Practicum 1: Oefening 4

Alexandre Segers & Lieven Clement

statOmics, Ghent University (https://statomics.github.io)

Contents

1 Invloed concentratie op reactiesnelheid

De reactiesnelheid van een proces met een enzyme als katalysator wordt opgemeten door het aantal radioactieve reactieproducten te tellen in functie van de substraatconcentratie. Dat wordt gedaan voor een reactiemengsel met Puromycine en zonder Puromycine.

We willen nagaan of er een lineair verband is tussen de gemiddelde reactiesnelheid en de substraatconcentratie voor zowel de groep die behandeld is met Puromycine als voor de controlegroep zonder Puromicine. Aangezien we de data zouden moeten analyseren met een meervoudige lineaire regressiemodel die het effect van de concentratie en de behandeling kan modelleren, beperken we ons voorlopig tot de data van de groep die behandeld is met Puromycine.

```
library(tidyverse)
library(ggplot2)
```

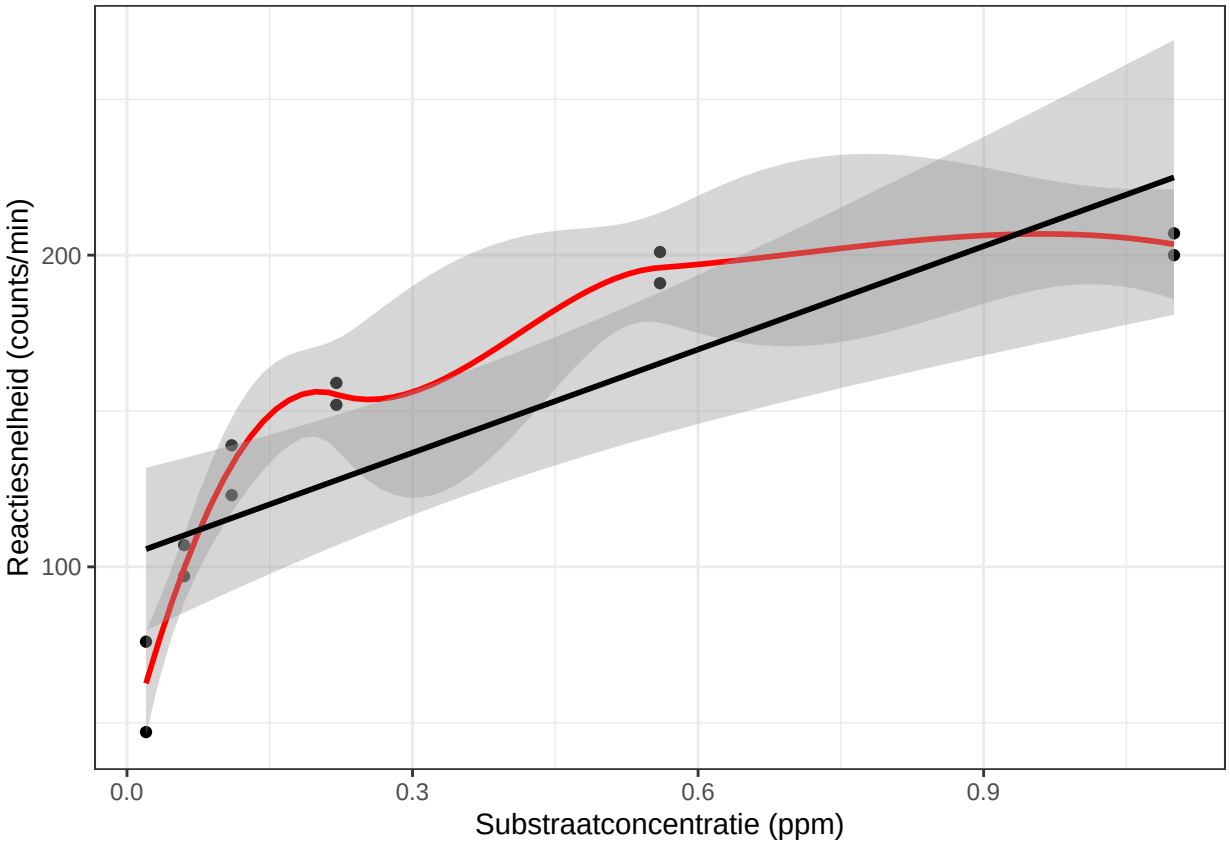
```
data(Puromycin)
Puromycin <- Puromycin %>% filter(state=="treated")
```

2 Data exploratie

We plotten de reactiesnelheid tegenover de concentratie om de data te exploreren.

```
Puromycin  %>%
  ggplot(aes(x=conc,y=rate)) +
  geom_point() +
  stat_smooth(method = "loess",col="red") + # fit een kromme door de punten (rode lijn)
  stat_smooth(method='lm',col="black") + # fit een rechte door de punten aan de hand van de kleinstekwa
  ylab("Reactiesnelheid (counts/min)") +
  xlab("Substraatconcentratie (ppm)") +
  theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```

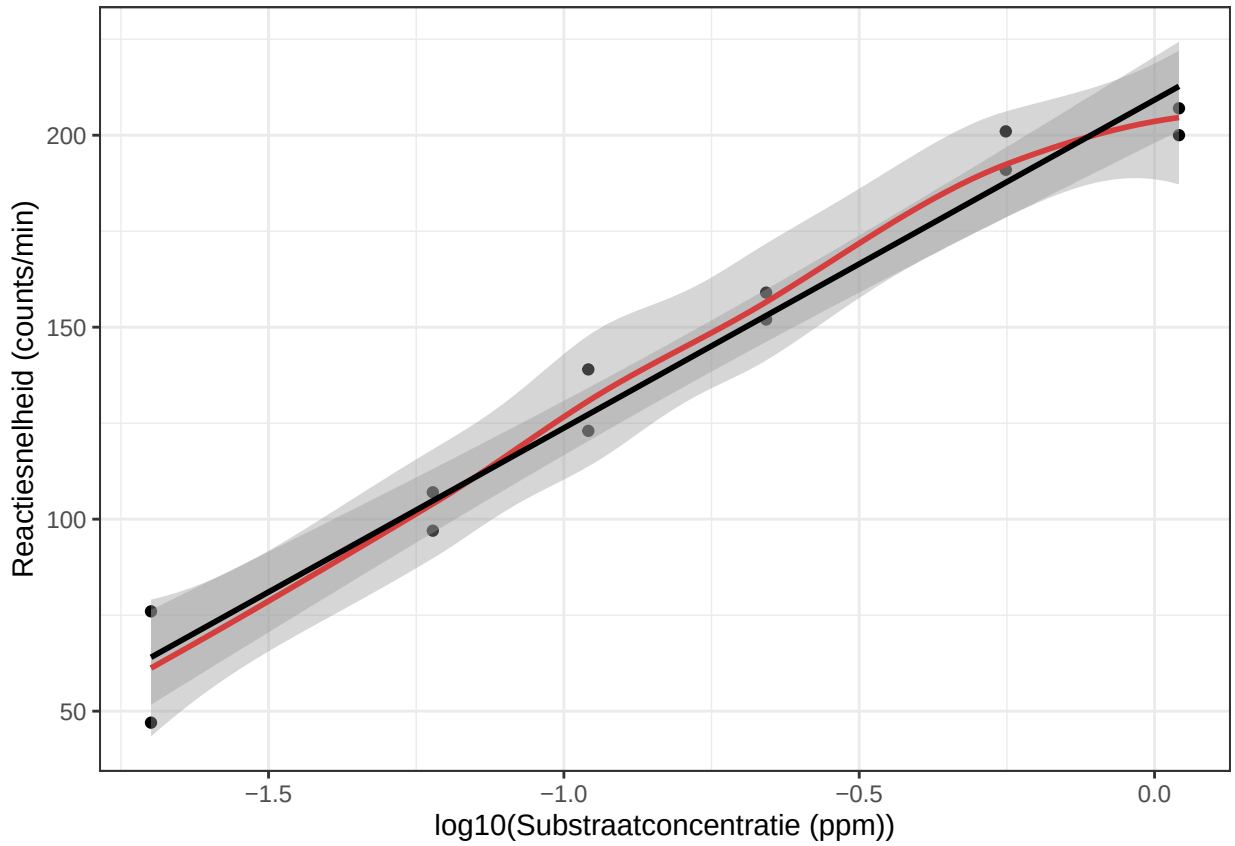


Het ziet ernaar uit dat de data geen lineaire trend volgt. We gaan nu het verband na log-transformatie van de substraatconcentratie. Gezien de substraat concentratie in ppm is gemeten zullen we een $\log_{10}$ transformatie gebruiken (een waarde van -2,-1,0 op log schaal is dan 0.01ppm, 0.1 ppm, 1 ppm).

```
Puromycin  %>%
  ggplot(aes(x=conc %>% log10,y=rate)) +
  geom_point() +
  stat_smooth(method = "loess",col="red") + # fit een kromme door de punten (rode lijn)
  stat_smooth(method='lm',col="black") + # fit een rechte door de punten aan de hand van de kleinstekwa
```

```
  ylab("Reactiesnelheid (counts/min)") +
  xlab("log10(Substraatconcentratie (ppm))") +
  theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



Het verband tussen de reactiesnelheid en het logaritme van de substraatconcentratie lijkt lineair. We zullen de reactiesnelheid dus verder modelleren in functie van de $\log_{10}$-substraatconcentratie.

3 Enkelvoudige lineaire regressie

Enkelvoudige lineaire regressie is een regressie waarbij een variabele gemodelleerd wordt in functie van slechts 1 variabele. De verwachtte reactiesnelheid wordt dus $E[Y_i] = \beta_0 + \beta_1 X_i$. In dit geval is Y de reactiesnelheid en X de log10-substraatconcentratie.

Het model wordt dan als volgt: $reactiesnelheid_i = \beta_0 + \beta_1 log_{10}(concentratie_i) + \epsilon_i$

met β_0 het (werkelijke) **intercept**, β_1 de (werkelijke) **helling** of meer specifiek het (werkelijk) **effect van log10(concentratie) op de gemiddelde reactiesnelheid**. Deze parameters gaan we schatten.

ϵ_i is een foutterm ("error term"), waarbij ϵ_i i.i.d. normaal verdeeld zijn met gemiddelde 0 en (constante) variantie σ^2.

3.1 Assumpties

Voordat we conclusies kunnen trekken uit het lineaire regressiemodel moeten we nagaan of er aan de assumpties voldaan zijn. Voor de lineaire regressie zijn dat volgende assumpties:

1. Onafhankelijke gegevens
2. Lineariteit tussen respons en predictor (impliceert dat residuen rond nul verdeeld zijn, zonder merkbaar resterend patroon tussen de residuen en de geschatte respons variabele)
3. Normaal verdeelde residuen
4. Gelijke variantie (homoscedasticiteit)

Onafhankelijke gegevens moeten we veronderstellen uit het experimenteel design. De andere assumpties moeten we controleren.

3.1.1 Lineariteit tussen reactiesnelheid en $\log_{10}$(substraatconcentratie):

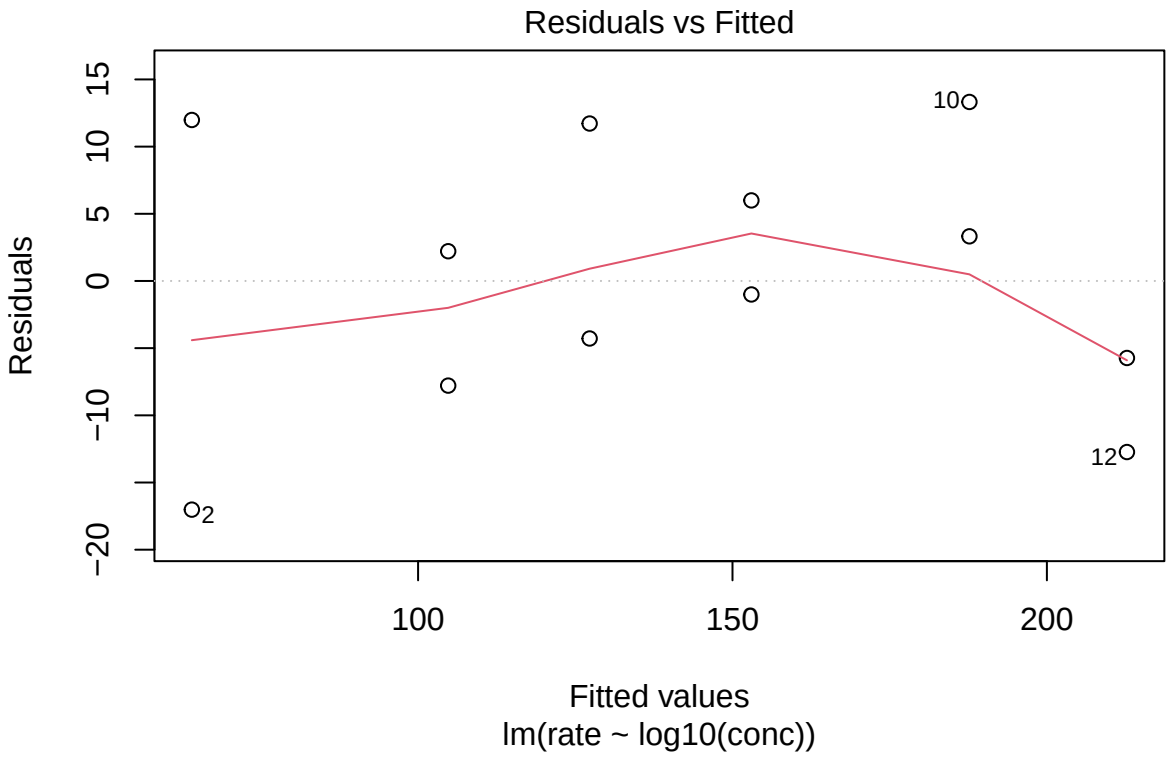
Zoals hierboven besproken lijkt het dat er een lineaire trend is tussen reactiesnelheid en $\log_{10}$(substraatconcentratie) in het volledige bereik van de data.

De lineariteitsassumptie impliceert dat de residuen willekeurig rond nul verdeeld zijn, onafhankelijk van waar we ons op de rechte bevinden. Dit kunnen we weergeven door een lineair model te fitten op de data en de residuen met een smoother weer te geven in functie van de gefitte responswaarden.

```
model <- lm(rate~log10(conc),data = Puromycin)
model
```

```
##
## Call:
## lm(formula = rate ~ log10(conc), data = Puromycin)
##
## Coefficients:
## (Intercept)  log10(conc)
##      209.19        85.45
```

```
plot(model,which=1)
```

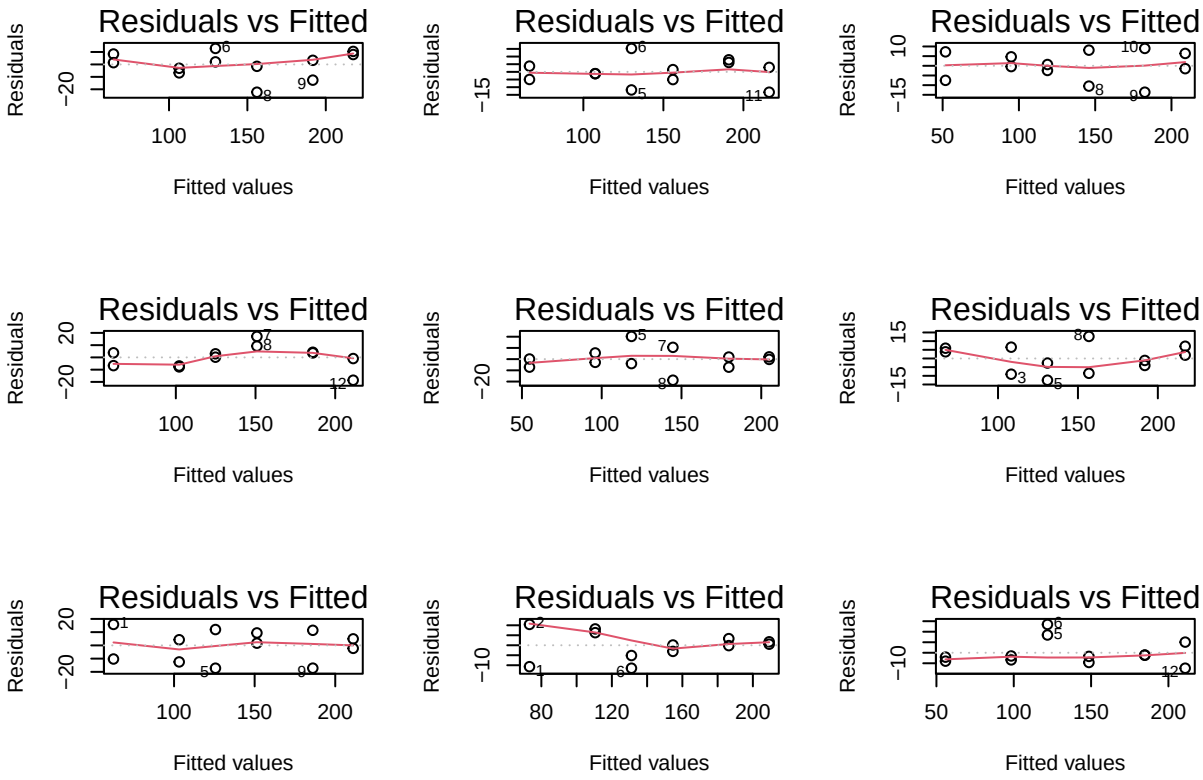


Er lijken kleine afwijkingen te zijn bij de residuen van hogere gefitted waarden. Er zijn echter niet zoveel observaties opgenomen in de studie en de smoother geeft sowieso onnauwkeurig schattingen op de eindpunten van het bereik.

Om na te gaan of de afwijkingen die we zien inderdaad plaucibel zijn en kunnen worden veroorzaakt door random variabiliteit kunnen we gebruik maken van simulaties waaruit we de data genereren onder de voorwaarden van het lineaire model.

We simuleren 9 datasets met hetzelfde aantal observaties, predictorwaarden, intercept, helling en standaarddeviatie. We fitten de modellen en maken de residuplot.

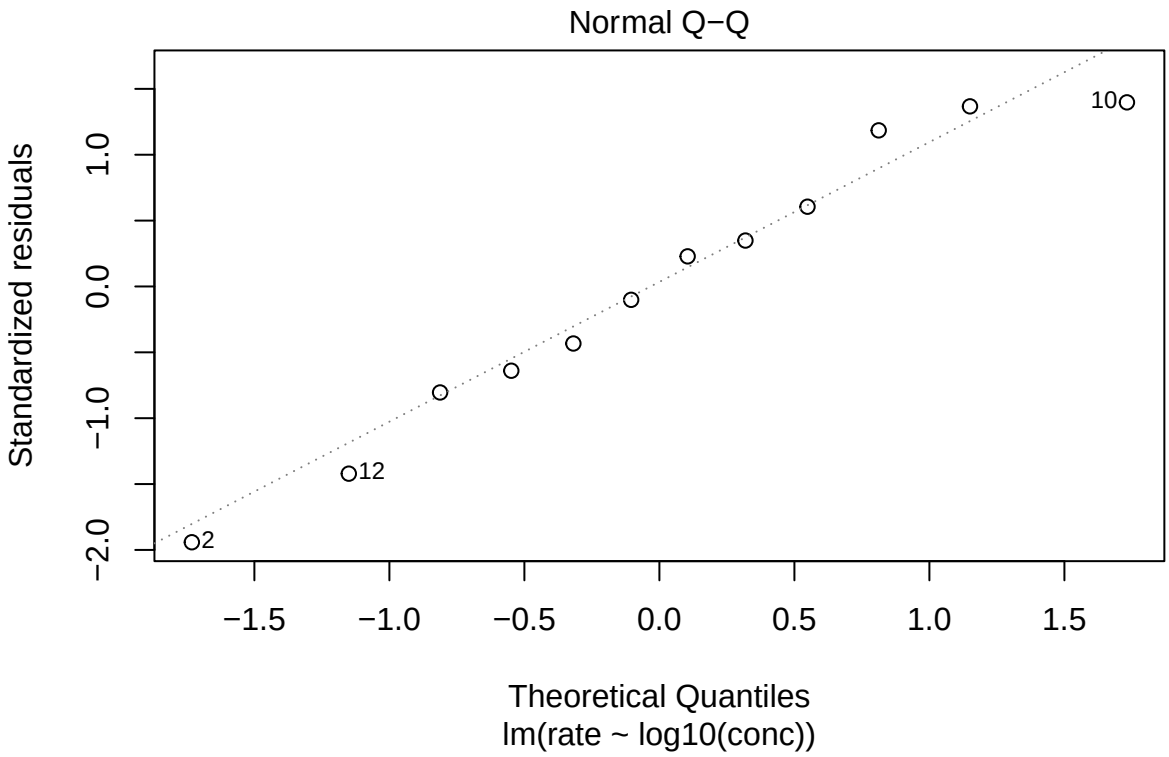
```
set.seed(1031)
betas <- model %>% coefficients
sigma <- model %>% sigma

simModels <- list()

par(mfrow=c(3,3))
for (i in 1:9)
{
  x <- Puromycin %>% pull("conc") %>% log10
  nobs <- Puromycin %>% nrow
  y <- betas[1] + betas[2] * x + rnorm(nobs, sd = sigma)
  simModels[[i]] <- lm(y~x)
  plot(simModels[[i]], which = 1)
}
```



3.1.2 Normaal verdeelde residuen

We gaan via een qq-plot na of de residuen normaal verdeeld zijn.

```
plot(model,which=2)
```



We zien dat er geen systematische afwijkingen zijn van normaliteit, en kunnen veronderstellen dat de kleine afwijkingen door toevallige steekproefvariabiliteit komen.

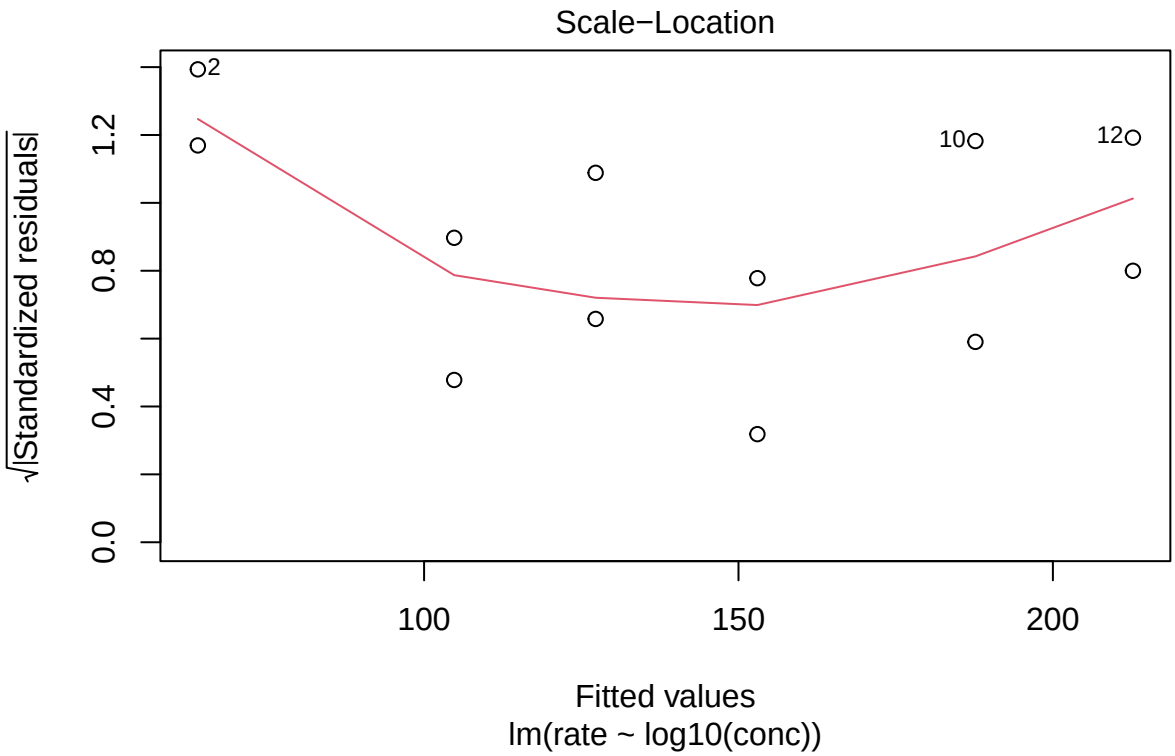
3.1.3 Gelijke variantie (homoscedasticiteit)

Bij lineaire regressie wil de assumptie van gelijkheid van variantie zeggen dat de variantie van de residuen rond de regressierechte hetzelfde is voor elke waarde van de predictor (predictorpatroon).

We kunnen dit opnieuw nagaan met de residu-plot. De spreiding van de residuen zou min of meer gelijk moeten zijn voor elke gefitte waarde.

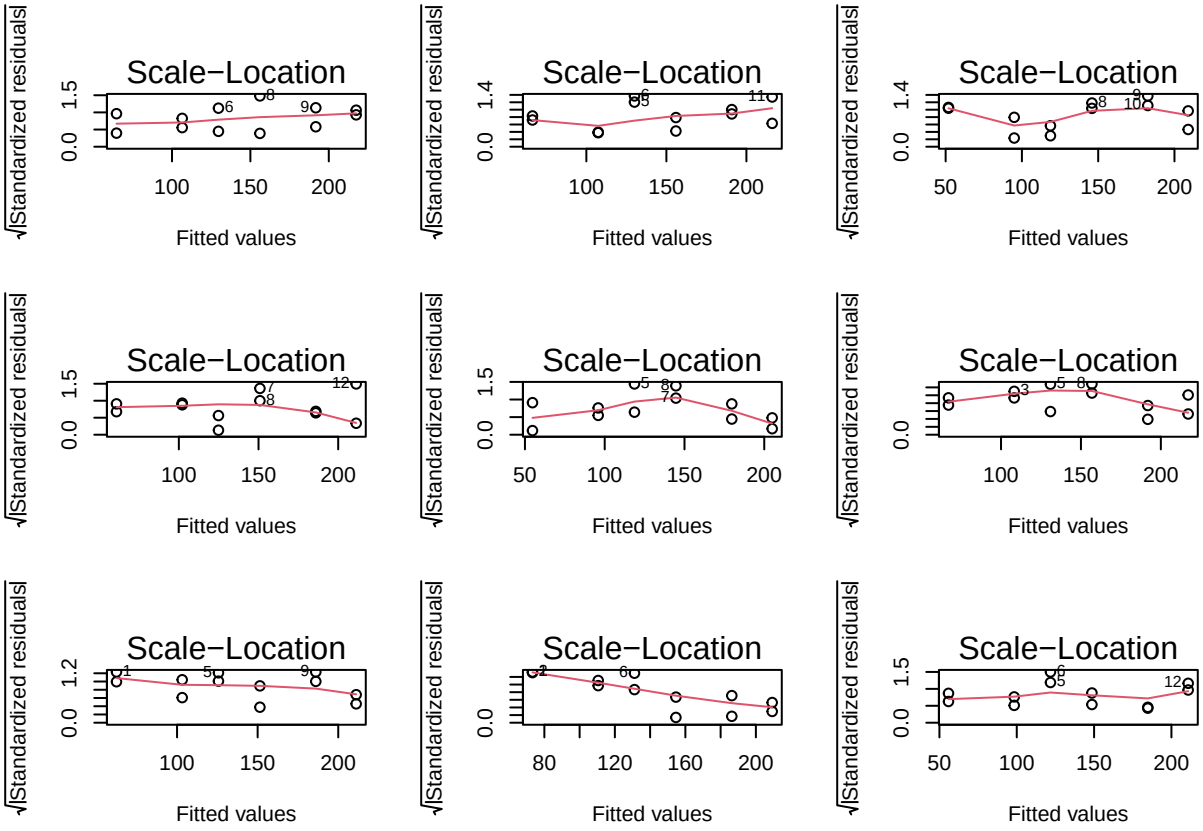
Een andere plot die we hiervoor kunnen gebruiken is een plot waar we de vierkantswortel van de absolute waarde van de gestandaardiseerde residuen plotten in functie van de gefitte waarden. Als we hier een smoother door trekken, zou de smoother een horizontaal verloop moeten hebben. Indien er afwijkingen zouden zijn, bv. er is een systematische trend waarbij de gestandaardiseerde residuen hoger/lager worden naarmate de *fitted values* hoger/lager worden, dan betekent het dat de variantie van de residuen hoger/lager wordt naarmate de geschatte respons hoger/lager wordt.

```
plot(model,which=3)
```



Hier zien we kleine afwijkingen van een horizontale lijn bij de uiterste waarden. Via simulatie zien we opnieuw dat deze afwijkingen plaucibel zijn in een experiment met ons design wanneer alle aannames geldig zijn.

```
par(mfrow = c(3,3))
for (i in 1:9)
  plot(simModels[[i]], which = 3)
```



We kunnen dus veronderstellen dus dat de varianties gelijk zijn.

3.2 Nul- en alternatieve hypothese:

We willen weten of er een lineaire associatie is tussen de reactiesnelheid en de $\log_{10}$ getransformeerde concentratie.

De nul- en alternatieve hypothese van het lineair model worden dus:

H_0: $\beta_1 = 0$

H_A: $\beta_1 \neq 0$

Met andere woorden stelt de nulhypothese dat er geen associatie is tussen de reactiesnelheid en de $\log_{10}$ concentratie, terwijl de alternatieve hypothese stelt dat er juist wel een associatie is.

3.3 Fit het lineair model en bespreek beide parameters, ga na of de nulhypothese verworpen wordt en maak een interpretatie van het betrouwbaarheidsinterval.

```
summaryModel <- summary(model)
summaryModel
```

```
##
## Call:
```

```
## lm(formula = rate ~ log10(conc), data = Puromycin)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.0176  -6.2455   0.6039   7.4262  13.3228
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  209.194      5.045   41.47 1.59e-12 ***
## log10(conc)   85.450      5.133   16.65 1.28e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.37 on 10 degrees of freedom
## Multiple R-squared:  0.9652, Adjusted R-squared:  0.9617
## F-statistic: 277.1 on 1 and 10 DF,  p-value: 1.28e-08
```

```
confintModel <- confint(model)
confintModel
```

```
##                 2.5 %    97.5 %
## (Intercept) 197.95335 220.43564
## log10(conc)  74.01255  96.88732
```

3.3.1 Conclusie

Er is een extreem significante lineaire associatie tussen de substraatconcentratie op logschaal en de reactiesnelheid (p « 0.001). Wanneer we de reactie laten doorgaan bij een substraatconcentratie die 10 keer hoger is, is de reactiesnelheid gemiddeld met 85.4 counts/min hoger (95% betrouwbaarheidsinterval [74, 96.9] counts/min).

3.3.2 Interpretatie

1. Het intercept is de geschatte gemiddelde reactiesnelheid bij een $\log_{10}$-concentratie van 0/een substraat concentratie van 1 ppm en is gelijk aan 209.2 counts/min.
2. Helling

- Log schaal: Wanneer we de reactie laten doorgaan bij een substraatconcentratie die 1 eenheid op $\log_{10}$ schaal hoger is, is de reactiesnelheid gemiddeld met 85.4 counts/min hoger.
- Originele schaal: Wanneer we de reactie laten doorgaan bij een substraatconcentratie die 10 keer hoger is, is de reactiesnelheid gemiddeld met 85.4 counts/min hoger.

3. 95% betrouwbaarheidsinterval: We hebben dus geschat dat het werkelijke verschil in reactiesnelheid tussen twee reacties die doorgaan onder een substraatconcentratie die een factor 10 verschillen met 95% kans ligt tussen [74, 96.9] counts/min; merk op dat de reactie sneller doorgaat in de reactie met de hoogste substraatconcentratie.

3.4 Schat de gemiddelde reactiesnelheid bij een substraat concentratie van 0.2ppm en geef een bijhorend 95%-betrouwbaarheidsinterval.

```
pred <- predict(model, newdata=data.frame(conc=0.2), interval="confidence")
pred
```

```
##        fit      lwr      upr
## 1 149.4676 142.7162 156.2189
```

De geschatte gemiddelde reactiesnelheid bij een substraatconcentratie van 10 ppm is 149.5 counts/min (95% betrouwbaarheidsinterval [142.7, 156.2] counts/min).

4 Algemene conclusie

Er is een extreem significante lineaire associatie tussen de substraatconcentratie op logschaal en de reactiesnelheid ($p \ll 0.001$). Wanneer we de reactie laten doorgaan bij een substraatconcentratie die 10 keer hoger is, is de reactiesnelheid gemiddeld met 85.4 counts/min hoger (95% betrouwbaarheidsinterval [74, 96.9] counts/min).