

# Thema\_Opdracht08\_introductie

Kasper Notebomer, Susan Reefman, Sander J. Bouwman

21/04/2021

## Introductie opdracht

Iedere groep krijgt een zak M&M's en een schaal. We gaan vervolgens de schaal vullen volgens het volgende principe: voeg 10 stuks toe en haal vervolgens 10% eraf (toevoeging =  $+10 - 0.1 * \text{nieuw volume}$ ) oftewel  $dY = 10 - 0.1 * (Y+10)$

[1] Welke parameters (constanten) zitten er in de vergelijking? De twee constanten zijn de  $+ 10$  en de  $- 10\%$

[2] Welke variabele uit de vergelijking verandert steeds?

Y.

[3] Wat is de initiele waarde van de variabele uit vraag 2?

0.

[4] Wat is het tijdsframe van dit experiment? ( $t = 0$  tot  $t = \dots\dots\dots$ ).

90.

```
count <- 0
dY <- 1
total <- 0
count_l <- c()
total_l <- c()
dY_l <- c()
dY <- 1

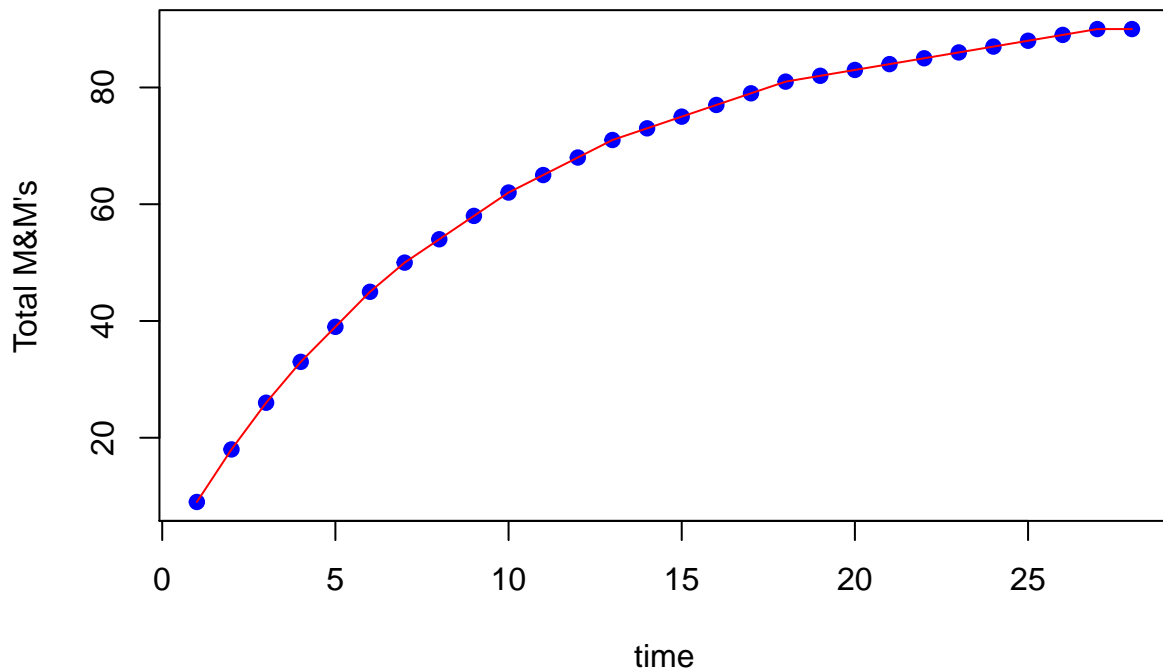
while (dY != 0) {
  count <- count + 1
  dY = 10 - 0.1 * (total+10)
  total <- total + ceiling(dY)
  count_l <- c(count_l, count)
  total_l <- c(total_l, total)
  dY_l <- c(dY_l, dY)
}

menm <- data.frame(time = count_l, tot = total_l, dY = dY_l)
cat("Evenwichtsmoment is bij: ", count, "\tCeiling is: " , total)
```

[5] Wanneer is het evenwichtsmoment?

## Evenwichtsmoment is bij: 28      Ceiling is: 90

```
plot(tot ~ time, data = menm, pch=19, col="blue", xlab="time", ylab="Total M&M's")
lines(menm$tot ~ menm$time, xlim=range(0), ylim=range(100), pch=16, col="red")
```



[6] Uit welke variabele uit de vergelijking kunnen we dat aflezen?  $dY = 0$

## Opdracht 2

```
library(deSolve)

parameters <- c(addVolume = 10, pV = 0.1)

# define model
volume <- function(t,y,parms){
  with(as.list(c(parms)),{
    dY <- addVolume - pV * (y+addVolume)
    return(list(c(dY)))
  })
}

#initial state
state <- c(Volume = 0)

#define time sequence you want to run the model
times <- seq(0, 100, by = 1)

# run simulation using continuous approach
out <- ode(times = times, y = state, parms = parameters, func = volume, method = "euler")
```

```
head(out)
```

```
##      time  Volume
## [1,]    0  0.0000
## [2,]    1  9.0000
## [3,]    2 17.1000
## [4,]    3 24.3900
## [5,]    4 30.9510
## [6,]    5 36.8559
```

```
plot(out)
```

