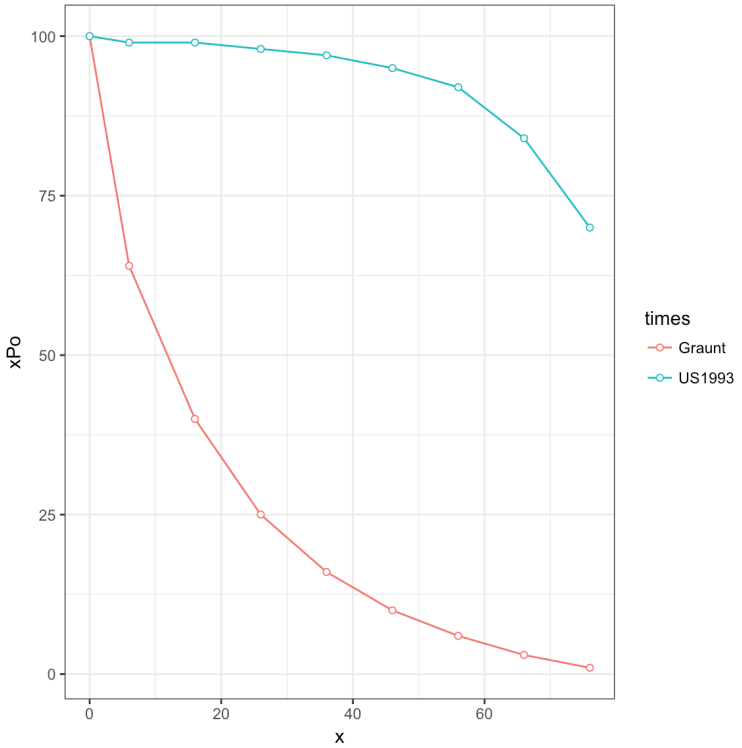
```
(gu1 <- ggplot() +
  geom_line(data = graunt.us.melt, aes(x = x, y = xPo, colour = times)))
```



```
(gu2 <- gu1 +
  geom_point(data = graunt.us.melt, aes(x = x, y = xPo, colour = times), shape = 21,
fill = "white"))
```



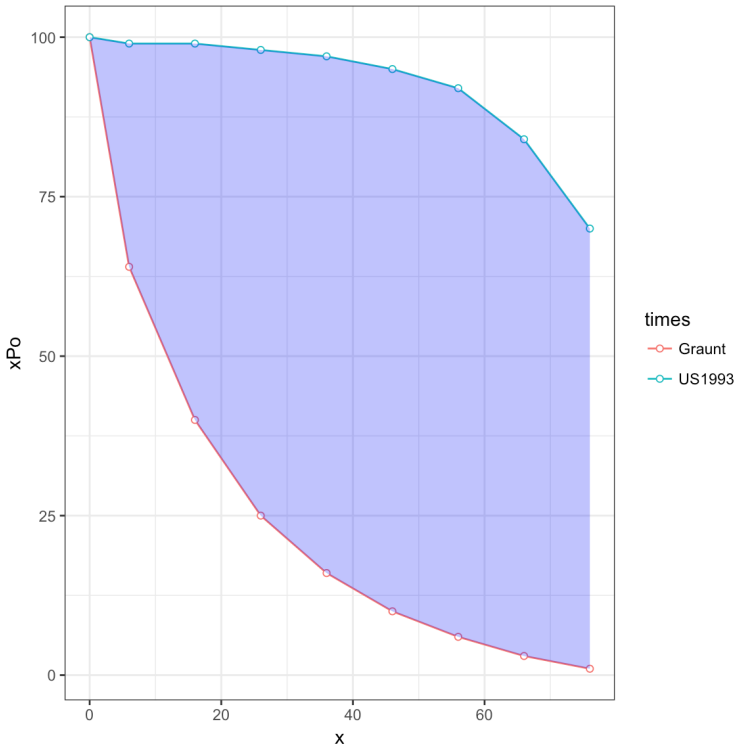

```
(gu3 <- gu2 +
  theme_bw())
```



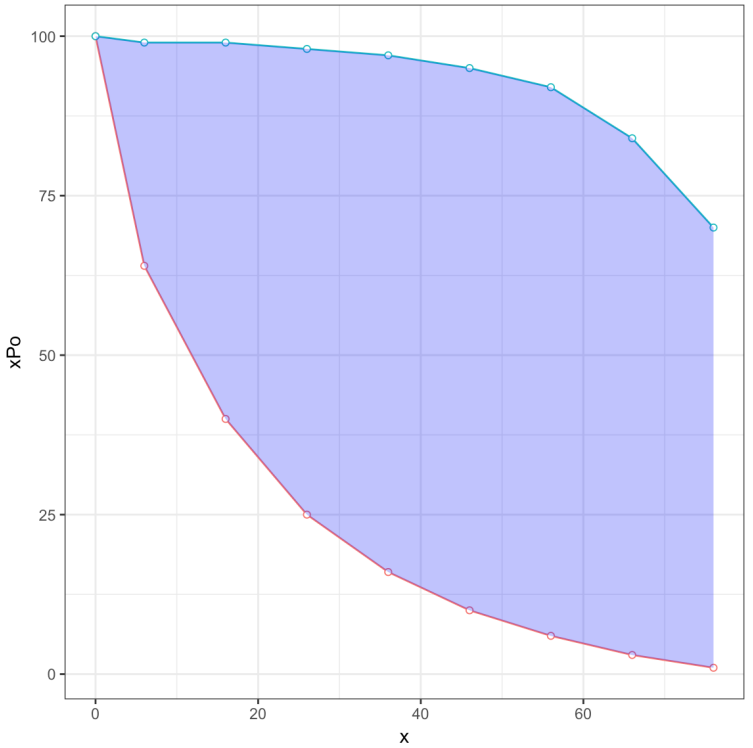
Polygon

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, aes(x = x, y = y), alpha = 0.3, fill = "blue"))
```




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

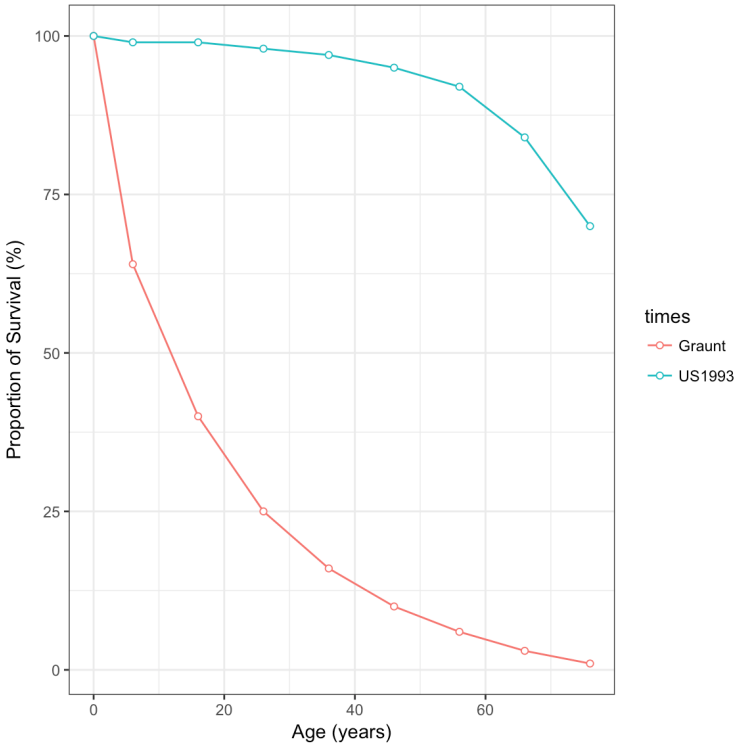


Change default annotations

Points and Lines

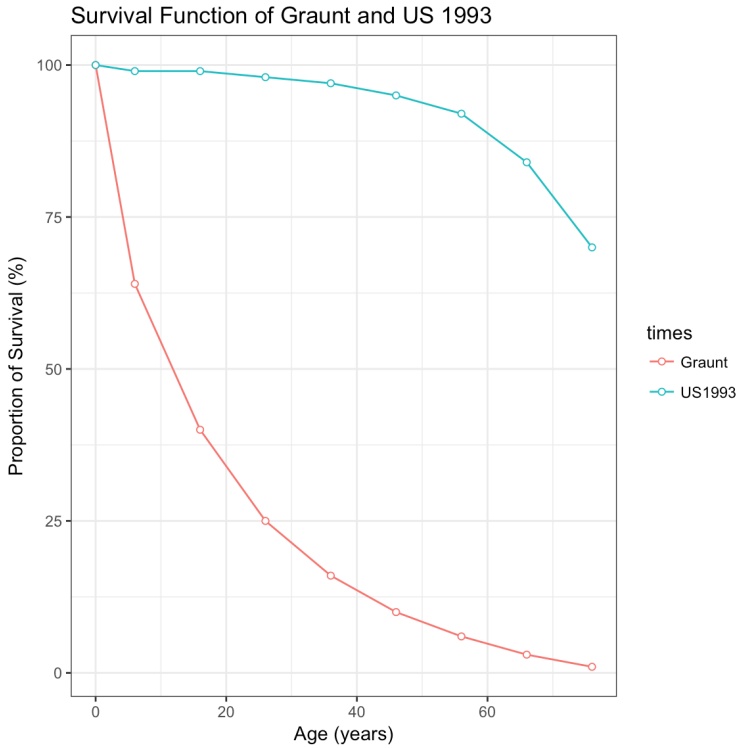
1. Change the x-axis and y-axis labels

```
(gu4 <- gu3 +
   xlab(x.lab) +
   ylab(y.lab))
```



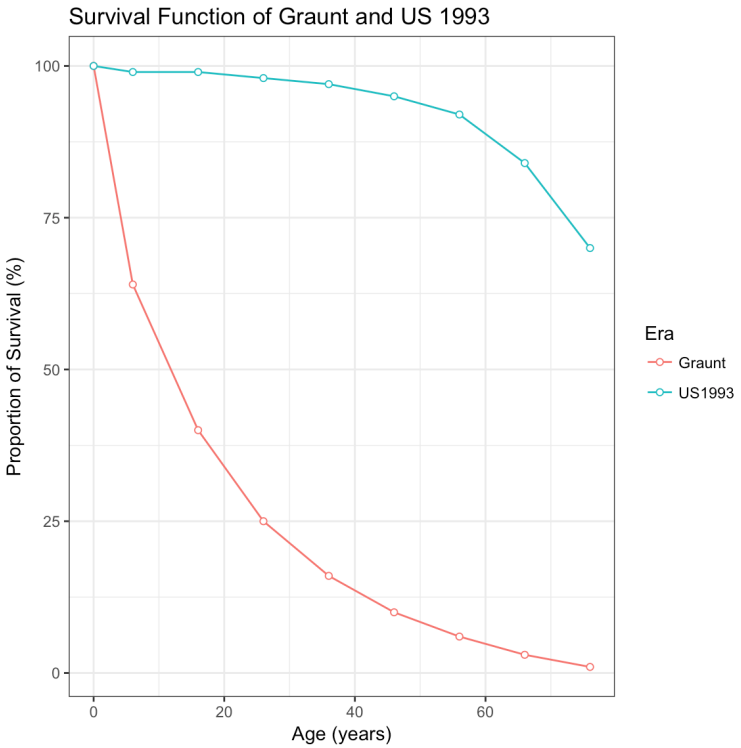
2. Add main title

```
(gu4 <- gu3 +
   xlab(x.lab) +
   ylab(y.lab) +
   ggtitle(main.title.g.us))
```



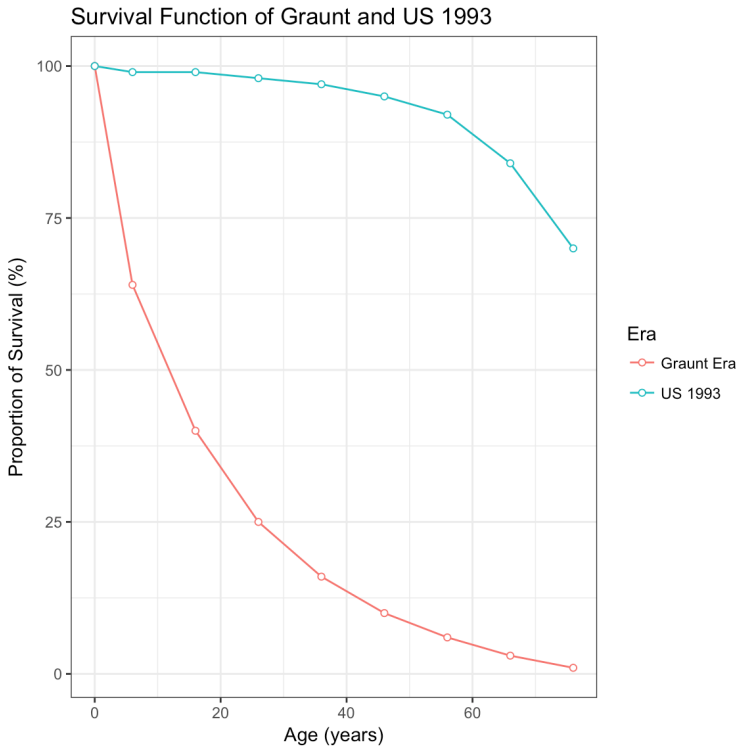
3. Change legend title

```
(gu4 <- gu3 +
   xlab(x.lab) +
   ylab(y.lab) +
   ggtitle(main.title.g.us) +
   labs(colour = "Era"))
```



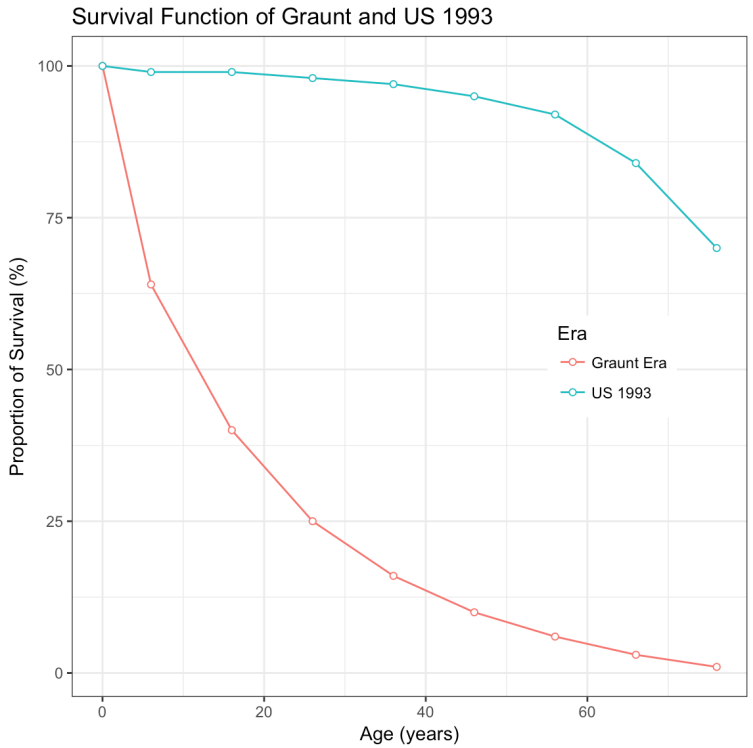
4. Change legends.

```
(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")))
```



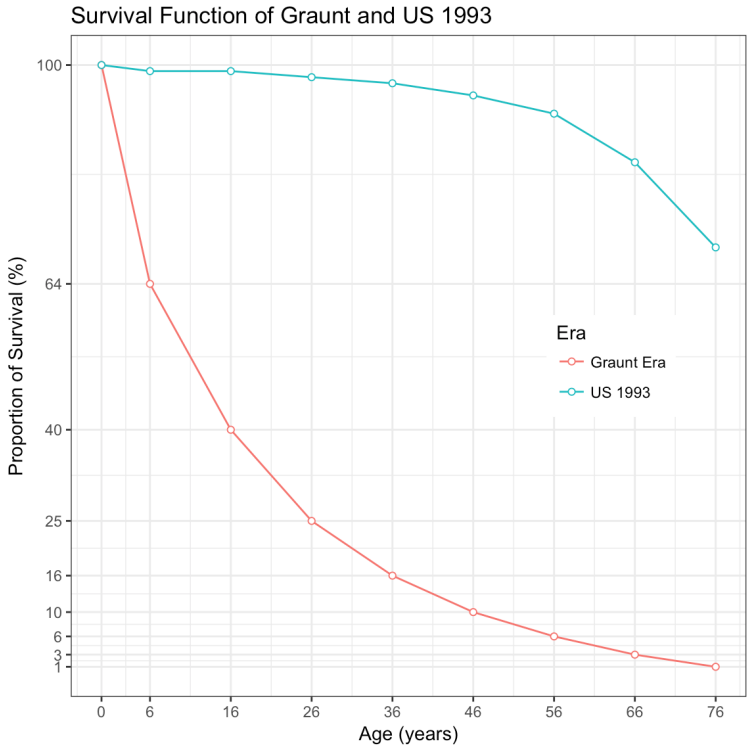
5. Place legends inside the plot

```
(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))
```



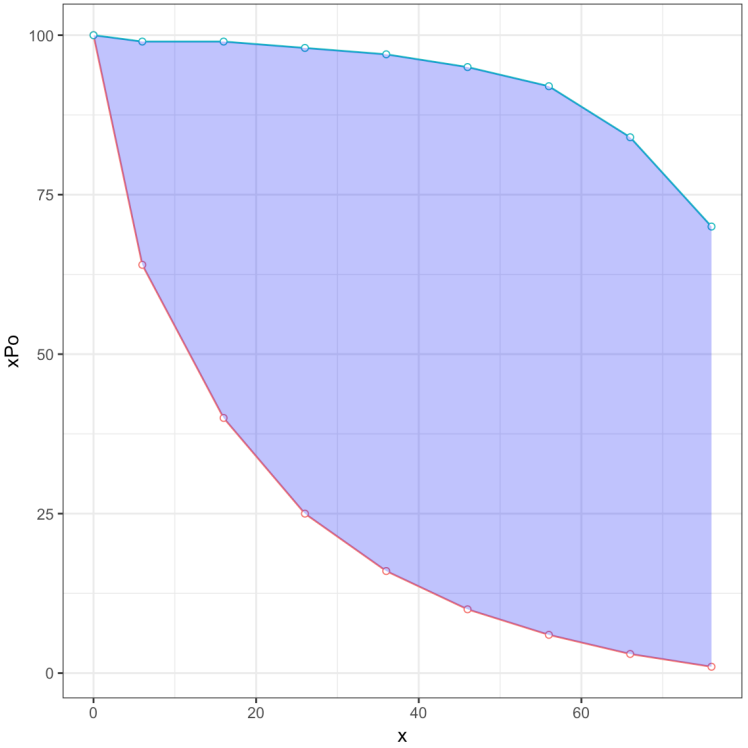

```
# ggsave("../pics/graunt_us_ggplot.png", gu6)
```

Polygon

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

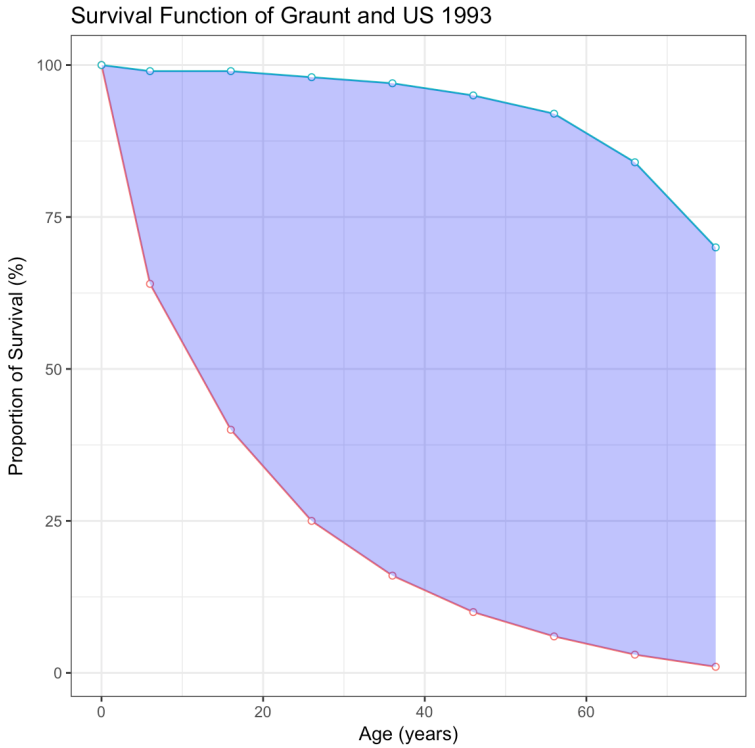
1. Start with `gup4`

```
gup4
```



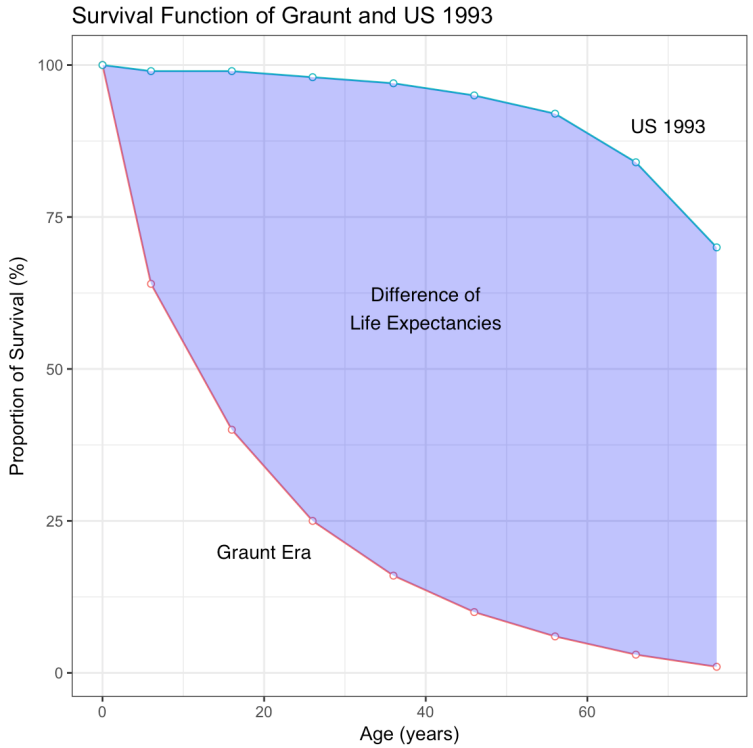
2. Main title, x-axis and y-axis labels

```
(gup5 <- gup4 +
   xlab(x.lab) +
   ylab(y.lab) +
   ggtitle(main.title.g.us))
```



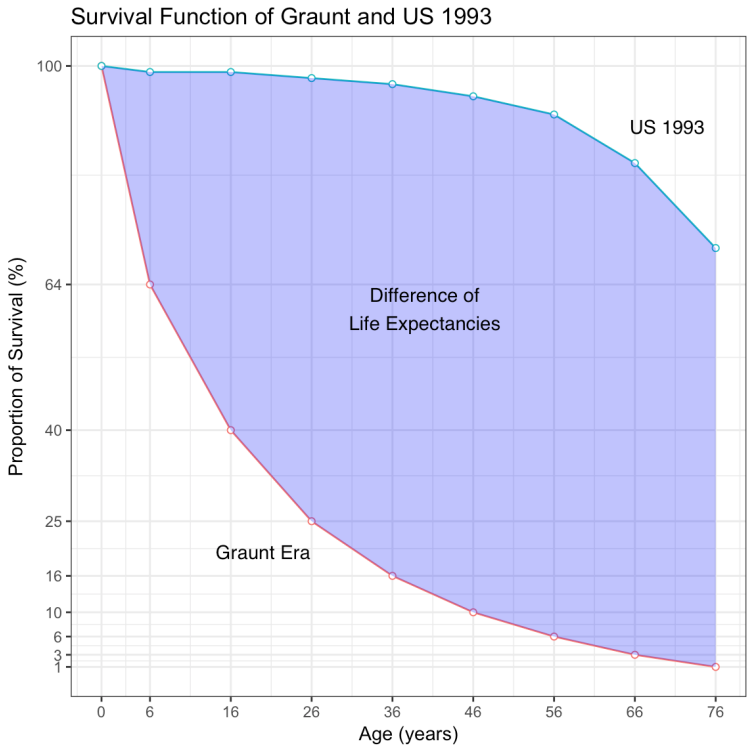
3. `"Graunt Era"`, `"US 1993"`, `"Difference of Life Expectancies"` at proper positions

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



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)
```

Graunt and Halley

Data Reshaping

Since the observed ages are different, we need final structure of the data frame to be melted. So, create copies of `graunt` and `halley` and extract parts of what we need and give feasible names.

```
graunt.halley.melt <- melt(list(graunt.2, halley.2), id.vars = "x", value.name = "xP
o", variable.name = "Who")
str(graunt.halley.melt)
```

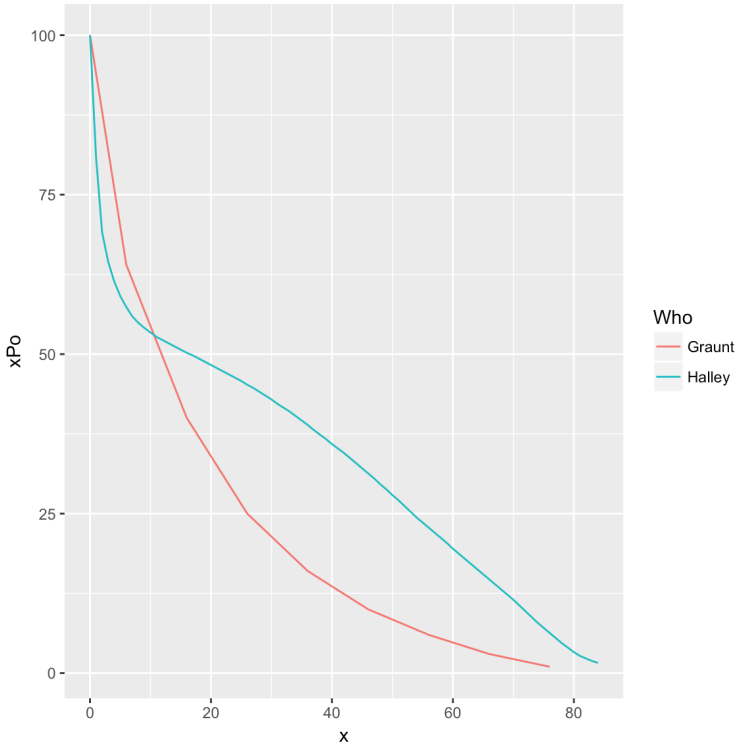
```
## 'data.frame':    94 obs. of  4 variables:
##  $ x  : num  0 6 16 26 36 46 56 66 76 0 ...
##  $ Who: Factor w/ 2 levels "Graunt","Halley": 1 1 1 1 1 1 1 1 1 2 ...
##  $ xPo: num  100 64 40 25 16 10 6 3 1 100 ...
##  $ L1 : int  1 1 1 1 1 1 1 1 1 2 ...
```

```
graunt.halley.melt <- graunt.halley.melt[-4]
(graunt.halley.melt.g <- subset(graunt.halley.melt, graunt.halley.melt$x %in%
graunt$x))
```

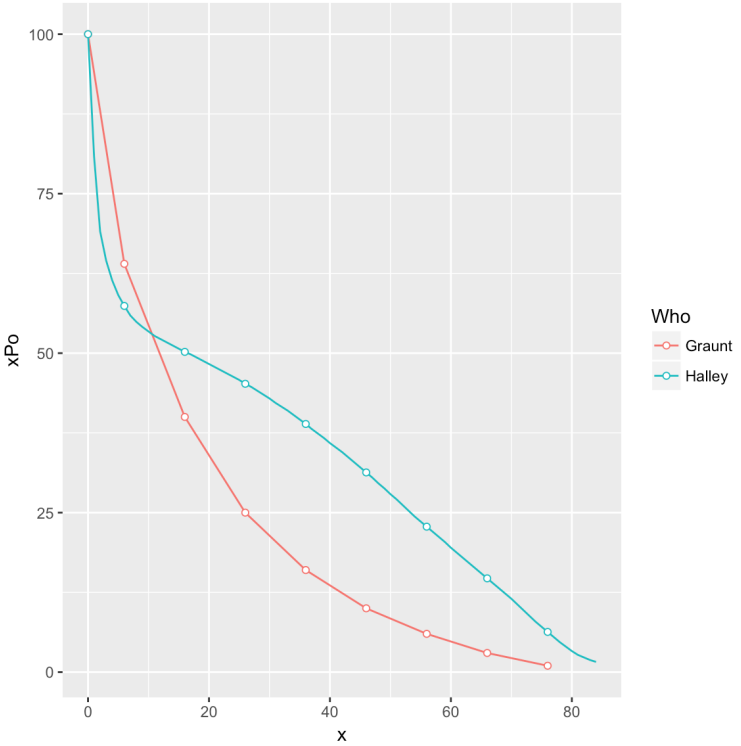
```
##     x    Who   xPo
## 1   0 Graunt 100.0
## 2   6 Graunt  64.0
## 3  16 Graunt  40.0
## 4  26 Graunt  25.0
## 5  36 Graunt  16.0
## 6  46 Graunt  10.0
## 7  56 Graunt   6.0
## 8  66 Graunt   3.0
## 9  76 Graunt   1.0
## 10  0 Halley 100.0
## 16  6 Halley  57.4
## 26 16 Halley  50.2
## 36 26 Halley  45.2
## 46 36 Halley  38.9
## 56 46 Halley  31.3
## 66 56 Halley  22.8
## 76 66 Halley  14.7
## 86 76 Halley   6.3
```

Survival Function, Step by Step

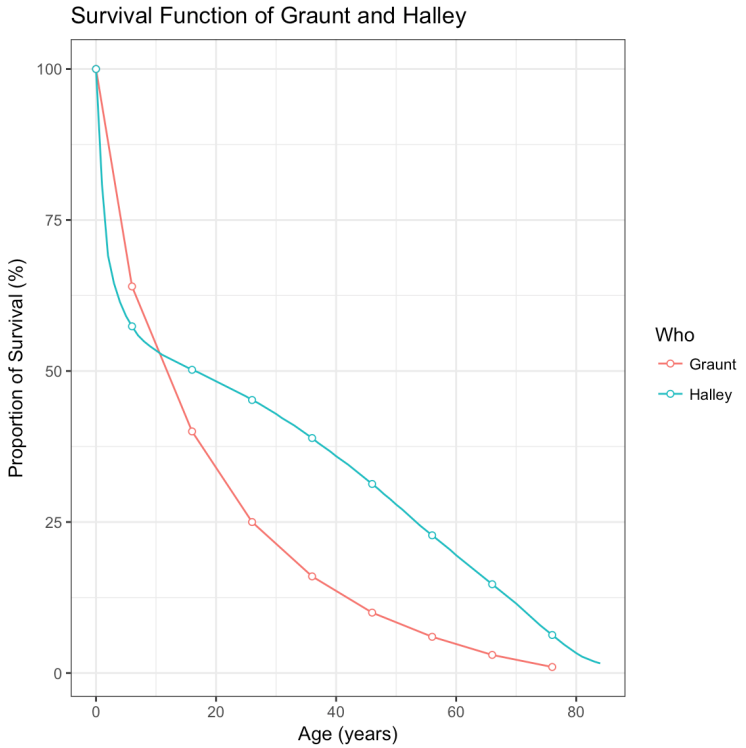
```
(gh1 <- ggplot() +
  geom_line(data = graunt.halley.melt, aes(x = x, y = xPo, colour = Who)))
```



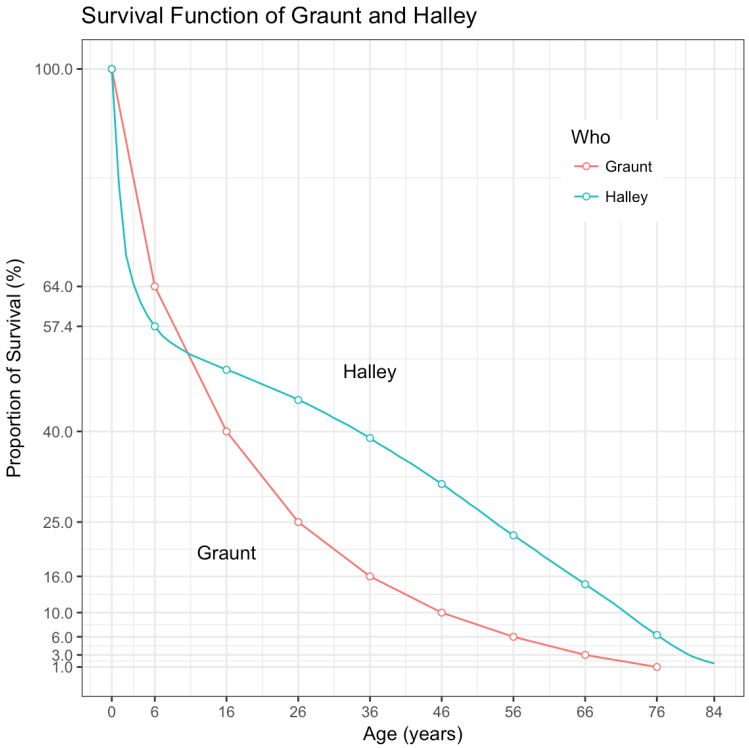
```
(gh2 <- gh1 +
  geom_point(data = graunt.halley.melt.g, aes(x = x, y = xPo, colour = Who), shape =
21, fill = "white"))
```




```
(gh3 <- gh2 +
  theme_bw() +
  xlab(x.lab) +
  ylab(y.lab) +
  ggtitle(main.title.2))
```



```
(gh4 <- gh3 +
  theme(legend.position = c(0.8, 0.8)) +
  annotate("text", x = c(16, 36), y = c(20, 50), label = c("Graunt", "Halley")) +
  scale_x_continuous(breaks = c(graunt$x, 84)) +
  scale_y_continuous(breaks = c(graunt$xPo.g, xPo.halley.age.6)))
```

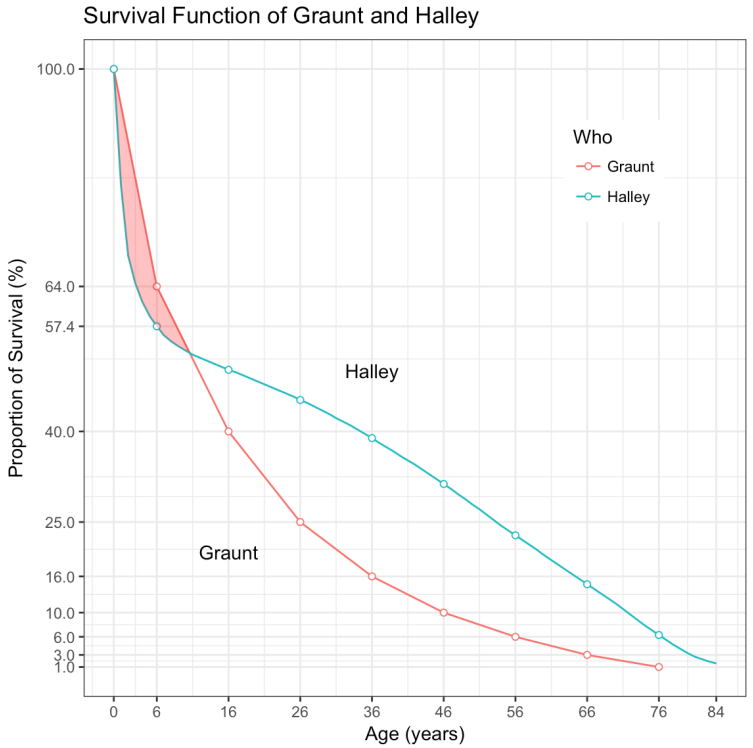



```
# ggsave("../pics/graunt_halley_ggplot.png", gh4)
```

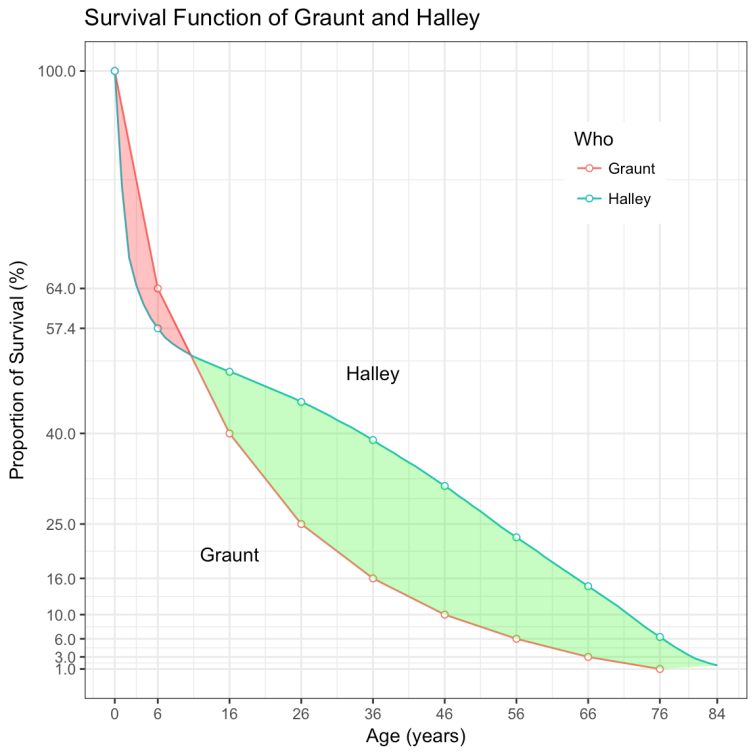
Polygon

Reuse `poly.upper` data frame and `poly.lower` data frame.

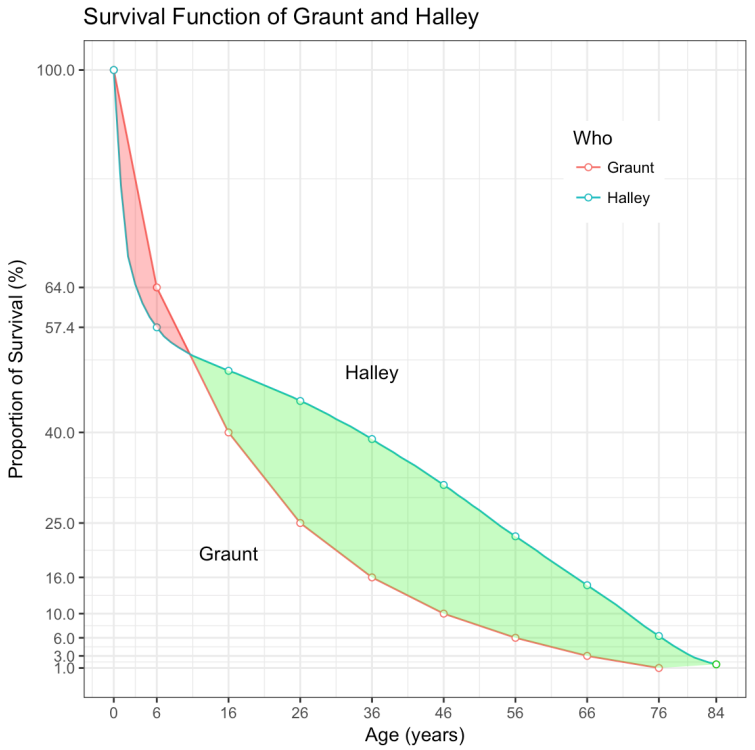
```
(ghp4 <- gh4 +
  geom_polygon(data = poly.upper, aes(x = x, y = y), alpha = 0.3, fill = "red"))
```



```
(ghp5 <- ghp4 +
  geom_polygon(data = poly.lower, aes(x = x, y = y), alpha = 0.3, fill = "green"))
```




```
(ghp5 <- ghp5 +
  geom_point(data = data.frame(x = 84, y = halley$xPo[85]), aes(x = x, y = y),  colou
r = 3, shape = 21, fill = "white"))
```




```
# ggsave("../pics/graunt_halley_poly_ggplot.png", ghp5)
```

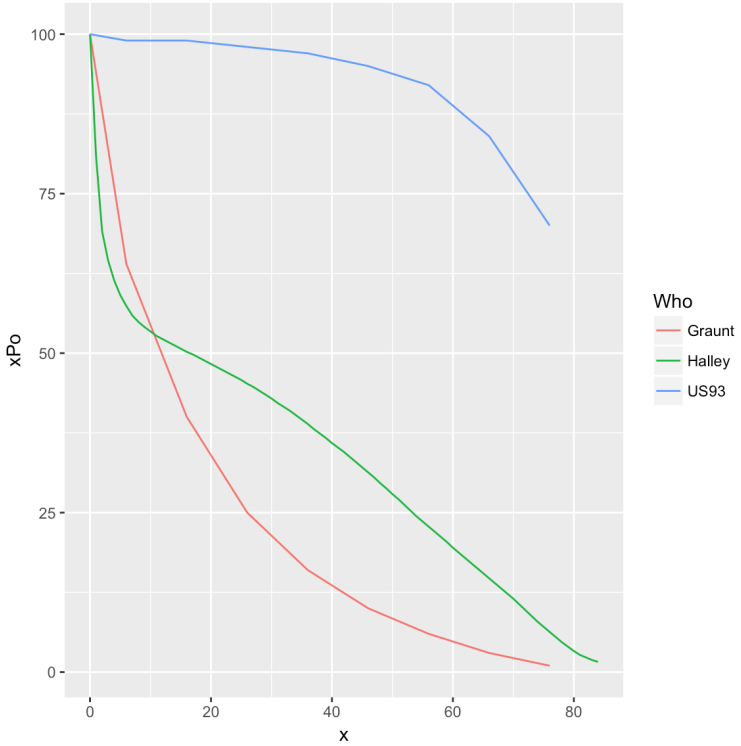
Graunt, Halley, and US93

Data Reshape

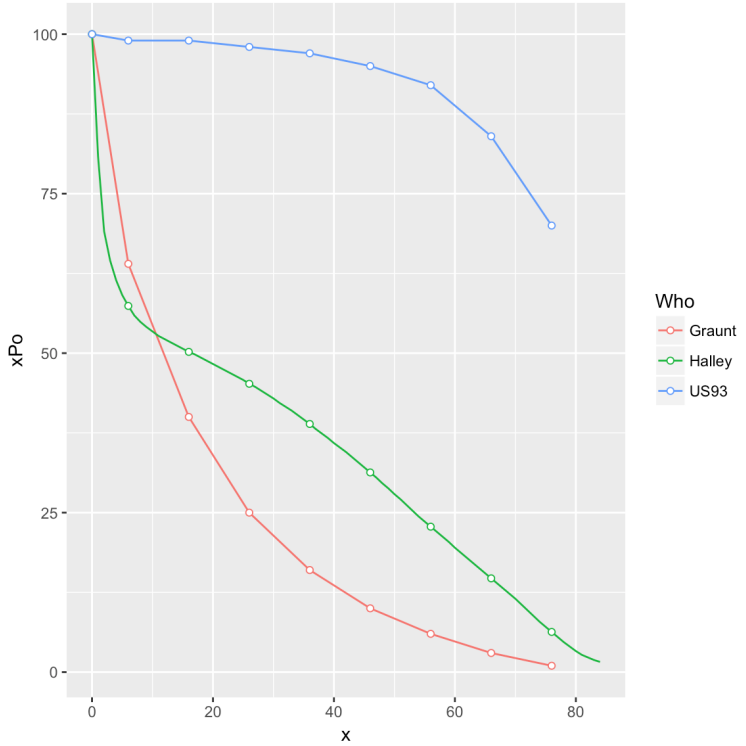
```
# us93.2 <- us93
# names(us93.2) <- c("x", "US93")
ghu.melt <- melt(list(graunt.2, halley.2, us93.2), id.vars = "x", value.name = "xPo",
 variable.name = "Who")
ghu.melt.g <- ghu.melt[ghu.melt$x %in% graunt$x, ]
# main.title.3 <- "Survival Function Plots"
```

Survival Function Plots with ggplot

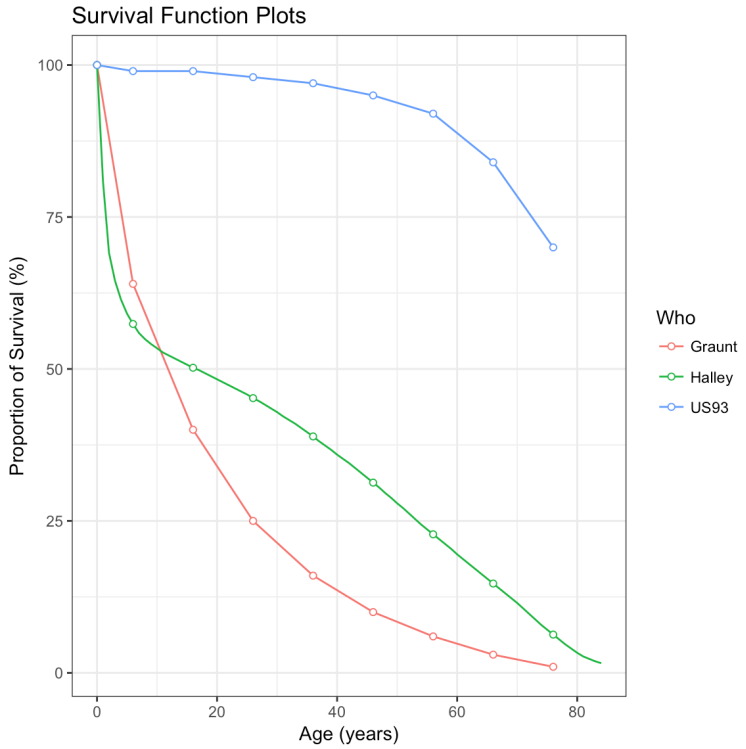
```
(ghu1 <- ggplot() +
  geom_line(data = ghu.melt, aes(x = x, y = xPo, colour = Who)))
```



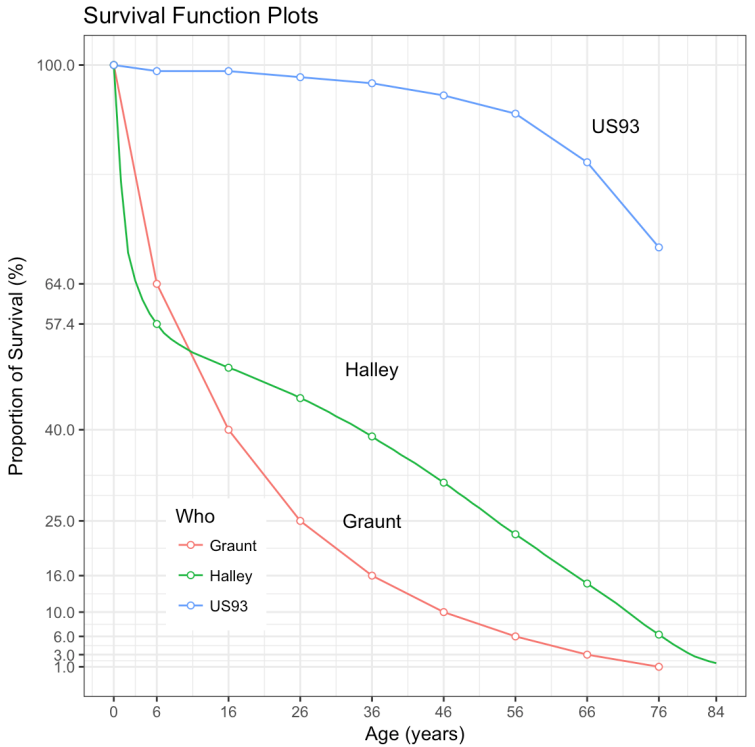

```
(ghu2 <- ghu1 +
  geom_point(data = ghu.melt.g, aes(x = x, y = xPo, colour = Who), shape = 21, fill =
 "white"))
```



```
(ghu3 <- ghu2 +
  theme_bw() +
  xlab(x.lab) +
  ylab(y.lab) +
  ggtitle(main.title.3))
```



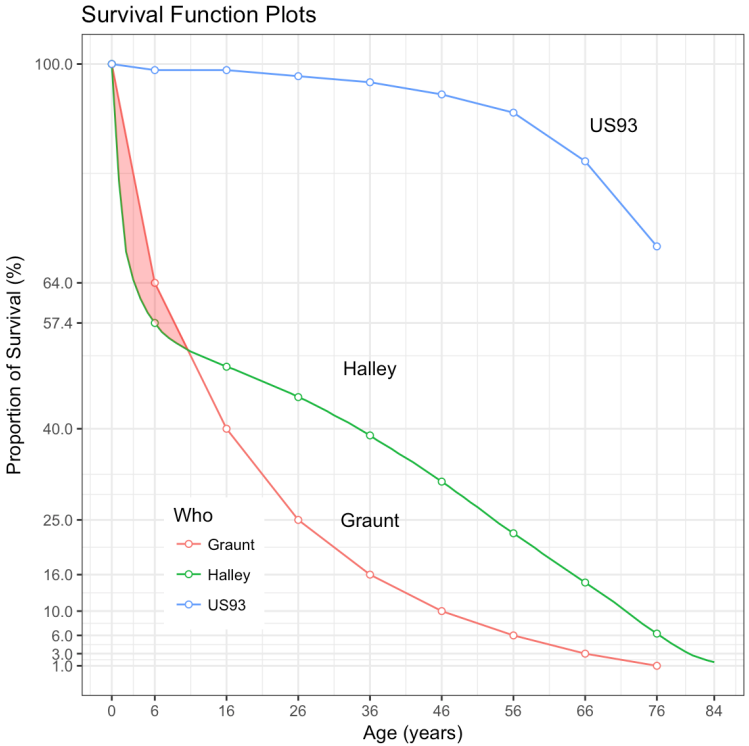

```
(ghu4 <- ghu3 +
  theme(legend.position = c(0.2, 0.2)) +
  annotate("text", x = c(36, 36, 70), y = c(25, 50, 90), label = c("Graunt",
"Halley", "US93")) +
  scale_x_continuous(breaks = c(graunt$x, 84)) +
  scale_y_continuous(breaks = c(graunt$xPo.g, xPo.halley.age.6)))
```



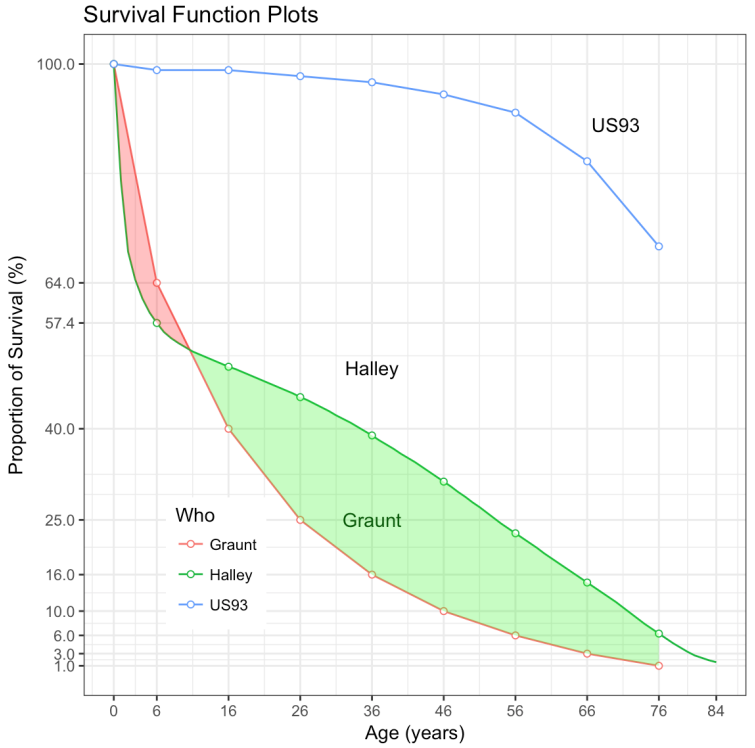

```
# ggsave("../pics/graunt_halley_us_ggplot.png", ghu4)
```

Polygon

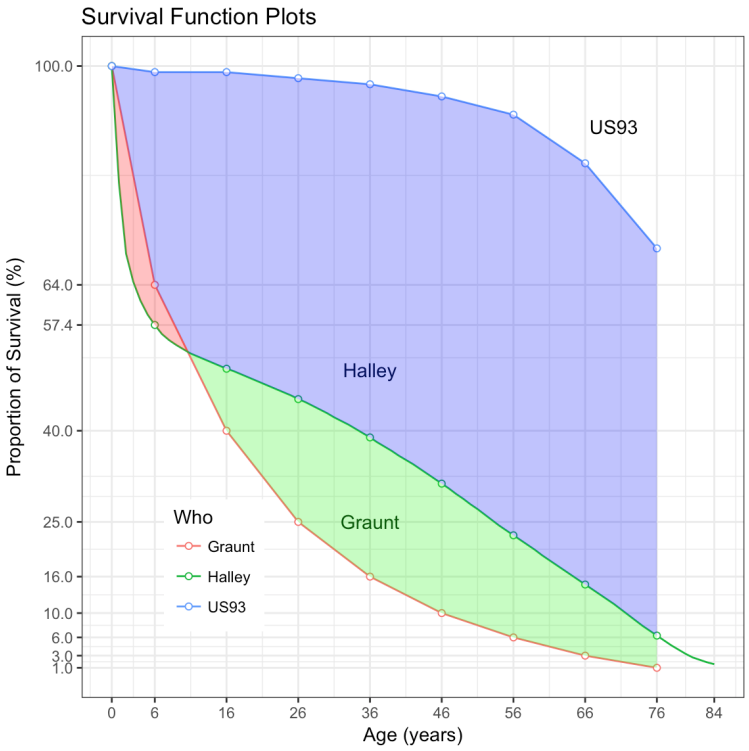
```
(ghup4 <- ghu4 +
  geom_polygon(data = poly.upper, aes(x = x, y = y), alpha = 0.3, fill = "red"))
```



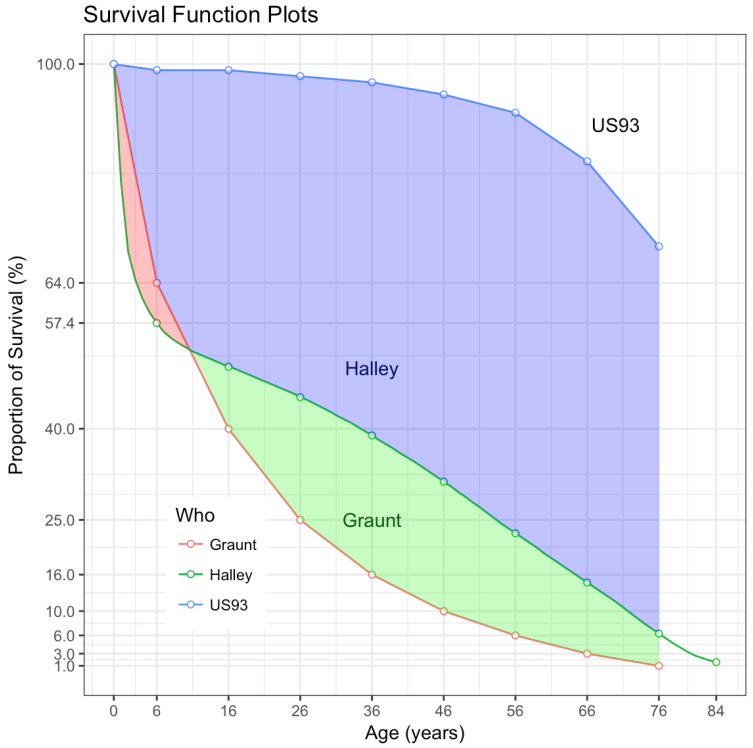

```
(ghup5 <- ghup4 +
  geom_polygon(data = poly.lower.76, aes(x = x, y = y), alpha = 0.3, fill = "green"))
```



```
(ghup6 <- ghup5 +
  geom_polygon(data = poly.us.76, aes(x = x, y = y), alpha = 0.3, fill = "blue"))
```




```
(ghup7 <- ghup6 +
  geom_point(data = data.frame(x = 84, y = halley$xPo[85]), aes(x = x, y = y),  colou
r = 3, shape = 21, fill = "white"))
```




```
# ggsave("../pics/graunt_halley_us_poly_ggplot.png", ghup7)
```

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_160406.rda")
```