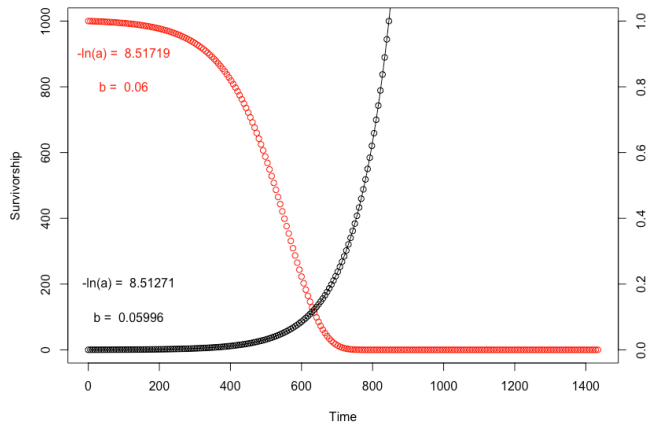
So,
Here is a small exercise to demonstrate the point.

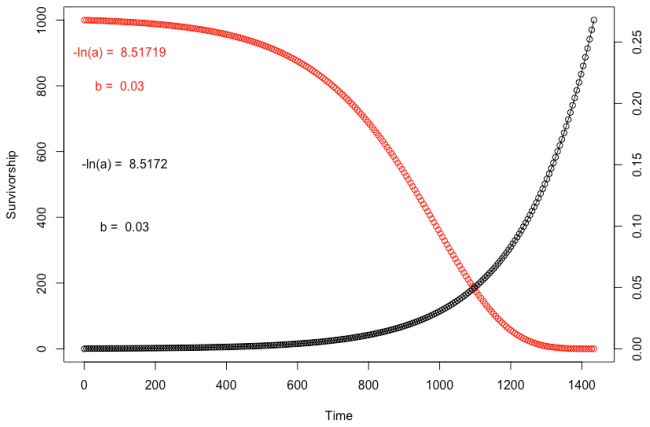
I wrote two functions.
First - a simple function (**Encoder**); it goes through time and kills off individuals (from the initial cohort of 1000) according to Gompertz hazard (death probability=a*exp(b*time). This function only generates survivorship curve (in red). This function also displays alpha (-ln(a)), and beta (b) that it operates with.

Second function (**Investigator**), only receives survivorship curve (nothing else). It converts survivorship curve into mortality data (black), fits exponential curve into that mortality, and extracts Gompertz coefficients. It also displays calculated from survivorship **alpha (-ln(a))**, and **beta (b)**, in black. As you can see in examples below, they coincide quite well, so all is working as it should. I for fun decreased **b** two fold, or decreased **a** few fold fold.

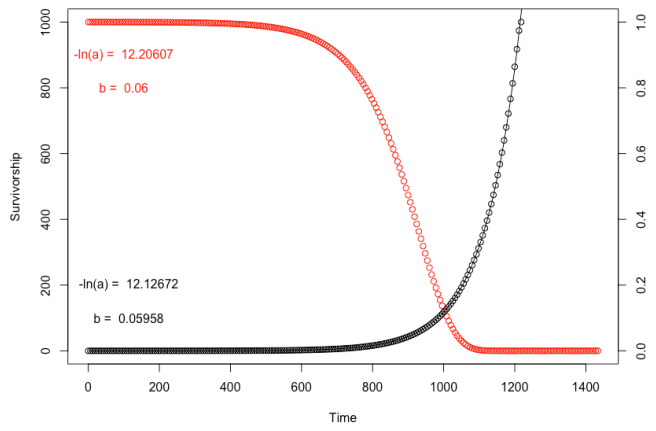
Few examples follow, the code for functions is at the end.




Rate of aging (b) is halved



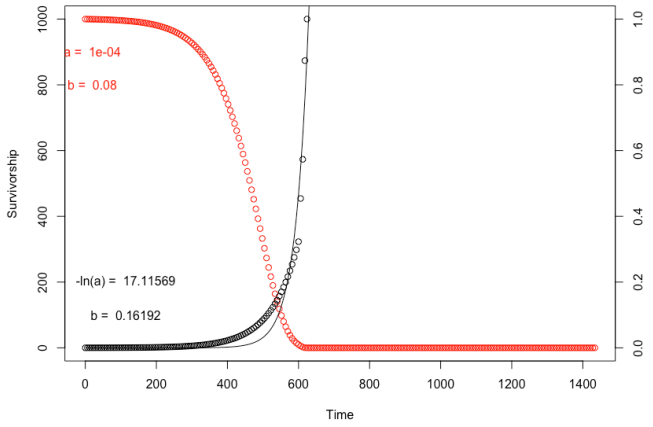

Mortality coefficient (a) is reduced

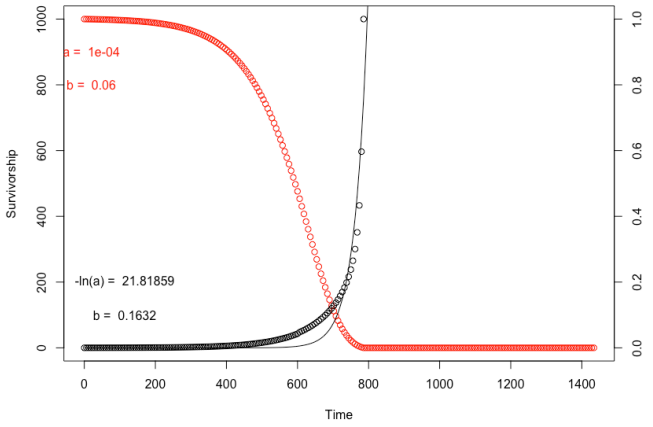



Notice that red and black coefficients are almost identical. Meaning that in this simple case, observer can easily and accurately calculate alpha and beta from survivorship curve.

Now, we introduce cancer (function, **encoder_cancer**). In this encoding, aging happens as before, however, depending on rate of aging, an individual can get cancer. Affected individual stays alive, but we assume that cancer grows linearly (there is a good evidence that volume of most cancers in mice increase linearly with time), and after certain time (100 computational steps) kills that individual. So, think of cancer in this example as death delayed by a defined amont of time.

Now, I show here two examples. Random baseline state (Encoder, and Investigator). And second example, I reduce **b (rate of aging**) by ~25% (from 0.8 to 0.6). However, if I run the output through the same **Investigator Function**, it suggests that **a**, and not **b** was changed. So, if cancer is responsible for the large portion of death, mortality analysis will be blind to increased or decreased rate of aging!! This can explain why analysis of numerous studies suggests that only **a**, and not **b** change in different mouse strains. Even though in reality, those mice might have quite different rates of aging.




If you compare red numbers, I changed b from 0.08 to 0.06, and didn't touch a. However, Gompertz analysis (black numbers, generated by Investigator function) suggested that b barely changed (from 0.162 to 0.163), however, a changed a lot.

**Encoder Function (in R)**

```
a<-0.000005
b<-0.06

or_size[1]<-1000
for (z in 2:240) {

 trout<-or_size[z-1]*(a*exp(b*z))
 if (trout<or_size[z-1]){or_size[z]<-((or_size[z-1]-trout))}
 else {or_size[z]<-0}
 }
plot(xz,or_size, pch=1, col='red', ylab='Survivorship', xlab='Time')
g<-round((-log(a)),5)
text(100,900,paste('-ln(a) = ',toString(g)), col='red')
text(100,800,paste('b = ',toString(b)), col='red')
```

**Investigator Function (in R)**

```
mort[1]<-0
temp_mort<-0
for (z in 2:240){
 mort[z]=NA
 rubic<-((or_size[z-1]-or_size[z])/or_size[z-1])
 if (!is.na(rubic)){
 if (rubic<temp_mort) {
  mort[z]=NA
  } else {
  mort[z]<-rubic
  temp_mort<-mort[z]
 }}
 #mort[z]<-max(mort[z-1],((or_size[z-1]-or_size[z])/or_size[z-1]))
 }
par(new = T)
plot(seq(1:240),mort, axes=F, xlab=NA, ylab=NA)
axis(side = 4)
mtext(side = 4, line = 3, 'Mortality')

boo <- data.frame(y = mort[1:240], x = seq(240))
mod <- nls(y ~ exp(a+b * x), data = boo, start = list(a = 0, b = 0))
lines(boo$x, predict(mod, list(x = boo$x)))
cc<-coef(mod)
text(20,0.2,paste('-ln(a) = ',toString(-round((cc[1]),5))))
text(20,0.1,paste('b = ',toString(round((cc[2]),5))))
```

**Encoder with Cancer**

```
a<-0.0001
b<-0.08
#0.08

or_size[1]<-1000
duration<-100
cancer<-rep(0, duration+2) #cancer extra death, delayed by 100 time units

for (z in 2:240) {
  trout<-or_size[z-1]*(a*exp(b*z))
  cancer[z+duration]<-or_size[z-1]*0.001*(exp(0.2*b*z))
  if ((trout+cancer[z])<or_size[z-1]){or_size[z]<-((or_size[z-1]-trout-cancer[z]))}
  else {or_size[z]<-0}
  }
plot(xz,or_size, pch=1, col='red', ylab='Survivorship', xlab='Time')
text(20,900,paste('a = ',toString(a)), col='red')
text(20,800,paste('b = ',toString(b)), col='red')
```