Week 1 opdracht

M van de Streek & L T Stein

2023-04-25

Contents

1 Introductie

In dit onderzoek wordt er gekeken naar een model die een evenwichtstoestand aanduid. Dit wordt gedaan met behulp van een praktijkvoorbeeld. In dit voorbeeld wordt een schaal gevuld met M&M's. Er wordt steeds een vast aantal M&M's toegevoegd en een vast percentage vanaf gehaald. Uiteindelijk komt de schaal in evenwicht. Om dit te experiment na te bootsen wordt een model gebruikt.

1.1 Doel

Het doel van dit onderzoek is om een model te maken/gebruiken die een verglijkbaar resultaat geeft als het reele verloop van het experiment. Dit doel willen we bereiken door beide resultaten weer te geven in een grafiek, en vervolgens onderling te vergelijken. Aan de hand van de verschillen tussen deze figuren kun je nagaan of het model juist is.

Aan de hand van het volgende model zullen de resultaten gelijk zijn aan het reele verloop van het experiment.

Model:

$$dY = 10 - 0.1 * (Y + 10)$$

1.2 Theorie

Tijdens dit model wordt een zak M&M's gebruikt. Uit deze zak worden steeds een constant aantal toegevoegd aan een schaal. Vervolgens wordt er steeds weer 10% aantal uit deze schaal gehaald. Vereenvoudigd is dit dus: voeg 10 stuks toe en haal vervolgens 10% eraf.

De variabeles in de formule veranderen niet, er komt steeds een absoluut aantal van 10 bij en er wordt een procentueel aantal van 10 afgehaald. Zoals [1] citeert: "An equilibrium of a dynamical system is a value of the state variables where the state variables do not change." De verglijking is dus in evenwicht omdat de staat van de variabeles niet veranderen.

1.2.1 Elementen

$$dY = 10 - 0.1 * (Y + 10)$$

Hierin is:

- dY = de toevoeging
- Y = nieuwe volume
- -0.1 = de 10% correctie die er steeds afgaat
- (+)10 = absolute aantal wat er in de schaal ligt

1.2.2 Transformaties

Hieronder worden de antwoorden op de vragen gegeven.

1. In de vergelijking zitten twee constanten: 10 en 0,1. Deze waardes veranderen niet en zijn dus niet afhankelijk van Y.
2. De variabele Y verandert steeds.
3. Variabele Y heeft als initiële waarde 9.
4. Het tijdsverloop begint bij t=0 en loopt tot t=100.
5. Als de 10% van het volume gelijk is aan een absoluut aantal wat er bij komt (=10) dan is de vergelijking in evenwicht.
6. Aan de variabele dY.

2 Methode

2.1 Het software model

Om dit model uit te werken zal gebruik gemaakt worden van R. In R gebruiken we het pakket deSolve. deSolve zal de differentiaal vergelijking oplossen met de eiegengeschreven functie. Onze functie heeft als doel om het nieuwe volume te berekenen.

```
library(deSolve)
```

```
params <- c(add.volume = 10, decrease.perc = 0.1)
```

```
volume <- function(t, y, params) {
  with(as.list(c(params)),{
  dY <- add.volume - decrease.perc * (y + add.volume)
  return(list(c(dY)))
  }
  )
}
```

```
state <- c(state = 0)
timeframe <- seq(0, 100, by = 1)
```

```
out <- ode(times = timeframe, y = state, parms = params,
           func = volume, method = "euler")
head(out)
```

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

Allereerst maken we dus onze functie. Vervolgens berekent deSolve de differentiaal verglijking. Hiervoor zijn een aantal parameters nodig: startwaarde en timeframe. Als de functie en parameters zijn gedefineerd kan ode functie van deSolve aangroepen worden. Ode staat voor ordinary differential equations.

DeSolve heeft zoals vermeld twee parameters nodig: startwaarde en timeframe. De startwaarde is 0, omdat de schaal natuurlijk leeg begint. Het timeframe loopt van 0 tot 100, want 10% van 100 is 10. 10 is de absolute waarde die er steeds bijkomt. Dit kunnen we defineren. De waardes worden opgeslagen in de variabele *out*.

2.2 Model configuratie

Table 1: Parameter Values

Parameter	Value	Unit
add.volume	10	volume
decrease.perc	0.1	percentage
state	0	volume
timeframe	0/100	per 1

De variabele add.volume is het absolute aantal wat er steeds bij wordt gedaan. Decreace.perc is het percentage wat er wordt afgehaald. Zoals bovenstaand al beschreven zijn dit respectievelijk 10 en 0.1%.

De state variabele geeft aan wat de startwaarde van Y is. Deze begint bij 0, want we willen met een lege schaal beginnen. Het timeframe loopt van t = 0 tot t = 100.

3 Resultaten

Alle waardes zijn opgegeslagen in out en kunnen dus geplot worden.

```
plot(
  out,
  main = "Verloop van de volume van de vergelijking",
  xlab = "Tijdsframe (t=0 tot t=100)",
  ylab = "Volume (aantal M&M's)",
  col = "blue",
  lwd = 3,
  ylim = c(0, 200))

lines(ode(times = timeframe, y = c(state=100), parms = params,
        func = volume, method = "euler"), col = "red", lwd = 3)

legend("topleft",
       legend = c("Originele model", "Initiële startwaarde van 100"),
       col = c("blue", "red"),
       lwd = 3)
```

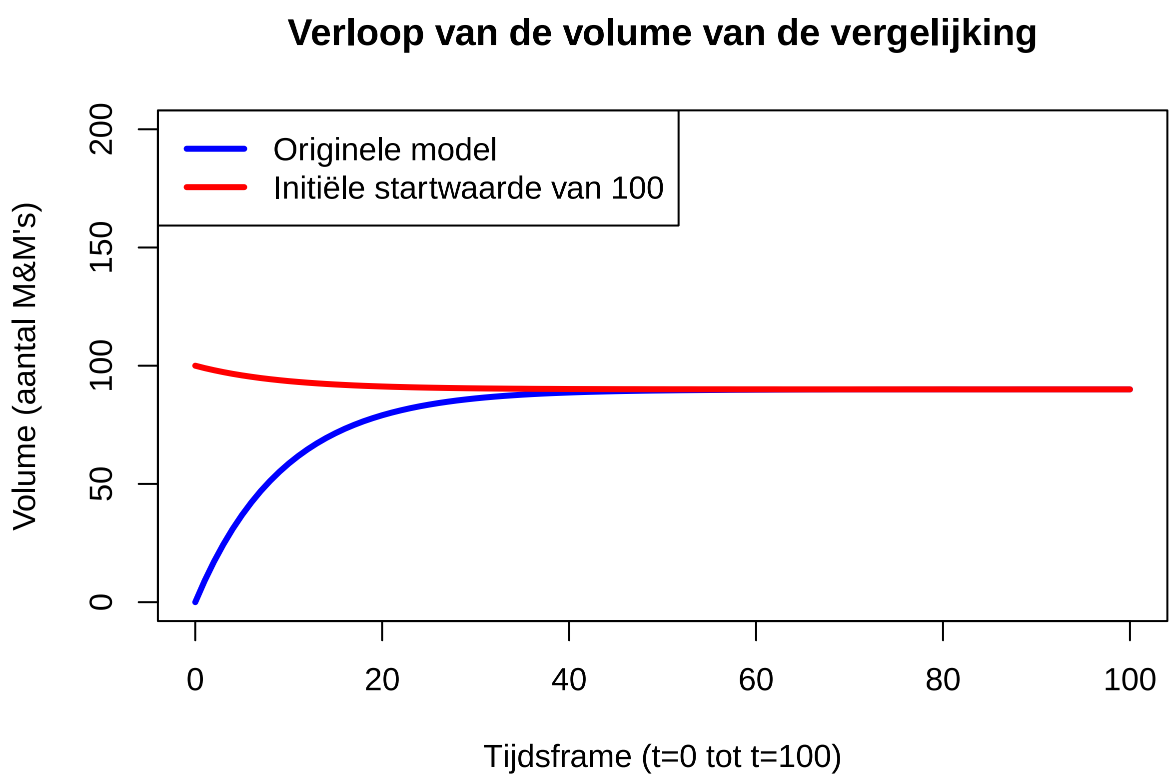

Figure 1: Verloop van de volume van het model

De bovenstaande grafiek geeft het verloop van de vergelijking weer die bovenstaand is gedefineerd/uitgelegd. Ook is het verloop geplot bij een initiële startwaarde van 100. Dit verloop heeft een rode kleur.

Uit de bovenstaande resultaten blijkt dat de verwachte hypothese uitkomt, het verloop van het model is gelijk aan het reële verloop. Echter is het wel goed om de verandering op te merken bij een startwaarde van 100 (zie model configuratie voor uitleg). Onderstaand wordt de uitleg hiervoor gegeven.

Antwoorden op een aantal vragen:

1. Er vindt geen verandering plaats bij het aanpassen van de parameters. Het blijven constante variabelen en dus verandert het verloop niet.
2. Als je de initiële waarde aanpast, geeft deze pas een ander effect bij 100. Dit is omdat het percentage hier velen malen hoger is dan het aantal wat er bijkomt.
3. Het tijdsframe zal geen effect hebben op het verloop. Er is alleen een effect als het *aantal* punten wordt aangepast in het tijdsframe. Het model heeft een x aantal tijdspunten nodig om op evenwicht te komen.
4. Bij het aanpassen van het model zodat de correctie is op basis van y is er geen effect. Er evenveel tijdspunten nodig om de evenwichtsstand te bereiken.

4 Discussie en Conclusie

4.1 Discussie

- Compare your results with what is expecting from the literature and discuss differences with them.
- Discuss striking and surprising results.
- Discuss weaknesses in your research and how they could be addressed.

In de resultaten hebben we al vermeld dat de hypothese uitkwam. Dit kan worden aangetoond door te kijken naar de grafiek van het reële experiment. Deze grafiek vertoond een vergelijkbaar verloop met de grafiek van het model die we gebruikt hebben.

Omdat bij dit onderzoek de hypothese uitkwam, waren er niet veel verrassende resultaten. Echter was het wel duidelijk om te zien dat de grafiek een ander verloop toonde bij een initiële startwaarde van 100.

Een verbeterplan voor het volgende onderzoek is om de parameters beter af te stemmen. Bij een initiële startwaarde van 100 werd duidelijk dat de grafiek een ander verloop had. Dit soort verlopen kunnen wellicht nog meer worden opgespoord door meer met de parameters te schuiven.

4.2 Algemene conclusie en perspectief

Het doel van dit onderzoek was om aan te tonen dat het model wat gebruikt wordt een vergelijkbaar patroon volgt als het daadwerkelijke onderzoek. In het eindresultaat hebben we dit kunnen aantonen. Ook hebben we het model iets nader uitgelegd.

Bij het vervolg onderzoek is het mogelijk om te kijken naar meer correcties. Zo kun je er achter komen welke factoren nou echt belangrijk zijn voor het model.

References

[1] Nykamp DQ, From Math Insight,: *Equilibrium Definition*, 2023.

5 Oefenen met de tutorial

Onderstaand volgen nog een aantal scripts met modellen. Deze scripts zijn afkomstig van de deSolve documentatie. Voor elke case worden de volgende vragen beantwoord:

1. Welke parameters (constantes) zitten er in de vergelijking(en)?
2. Wat is/zijn de initiële waarde(s)?
3. Wat is het tijdsframe van dit experiment?

Ook wordt er per plot beschreven wat de grafiek allemaal laat zien.

5.1 Lorenz model

```r
parameters <- c(a = -8/3, b = -10, c = 28)
state <- c(X = 1, Y = 1, Z = 1)

Lorenz<-function(t, state, parameters) {
    with(as.list(c(state, parameters)),{
    dX<-a*X+Y*Z
    dY<-b*(Y-Z)
    dZ<--X*Y+c*Y-Z
    # return the rate of change
    list(c(dX, dY, dZ))
  }) # end with(as.list ... +}
}

times <- seq(0, 100, by = 0.01)

out <- ode(y = state, times = times, func = Lorenz, parms = parameters)
head(out)
```

```
##      time         X        Y        Z
## [1,] 0.00 1.0000000 1.000000 1.000000
## [2,] 0.01 0.9848912 1.012567 1.259918
## [3,] 0.02 0.9731148 1.048823 1.523999
## [4,] 0.03 0.9651593 1.107207 1.798314
## [5,] 0.04 0.9617377 1.186866 2.088545
## [6,] 0.05 0.9638068 1.287555 2.400161
```

```r
par(oma = c(0, 0, 3, 0))
plot(out, xlab = "time", ylab = "-")
plot(out[, "X"], out[, "Z"], pch = ".")
mtext(outer = TRUE, side = 3, "Lorenz model", cex = 1.5)
```

In de vergelijking zijn de volgende constantes aanwezig:

Paramater	Value
a	-8/3
b	-10
c	28
timeframe	0/100

De initiële waardes zijn:

Paramater	Value
X	1
Y	1
Z	1

Het tijdsframe voor dit experiment loopt van t = 0 tot t = 100, met stappen van 0.01.

De eerste drie plots zijn drie verschillende vergelijkingen. Ze starten alle drie wel met dezelfde initiële waardes. Deze drie vegelijking samen geven het gedrag van de atmosfeer van de aarde. Bij het vierde plot worden de X en Z gecombineerd, en dus het twee dimensionale model gegeven.

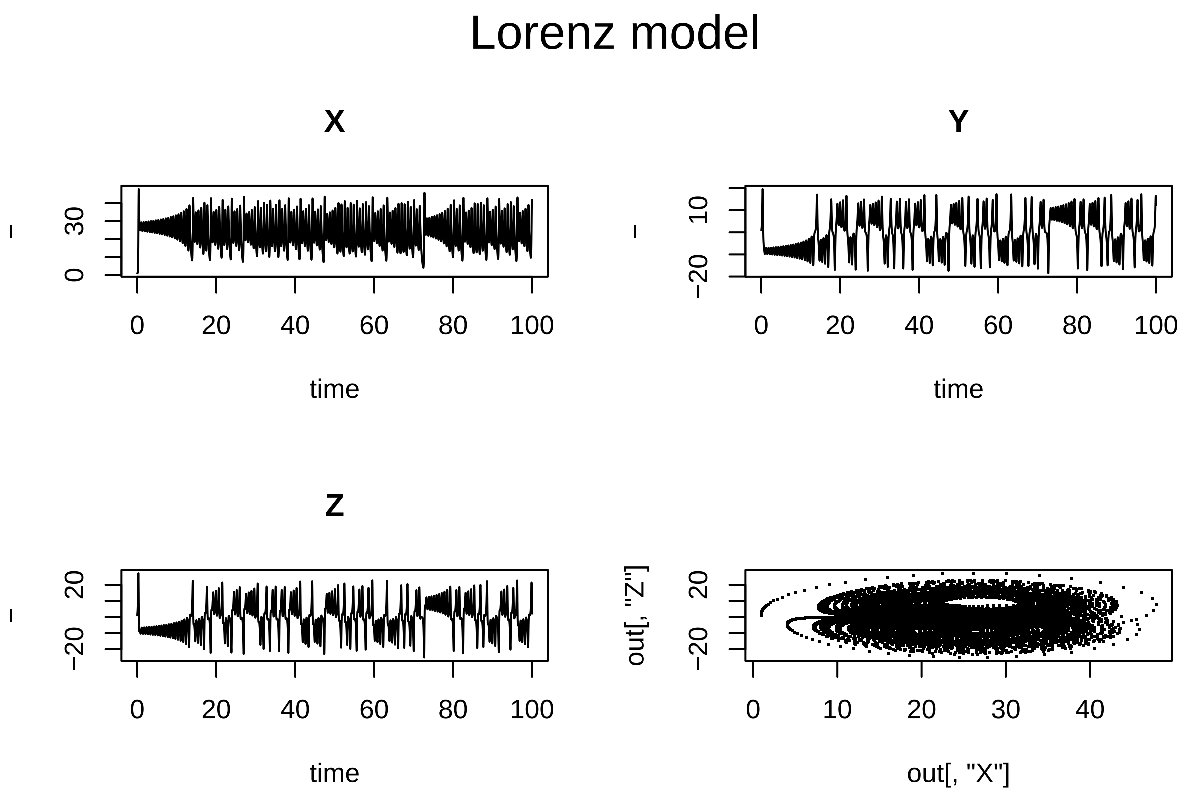

Figure 2: Lorenz Model

5.2 Combustion model

In deze sectie wordt er gekeken naar één model, maar met vier verschillende initiële waardes.

```
combustion <- function (t, y, parms) {
   list(y^2 * (1-y) )
}

yini  <- 0.01
times <- 0 : 200
out  <- ode(times = times, y = yini, parms = 0, func = combustion)
out2 <- ode(times = times, y = yini*2, parms = 0, func = combustion)
out3 <- ode(times = times, y = yini*3, parms = 0, func = combustion)
out4 <- ode(times = times, y = yini*4, parms = 0, func = combustion)

plot(out, out2, out3, out4, main = "combustion")
legend("bottomright", lty = 1:4, col = 1:4, legend = 1:4, title = "yini*i")
```

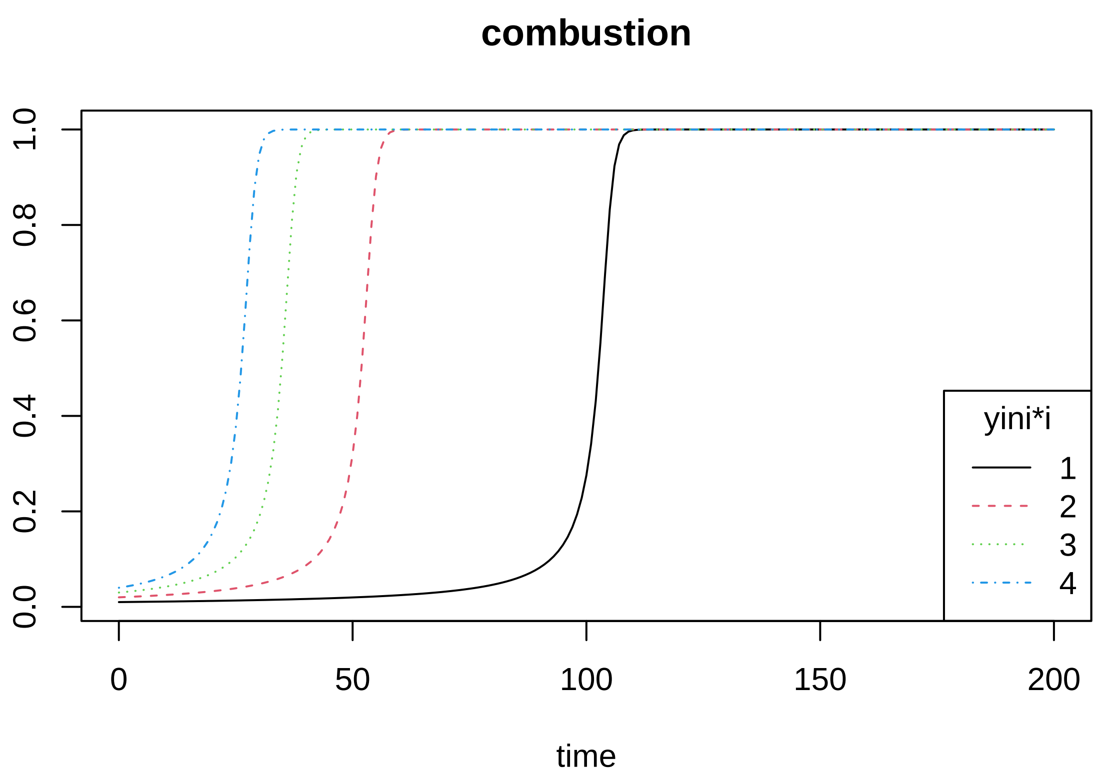

Figure 3: Combustion Model

De volgende constantes zijn aanwezig:

Paramater	Value
timeframe	0/200

De initiële waardes zijn:

Paramater	Value
yini	0.01
yini * 1	0.02
yini * 2	0.03
yini * 3	0.03

In de grafiek wordt vier keer hetzelfde model geplot, maar elke keer met een andere initiële waarde. Daarom hebben de verschillende lijnen ook een ander verloop. Je kunt zien naar mate de initiële waarde groter wordt, de stijging eerder begint.

5.3 Ccl4 Model

In deze case wordt er gekeken naar het ccl4model. Dit is een model die in het deSolve paket zit. Dit model zegt iets over ratten die gedrugged zijn met ccl4 in een kamer met toxische concentraties.

```r
obs <- subset (ccl4data, animal == "A", c(time, ChamberConc))
names(obs) <- c("time", "CP")

# Het defineren van de constanten
parms <- c(0.182, 4.0, 4.0, 0.08, 0.04, 0.74, 0.05, 0.15, 0.32, 16.17,
281.48, 13.3, 16.17, 5.487, 153.8, 0.04321671, 0.40272550, 951.46, 0.02, 1.0, 3.80000000)

# Het defineren van de initiële waardes
yini <- c(AI = 21, AAM = 0, AT = 0, AF = 0, AL = 0, CLT = 0, AM = 0)

out <- ccl4model(times = seq(0, 6, by = 0.05), y = yini, parms = parms)
par2 <- parms
par2[1] <- 0.1
out2 <- ccl4model(times = seq(0, 6, by = 0.05), y = yini, parms = par2)
par3 <- parms
par3[1] <- 0.05
out3 <- ccl4model(times = seq(0, 6, by = 0.05), y = yini, parms = par3)

plot(out, out2, out3, which = c("AI", "MASS", "CP"),
     col = c("black", "red", "green"), lwd = 2,
     obs = obs, obspar = list(pch = 18, col = "blue", cex = 1.2))

legend("topright", lty = c(1,2,3,NA), pch = c(NA, NA, NA, 18),
       col = c("black", "red", "green", "blue"), lwd = 2,
       legend = c("par1", "par2", "par3", "obs"))
```

```r
obs2 <- data.frame(time = 6, MASS = 12)
```

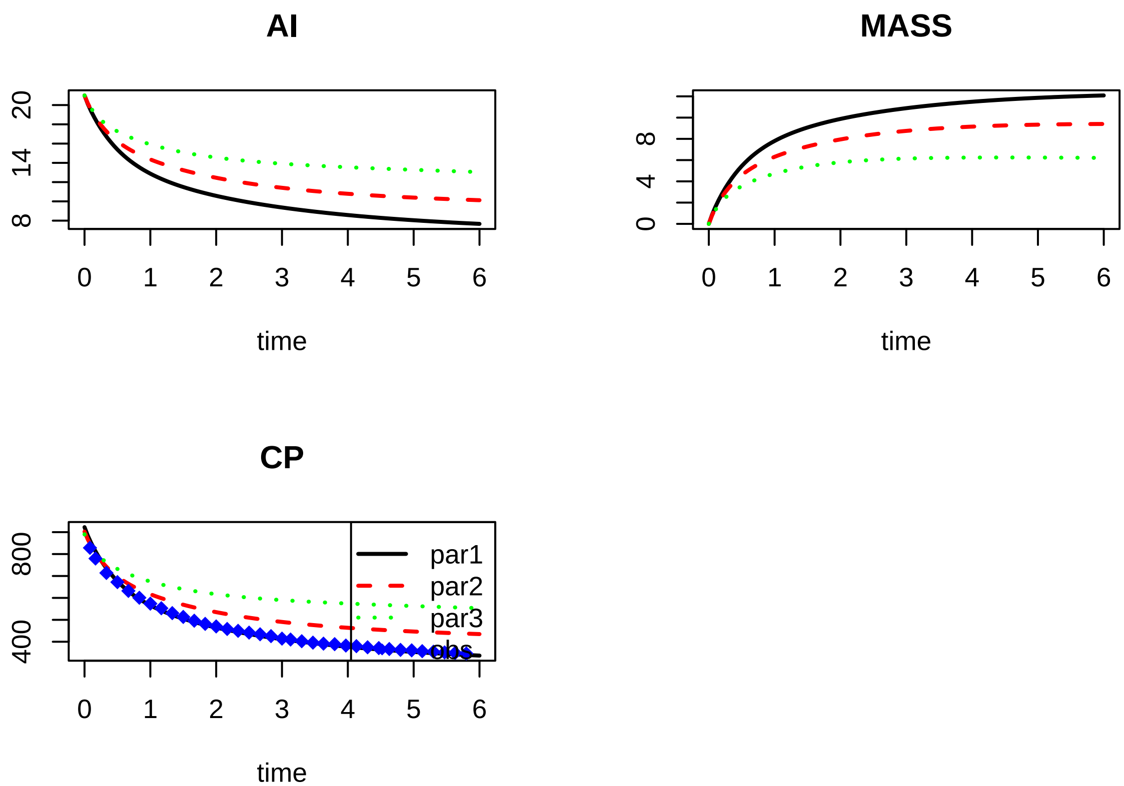

Figure 4: CCL4 model

```
plot(out, out2, out3, lwd = 2,
     obs = list(obs, obs2),
     obspar = list(pch = c(16, 18), col = c("blue", "black"),
                   cex = c(1,2, 2))
     )
```

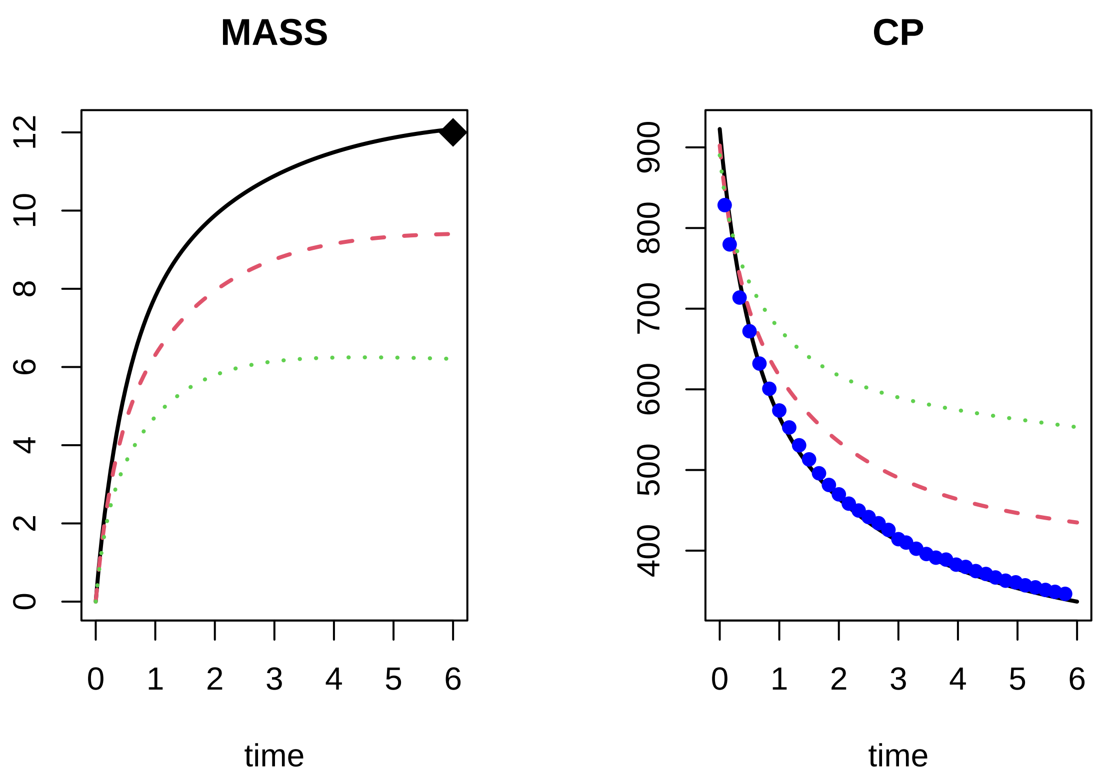

Figure 5: Verloop van dagnummer 6

In dit model zijn heel veel paramaters aanwezig. 21 om precies te zijn. Bovenstaand in het code block worden deze allemaal gedefineerd. De initiële waardes zijn als volgt:

Paramater	Value
AI	21
AAM	0
AT	0
AF	0
AL	0
CLT	0
AM	0

De drie grafieken zijn allemaal van hetzelfde model. Ze worden alleen steeds aangeroepen met één constante, die een andere waarde krijgt. Deze constante is het gewicht van de rat. Het gewicht is dus een belangrijke constante bij het model. CP is de concentratie van de toxische stoffen in de kamer. M is de massa van het dier. AI is de concentratie CCL4 in de rat.

Bij de laatste twee plots wordt er alleen gekeken naar dag nummer 6. Bij de eerste drie plots wordt er gekeken naar alle dagen waarop er data aanwezig is.

Het is ook mogelijk om van alle initiële waardes de histogrammen te plotten. Dit wordt onderstaand gedaan.

```
hist(out, col = grey(seq(0, 1, by = 0.1)), mfrow = c(3,4))
```

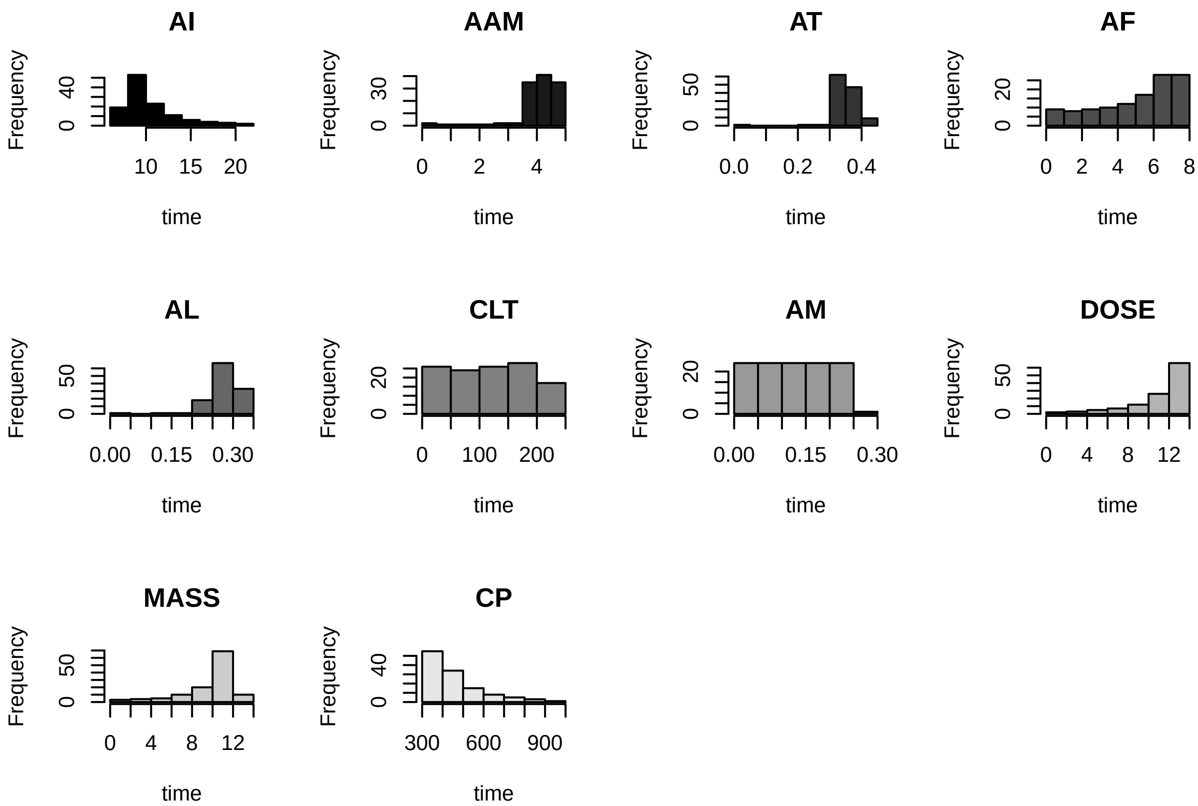

Figure 6: histogram van alle initiële waardes

In deze figuren is het mogelijk om individueel te kijken naar de initiële waardes. Je kunt bijvoorbeeld duidelijk zien dat de dose hoger wordt. Dit klopt, want CCL4 wordt gemetaboliseerd in verloop van tijd. De dose stijgt dus.