Non linear regression with factors

Cultivation of xxxxxx

Dr Giorgio Pioda

Last compilation: 23 marzo, 2023

SSSMT - Locarno

Data set with factors and interactions

- Experiments about in vitro cultivation of xxxxxx. The response value is the yield ($r =$ "*resa*").
- Cultivations take place in presence of inhibiting antibodies at 8 different concentrations levels.
- Extractions of the xxxxxx have been performed with different kits.
- The cultivations have been performed with or without an additional dilution factor.
- The number of replicas are not symmetric across all factors.

Original *look* of the data set

- Misleadingly, the data set was delivered initially in wide format, one column for each dilution series.
- After a long discussion with the experimentator it came out that there were no connection at all in the dilution series.
- All yield data points were completely independent. Not even multipipetting pairing was possible.

```
## 'data.frame':    128 obs. of  4 variables:
##  $ c     : num  40 20 10 5 2.5 ...
##  $ r     : num  0 0 0 0 382226 ...
##  $ mAb   : Factor w/ 2 levels "a1","a11": 1 1 1 1 1 1 1 1 1 1 ...
##  $ status: Factor w/ 2 levels "dil","ndil": 2 2 2 2 2 2 2 2 2 2 ...
```

Plot of the raw data

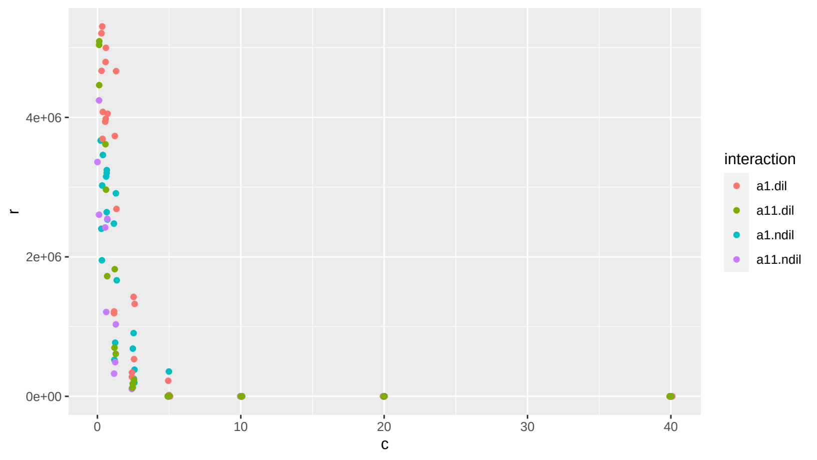


Figure 1: Yields by antibody and dilution interactions

Plot of the logtransformed data (loglogistic)

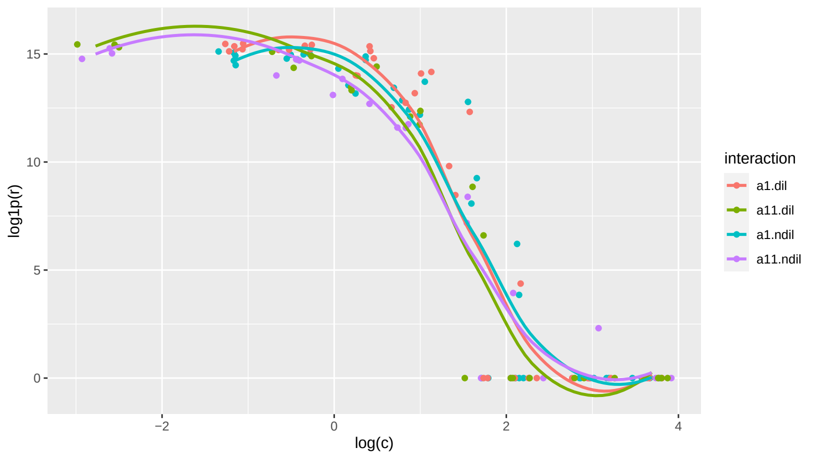


Figure 2: Yields by antibody and dilution interactions

- Inspiring for asymptotic levels *Asym* and *xmids* (*ED50*). Less for the scaling *scal* "slope"

At first glance

- The response value needs a log-transformation.
- The concentration of the inhibitor also need a log-transformation (loglogistic model).
- 3 parameter logistic model with (Asym, xmid, scale).
- There are a lot of zeros in the response value. Experimental nonsense since the initial concentration of the xxxxxx is not zero.
- The use of log1p is justified.

Possible tackling strategies for model fitting / 1

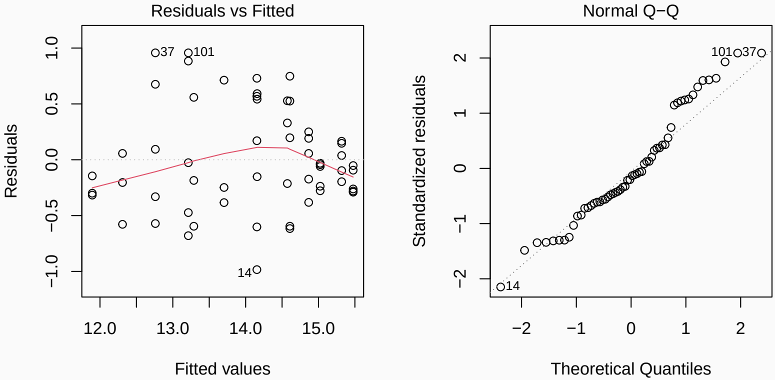
- Concentrate the analysis only on the asymptotic yield *r*. Use the inhibitor concentration as a factor: conventional regression with factors and interactions.
  - Pros: Easy approach, standard ANOVA.
  - Cons: The role of the inhibitor concentration is lost, no determination of the effective dose *ED50* (*xmid*) possible.
  - Cons: Half of the data, at low inhibitor concentration, is discarded.

Possible tackling strategies for model fitting / 2

- Summary method for *xmid* and *scal* is not viable since there is **no experimental pairing in the dilution series** to allow a separate non linear regression for each dilution serie.
- Non linear regression with standard *nls()*, separated for each interaction (4) and comparison of the obtained CI ? Bad idea. No common error management.
- Nonlinear regression with factors through *gnls()* from the *{nlme}* package. The end choice.

Linear regression approach / 1

```
r.dnd<-subset(dnd,c < 5)
rr.dnd<-subset(r.dnd,c > 0.1)
options(contrasts = c("contr.sum", "contr.poly"))
s.fit<-lm(log1p(r) ~ as.factor(c) + mAb * status,data=rr.dnd )
par(mfrow=c(1,2))
plot(s.fit,which=1:2)
```



Linear regression approach / 2

```
drop1(s.fit,test="F",scope=~.)
```

```
## Single term deletions
##
## Model:
## log1p(r) ~ as.factor(c) + mAb * status
##              Df Sum of Sq    RSS     AIC F value    Pr(>F)
## <none>                    11.891 -77.913
## as.factor(c)  3    47.609 59.499   9.480 68.0664 < 2.2e-16 ***
## mAb           1     8.811 20.701 -47.755 37.7899 1.205e-07 ***
## status        1     2.338 14.228 -69.503 10.0264  0.002605 **
## mAb:status    1     0.004 11.895 -79.891  0.0189  0.891201
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Reasonables expectations for the asymptotic level (*Asym*) parameter. Significant antibody and dilution factors. Nonsignificant interaction.

Non linear approach based on *gnls()*, *{nlme}* package

- A lot of different non linear parametrisations with self starting functions. No need to find starting values.
- More self starting functions are available in add on packages
- Multiple factors with interaction are possible.
- Weighting factors with internal optimization to adjust for heteroscedasticity.
- Random effect are also available via *nlme()*. Here not used since no real pairing of repeated measures is present in the data set.

The 3 parameter logistic regression alternatives

- *L.3()* package *{drc}*

$$Y(x) = \frac{d}{1 + exp(b \cdot (x - e))}$$

- *SSlogis()* package *{nlme}*

$$Y(x) = \frac{Asym}{1 + exp(\frac{xmid - x}{scal})}$$

- *NLS.L3()* from package *{aomisc}* which has the “same” (b sign mismatch!) parametrization found in the *{drc}* package

$$Y(x) = \frac{d}{1 + exp(-b \cdot (x - e))}$$

Thus $scal = -\frac{1}{b}$ for *L.3()* but $scal = \frac{1}{b}$ for *NLS.L3()*

Pararametrization and convergency errors

Typical error during non linear fit:

“Error in gnls(rlog1p ∼ SSlogis(clog, Asym, xmid, scal), data = new.dnd, : step halving factor reduced below minimum in NLS step”

Two alternatives:

- Check starting values
- Change parametrization
- (Sometimes an increase of the tolerance parameter helps)

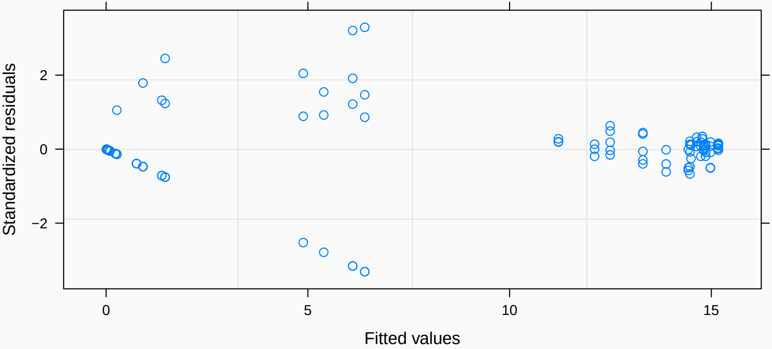
In this case switched to *NLS.L3()*

Using {drc} to get starting values

- *drm()* from *{drc}* can self start with factors and interactions. *{nlme}* cannot.
- The *{drc}* *L.3()* is almost the "same" as *NLS.L3()* in *{aomisc}* written to be used with *{nlme}*.

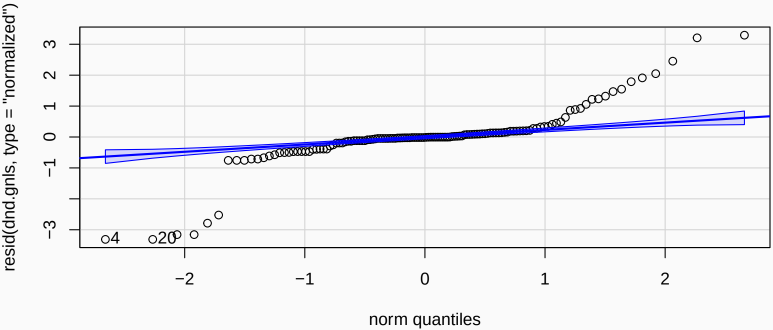
```r
options(contrasts = c("contr.treatment", "contr.poly"))
library(nlme)
library(aomisc) # Additional self starting sunction package
library(drc) # Dose Response package
new.dnd<-groupedData(rlog1p~clog | interaction ,data=dnd) # gnls needs is,
                                                          # althogh no random effect is used
drm.dnd <- drm(rlog1p ~ clog, fct = L.3(), data = new.dnd,
                 curveid = interaction,
                 pmodels = c( ~ mAb * status,  ~ mAb * status,   ~ mAb * status))

newstart<-coef(drm.dnd) # save starting values for gnls()
newstart<-coef(drm.dnd)*c(rep(-1,4),rep(1,8)) # correct for b sign

dnd.gnls<-gnls(rlog1p ~ NLS.L3(clog,b,d,e) , data=new.dnd,
                params= b + d + e ~ mAb * status,
                start=newstart)
```

Residuals of the complete model with interactions / T-A



The homoscedasticity is NOT satisfied.

Residuals of the complete model with interactions / Q-Q

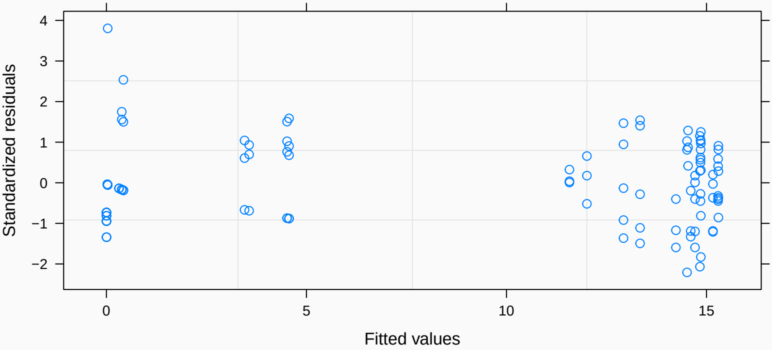


Both T-A and Q-Q look pretty bad. A correction is needed. The best option is to have a weighting factor for each inhibitor concentration level.

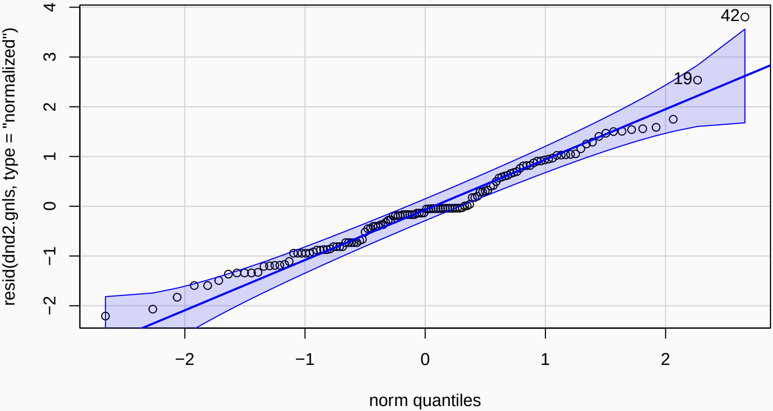
Complete model with weighting factors / T-A

{nlme} **big strength**. Different weighting methods available!

```
newstart<-coef(dnd.gnls) # storing previous starting values
dnd2.gnls<-update(dnd.gnls , data=new.dnd,
                weights=varIdent(form= ~ 1 | as.factor(clog)), # here the magic happens
                start=newstart)
plot(dnd2.gnls)
```



Complete model with weighting factor / Q-Q



Much better. . .

Is the weigthing worth ? Model comparison

ANOVA III - loglikelihood test

- Under H_0 for the equivalence of two nested models (B: bigger, R: reduced) and d as the degrees of freedom difference.

$$2(ll^B - ll^R) \sim \chi_d$$

```
anova(dnd2.gnls,dnd.gnls)
```

```
##           Model df      AIC      BIC     logLik   Test  L.Ratio p-value
## dnd2.gnls     1 21 150.5069 210.3996  -54.25346
## dnd.gnls      2 13 545.6810 582.7574 -259.84051 1 vs 2 411.1741  <.0001
```

- 8 degree of freedom used for the weighting.
- AIC, BIC and also the P-values are very clear. The correction is beneficial.

Testing ANOVA I vs ANOVA III

- Pinheiro & Bates suggest the following:
    - ANOVA I should be preferred over ANOVA III (loglikelyhood) test especially for fixed factors.
    - If the fixed factors are constant then ANOVA III is OK
    - The reason behind this is that in P-P plots, especially if the fixed factor has many levels, the ANOVA III tends to be too optimistic (or not enough conservative)
- But in this case:
    - There are no random effect
    - The fit is ML and not ReML
    - The suggestions above are referred about a ReML model with random effects.

Classical testing confints

- *{nlme}* Confints are based on the entries of the variance covariance matrix

$$\hat{\theta}_j \overset{approx.}{\sim} \mathcal{N}(\theta_j, V_{jj})$$

$$\frac{\hat{\theta}_j - \theta_j}{\sqrt{V_{jj}}} \overset{approx.}{\sim} t_{n-p}$$

- *{nlraa}* Add-on package provides bootstrap and thus bootstrapped CI (centered residual bootstrap)
- Remark: "***" < 0.02, "*" < 0.05, "°" < 0.1

ANOVA I output, weighted vs unweighted

Contrasts check

```
## $contrasts
## [1] "contr.treatment" "contr.poly"
```

Table 1: Weighted vs unweighted regressions ANOVA I

	numDF	Unw F-value	Unw p-value		W F-value	W p-value	
b.(Intercept)	1	1014.1475	0.0000	***	19645.6404	0.0000	***
b.mAb	1	4.4518	0.0370	*	352.0521	0.0000	***
b.status	1	3.6315	0.0592	°	900.8697	0.0000	***
b.mAb:status	1	0.1757	0.6759		47.5678	0.0000	***
d.(Intercept)	1	6090.1872	0.0000	***	160160.9470	0.0000	***
d.mAb	1	1.4269	0.2347		5.9512	0.0162	***
d.status	1	0.2287	0.6334		11.0871	0.0012	***
d.mAb:status	1	0.1239	0.7255		4.2710	0.0410	*
e.(Intercept)	1	1242.5573	0.0000	***	5520.7044	0.0000	***
e.mAb	1	1.5496	0.2157		7.8666	0.0059	***
e.status	1	0.0228	0.8802		0.0177	0.8945	
e.mAb:status	1	0.0053	0.9422		0.0113	0.9156	

Confint output, weighted vs unweighted

Table 2: Weighted vs unweighted regressions confidence intervals

	Unw 2.5%	Unw 97.5%		W 2.5%	W 97.5%	
b.(Intercept)	-4.8760	-1.9182	***	-4.4587	-3.5874	***
b.mAba11	-2.0125	2.4097		-0.4465	0.8882	
b.statusndil	-1.2431	2.4453		-0.4478	0.6665	
b.mAba11:statusndil	-2.6327	2.8539		-0.7097	0.9244	
d.(Intercept)	14.1402	16.2145	***	15.1647	15.4311	***
d.mAba11	-2.1330	1.3326		-0.7182	-0.1713	***
d.statusndil	-1.8144	1.1913		-0.6271	-0.2503	***
d.mAba11:statusndil	-2.3664	2.7085		-0.2627	0.5120	
e.(Intercept)	1.3552	1.6315	***	1.3178	1.4684	***
e.mAba11	-0.3437	0.1349		-0.2072	0.0116	°
e.statusndil	-0.1941	0.2281		-0.0957	0.1128	
e.mAba11:statusndil	-0.3814	0.3554		-0.1576	0.1421	

- Focus on the *d (Asym)* parameter!

ANOVA III / parameter *e* (ED50)

ANOVA III is boring with gnls(); no automatic *drop1()* is available. Each model have to be tested separatedly.

- Remove interaction for *e*

```
##              Model df      AIC      BIC    logLik   Test   L.Ratio p-value
## dnd2e.gnls     1 20 148.9730 206.0136 -54.48651
## dnd2.gnls      2 21 150.5069 210.3996 -54.25346 1 vs 2 0.4661079  0.4948
```

- Main factor dilution status for *e*

```
##                 Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2e.gnls        1 20 148.9730 206.0136 -54.48651
## dnd2edil.gnls     2 19 144.8375 199.0261 -53.41874 1 vs 2 2.135546  0.1439
```

- Main factor inhibiting antibody for *e*

```
##                 Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2e.gnls        1 20 148.9730 206.0136 -54.48651
## dnd2emAb.gnls     2 19 159.7536 213.9422 -60.87683 1 vs 2 12.78062   4e-04
```

ANOVA III / parameter *d* (Asyn)

- Remove interaction for *d*

```
##            Model df      AIC      BIC    logLik   Test   L.Ratio p-value
## dnd2d.gnls     1 20 148.0194 205.0600 -54.00968
## dnd2.gnls      2 21 150.5069 210.3996 -54.25346 1 vs 2 0.4875581   0.485
```

- Main factor dilution status for *d*

```
##               Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2ddil.gnls     1 19 169.0554 223.2439 -65.52767
## dnd2d.gnls        2 20 148.0194 205.0600 -54.00968 1 vs 2 23.03598  <.0001
```

- Main factor inhibiting antibody for *d*

```
##               Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2dmAb.gnls     1 19 161.3896 215.5781 -61.69477
## dnd2d.gnls        2 20 148.0194 205.0600 -54.00968 1 vs 2 15.37019   1e-04
```

ANOVA III / parameter *b* (scal)

- Remove interaction for *b*

```
##            Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2b.gnls     1 20 144.6316 201.6722 -52.31580
## dnd2.gnls      2 21 150.5069 210.3996 -54.25346 1 vs 2 3.875323   0.049
```

- Main factor dilution status for *b*

```
##               Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2bdil.gnls     1 19 137.8140 192.0025 -49.90698
## dnd2b.gnls        2 20 144.6316 201.6722 -52.31580 1 vs 2 4.817638  0.0282
```

- Main factor inhibiting antibody for *b*

```
##               Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2bmAb.gnls     1 19 134.3744 188.5630 -48.18719
## dnd2b.gnls        2 20 144.6316 201.6722 -52.31580 1 vs 2 8.257219  0.0041
```

Weighted full model Anova I vs Anova III vs CI {#(AiAIIICI-Fullcomp)}

Table 3: Anova I vs Anova III vs CI

	A-I p-value		A-III p-value		CI 2.5%	CI 97.5%	
b.(Intercept)	0.0000	***	NA	NA	-4.4587	-3.5874	***
b.mAb	0.0000	***	0.0041	***	-0.4465	0.8882	
b.status	0.0000	***	0.0282	*	-0.4478	0.6665	
b.mAb:status	0.0000	***	0.0490	*	-0.7097	0.9244	
d.(Intercept)	0.0000	***	NA	NA	15.1647	15.4311	***
d.mAb	0.0162	***	0.0001	***	-0.7182	-0.1713	***
d.status	0.0012	***	0.0000	***	-0.6271	-0.2503	***
d.mAb:status	0.0410	*	0.4850		-0.2627	0.5120	
e.(Intercept)	0.0000	***	NA	NA	1.3178	1.4684	***
e.mAb	0.0059	***	0.0004	***	-0.2072	0.0116	°
e.status	0.8945		0.1439		-0.0957	0.1128	
e.mAb:status	0.9156		0.4948		-0.1576	0.1421	

- *b* parameters are strange.
- ANOVA III seems more conservative.

Model without interaction

- It seems that all the interactions can be eliminated.

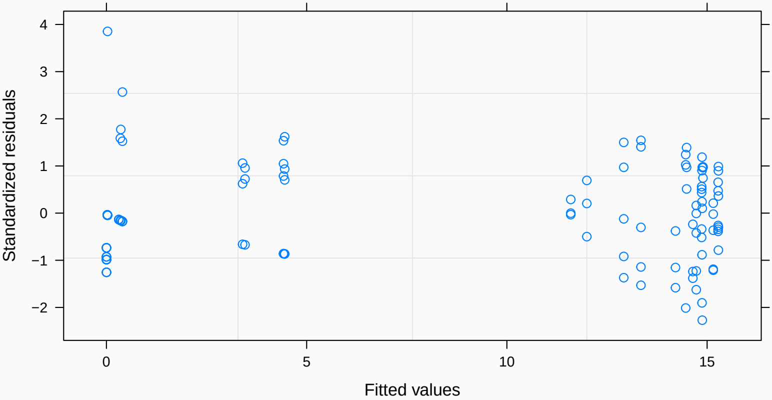
```
##            Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2ni.gnls     1 18 140.8578 192.1944 -52.42891
## dnd2.gnls       2 21 150.5069 210.3996 -54.25346 1 vs 2 3.649094  0.3019
```

Table 4: Anova I vs Anova III, no interactions

	A-I p-value		A-III p-value		CI 2.5%	CI 97.5%	
b.(Intercept)	0.0000	***	NA	NA	-4.4430	-3.7223	***
b.mAb	0.0000	***	0.0041	***	-0.1000	0.6539	
b.status	0.0000	***	0.0282	*	-0.2601	0.5219	
d.(Intercept)	0.0000	***	NA	NA	15.1581	15.4070	***
d.mAb	0.0080	***	0.0001	***	-0.5777	-0.1929	***
d.status	0.0004	***	0.0000	***	-0.5730	-0.2456	***
e.(Intercept)	0.0000	***	NA	NA	1.3262	1.4526	***
e.mAb	0.0054	***	0.0004	***	-0.1718	-0.0283	***
e.status	0.8927		0.1439		-0.0673	0.0769	

Results are almost consistent with previous tests. The factors influx on *b* still remains puzzling. **All significant P-values but with non significant CI**.

Residuals of the model without interactions / T-A



- No big differences from full model.

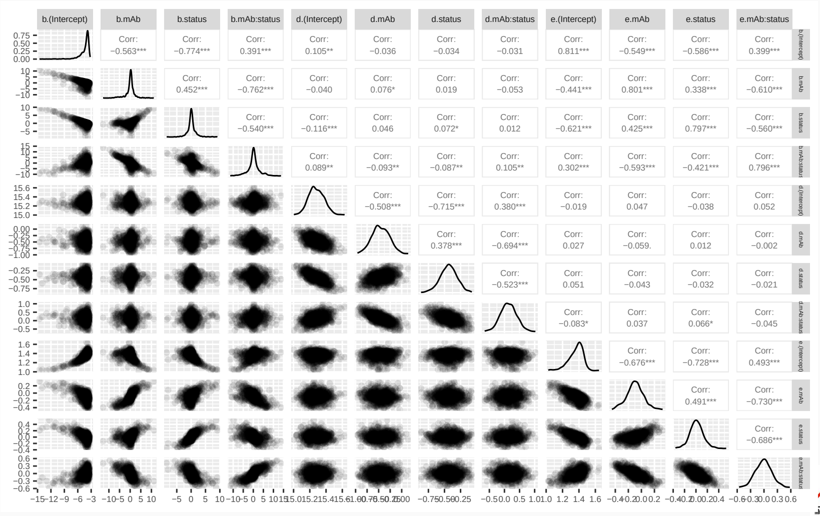
Bootstrap: general idea

- Modern way to asses bias, variance and CI when no distribution data is available.
- Basic idea: resample the data set **with repetitions**. For each new sample the test / regression is performed again collecting all the parameters.
- Repeat it at least 200-300 times to asses parameter variance. Repeat it at least 1000 times to asses CI (for example with quantiles)
- Different types of resampling possible. For nonlinear regression the centered residuals get resampled and added to the response value; all the rows of the data set are used in each repetition.

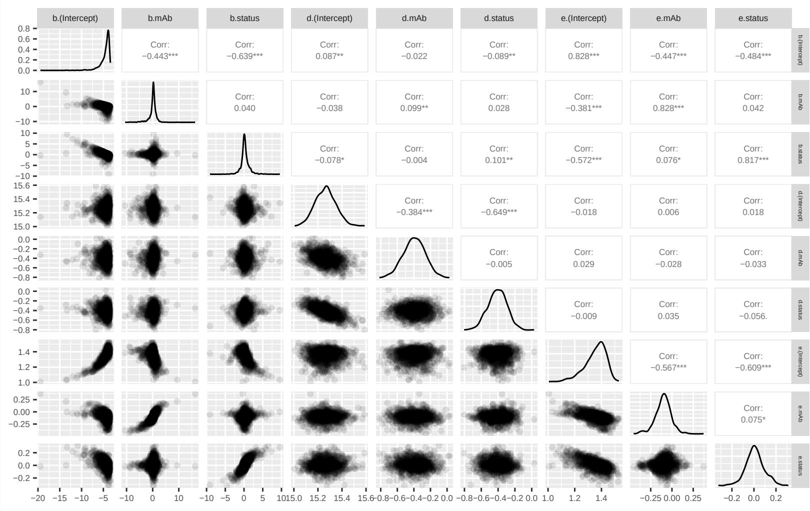
Bootstrap benefit for non lon linear regression

- Assessment of parameter variances.
- Alternative way to get CI without normality assumptions.
- Collinearity check of the parameters via pairs plot.

Bootstrap with *{nlraa}* package / full model -> s46



Bootstrap with *{nlraa}* package / model without interactions -> s49



Idea from pairs plot: keep *b* fixed without interaction.

```
dnd2nibf.gnls<-update(dnd2.gnls,
                 params= c(b  ~ 1, d  ~ mAb + status, e  ~ mAb + status),
                 weights=varIdent(form= ~ 1 | as.factor(clog)),
                 start =newstart[c(1,5:7,9:11)])

dnd2nibfUnw.gnls<-update(dnd.gnls,
                 params= c(b  ~ 1, d  ~ mAb + status, e  ~ mAb + status),
                 #weights=varIdent(form= ~ 1 | as.factor(clog)),
                 start =newstart[c(1,5:7,9:11)])

anova(dnd2nibf.gnls,dnd2ni.gnls)
```

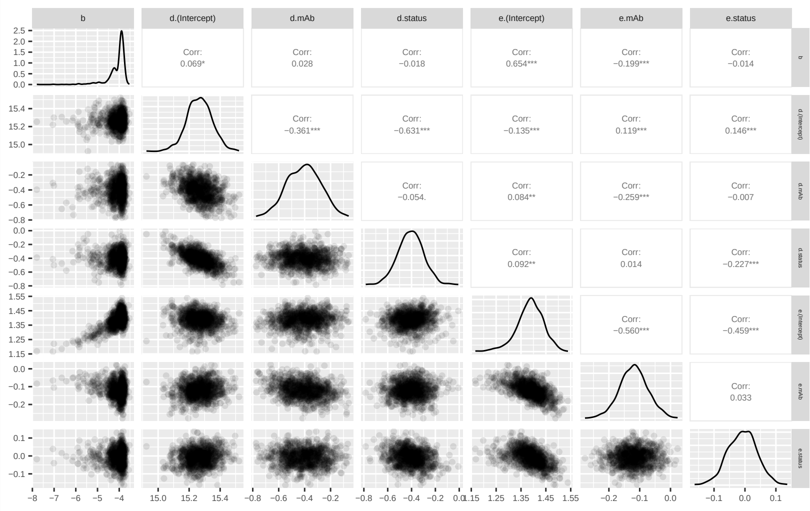
```
##               Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2nibf.gnls     1 16 127.3613 172.9938 -47.68066
## dnd2ni.gnls       2 18 140.8578 192.1944 -52.42891 1 vs 2 9.496499  0.0087
```

- And the comparison with the full model

```
##               Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## dnd2nibf.gnls     1 16 127.3613 172.9938 -47.68066
## dnd2.gnls         2 21 150.5069 210.3996 -54.25346 1 vs 2 13.14559  0.0221
```

- The model is different. But the collinearity on *b* must be fixed

Bootstrap with *{nlraa}* package / model without interactions and fixed *b* -> s51



Bootstrap: comments

- Very clearly there is something odd about the *b* parameter.
- The bootstrap clouds are by no mean round or elliptic.
- It can be suggested that a strong collinearity in present in *b* between the factors.
- Simplified model with *b* fixed looks better.
- Would be nice to see profile traces (see later with *nls()*)

Bootstrapped quantile CI / full model

Table 5: Bootstrapped vs classical confidence intervals, weighted

	Boots. 2.5%	Boots. 97.5%		W 2.5%	W 97.5%	
b.(Intercept)	-8.0083	-3.4751	***	-4.4587	-3.5874	***
b.mAba11	-6.1144	3.9552		-0.4465	0.8882	
b.statusndil	-2.8934	3.7191		-0.4478	0.6665	
b.mAba11:statusndil	-6.7247	6.7100		-0.7097	0.9244	
d.(Intercept)	15.1063	15.4619	***	15.1647	15.4311	***
d.mAba11	-0.8122	-0.1014	***	-0.7182	-0.1713	***
d.statusndil	-0.7153	-0.1855	***	-0.6271	-0.2503	***
d.mAba11:statusndil	-0.3871	0.6627		-0.2627	0.5120	
e.(Intercept)	1.1596	1.4932	***	1.3178	1.4684	***
e.mAba11	-0.3569	0.1541		-0.2072	0.0116	°
e.statusndil	-0.2323	0.2633		-0.0957	0.1128	
e.mAba11:statusndil	-0.3491	0.3596		-0.1576	0.1421	

Good coherence. But *e.mAba11* parameters is still divergent for CI and ANOVA III, see Table 3 / s28

Bootstrapped quantile CI / model without interactions

Table 6: Bootstrapped vs classical confidence intervals, weighted

	Boots. 2.5%	Boots. 97.5%		W 2.5%	W 97.5%	
b.(Intercept)	-7.7189	-3.5802	***	-4.4430	-3.7223	***
b.mAb	-4.3103	2.2860		-0.1000	0.6539	
b.status	-1.9829	2.9010		-0.2601	0.5219	
d.(Intercept)	15.0964	15.4342	***	15.1581	15.4070	***
d.mAb	-0.6732	-0.1266	***	-0.5777	-0.1929	***
d.status	-0.6318	-0.1868	***	-0.5730	-0.2456	***
e.(Intercept)	1.1465	1.4738	***	1.3262	1.4526	***
e.mAb	-0.3331	0.0765		-0.1718	-0.0283	***
e.status	-0.1678	0.1826		-0.0673	0.0769	

e.mAba11 parameters is still divergent, see [Table 4] / s29

Bootstrapped quantile CI / model without interactions and fixed *b*

Table 7: Bootstrapped vs classical confidence intervals, weighted

	Boots. 2.5%	Boots. 97.5%		W 2.5%	W 97.5%	
b	-5.3366	-3.7373	***	-4.2387	-3.8588	***
d.(Intercept)	15.0959	15.4221	***	15.1618	15.4076	***
d.mAb	-0.6619	-0.1588	***	-0.5927	-0.2136	***
d.status	-0.6260	-0.2007	***	-0.5731	-0.2505	***
e.(Intercept)	1.2588	1.4850	***	1.3339	1.4458	***
e.mAb	-0.2138	-0.0272	***	-0.1845	-0.0533	***
e.status	-0.1014	0.0843		-0.0718	0.0580	

Coherent result ! This last result is also coherent with the ANOVA III testing. See [Table 8] / s41

ANOVA I vs III vs CI, *b* fixed, no interaction,

Table 8: Anova I vs Anova III

	A-I p-value		A-III p-value		CI 2.5%	CI 97.5%	
b	0.0000	***	NA	NA	-4.2387	-3.8588	***
d.(Intercept)	0.0000	***	NA	NA	15.1618	15.4076	***
d.mAb	0.0130	***	0.0026	***	-0.5927	-0.2136	***
d.status	0.0000	***	0.0000	***	-0.5731	-0.2505	***
e.(Intercept)	0.0000	***	NA	NA	1.3339	1.4458	***
e.mAb	0.0004	***	0.0000	***	-0.1845	-0.0533	***
e.status	0.8304		0.5220		-0.0718	0.0580	

The *nls()* world

- Bootstrap shows that the *b* parameter has some oddities
- ANOVA III vs (confints and ANOVA II) divergency need an explication
- Profiling is a better alternative as classical confint
- Compare bootstrap CI with profiled CI
- Compare bootstrap pairs plot with trace plots
- *nls()* with factors is tricky but feasable.

Profiling only available in *nls()*

- For whole vector $H_0 : \theta = \theta^*$

$$T = (\frac{n-p}{p})\frac{S(\theta^*) - S(\hat{\theta})}{S(\hat{\theta})} \overset{approx.}{\sim} F_{p,n-p}$$

- For a single parameter test $H_0 : \theta_k = \theta_k^*$ fixing $\theta_k = \theta_k^*$ and minimizing $S(\theta)$ for $\theta_j, j \neq k$ called $\tilde{S}(\theta_k^*)$

$$\tilde{T}(\theta_k^*) = (n-p)\frac{\tilde{S}(\theta_k^*) - S(\hat{\theta})}{S(\hat{\theta})} \overset{approx.}{\sim} F_{1,n-p}$$

$$\tilde{T}(\theta_k^*) = sign(\theta_k - \theta_k^*)\frac{\sqrt{\tilde{S}(\theta_k^*) - S(\hat{\theta})}}{\hat{\sigma}} \overset{approx.}{\sim} t_{n-p}$$

nls() full model

- A little nightmare for the parenthesis typing. . .
- Full model without weighting. No automatic weighting system available like in *nlme()*
- Start values must be nameless

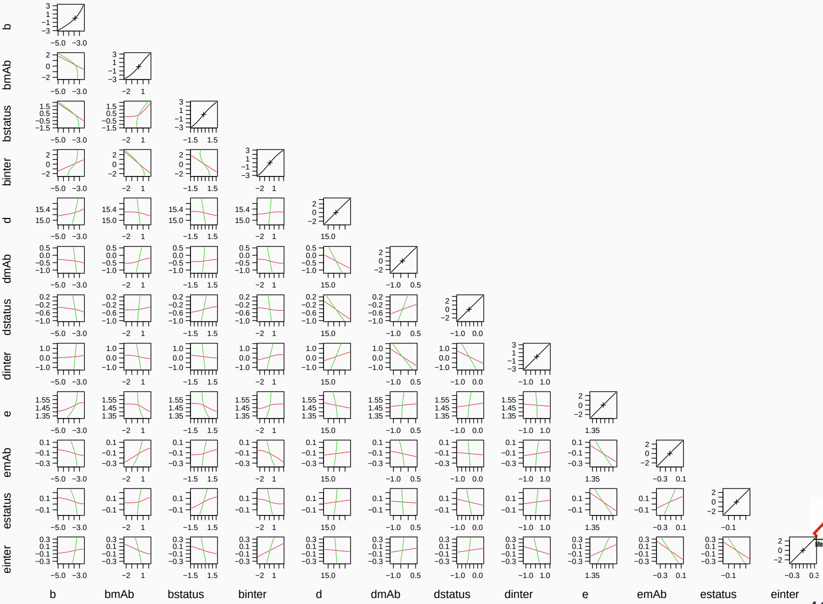
```
f.nls<-nls(rlog1p ~ (d + dmAb * (mAb == "a11") + dstatus * (status == "ndil") +
                       dinter * ((mAb == "a11") * (status == "ndil")) ) /
             (1 + exp( -(b + bmAb * ( mAb == "a11" ) + bstatus * ( status == "ndil" ) +
                           binter * ((mAb == "a11") * (status=="ndil")) ) *
             ( clog - ( e + emAb * ( mAb == "a11" ) + estatus * ( status == "ndil" ) +
                          einter * ( (mAb == "a11") * (status=="ndil")) ) ) ) ),
  data = dnd,
  #control = list(tol=5e-05),    # Only for profiling
  start = s.f.nls  )
```

nls() full model with weights

- *nls()* can receive weights
- *gnls()* weights exported and imported into the *nls()* function

```
w.f<-as.numeric(attr(dnd2.gnls$model$varStruct,"weights")) #export weights

fW.nls<-nls(rlog1p ~ (d + dmAb * (mAb == "a11") + dstatus * (status == "ndil") +
                       dinter * ((mAb == "a11") * (status == "ndil")) ) /
             (1 + exp( -(b + bmAb * ( mAb == "a11" ) + bstatus * ( status == "ndil" ) +
                           binter * ((mAb == "a11") * (status=="ndil")) ) *
             ( clog - ( e + emAb * ( mAb == "a11" ) + estatus * ( status == "ndil" ) +
                          einter * ( (mAb == "a11") * (status=="ndil")) ) ) ) ),
  data = dnd,
  weights = w.f, # import weights
  control = list(tol=5e-05),    # Only for profiling
  start = s.f.nls  )
```

Collinearity check with profile traces / comp s33



Comments on the profile traces

- Easier to read as bootstrap pairs plot?
- Complementary to bootstrap pairs plot. Not all violations are detected well. It's better to plot both.
- Collinearity of *b* factors in evidence.
- Other correlations also better detectable. Especially for a couple of interactions.
- On the diagonal the profile plots suggest the CI limits.

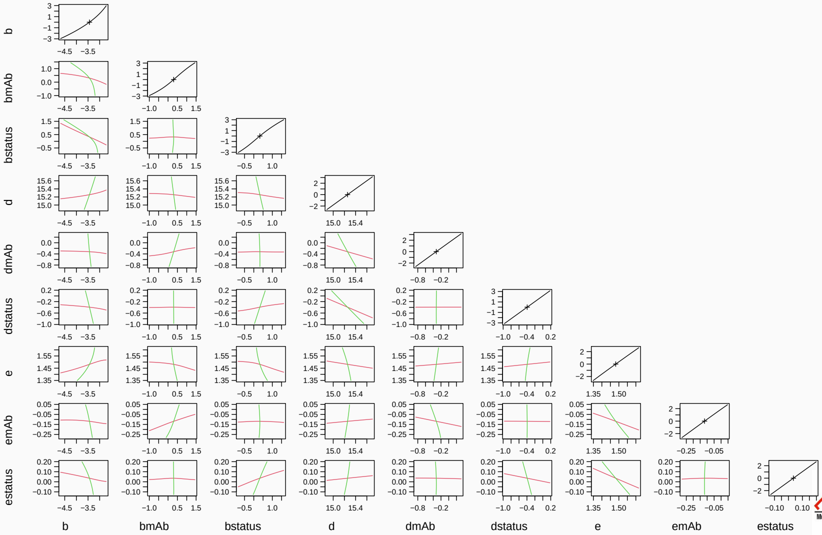
CI comparison for the full weighted model

Table 9: Bootstrapped vs classical vs profiled CI, weighted

	B 2.5%	B 97.5%		W 2.5%	W 97.5%		P 2.5%	P 97.5%	
b.(Intercept)	-8.008	-3.475	***	-4.459	-3.587	***	-4.386	-2.842	***
b.mAba11	-6.114	3.955		-0.446	0.888		-1.249	1.427	
b.statusndil	-2.893	3.719		-0.448	0.667		-0.714	1.360	
b.mAba11:statusndil	-6.725	6.710		-0.710	0.924		-1.375	1.774	
d.(Intercept)	15.106	15.462	***	15.165	15.431	***	14.971	15.581	***
d.mAba11	-0.812	-0.101	***	-0.718	-0.171	***	-0.964	0.192	
d.statusndil	-0.715	-0.186	***	-0.627	-0.250	***	-0.862	0.000	°
d.mAba11:statusndil	-0.387	0.663		-0.263	0.512		-0.687	0.956	
e.(Intercept)	1.160	1.493	***	1.318	1.468	***	1.368	1.597	***
e.mAba11	-0.357	0.154		-0.207	0.012	°	-0.294	0.062	
e.statusndil	-0.232	0.263		-0.096	0.113		-0.127	0.201	
e.mAba11:statusndil	-0.349	0.360		-0.158	0.142		-0.258	0.246	

Bootstrapped CI and profiled CI are not really coherent to each other. Is profiling a little more conservative?

No interaction model / comp s34

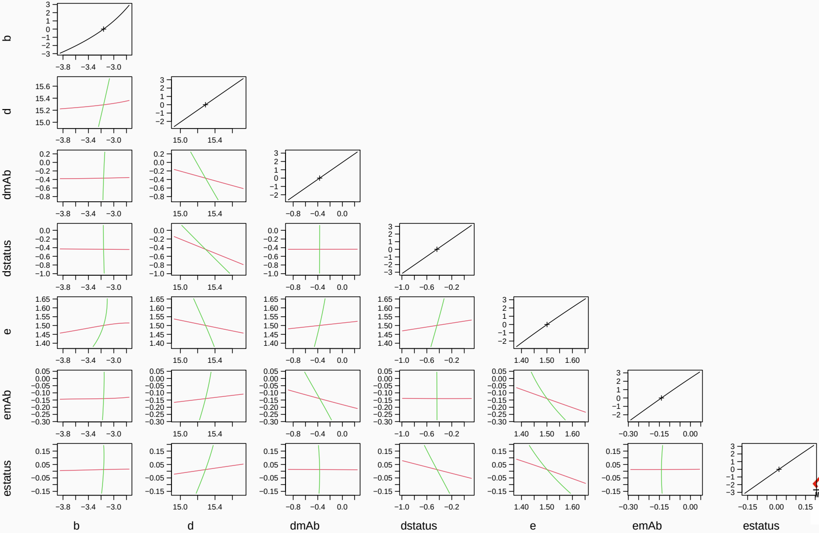


No interactions model confints

Table 10: Bootstrapped vs classical vs profiled CI, weighted

	B 2.5%	B 97.5%		W 2.5%	W 97.5%		P 2.5%	P 97.5%	
b	-6.652	-3.517	***	-4.443	-3.722	***	-4.204	-2.910	***
bmAb	-3.808	2.408		-0.100	0.654		-0.480	0.992	
bstatus	-2.163	2.575		-0.260	0.522		-0.379	1.117	
d	15.140	15.510	***	15.158	15.407	***	14.977	15.536	***
dmAb	-0.649	-0.108	***	-0.578	-0.193	***	-0.725	0.085	
dstatus	-0.649	-0.201	***	-0.573	-0.246	***	-0.756	-0.031	*
e	1.227	1.511	***	1.326	1.453	***	1.382	1.583	***
emAb	-0.339	0.069		-0.172	-0.028	***	-0.241	0.005	°
estatus	-0.171	0.182		-0.067	0.077		-0.087	0.158	

Boiling down to the simplest model, *b* fixed and no interactions / comp s36



Centro Professionale Sociosanitario
Medico-Tecnico Locarno

b fixed, complete CI comparison

Table 11: Bootstrapped vs classical vs profiled CI, weighted

	B 2.5%	B 97.5%		W 2.5%	W 97.5%		P 2.5%	P 97.5%	
b	-5.337	-3.737	***	-4.239	-3.859	***	-3.583	-2.866	***
d.(Intercept)	15.096	15.422	***	15.162	15.408	***	15.017	15.565	***
d.mAb	-0.662	-0.159	***	-0.593	-0.214	***	-0.755	0.018	◦
d.status	-0.626	-0.201	***	-0.573	-0.251	***	-0.787	-0.089	***
e.(Intercept)	1.259	1.485	***	1.334	1.446	***	1.411	1.595	***
e.mAb	-0.214	-0.027	***	-0.184	-0.053	***	-0.253	-0.024	***
e.status	-0.101	0.084		-0.072	0.058		-0.100	0.126	

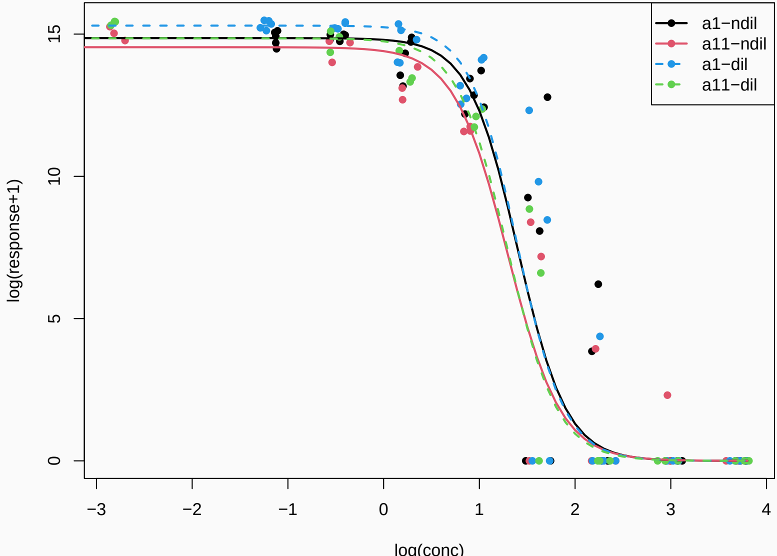
Profiling still looks more conservative.

ANOVA *b* fixed, no interaction

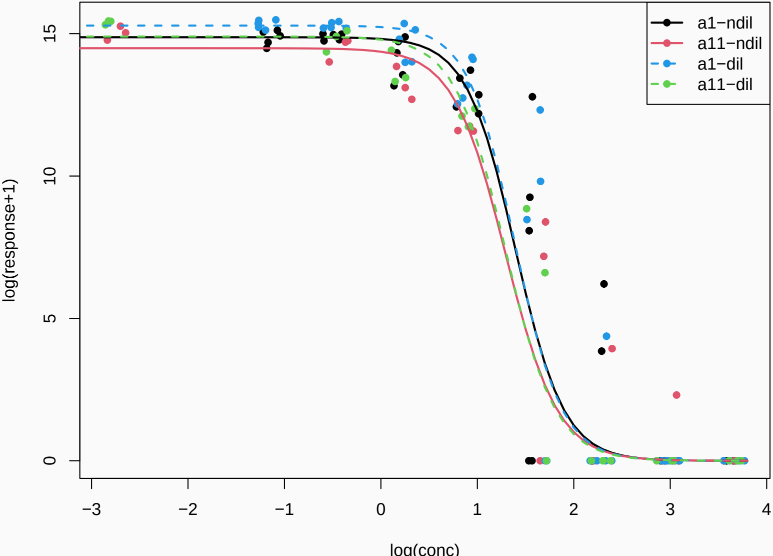
Table 12: Anova I vs Anova III

	A-I p-value		A-III p-value		CI 2.5%	CI 97.5%	
b	0.0000	***	NA	NA	-4.2387	-3.8588	***
d.(Intercept)	0.0000	***	NA	NA	15.1618	15.4076	***
d.mAb	0.0130	***	0.0026	***	-0.5927	-0.2136	***
d.status	0.0000	***	0.0000	***	-0.5731	-0.2505	***
e.(Intercept)	0.0000	***	NA	NA	1.3339	1.4458	***
e.mAb	0.0004	***	0.0000	***	-0.1845	-0.0533	***
e.status	0.8304		0.5220		-0.0718	0.0580	

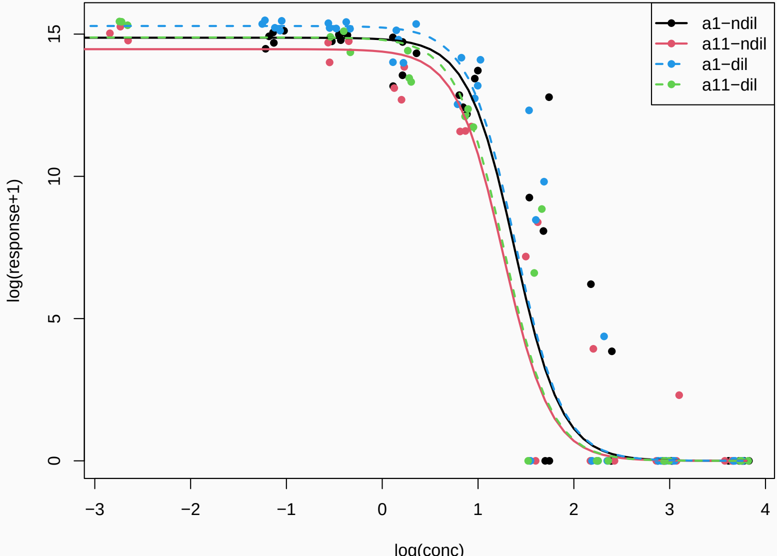
Plot of the fit / full model



Plot of the fit / model without interactions



Plot of the fit / model without interactions and *b* fixed



Take home messages

- Non linear regression can be daunting complex. If possible, use summary methods!
- Residual analysis is **very important** for nonlinear regression. Residuals have the same requisites as in the linear regression. Heteroscedasticity must be managed/corrected.
- *{nlme}* and *{drc}* have different parametrizations. Sometimes it is useful to switch.
- *{nlme}* can optimize residuals for weighted regression. The others alternative simply *do not have this ability*.
- *{nlme}* can fit nonlinear models with random effects. Other packages can not.
- *{nlme}* is lacking profiling tools. (Let's put it into Santa's wishlist).

Take home messages

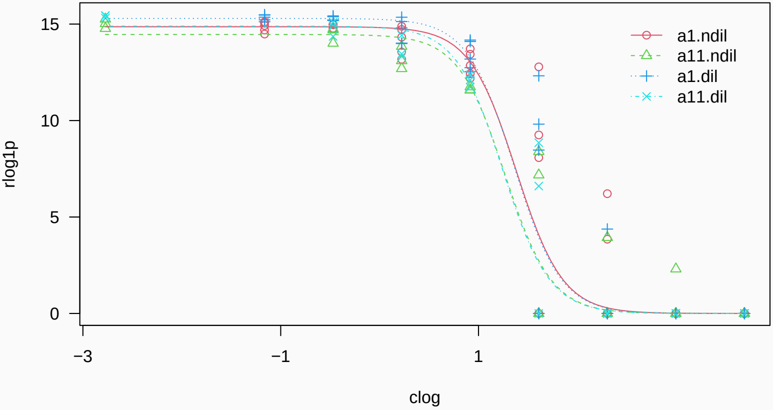
- *{nlraa}* provides a handy bootstrap infrastructure for *{nlme}*
- *{medrc}* is a package bridge package between *{drc}* and *{nlme}* to provide random effects for *{drc}* models.
- *{lme4}* via *nlmer()* also provide nonlinear regression with factors but needs random effects.
- I didn't dig into Bayesian methods, but there are viable approaches there too with packages like *{brms}*, *{rstanarm}* and perhaps others.

{drc} with weights / *b* fixed without interaction

```r
drm.s <- drm(rlog1p ~ clog, fct = L.3(), data = new.dnd,
                 curveid = interaction,
                 pmodels = c( ~ 1,  ~ mAb + status,   ~ mAb + status)) # Self starting with factors !

sW.drm <- coef(drm.s) # To be on the safe side, starting values for the weighted regression

drm.sW <- drm(rlog1p ~ clog, fct = L.3(), data = new.dnd,
                 curveid = interaction,
                 pmodels = c( ~ 1,  ~ mAb + status,   ~ mAb + status),
                 weights = w.f, # Imported weights, like for nls()
                 start = sW.drm
              )
```

Plot with *{drc}* (very handy)

```
plot(drm.sW,log="",type="all",col=2:5)
```



Comparison of the coefficients *{nlme}* vs *{drc}* vs *nls()*

Table 13: Coefficient comparison, unweighted regressions

	nlme	drc	nls
b:(Intercept)	-3.0035	3.0061	-3.0035
d:(Intercept)	15.3065	15.3087	15.3065
d:mAba11	-0.4308	-0.4357	-0.4308
d:statusndil	-0.5124	-0.5165	-0.5124
e:(Intercept)	1.4811	1.4803	1.4811
e:mAba11	-0.1007	-0.1001	-0.1007
e:statusndil	0.0341	0.0355	0.0341

Table 14: Coefficient comparison, weighted regressions

	nlme	drc	nls
b:(Intercept)	-4.0488	4.2447	-3.1605
d:(Intercept)	15.2847	15.2923	15.2904
d:mAba11	-0.4031	-0.4004	-0.3693
d:statusndil	-0.4118	-0.4304	-0.4378
e:(Intercept)	1.3898	1.3584	1.5005
e:mAba11	-0.1189	-0.1137	-0.1401
e:statusndil	-0.0069	0.0211	0.0122