

Glucocorticoid receptor dynamics, a closer look

L T Stein & M van de Streek

2023-05-22

Contents

1	Afkortingen	3
2	Introductie	4
2.1	Doel	4
2.2	Theorie	5
3	Methode	7
3.1	Het software model	7
3.2	Model configuratie	10
4	Resultaten	13
4.0.1	Median experimental data (1.1)	13
4.0.2	Adjusting D value (1.2)	16
4.0.3	Results in line with experiment (1.3)	20
4.0.4	No auto-regulation (2.1)	20
4.0.5	Steady state (2.2)	20
4.0.6	Changing Kre/Kon (2.3)	24
4.0.7	Blocking synthesis (2.4)	26
4.0.8	Changing Ks_Rm en Kd_Rm (2.5)	27
5	Discussie en Conclusie	33
5.1	Discussie	33
5.2	Algemene conclusie en perspectief	33

1 Afkortingen

- Glucocorticoïdereceptor (GR)
- Methylprednisolon (MPL)
- Receptor (R)
- MPL (D)
- MPL-receptor complex (DR)
- Geactiveerd of in de celkern MPL-receptor complex (DR_N)

2 Introductie

In dit verslag zal er nader gekeken worden naar de werking van glucocorticosteroiden. Er zal worden gekeken naar de dynamica hiervan. Het model wat gebruikt wordt, zal nader worden uitgelegd in de theorie. De belangrijkste termen die hier voorbij zullen komen worden alvast kort uitgelegd.

Glucocorticosteroiden (D) zijn hormonen die worden geproduceerd in de bijnierschors. Het is een samenstelling van glucose, cortex en steroïde. glucocorticosteroiden worden ook vaak gebruikt in de remming van ontstekingen. De glucocorticosteroiden binden aan receptoren (R), er ontstaat dan een glucocorticoidereceptorcomplex (DR). Dit complex verplaatst naar de celkern en bindt hier aan het DNA (DR_N). De transcriptie van genen die ontstekingen veroorzaken worden tegengehouden door de complexen. Tegelijkertijd worden de genen die ontsteking remmen juist gepromoveerd. Er komen dus meer eiwitten die de ontsteking onderdrukken.

Bij het onderzoek zal gebruik gemaakt worden van experimentele data. Deze data is afkomstig van ratten. De ratten zijn zeven dagen lang geïnfecteerd met methylprednisolon (een vorm van een glucocorticosteroid). De ratten zijn geïnfecteerd met twee verschillende doses: 0.1 en 0.3 mg drug/kg rat/h. Hierover zal meer verteld worden in de theorie.

Bij het nader kijken wordt er bedoeld dat er aantal situaties worden uitgewerkt. Denk hierbij bijvoorbeeld aan: wat gebeurt er met het systeem als ik een constante vijf keer zo groot maak? Ook wordt er gekeken naar het verloop van het systeem bij een doseringstop na een bepaalde tijd. Al deze situaties worden vergeleken met de experimentele data.

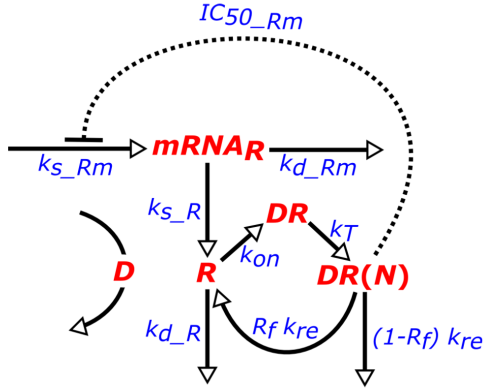
2.1 Doel

Het doel van dit onderzoek is om erachter te komen welke elementen belangrijk zijn in het model. Dus de elementen die van grote invloed zijn op bepaalde waarden. Aan de hand hiervan kan er ook iets gezegd worden over de hoeveelheid drug die kan toe worden gediend.

Dit doel kan bereikt worden door aanpassingen te doen aan het model en vervolgens te simuleren. Deze simulaties kunnen vergeleken worden met de experimentele data. Aan de hand hiervan kan worden vastgesteld of een aanpassing/weglating van een element in het model van grote invloed is.

Er zijn een aantal verwachtingen bij het model. Allereerst wordt er verwacht dat het aantal receptoren een belangrijke factor zal zijn. Hier bindt de drug aan en zal dus belangrijk zijn. Verder zal ook het aantal drug een belangrijke rol spelen. Zoals eerder vermeld werden de ratten geïnfecteerd met een dosis van 0.1 (laag) en 0.3 (hoog). De verwachting is dat ergens tussen deze waarden een ideale dosering ligt.

2.2 Theorie



Afbeelding 1: *Biologisch model van Glucocorticoid receptor dynamica*

In afbeelding 1 is het biologische model van glucocorticoïde dynamica te zien. Hierbij zijn veel constanten betrokken, die invloed uitoefenen op de hoeveelheid van het eindproduct. Deze worden verder op in dit hoofdstuk beschreven. “Glucocorticoids reverse this effect, resulting in rapid degradation of mRNA and reduced inflammatory protein secretion” [1].

Het artikel beschrijft dat glucocorticoïden (drug) zorgt voor degradatie, ofwel vermindering van mRNA, wat resulteert in ontbrekende eiwitten betrokken bij ontstekingsreacties. Het biologische model werkt deze constatering uit door naast de formule dat verandering in mRNA berekent over tijd, ook drie andere formules geeft die helpen bij het argumenteren wat er daadwerkelijk gebeurt met de receptoren die worden aangemaakt, het MPL-complex en vervolgens wat er dan gebeurt wanneer het MPL-complex in de celkern bevindt. In hoofdstuk Resultaten, is goed te zien hoe alle formules van elkaar afhangen.

In het biologische model wordt de samenhang van vier vergelijkingen getoond, waar drug (D) reageert met receptor (R) tot een Glucocorticoidereceptorcomplex (DR), waar D dus glucocorticoïde is. In totaal zijn er veertien variabelen betrokken waar dieper op in wordt gegaan bij hoofdstuk “Model configuratie”.

Elke formule hieronder beschreven hangt af van aanmaak en afbraak. Aan de linker kant wordt er stof X in context Y aangemaakt, wat afhankelijk is van een aantal parameters en daartegenover is aan de rechter kant van de minus de hoeveelheid dat ervan afgebroken wordt.

De formule waar in dit onderzoek interesse op ligt, berekent de verandering in hoeveelheid mRNA dat getranscribeerd wordt tot receptoren afhankelijk van de verandering in tijd. Aan de linkerkant van de formule wordt mRNA_R aangemaakt: mRNA dat codeert voor receptoren. Snelheidsconstante ks_Rm beïnvloed het tempo hiervan. Daartegenover beïnvloedt kd_Rm aan de rechterkant de afbraak.

$$\frac{dmRNA_R}{dt} = ks_Rm \cdot \left(1 - \frac{DR(N)}{IC50_{Rm} + DR(N)}\right) - kd_Rm \cdot mRNA_R$$

De volgende formule bepaald hoe de verandering in vrije receptoren afhankelijk is van de verandering in tijd.

Vervolgens is in de tweede formule op receptor niveau de snelheidsconstante ks_R voor aanmaak van receptoren. Constante k_{on} bepaalt de snelheid dat een MPL-receptor complex (DR) gemaakt, wat dus R kost en kd_R geeft het tempo van afbraak van receptoren aan.

$$\frac{dR}{dt} = k_{s_R}.mRNA_R + Rf.k_{re}.DR(N) - k_{on}.D.R - k_{d_R}R$$

Verder wordt er gekeken naar hoe de verandering in glucocorticoïde (D) dat reageert met het aantal receptoren (R) tot DR afhankelijk is van de verandering in tijd. Aan de linkerkant is de productiesnelheid van DR afhankelijk van k_{on} en rechts beïnvloedt k_T de vermindering van DR in de omgeving. K_T geeft het tempo aan dat het MPL-receptor complex verplaatst richting de nucleus, wat betekent dat de concentratie DR in de omgeving omlaag gaat.

$$\frac{dDR}{dt} = k_{on}.D.R - k_T.DR$$

Als laatste wordt de verandering in hoeveelheid DR in de nucleus (DR(N)) gezet tegen verandering in tijd. De concentratie DR(N) in de nucleus neemt toe door k_T . Vervolgens worden er MPL-receptoren van het complex afgehaald. K_{re} geeft het tempo aan, waarmee deze receptoren terug worden getransporteerd naar het cytosol.

$$\frac{dDR(N)}{dt} = k_T.DR - k_{re}.DR(N)$$

In dit onderzoek worden er doses van methylprednisolon (MPL) toegepast van $D_1 = 0.1$ en $D_2 = 0.3$ mg drug/kg rat/h. De gemeten concentraties van deze dosissen worden gebruikt met $D_1 = 14.59$ ng/ml en $D_2 = 39.925$ ng/ml, maar deze moeten eerst nog worden omgerekend naar nmol / L. Daarop sluiten de volgende berekeningen. D_1 wordt $(14.59.1000)/374.471$ en geeft 39.0 nmol / L. D_2 wordt $(39.925.1000)/374.471$ en geeft 107 nmol / L.

3 Methode

3.1 Het software model

Om de dynamica van de systemen eenvoudig weer te geven, moeten de differentiaal vergelijkingen worden uitgevoerd met verschillende waarden. Dit zal gedaan worden met het deSolve pakket [3] (versie 1.35) in R (versie 4.2.2).

Er worden twee scenario's behandeld. Er wordt allereerst gekeken naar de validiteit van het model. Vervolgens zal er naar simulatie scenario's worden gekeken, d.w.z. wat gebeurt er met het systeem als een waarde wordt aangepast of weggelaten.

Bij beide scenario's zijn ook vragen. De antwoorden hierop worden in de resultaten sectie gegeven. Hier worden ook alle grafieken geplott.

```
# Inladen van het deSolve pakket
library(deSolve)

# Inladen van de experiment data
experiment.data <- read.csv("../MPL.csv", header = T, sep = ",")
medians <- aggregate(experiment.data[,c("MPL_conc", "mRNA", "Free_receptor")],
                      list(experiment.data$dose, experiment.data$time),
                      median, na.rm=TRUE)

# Goede namen geven aan de kolommen
names(medians)[1:2] <- c("dose", "time")
```

Onderstaand een functie die wordt gebruikt voor een legenda bij elke grafiek.

```
legend.graph <- function(){
  return(
    legend("topright",
           legend = c("Model", "Median 0.1", "Median 0.3", "Experiment"),
           col = c("#21A9AD", "darkred", "darkgreen", "grey"),
           pch = c(NA, NA, NA, 1),
           lwd = c(2, 2, 2, NA)
    )
  )
}
```

Functie die de verandering berekend. Deze zal worden gebruikt voor scenario één.

```
verandering_dt.1 <- function (time, parameters, initial.values, D) {  
  with(as.list(c(parameters, initial.values)),{  
    # Vergelijking van mRNAr  
    mRNAr.dt <- Ks_Rm * (1 - (DR_N / (IC50_Rm + DR_N))) - Kd_Rm * mRNAr  
    # Vergelijking van R  
    R.dt <- Ksr * mRNAr + RF * Kre * DR_N - Kon * D * R - Kd_R * R  
    # Vergelijking van DR  
    DR.dt <- Kon * D * R - KT * DR  
    # Vergelijking van DR(N)  
    DR_N.dt <- KT * DR - Kre * DR_N  
  
    # Retourneren van de initiële waarden  
    return(list(c(DR_N.dt, mRNAr.dt, R.dt, DR.dt)))  
  })  
}
```

Onderstaand de functie die wordt gebruikt bij vraag 1 van scenario twee.

```
verandering_dt.2.1 <- function (time, parameters, initial.values, D) {  
  with(as.list(c(parameters, initial.values)),{  
    # Vergelijking van mRNAr  
    mRNAr.dt <- Ks_Rm * (1 - (DR_N / (IC50_Rm + DR_N))) - Kd_Rm * mRNAr  
    # Vergelijking van R  
    R.dt <- Ksr * mRNAr + RF * Kre * DR_N - Kon * R - Kd_R * R  
    # Vergelijking van DR  
    DR.dt <- Kon * D * R - KT * DR  
    # Vergelijking van DR(N)  
    DR_N.dt <- KT * DR - Kre * DR_N  
  
    # Retourneren van de initiële waarden  
    return(list(c(DR_N.dt, mRNAr.dt, R.dt, DR.dt)))  
  })  
}
```


Bij vraag 2 van scenario twee wordt er gekeken wat er gebeurt met het verloop als de toevoeging van D stopt na verloop van tijd. Onderstaand wordt hiervoor de gebruikte functie gedefinieerd.

```
verandering_dt.2.2 <- function (time, parameters, initial.values, D) {
  with(as.list(c(parameters, initial.values)),{
    # D steady
    if (time>35) {
      D <- 0
    }
    # Vergelijking van mRNAr
    mRNAr.dt <- Ks_Rm * (1 - (DRN / (IC50_Rm + DRN))) - Kd_Rm * mRNAr
    # Vergelijking van R
    R.dt <- Ksr * mRNAr + RF * Kre * DRN - Kon * D * R - Kd_R * R
    # Vergelijking van DR
    DR.dt <- Kon * D * R - KT * DR
    # Vergelijking van DR(N)
    DRN.dt <- KT * DR - Kre * DRN

    # Retourneren van de initiële waarden
    return(list(c(DRN.dt, mRNAr.dt, R.dt, DR.dt)))
  })
}
```

Voor vraag 4 van scenario twee wordt er gekeken naar het stopzetten van de synthese van R. Onderstaand de juiste functie en berekening.

```
verandering_dt.2.4 <- function (time, parameters, initial.values, D, Ks_Rm) {
  with(as.list(c(parameters, initial.values)),{
    # Vergelijking van mRNAr
    mRNAr.dt <- Ks_Rm * (1 - (DRN / (IC50_Rm + DRN))) - Kd_Rm * mRNAr
    # Vergelijking van R
    R.dt <- Ksr * mRNAr + RF * Kre * DRN - Kon * D * R - Kd_R * R
    # Vergelijking van DR
    DR.dt <- Kon * D * R - KT * DR
    # Vergelijking van DR(N)
    DRN.dt <- KT * DR - Kre * DRN

    # Retourneren van de initiële waarden
    return(list(c(DRN.dt, mRNAr.dt, R.dt, DR.dt)))
  })
}
```

3.2 Model configuratie

```
# Pakket inladen voor mooiere tabel
library(pander)
```

Tabel 1: Variabelen betrokken bij opzetten experiment opgedeeld in zeven konstantes, drie parameters en vier initiële waarden.

```
# Drie categorieën aan soorten input waarden
Categorie <- c(rep("Initiële waarden", 4),
               rep("Parameters", 10))

# Waarden uit bovenstaande categorieën verkregen van experiment
# Omdat DR een reactie is tussen D en R gaan de waarden keer elkaar
Waarden<- c("mRNAr"=4.74, "DR_N"=0, "R"=267, "DR"=0, "ks_Rm"=2.90,
            "kd_Rm"= 0.612, "k_re"=0.57, "k_T"=0.63, "k_on"= 0.00329,
            "k_d_R"=0.0572, "k_s_r"=3.22, "IC50_Rm"=26.2, "RF"=0.49, "D"=53.4)

# Eenheden in een vector verwerkt. Met functie 'rep()' kan een iterable X
# keer herhaald worden.
Eenheden <- c("fmol / g liver", rep("fmol/mg protein", 3),
              rep("fmol/g liver/h", 2), rep("1 / h", 2),
              "L/nmol/h", rep("1 / h", 2), "fmol/mg protein",
              "", "nmol/L")

# Dataframe waar vector namen in de header komen, waarde namen als rijnamen
# en de daadwerkelijke waarden daarin verwerkt.
model.conf <- data.frame(Waarden, Categorie, Eenheden)
# Tekenen van tabel
pander(model.conf)
```

	Waarden	Categorie	Eenheden
mRNAr	4.74	Initiële waarden	fmol / g liver
DR_N	0	Initiële waarden	fmol/mg protein
R	267	Initiële waarden	fmol/mg protein
DR	0	Initiële waarden	fmol/mg protein
ks_Rm	2.9	Parameters	fmol/g liver/h
kd_Rm	0.612	Parameters	fmol/g liver/h
k_re	0.57	Parameters	1 / h
k_T	0.63	Parameters	1 / h
k_on	0.00329	Parameters	L/nmol/h
k_d_R	0.0572	Parameters	1 / h
k_s_r	3.22	Parameters	1 / h
IC50_Rm	26.2	Parameters	fmol/mg protein
RF	0.49	Parameters	
D	53.4	Parameters	nmol/L

Zoals eerder vermeld worden er twee situaties uitgewerkt. Er wordt allereerst gekeken naar de validiteit van het model. Hierbij worden een aantal vragen beantwoord. De antwoorden op de vragen bevinden zich in de resultaten sectie. Er moet wel veel verschuiving in de parameters plaatsvinden, dit wordt allemaal in de onderstaande alinea('s) weergegeven.

Hieronder staan de parameters die gebruikt zijn bij scenario één. Wat belangrijk is om op te merken is dat D niet in de vectoren wordt verwerkt, omdat de concentraties in het onderzoek variëren van 53.4 tot 39 (dosis 0.1) tot 107 (dosis 0.3). Als oplossing worden de concentraties direct aan functie ode() gegeven.

```
# Definieren van de parameters
parameters.1 <- c(Ks_Rm = 2.90, IC50_Rm = 26.2,
                  Kon = 0.00329, KT = 0.63,
                  Kre = 0.57, RF = 0.49,
                  Kd_R = 0.0572, Kd_Rm = 0.612,
                  Ksr = 3.22)

# 53.4
# Tijdstippen definiëren
times.1 <- 0 : 200

# Het definiëren van de initiële waarden
initial.values.1 <- c(DR_N = 0, mRNAr = 4.74, R = 267, DR = 0)
```

tijdverloop voor vraag 1 van scenario twee.

```
times.2.1 <- 0 : 48
```

Bij scenario twee (simulatie scenario's) zijn veel verschillende parameters gebruikt. In opdracht drie blijven de waarden van tijd en D hetzelfde. Echter veranderen parameters Kre en Kon bij concentraties R en mRNAr. Per concentratie zullen de concentraties dissociëren en associëren met vijf en twee door de volgende formules. Waar dissociatie wordt bepaald door deling met vijf en twee, associatie wordt bepaald door vermenigvuldiging met twee en vijf.

```
# Kon / 5
parameters.1["Kon"] = 0.00329 / 5
dataset.kon.g5 <- ode(initial.values.1, times.1,
                      verandering_dt.1, parameters.1, D = 53.4)

# Kon / 2
parameters.1["Kon"] = 0.00329 / 2
dataset.kon.g2 <- ode(initial.values.1, times.1,
                      verandering_dt.1, parameters.1, D = 53.4)

# Kon * 5
parameters.1["Kon"] = 0.00329 * 5
dataset.kon.k5 <- ode(initial.values.1, times.1,
                      verandering_dt.1, parameters.1, D = 53.4)

# Kon * 2
parameters.1["Kon"] = 0.00329 * 2
dataset.kon.k2 <- ode(initial.values.1, times.1,
                      verandering_dt.1, parameters.1, D = 53.4)

# Kre variabele
# Kre / 5
parameters.1["Kre"] = 0.57 / 5
dataset.kre.g5 <- ode(initial.values.1, times.1,
                      verandering_dt.1, parameters.1, D = 53.4)

# Kre / 2
parameters.1["Kre"] = 0.57 / 2
dataset.kre.g2 <- ode(initial.values.1, times.1,
```

```

        verandering_dt.1, parameters.1, D = 53.4)

# Kre * 5
parameters.1["Kre"] = 0.57 * 5
dataset.kre.k5 <- ode(initial.values.1, times.1,
        verandering_dt.1, parameters.1, D = 53.4)

# Kre * 2
parameters.1["Kre"] = 0.57 * 2
dataset.kre.k2 <- ode(initial.values.1, times.1,
        verandering_dt.1, parameters.1, D = 53.4)

```

In opgave vier blijft de tijdsequentie en D hetzelfde, maar Ks_Rm wordt gezet naar nul. Dit zorgt ervoor dat de synthese van R tot stop wordt gezet.

```

Ks_Rm <- 0
D <- 53.4
times <- 0:48

```

In opgave vijf veranderen de parameters Ks_Rm en Kd_Rm op basis van toename (keersom) van twee en vijf of een afname (deling) van twee en vijf. Echter is Kd_Rm bij iedere toename of afname nog afhankelijk van de hoeveelheid mRNA_r vanaf t = 0, wat 4.74 is. Door het veranderen van deze snelheidsconstanten wordt de aanmaak van mRNA_r beïnvloed.

```

parameters.2 <- c(Ks_Rm = 2.90, IC50_Rm = 26.2,
        Kon = 0.00329, KT = 0.63,
        Kre = 0.57, RF = 0.49,
        Kd_R = 0.0572, Kd_Rm = 0.612,
        Ksr = 3.22)

parameters.2["Ks_Rm"] = 2.9 / 5
parameters.2["Kd_Rm"] = 2.9 / 5 / 4.74
dataset.g5 <- ode(initial.values.1, times.2.1, verandering_dt.2.4, parameters.2, D = 53.4)

parameters.2["Ks_Rm"] = 2.9 / 2
parameters.2["Kd_Rm"] = 2.9 / 2 / 4.74
dataset.g2 <- ode(initial.values.1, times.2.1, verandering_dt.2.4, parameters.2, D = 53.4)

parameters.2["Ks_Rm"] = 2.9 * 5
parameters.2["Kd_Rm"] = 2.9 * 5 / 4.74
dataset.k5 <- ode(initial.values.1, times.2.1, verandering_dt.2.4, parameters.2, D = 53.4)

parameters.2["Ks_Rm"] = 2.9 * 2
parameters.2["Kd_Rm"] = 2.9 * 2 / 4.74
dataset.k2 <- ode(initial.values.1, times.2.1, verandering_dt.2.4, parameters.2, D = 53.4)

```

4 Resultaten

In deze sectie zal er worden gekeken naar de resultaten. Omdat er vragen worden beantwoordt, zal per code worden aangegeven bij de titel om welke vraag(nummer) het gaat.

4.0.1 Median experimental data (1.1)

```
verloop.dataset <- ode(initial.values.1, times.1,
                        verandering_dt.1, parameters.1, D = 53.4)
verloop.dataset.lage.doses <- ode(initial.values.1,
                                   times.1, verandering_dt.1, parameters.1, D = 39)
verloop.dataset.hoge.doses <- ode(initial.values.1,
                                   times.1, verandering_dt.1, parameters.1, D = 107)

# mRNA plot - experiment - model

plot(verloop.dataset[, "mRNAr"],
     type="l",
     main = "Verloop van mRNAr (zowel model als experiment)",
     xlab="Time", ylab="mRNAr",
     lwd = 2, ylim = c(0, 8), xlim = c(0, 150),
     col = "#21A9AD")

lines(medians$mRNA[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
      type = "l", lwd = 2, col = "darkred")

lines(medians$mRNA[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      type = "l", lwd = 2, col = "darkgreen")

points(experiment.data$mRNA,
       col = "black", pch = 1, cex = .7)

legend.graph()

# Free receptor R plot - experiment - model

plot(verloop.dataset[, "R"],
     type="l",
     main = "Verloop van R (zowel model als experiment)",
     xlab="Time", ylab="vrije receptor (R)",
     lwd = 2, ylim = c(0, 500), xlim = c(0, 150),
     col = "#21A9AD")

lines(medians$Free_receptor[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
      type = "l", lwd = 2, col = "darkred")

lines(medians$Free_receptor[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      type = "l", lwd = 2, col = "darkgreen")

points(experiment.data$Free_receptor,
       col = "black", pch = 1, cex = .7)

legend.graph()
```

Verloop van mRNA_r (zowel model als experiment)

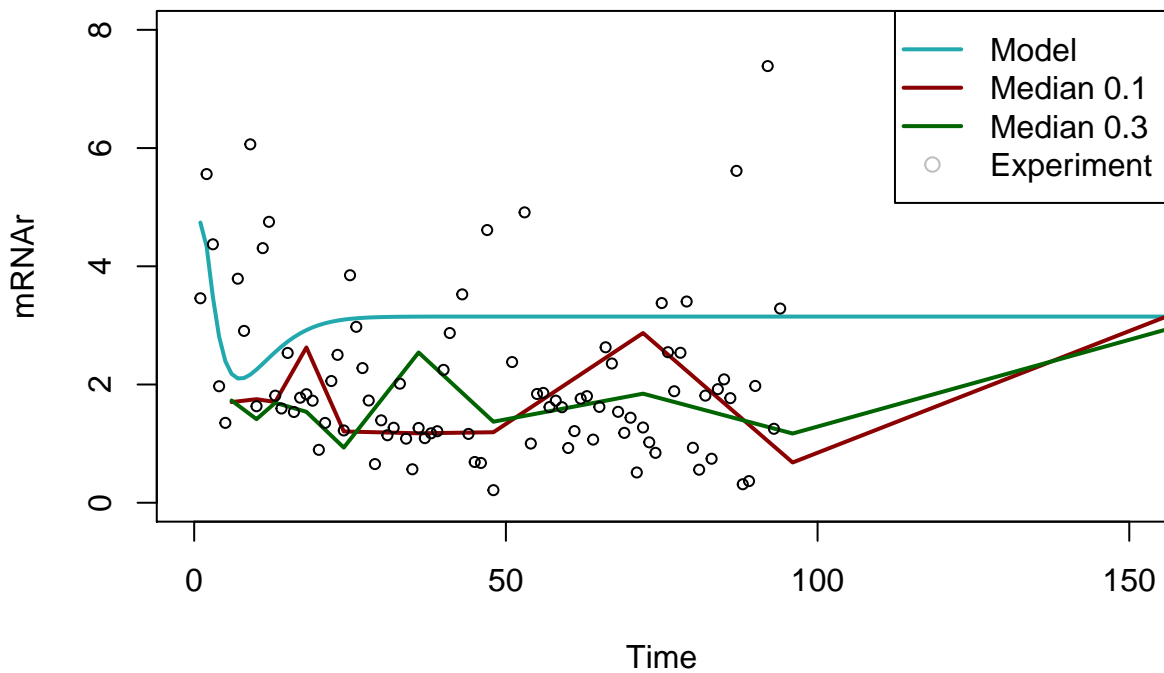


Figure 1: De bovenstaande figuur geeft het model en de experimentele waardes weer. De verloop van het model is een vloeiende lijn die eerste daalt en vervolgens weer iets stijgt en naar een constante waarde gaat. De rode lijn geeft de experimentele data weer van de lage dosis. De groene lijn geeft de hoge dosis weer. Beide lijnen zijn schokkerige lijnen. De waardes van de experimentele data liggen iets onder de model waardes. De lijnen voor de vrije receptor (figuur 2) zitten ongeveer op dezelfde waardes, halverwege vertoont het experiment een kleine piek.

Verloop van R (zowel model als experiment)

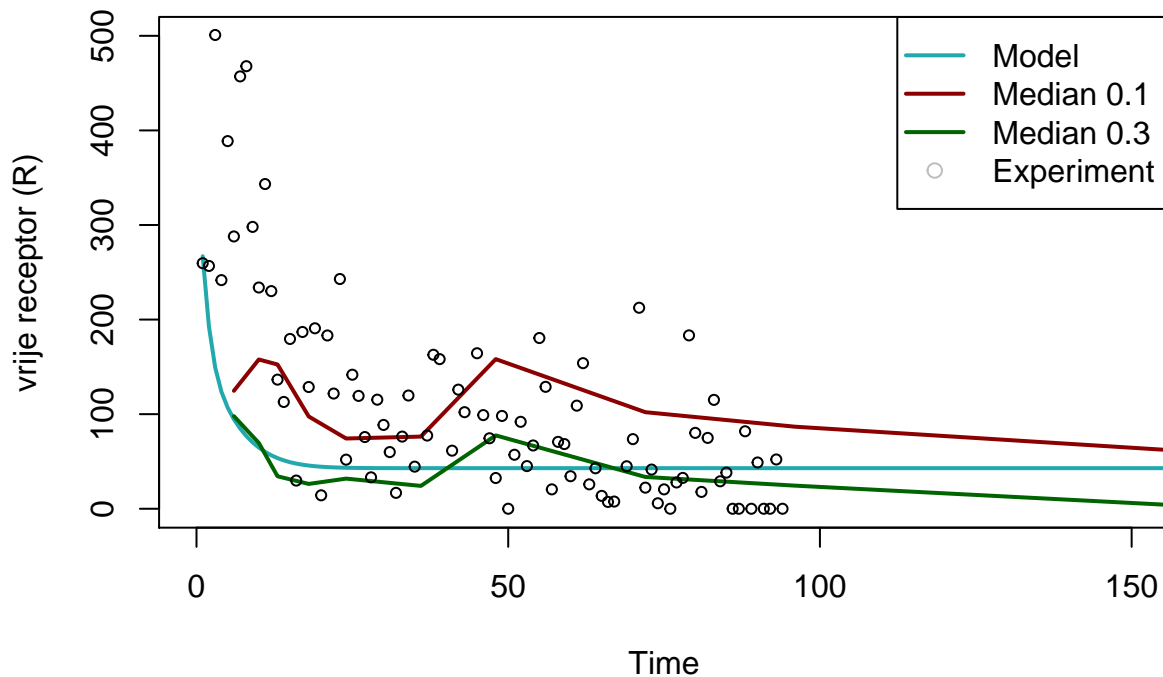


Figure 2: De bovenstaande figuur geeft het model en de experimentele waardes weer. De verloop van het model is een vloeiende lijn die eerste daalt en vervolgens weer iets stijgt en naar een constante waarde gaat. De rode lijn geeft de experimentele data weer van de lage dosis. De groene lijn geeft de hoge dosis weer. Beide lijnen zijn schokkerige lijnen. De waardes van de experimentele data liggen iets onder de model waardes. De lijnen voor de vrije receptor (figuur 2) zitten ongeveer op dezelfde waardes, halverwege vertoont het experiment een kleine piek.

Door het plotten van een mediaan wordt het midden van de data gevisualiseerd. De helft van de data ligt boven de lijn en de andere helft eronder. De data wordt niet naar beneden (getrokken) door uitschieters, daarom is het een mooie methode om het verloop te zien.

4.0.2 Adjusting D value (1.2)

```
par(mfrow = c(1,2))

# Verloop van mRNAr

plot(verloop.dataset.lage.doses[, "mRNAr"],
     main = "mRNAr met lage dosis",
     type="l", xlab="Time", ylab="mRNAr",
     lwd = 2, xlim = c(0, 150), ylim = c(0, 8), col = "#21A9AD")

lines(medians$mRNA[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
      type = "l", lwd = 2, col = "darkred")

lines(medians$mRNA[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      type = "l", lwd = 2, col = "darkgreen")

points(experiment.data$mRNA, col = "black", pch = 1, cex = .7)

legend.graph()

plot(verloop.dataset.hoge.doses[, "mRNAr"],
     main = "mRNAr met hoge dosis",
     type="l", xlab="Time", ylab="Free receptor",
     lwd = 2, xlim = c(0, 150), ylim = c(0, 8),
     col = "#21A9AD")

lines(medians$mRNA[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
      type = "l", lwd = 2, col = "darkred")

lines(medians$mRNA[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      type = "l", lwd = 2, col = "darkgreen")

points(experiment.data$mRNA, pch = 1, cex = .7,
      col = "black")

legend.graph()

# Verloop bij vrije receptoren

par(mfrow = c(1,2))
plot(verloop.dataset.lage.doses[, "R"],
     main = "Free receptor (R) met lage dosis",
     type="l", xlab="Time",
     ylab="Free receptor",
     lwd = 2, xlim = c(0, 140),
     ylim = c(0, 500),
     col = "#21A9AD")

lines(medians$Free_receptor[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
```

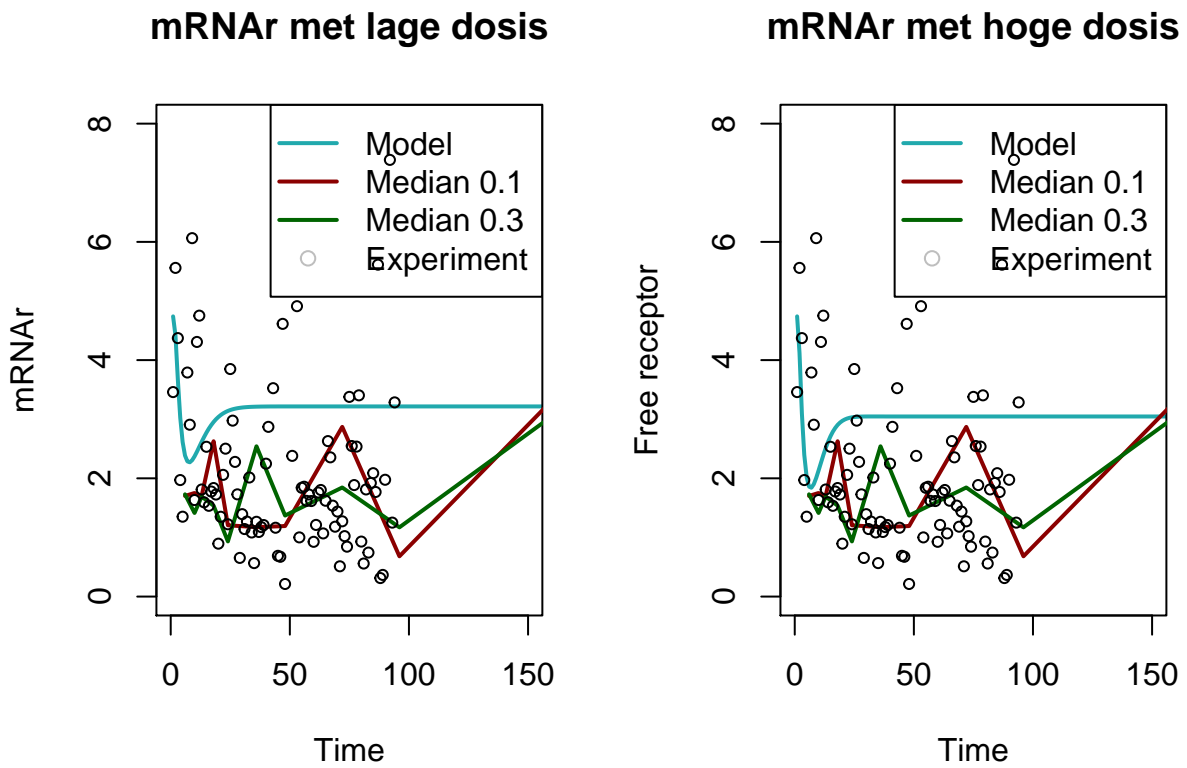



Figure 3: Voor figuur 3 en 4 geldt hetzelfde verhaal als voor figuur 1 en 2. Er wordt nu voor zowel mRNA als R een grafiek weergegeven voor een lage dosis D en een hoge dosis D . Bij de grafieken van R zijn de pieken halverwege hoger

```

    type = "l",
    lwd = 2,
    col = "darkred")

lines(medians$Free_receptor[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      type = "l",
      lwd = 2,
      col = "darkgreen")

points(experiment.data$Free_receptor,
       col = "black",
       pch = 1,
       cex = .7)

points(medians$MPL_conc[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
      lwd = 2, col = "orange")

legend.graph()

plot(verloop.dataset.hoge.doses[, "R"],
     main = "Free receptor (R) met hoge dosis",
     type = "l",
     xlab = "Time",
     ylab = "Free receptor",
     lwd = 2, xlim = c(0, 140),
     ylim = c(0, 500),
     col = "#21A9AD")

lines(medians$Free_receptor[medians$dose == 0.1] ~ medians$time[medians$dose == 0.1],
      type = "l",
      lwd = 2, col = "darkred")

lines(medians$Free_receptor[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      type = "l",
      lwd = 2,
      col = "darkgreen")

points(experiment.data$Free_receptor,
       col = "black",
       pch = 1,
       cex = .7)

points(medians$MPL_conc[medians$dose == 0.3] ~ medians$time[medians$dose == 0.3],
      lwd = 1, col = "orange", type = "b")

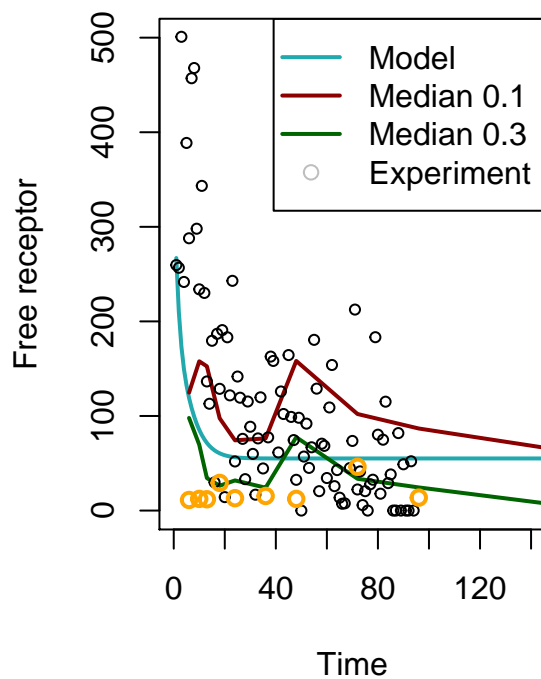
legend.graph()

```

In het verloop van mRNA is weinig verandering te zien afhankelijk van de dosis. Er is weliswaar bij een hoge dosis een lager aantal mRNA, dit is absoluut niet significant. Verder is er wel een groot verschil bij het aantal vrije receptoren door een hogere dosis drug (D).

Dit is goed te verklaren doordat bij een hogere hoeveelheid er meer MPL bindt aan de receptoren, dus dit betekent dat de concentratie van het MPL-complex omhoog gaat.

Free receptor (R) met lage dosis



Free receptor (R) met hoge dosis

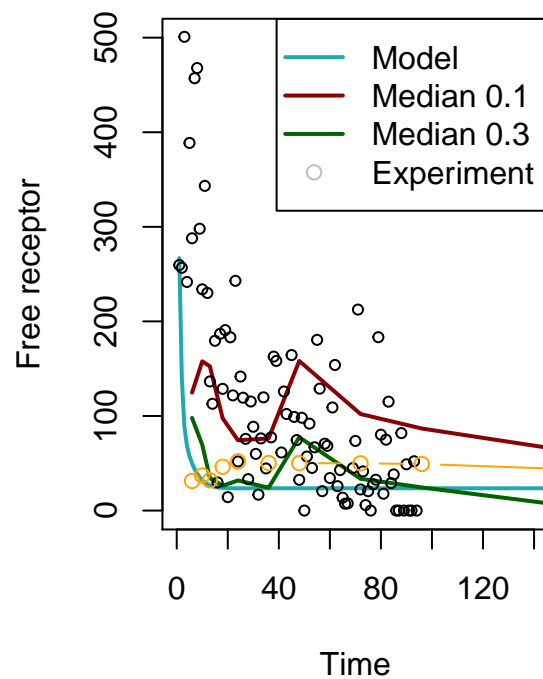


Figure 4: Voor figuur 3 en 4 geldt hetzelfde verhaal als voor figuur 1 en 2. Er wordt nu voor zowel mRNA als R een grafiek weergegeven voor een lage dosis D en een hoge dosis D . Bij de grafieken van R zijn de pieken halverwege hoger

4.0.3 Results in line with experiment (1.3)

De resultaten van het model liggen over het algemeen op een goede lijn met de experimentele data. Bij het aantal vrije receptoren ligt het verloop van het model tussen de twee medianen, dus overlapt het model goed met experimentele data. Echter is er wel een lagere werkelijkheid bij het aantal mRNA. Dat wil zeggen het aantal mRNA wordt hoger geschat in het model.

4.0.4 No auto-regulation (2.1)

```
# Het aanroepen van de ode functie, waarden worden opgeslagen
verloop.dataset.r.niet.afhankelijk <- ode(initial.values.1,
                                         times.2.1, verandering_dt.2.1, parameters.1, D=53.4)

# plotten van vier formules in vier grafieken
par(mfrow = c(1,2))
plot(verloop.dataset.r.niet.afhankelijk[, "DR_N"],
     lwd = 2.5,
     main = "Figuur A",
     xlab = "Tijd (in uren)",
     ylab = "Aantal stof (in fmol/mg eiwit)",
     col = "darkblue",
     type = "l")

plot(verloop.dataset.r.niet.afhankelijk[, "mRNAr"],
     lwd = 2.5,
     main = "figuur B",
     xlab = "Tijd (in uren)",
     ylab = "Aantal stof (in fmol/g lever)",
     col = "blue",
     type = "l")
```

In figuur 5 wordt het model geplot nadat D weggehaald wordt uit de formule van “R.dt” uit de keersom met $Kon * R$. Het doel is om te kijken wat er gebeurt met de concentratie DR_N wanneer R niet afhankelijk is aan D. We halen D uit de formule, omdat het de synthese van mRNA voor vrije receptor betreft.

Op de x-as staat de hoeveelheid stof in fmol/mg eiwit en de y-as geeft de tijd in uren. In grafiek A is het verloop van DR_N te zien, die eerst snel toeneemt en vervolgens minder snel toeneemt in een lineaire lijn. Tevens is in figuur B het verloop van mRNA receptoren te zien, waar het eerst sterk afneemt en vervolgens langzaam lineair door blijft dalen.

4.0.5 Steady state (2.2)

```
verloop.dataset.steady <- ode(initial.values.1, times.2.1,
                              verandering_dt.2.2, parameters.1, D=53.4)

plot(verloop.dataset.steady[, "DR_N"],
     type = "l", main = "Figuur C",
     xlab = "Tijd (in uren)",
     ylab = "Hoeveelheid DR_N (n)")

plot(verloop.dataset.steady[, "mRNAr"],
     type = "l", main = "Figuur D",
     xlab = "Tijd (in uren)",
     ylab = "Hoeveelheid mRNAr (n)")
```

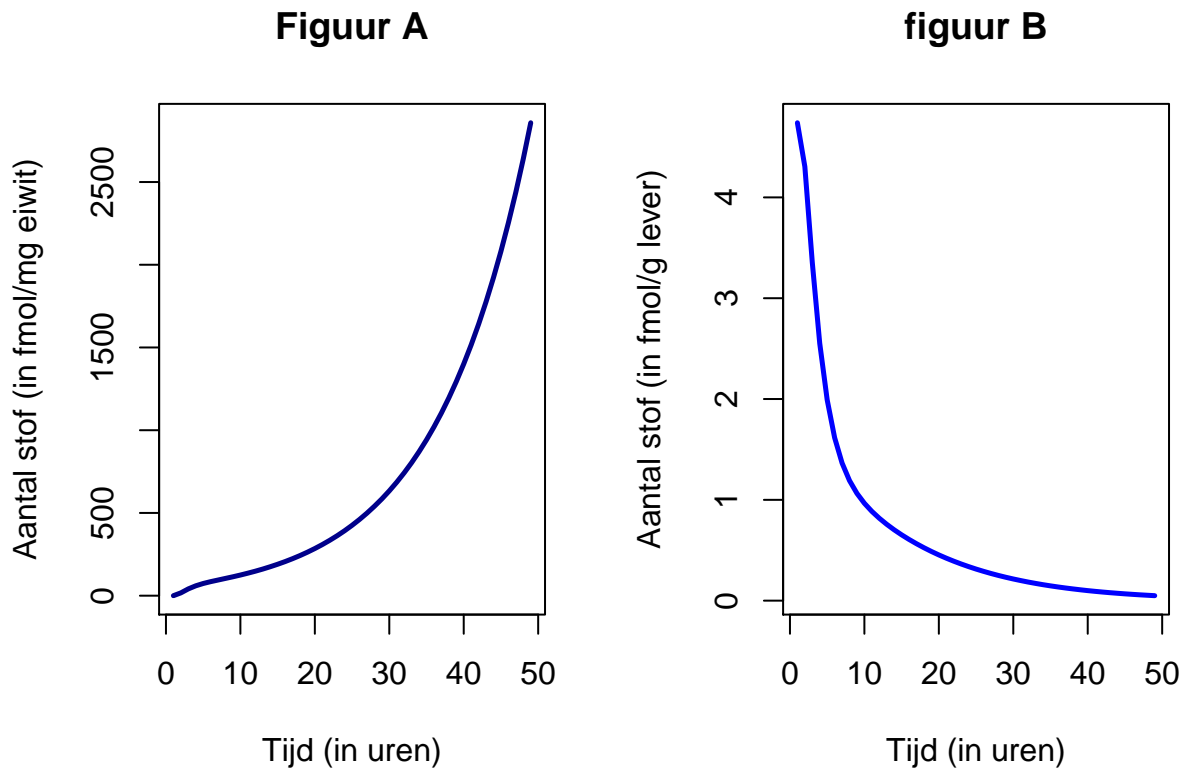


Figure 5: *Figuren die het verloop van de concentraties weergeven over tijd. Bij elke figuur is op de x-as de tijd (in uren) zichtbaar. Op de y-as staat de concentratie in de juiste eenheid. Grafiek 1 geeft het verloop van de concentratie MPL-complex in de celkern weer. Grafiek 2 geeft de concentratie mRNA receptoren weer. Grafiek 3 geeft het verloop van vrije receptoren weer. En de laatste grafiek (4) geeft het verloop van het MPL-complex weer wat niet geactiveerd is.*

Figuur C

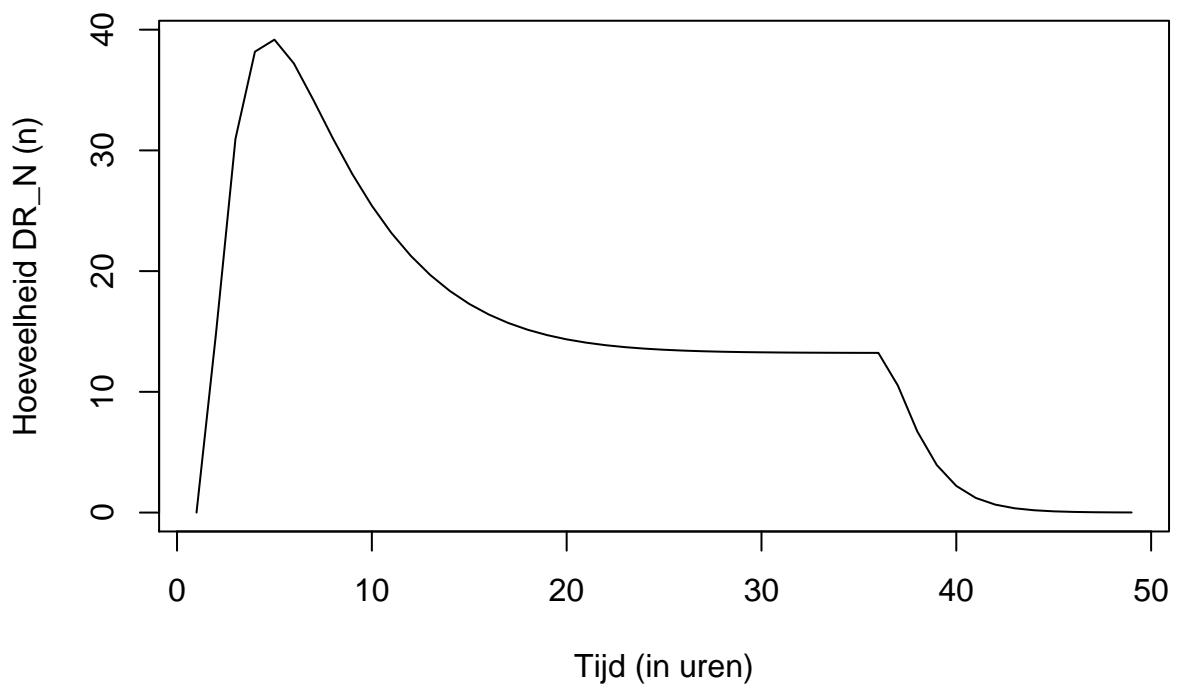


Figure 6: In figuur 6/7 is het verloop te zien van grafiek C en D (DR_N en $mRNA_r$) waarbij doses (D) naar nul gaat zodra de steady state wordt bereikt bij $t = 35$. Op de x-as staat de hoeveelheid stof in fmol/mg eiwit en de y-as geeft de tijd in uren.

Figuur D

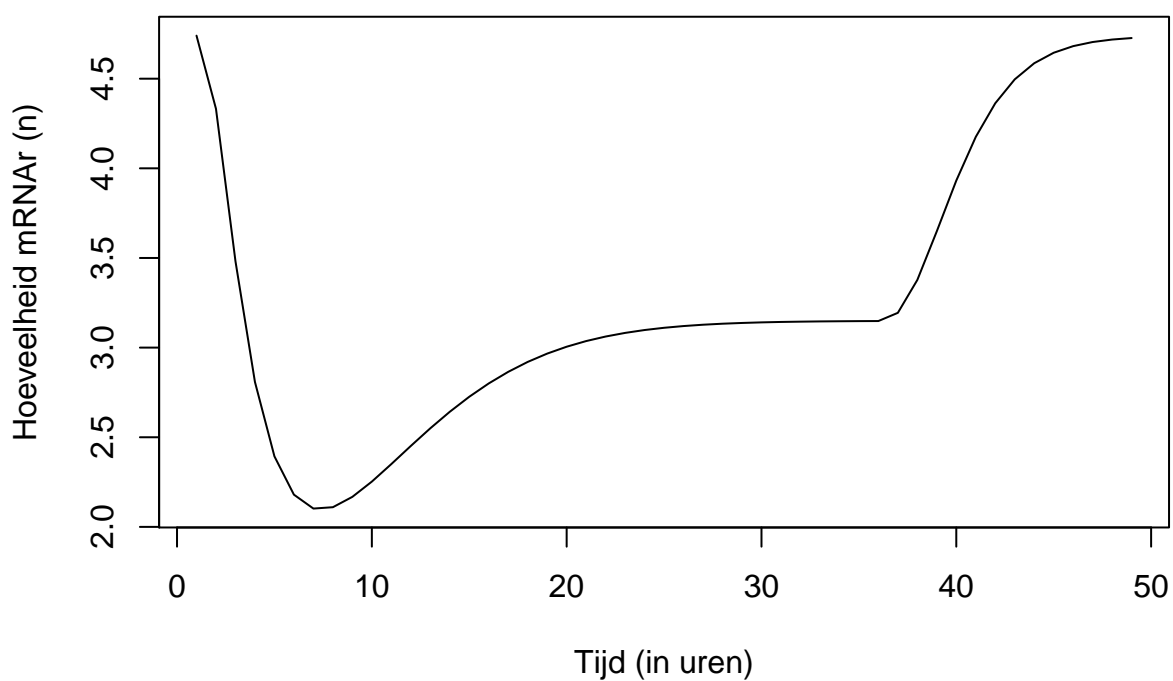


Figure 7: In figuur 6/7 is het verloop te zien van grafiek C en D (DR_N en $mRNA_r$) waarbij doses (D) naar nul gaat zodra de steady state wordt bereikt bij $t = 35$. Op de x-as staat de hoeveelheid stof in fmol/mg eiwit en de y-as geeft de tijd in uren.

Tabel 2: Tabel waarin is te zien dat er ongeveer een steady state is bereikt bij $t = 35$.

```
pander(verloop.dataset[30:40,])
```

time	DR_N	mRNAr	R	DR
29	13.27	3.141	42.95	23.99
30	13.26	3.143	42.93	23.97
31	13.25	3.145	42.92	23.96
32	13.24	3.146	42.91	23.95
33	13.23	3.147	42.91	23.94
34	13.23	3.148	42.9	23.93
35	13.23	3.148	42.9	23.93
36	13.23	3.149	42.9	23.93
37	13.22	3.149	42.9	23.93
38	13.22	3.149	42.9	23.93
39	13.22	3.149	42.9	23.93

Het is duidelijk te zien in beide grafieken wanneer de drug dosis naar 0 gaat.

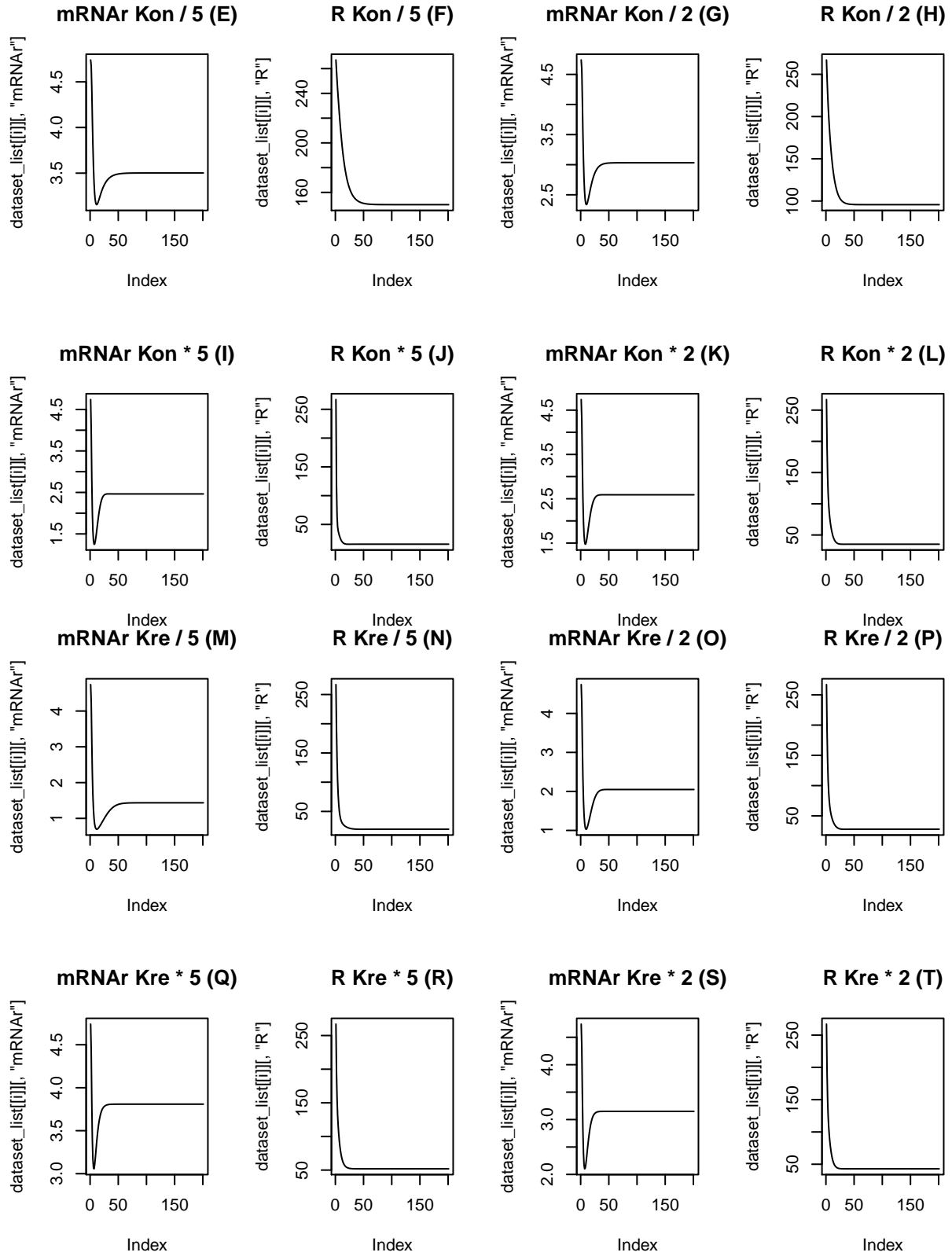
4.0.6 Changing Kre/Kon (2.3)

```
# Stel de datasetnamen in een lijst
dataset_list <- list(dataset.kon.g5, dataset.kon.g2,
                    dataset.kon.k5, dataset.kon.k2, dataset.kre.g5,
                    dataset.kre.g2, dataset.kre.k5, dataset.kre.k2)

# Stel de plotlabels in een lijst
label_list <- c("mRNAr Kon / 5 (E)", "R Kon / 5 (F)", "mRNAr Kon / 2 (G)",
               "R Kon / 2 (H)", "mRNAr Kon * 5 (I)", "R Kon * 5 (J)",
               "mRNAr Kon * 2 (K)", "R Kon * 2 (L)",
               "mRNAr Kre / 5 (M)", "R Kre / 5 (N)", "mRNAr Kre / 2 (O)" ,
               "R Kre / 2 (P)", "mRNAr Kre * 5 (Q)", "R Kre * 5 (R)",
               "mRNAr Kre * 2 (S)", "R Kre * 2 (T)")

par(mfrow = c(2, 4))

# Maak een lus om de plots te maken voor elke dataset
for (i in 1:length(dataset_list)) {
  plot(dataset_list[[i]][, "mRNAr"], type = "l", main = label_list[2*i-1])
  plot(dataset_list[[i]][, "R"], type = "l", main = label_list[2*i])
}
```

Voor zowel mRNAr en R wordt het verloop weergegeven met aangepaste parameters. De parameters 'Kon' en 'Kre' worden aangepast op de volgende manier. *Kre* of *Kon* worden gedeeld door vijf en twee, ook worden ze vermenigvuldigd met vijf en twee.

Allereerste wordt er gekeken naar de verandering in mRNAr. Bij een deling van vijf bij *Kon* daalt de grafiek sneller aan het begin tot $t = 10$. Vervolgens neemt het minder snel toe. Bij de deling door twee is er een minuscule verschil. De vermenigvuldiging van *Kon* vertoont bij vijf een snellere daling. Het tweede deel van de grafiek vertoont geen verschil met de originele waarde. Dit geldt ook bij de vermenigvuldiging van twee.

Verder is bij een deling van vijf voor *Kre* een groter verschil. De daling aan het begin lijkt identiek te zijn, maar de toename neemt minder snel toe tot bijna een rechte lijn. Bij de deling door twee is er het begin ook bijna identiek, maar is de stijging minder snel. Voor de vermenigvuldiging van *Kre* met vijf is er een sneller daling aan het begin en vervolgens een snellere stijging. De steady state is ook hoger dan in het originele model. Bij de vermenigvuldiging van twee geldt hetzelfde verhaal, maar dan in mindere mate, d.w.z. de stijging is iets minder snel dan bij de vermenigvuldiging van vijf.

Nu de mRNA behandeld is, kan er gekeken worden naar het verloop van de vrije receptoren.

Alle grafieken van R vertonen kleine veranderingen. Om te beginnen met de deling van vijf voor *Kon*. Bij deze grafiek is de afname minder snel dan bij het originele. De deling van twee vertoont bijna een identiek beeld met de deling van vijf. Voor de vermenigvuldiging geldt een omgekeerde beeld. De afname is voor zowel, vijf als twee, sneller. De vermenigvuldiging van vijf vertoont een significant sneller beeld.

Voor de correcties van *Kre* heeft de deling meer effect dan de vermenigvuldiging. Bij de vermenigvuldiging (zowel vijf als twee) ontstaan grafieken die bijna hetzelfde beeld vertonen als het originele model. Alleen de stijging is een fractie sneller. De deling komt wel een ander beeld naar voren. De stijging is bij de vermenigvuldiging van vijf een stuk sneller. De steady state een stuk lager als bij het originele model. Voor de vermenigvuldiging van twee is de stijging sneller, maar niet sneller dan die van vijf. De steady state komt ook hier lager uit.

4.0.7 Blocking synthesis (2.4)

```
dataset.geen.R.synthese <- ode(initial.values.1, times.2.1,
                               verandering_dt.2.4, parameters.1, D = 53.4, Ks_Rm = 0)
par(mfrow = c(1,2))
plot(dataset.geen.R.synthese[, "R"],
      xlab = "Tijd (in uren)",
      ylab = "hoeveelheid R",
      type = "l",
      main = "vrije R synthese (U)")

plot(dataset.geen.R.synthese[, "mRNAr"],
      xlab = "Tijd (in uren)",
      ylab = "hoeveelheid mRNAr",
      type = "l",
      main = "mRNAr productie (V)")
```

In figuur 10 is het verloop te zien van de hoeveelheid vrije R in grafiek U. Om deze grafiek te maken is parameter *Ks_Rm* naar nul veranderd. *Ks_Rm* is verantwoordelijk voor de aanmaak per tijdeenheid van mRNAr. Door deze parameter naar nul te veranderen, zal in verloop van tijd geen mRNAr meer worden aangemaakt. Dit is te zien in grafiek V waar de hoeveelheid mRNAr bij $t \sim 30$ constant blijft. In grafiek V is te zien dat de hoeveelheid mRNAr is afneemt via de y-as en over tijd zal de toename tot bijna constant verlopen.

Daarom is in grafiek U te zien dat de hoeveelheid vrije R blijft afnemen, dat verloopt tot een constante lijn bij $y \sim 40-50$ en bij $t \sim 20$. Omdat vrije R hergebruikt wordt zal er altijd een beetje aanwezig blijven en de lijn dus nooit 0 raken.

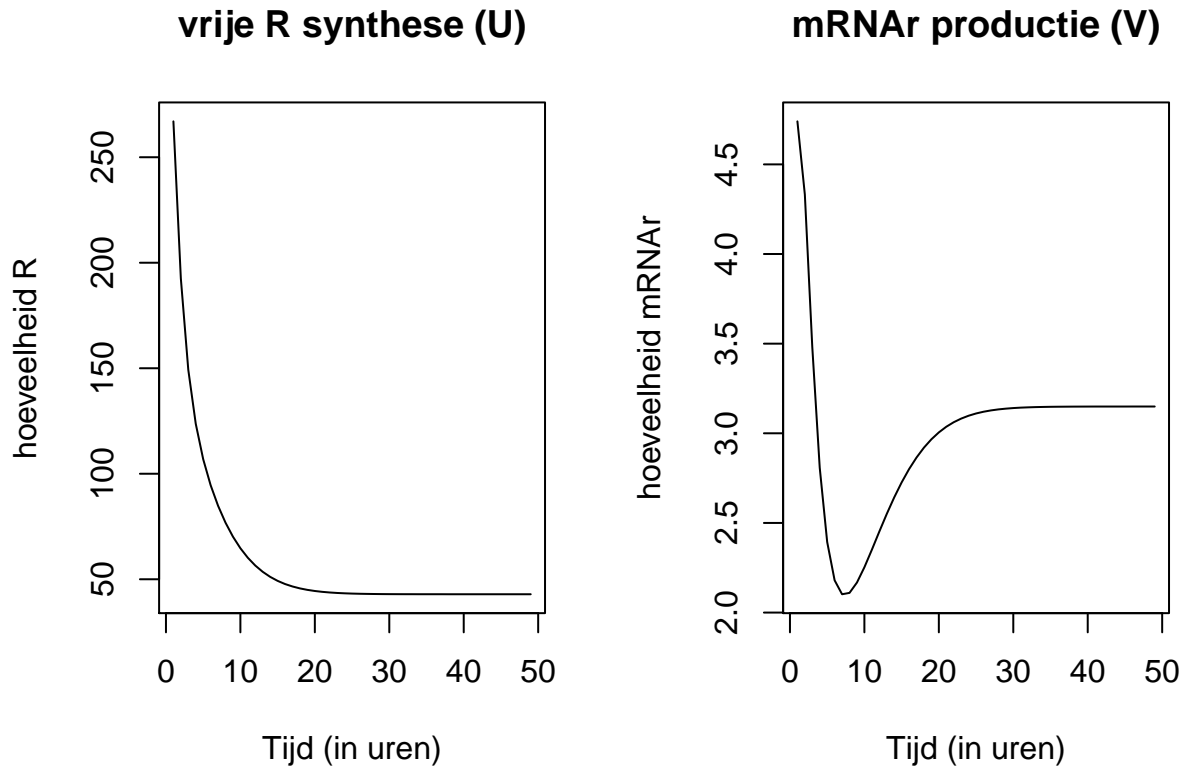


Figure 8: In figuur 10 is het verloop te zien van de hoeveelheid vrije R en mRNAr waarbij de constante kS_{rm} is veranderd naar 0. Beide grafieken lijken op het 'normale/originele model'

4.0.8 Changing Ks_{Rm} en Kd_{Rm} (2.5)

In de plotten beneden wordt er gekeken naar wat er gebeurt in het systeem, als de snelheidsconstanten Ks_{Rm} en Kd_{rm} die de aanmaak en afbraak van mRNAr beïnvloeden toenemen of afnemen.

```
plot(dataset.g2, main = c("DR_N (A)", "mRNAr (B)",
                        "R (C)", "DR (D)"),
     ylab = c("hoeveelheid DR_N (n)", "hoeveelheid mRNAr (n)",
              "hoeveelheid R (n)", "hoeveelheid DR (n)"),
     xlab = "Tijd (in uren)")
mtext("Ks_Rm en Kd_Rm gedeeld door vijf",
     side = 3,
     line = - 1,
     outer = TRUE)
```

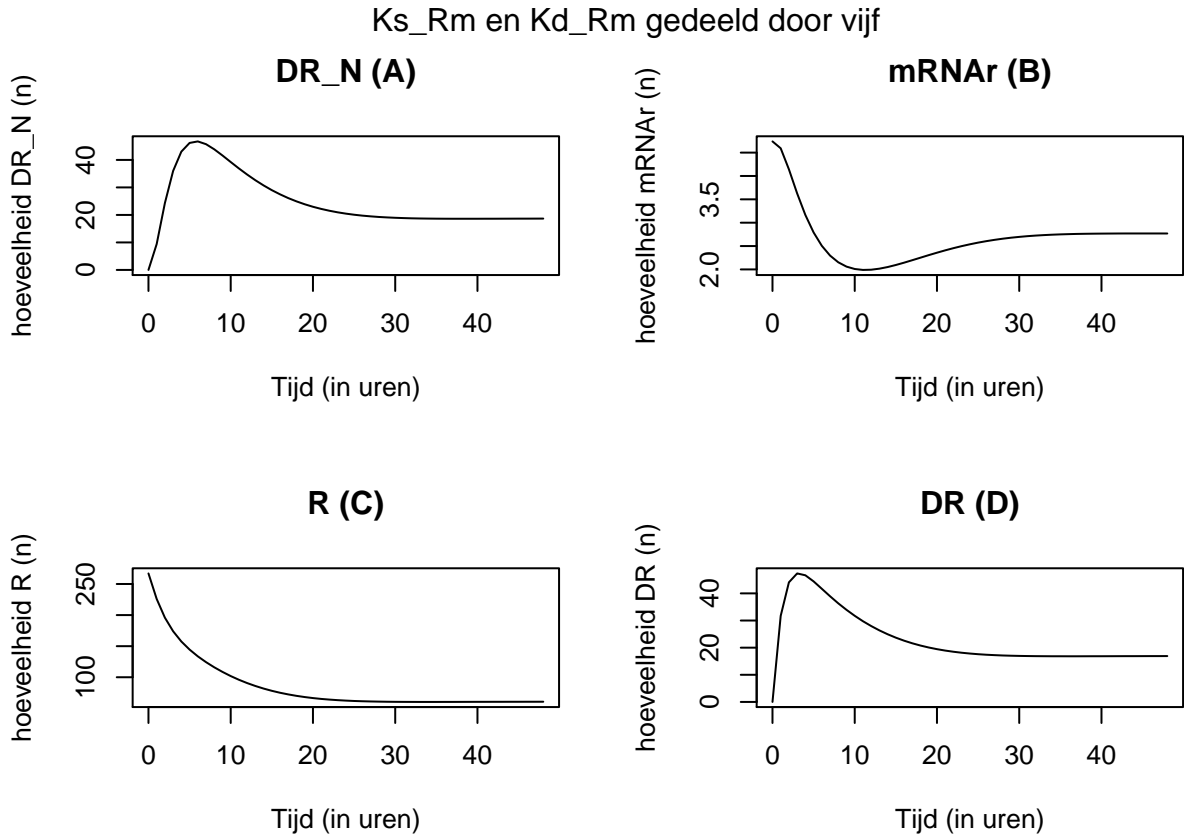


Figure 9: In de bovenstaande figuur staan in de vier grafieken op de x-as het verloop van tijd in uren en op de y-as de verandering in stof verschillend per context. In grafiek B is te zien hoe de hoeveelheid mRNA eerst afneemt bij omzetting naar R. Vervolgens verloopt de aanmaak mRNAr richting een bijna constante lijn vanaf $t \sim 10$. De hoeveelheid R in grafiek C neemt eerst alleen af, omdat er DR vormt. Over tijd zal R (n) als gevolg van de monotone productie in mRNAr, geen veranderingen meer hebben in toename of afname. Verder is te zien dat de concentraties MPL-complex in grafieken A en D eerst toenemen, maar vervolgens zal de productie ook niet meer veranderen. Op basis van de bovenstaande bevindingen is te zien, dat het systeem reageert zoals verwacht op het stilliggende gedrag van de productie in mRNAr.

```

plot(dataset.g5, main = c("DR_N (A)", "mRNAr (B)",
                        "R (C)", "DR (D)"),
      ylab = c("hoeveelheid DR_N (n)", "hoeveelheid mRNAr (n)",
              "hoeveelheid R (n)", "hoeveelheid DR (n)"),
      xlab = "Tijd (in uren)")
mtext("Ks_Rm en Kd_Rm gedeeld door twee",
      side = 3,
      line = - 1,
      outer = TRUE)

```

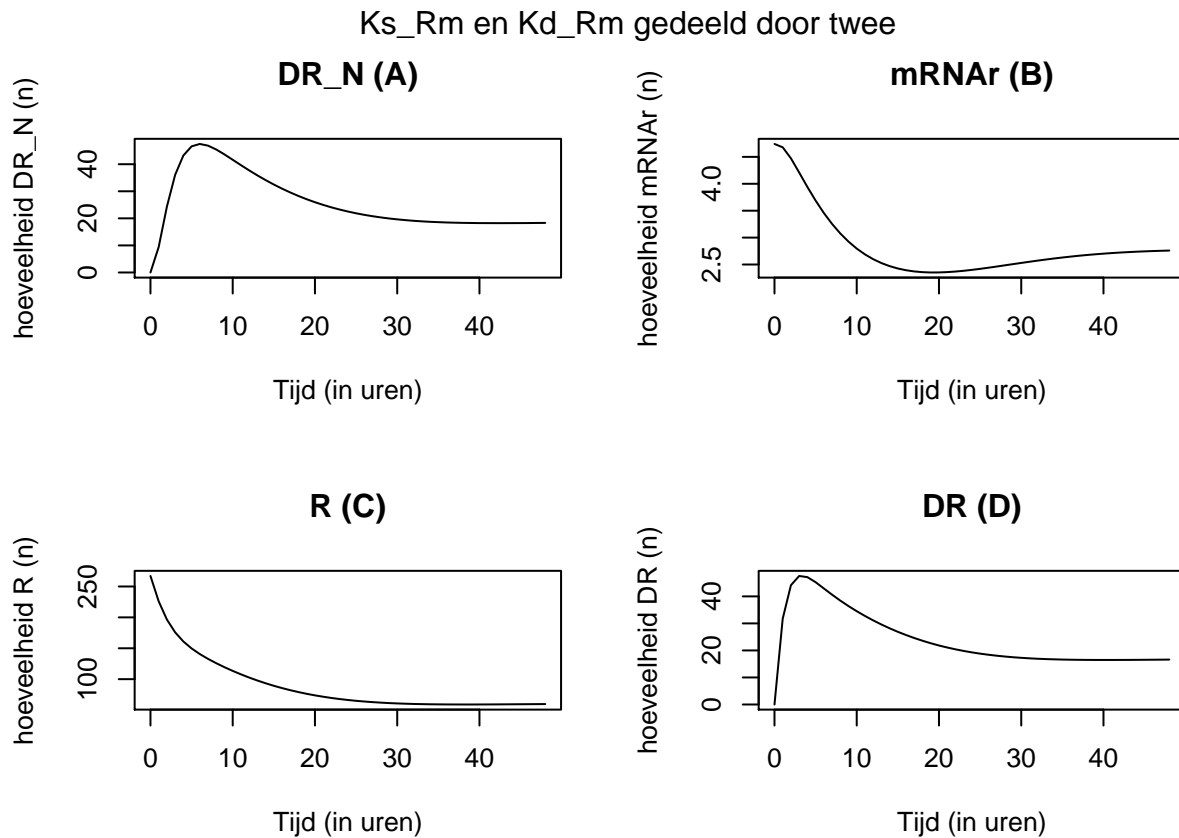


Figure 10: In grafiek B is te zien hoe de hoeveelheid mRNA langzamer afneemt dan bij figuur 9. Vervolgens verloopt de aanmaak vanaf $t \sim 20$ mRNAr minder snel richting een constante lijn. De hoeveelheid R in grafiek C neemt minder snel af in het begin. De rest van de curve verloopt hetzelfde. Grafieken A en D verlopen nagenoeg hetzelfde.

Het wordt duidelijk dat bij een deling door twee, minder grote impact is op de aanmaak van receptoren dan bij een deling van 5.

```

plot(dataset.k5, main = c("DR_N (A)", "mRNAr (B)",
                        "R (C)", "DR (D)"),
      ylab = c("hoeveelheid DR_N (n)", "hoeveelheid mRNAr (n)",
                "hoeveelheid R (n)", "hoeveelheid DR (n)"),
      xlab = "Tijd (in uren)")
mtext("Ks_Rm en Kd_Rm keer vijf",
      side = 3,
      line = - 1,
      outer = TRUE)

```

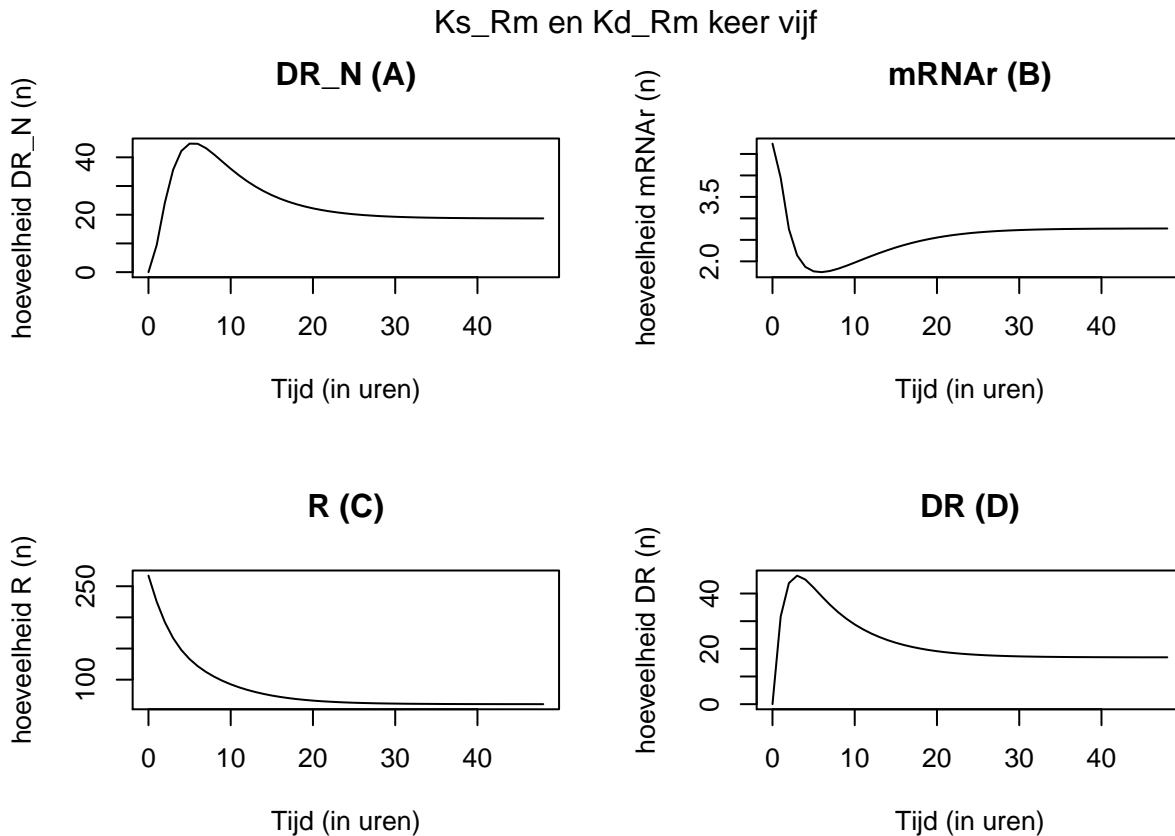


Figure 11: In grafiek B is te zien hoe de hoeveelheid mRNA eerst stijl afneemt bij omzetting naar R. Vervolgens verloopt de aanmaak mRNAr richting een constante lijn, op een gelijke manier als bij figuur 9. Echter begint de aanmaak al bij $t \sim 5$. In grafiek C neemt R (n) steiler af. Grafieken A en D reageren als gevolg van het gedrag van grafiek B op dezelfde wijze als in figuur 9.

Zoals verwacht daalt de hoeveelheid mRNAr veel sneller aan het begin, omdat de afname constante verhoogt is. Tevens begint de toename veel eerder door een verhoogte toename constante.

```

plot(dataset.k2, main = c("DR_N (A)", "mRNAr (B)",
                        "R (C)", "DR (D)"),
      ylab = c("hoeveelheid DR_N (n)", "hoeveelheid mRNAr (n)",
                "hoeveelheid R (n)", "hoeveelheid DR (n)"),
      xlab = "Tijd (in uren)")
mtext("Ks_Rm en Kd_Rm keer twee",
      side = 3,
      line = - 1,
      outer = TRUE)

```

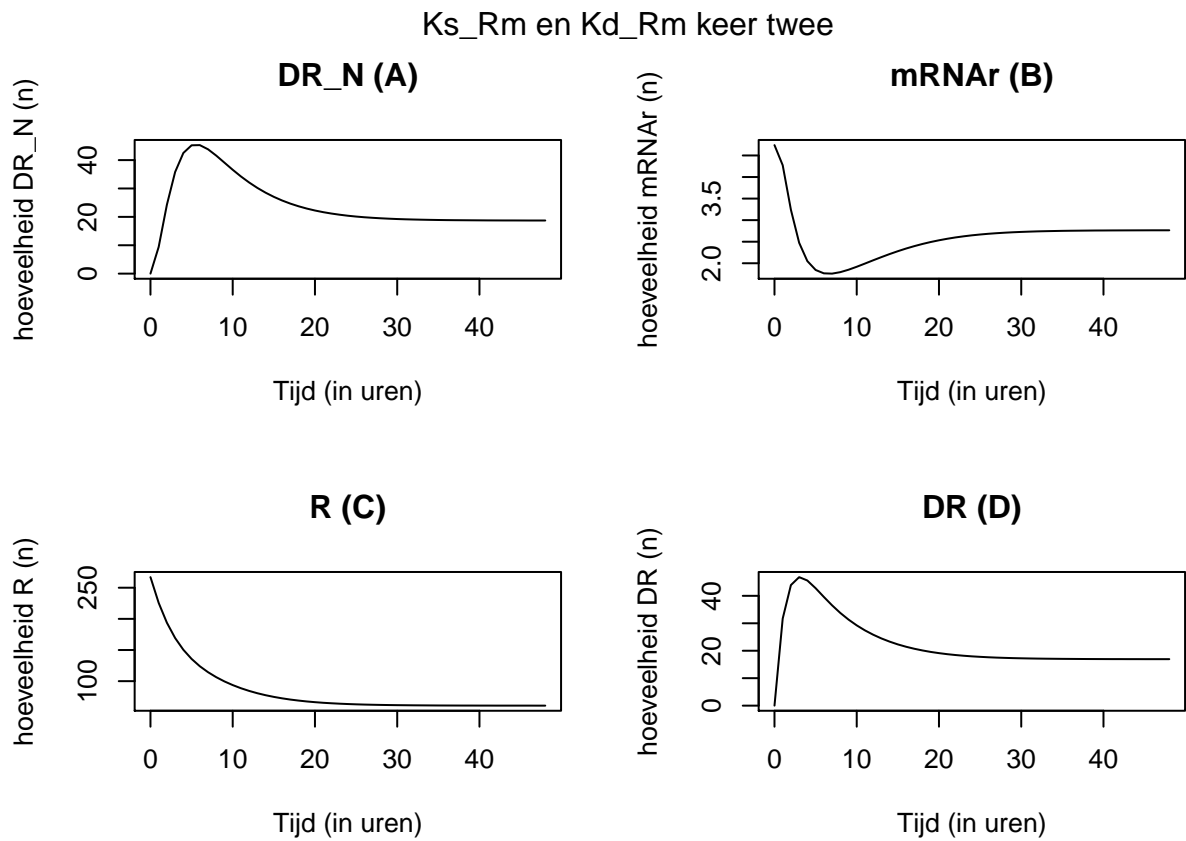


Figure 12: In grafiek B is te zien hoe de hoeveelheid mRNA iets minder stijl afneemt bij omzetting naar R dan bij figuur 11. Vervolgens verloopt de aanmaak mRNAr iets later, vanaf $t \sim 7$. Grafieken A, C en D reageren wederom op dezelfde manier.

Het feit dat de afname minder steil verloopt voldoet aan de verwachting dat, keer twee minder grote invloed uitoefent dan keer vijf. Ook de toename wordt met dezelfde logica beïnvloed en neemt iets later pas toe.

5 Discussie en Conclusie

5.1 Discussie

Uit de resultaten sectie zijn behoorlijke wat resultaten naar voren gekomen. Er waren ook ‘weinig’ verwachtingen voor het onderzoek, omdat we natuurlijk niet wisten wat het model ging doen bij een specifieke aanpassing. De verwachting was dat het aantal vrije receptoren een belangrijke rol zal spelen. Het bewijs voor deze verwachting werd getoond in (2.1). De grafiek van DR_N vertoont een behoorlijk ander verloop. De stijging is namelijk veel sneller. Als de drug geen effect heeft op de synthese van R, komt er een veel hogere concentratie van R. Hierdoor komt er ook veel meer DR_N en dus ook veel meer bijwerkingen. R speelt dus een belangrijke rol.

Verder werd er ook verwacht dat het aantal D een belangrijke rol speelt. Al snel bleek dat na een bepaalde drug ($D=53.4$) de werking niet meer snel stijgt. De grafieken voor een hoge dosis D geven niet een significante hogere uitval.

Bij vraag (2.4) over het stoppen van het toedienen van D, na het bereiken van de steady state, viel iets op. De grafieken stegen of daalden explosief na het stoppen van het toedienen. Achteraf is dit natuurlijk logisch te verklaren. De hoeveelheid D speelt in alle vergelijkingen door, als deze dus plotseling naar 0 zal gaan, is dit van directe invloed op de andere vergelijkingen.

In resultaten (2.5) verliepen de grafieken zoals er wiskundig verwacht werd. Bij twee was er minder invloed op de toename of afname van de productie factoren voor mRNA_r dan bij vijf. Tevens was op logische wijze te zien dat bij een keersom een versterkende invloed werkte en bij een deling het verzachtend effect.

5.2 Algemene conclusie en perspectief

Het uiteindelijke doel van het onderzoek was om iets te kunnen zeggen over de ‘juiste’ hoeveelheid drug. De verwachting was dat deze ergens tussen de lage- en hoge dosis van het experiment zat. Uit de grafieken is gebleken dat bij een hoeveelheid drug dat rond de middengrens van de experimentele zit, een ideaal verloop is voor alle variabelen. Bij een hogere dosis is er niet veel meer resultaat. In combinatie met bijwerkingen is het dus niet noodzakelijk om een hoge dosis te nemen.

Een verbeterpunt voor het volgende onderzoek is het gebruik van de experimentele data. Deze data zou veel meer gebruikt kunnen worden in de figuren. Op deze manier is het nog duidelijker om aan te tonen hoe iets ‘echt’ zou verlopen of om juist aan te tonen dat iets significant anders zou verlopen.

References

- [1] Bergmann MW, Staples KJ, Smith SJ, Barnes PJ, Newton R (2004)., Am J Respir Cell Mol Biol 30: 555–563.: *Glucocorticoids effects in degradation of mRNA*
- [2] Daróczi G, Tsegelskyi R (2022).: *pander: An R 'Pandoc' Writer*. R package version 0.6.5, <<https://CRAN.R-project.org/package=pander>>.
- [3] Soetaert, K., Petzoldt, T., and Woodrow Setzer, R.: *Solving differential equations in R: package deSolve*, J. Stat. Softw., 33, 1-25, 2010.