

Exponential Growth COVID 19 in Switzerland

Andre Meichtry

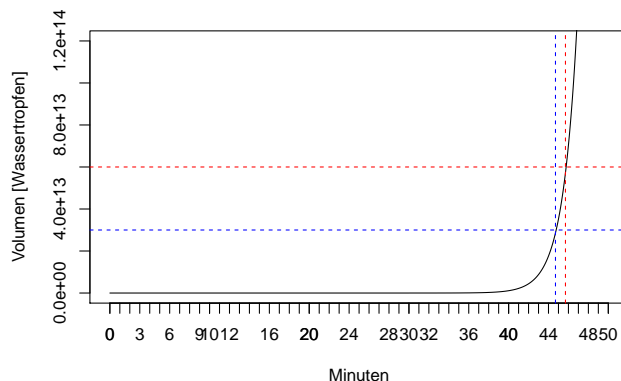
2020-11-02

The greatest shortcoming of the human race is the inability to understand the exponential function. (Al Bartlett)

Problem

Sie sitzen zuoberst in einem Fussball-Stadion; es hat Platz für 6×10^{13} Wassertropfen. Es beginnt mit einem Tropfen, der in der ersten Minute ins Stadion tröpfelt. Jede Minute *verdoppelt* sich die Anzahl Tropfen. Lange passiert nichts Besonderes. Sie sehen die Gefahr nicht kommen. Von der *Hälfte* bis *ganz* oben geht es plötzlich sehr schnell. Es würde auch nicht viel bringen, wenn das Stadion noch viel grösser wäre; wir haben **exponentielles Wachstum**.

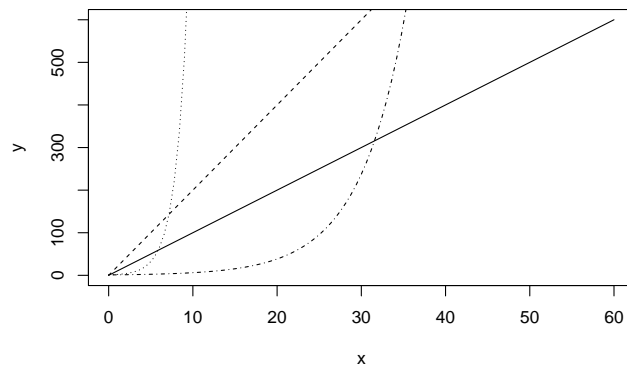
```
max <- 3e06*20e06 ##Volumen Allianz-Arena
halb <- max/2
max2<-max*2
tmax<-50
t<-seq(0,tmax,by=0.1)
expwachs <- function(t) {y<-1*2^(t/1)}
plot(t,expwachs(t),type="l",ylim=c(0,max2),main="",ylab="Volumen [Wassertropfen]", xlab="Minuten")
axis(side = 1, at = seq(0,tmax,by=1))
abline(h=max,lty=2,col="red")
abline(v=t[458],lty=2,col="red")
abline(h=halb,lty=2,col="blue")
abline(v=t[448],lty=2,col="blue")
```



Exponentialfunktion

Jede Exponentialfunktion a^x mit $a > 1$ wächst ab einem gewissen x schneller als jede lineare Funktion $a + bx$.

```
curve(10*x,from=0,to=60,ylab="y")
curve(20*x,add=TRUE,lty=2)
curve(2^x,add=TRUE,lty=3)
curve(1.2^x,add=TRUE,lty=4)
```



Exponentielles Wachstum

Exponential growth is defined by

$$x(t) = x_0 e^{kt} = x_0 e^{t/\tau} = x_0 2^{t/T},$$

with t : time, τ : e -folding time, T : doubling time, k : growth constant. ¹

- Bei Verdoppelung alle 2 Tage: $2^{t/2} = (2^{1/2})^t = 1.41^t$
- Bei Verdoppelung alle 3 Tage: $2^{t/3} = (2^{1/3})^t = 1.26^t$
- Bei Verdoppelung alle 7 Tage: $2^{t/7} = (2^{1/7})^t = 1.1^t$
- Bei Verdoppelung alle 10 Tage: $2^{t/10} = (2^{1/10})^t = 1.07^t$

Covid-19

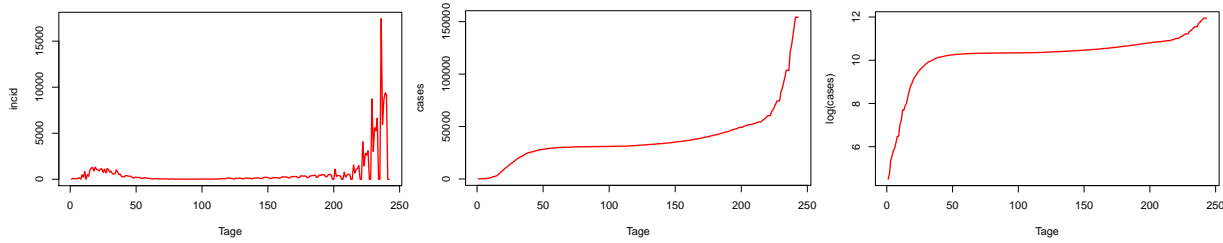
Inzidenz und kumulierte Inzidenz

```
data<-read.csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv")
sw<-data[data$Country/Region=="Switzerland",-c(1,2,3,4)]
cases<-as.numeric(sw[-c(1:42)])
incid<-diff(cases)
plot(1:length(incid),incid,type="l",col="red",lwd=2,xlab="Tage")
```

¹Proof:

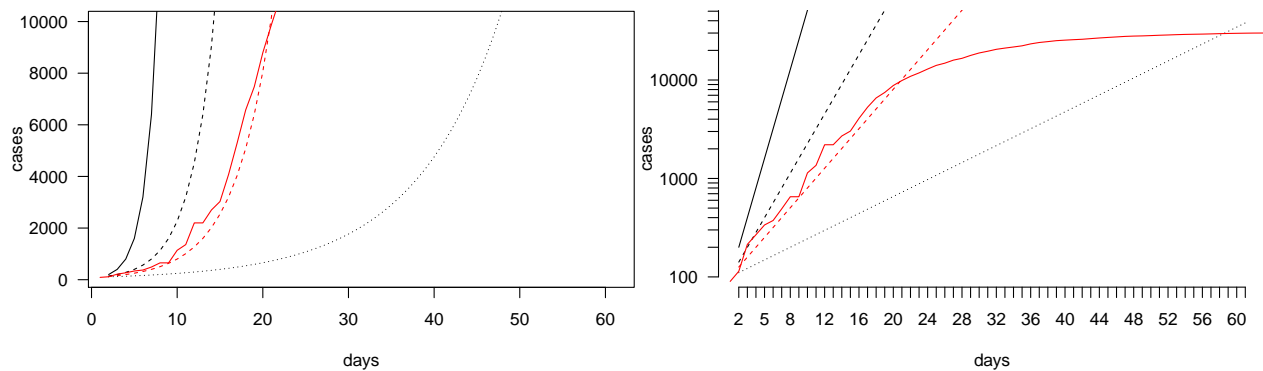
$$x(t) = x_0 e^{kt} = x_0 e^{t/\tau} = x_0 2^{\frac{t}{\tau} \log_2 e} = x_0 2^{\frac{t}{\tau \log_2 2}} = x_0 2^{t/T}$$

```
plot(1:length(cases),cases,type="l",col="red",lwd=2,xlab="Tage")
plot(1:length(cases),log(cases),type="l",col="red",lwd=2,xlab="Tage")
```



Erste Welle Covid-19

```
swisspop<-8e6
time<-seq(1,60,by=1)
tag<-1:length(cases)
T1<-1
T2<-2
T3<-3
T7<-7
x0<-100
Y1<-x0*2^(time/T1)
Y2<-x0*2^(time/T2)
Y3<-x0*2^(time/T3)
Y7<-x0*2^(time/T7)
time<-time+1
plot(time,Y1,type="l",ylab="cases",ylim=c(100,10000),xlab="days",las=1)
lines(time,Y3,col="red",lty=2)
lines(time,Y2,lty=2)
lines(time,Y7,lty=3)
abline(h=swisspop,lty=5,col="red")
points(tag,cases,type="l",col="red")
plot(time,Y1,log="y",type="l",ylab="cases",xlab="days",axes=FALSE,ylim=c(100,40000))
at.y <- outer(1:9, 10^(2:9))
lab.y <- ifelse(log10(at.y) %% 1 == 0, at.y, NA)
axis(2, at=at.y, labels=lab.y, las=2)
axis(1,time)
lines(time,Y2,lty=2)
lines(time,Y7,lty=3)
lines(time,Y3,col="red",lty=2)
abline(h=swisspop,col="red",lty=3)
points(tag,cases,type="l",col="red")
```



Erste Welle. Example of doubling times: 1 day (solid), 2 days (dashed), 3 days (red), seven days (dotted), with reported cases Covid19 in Switzerland. Horizontal line: swiss population. On a logarithmic scale, a straight line indicates exponential growth. Quelle.

Auswirkung Vorfaktor

Annahme: Verdoppelung alle drei Tage, 10 Prozent der Infizierten müssen ins Spital. Die Anzahl Cases von heute sind die Anzahl Spitalpatienten in 9 Tagen, **wenn man nichts macht**.

$$0.1 \times 2^{0.33t} = 0.1 \times (2^{0.33})^t = 0.1 \times 1.3^t = 1.3^{\log_{1.3} 0.1} 1.3^t = 1.3^{t + \log_{1.3} 0.1} = 1.3^{t-8.776}$$

Analog kann man zeigen: Wenn die Mortalitätsrate bei einem Prozent der bestätigten Fälle liegt, dann ist die Anzahl der bestätigten Fälle die zu erwartende Anzahl der Todesfälle ca. 18 Tage später, **wenn man nichts macht**.

$$0.01 \times 2^{0.33t} = 0.01 \times (2^{0.33})^t = 0.01 \times 1.3^t = 1.3^{\log_{1.3} 0.01} 1.3^t = 1.3^{t + \log_{1.3} 0.01} = 1.3^{t-17.552}$$

```
delay<-log(0.1)/log(1.3)
```

```
delay
```

```
[1] -8.7763
```

```
delay2<-log(0.01)/log(1.3)
```

```
delay2
```

```
[1] -17.553
```

```
plot(time,2^(0.33*time),ylab="cases",xlab="days",type="l",ylim=c(0,10000),col=3)
```

```
lines(time,0.1*2^(time/3),lty=1,col=1)
```

```
lines(time,0.01*2^(time/3),lty=1,col=2,lwd=2)
```

