Fitting nonlinear mixed models

February 14, 2013

Abstract

The nlraa package illustrates how to fit nonlinear mixed models with nlme and related packages. It also adds a few selfStart functions such as the beta growth function (Yin et al. 2003).

1 Introduction

The nlraa is distributed as part of a publication that illustrates the fit of nonlinear mixed models to be published in Agronomy Journal.

2 Example

We start by looking at biomass accumulation data from an experiment conducted in Greece. The dataset sm is included as part of the nlraa.

```
> data(sm)
> str(sm)

'data.frame':        236 obs. of  5 variables:
 $ DOY  : int  141 141 141 141 141 141 141 141 141 141 ...
 $ Block: int  1 2 3 4 1 2 3 4 1 2 ...
 $ Input: int  2 2 2 2 1 1 1 1 2 2 ...
 $ Crop : Factor w/ 3 levels "F","M","S": 2 2 2 2 2 2 2 2 3 3 ...
 $ Yield: num  0 0 0 0 0 0 0 0 0 0 ...

> sm
```

	DOY	Block	Input	Crop	Yield
1	141	1	2	M	0.00000
2	141	2	2	M	0.00000
3	141	3	2	M	0.00000
4	141	4	2	M	0.00000
5	141	1	1	M	0.00000
6	141	2	1	M	0.00000
7	141	3	1	M	0.00000
8	141	4	1	M	0.00000
9	141	1	2	S	0.00000
10	141	2	2	S	0.00000
11	141	3	2	S	0.00000
12	141	4	2	S	0.00000
13	141	1	1	S	0.00000
14	141	2	1	S	0.00000
15	141	3	1	S	0.00000
16	141	4	1	S	0.00000
17	141	1	2	F	0.00000
18	141	2	2	F	0.00000
19	141	3	2	F	0.00000
20	141	4	2	F	0.00000
21	141	1	1	F	0.00000
22	141	2	1	F	0.00000
23	141	3	1	F	0.00000
24	141	4	1	F	0.00000
25	168	1	2	M	0.42636
26	168	2	2	M	0.47368
27	168	3	2	M	0.44427
28	168	4	2	M	0.44810
29	168	1	1	M	0.42636
30	168	2	1	M	0.47368
31	168	3	1	M	0.44427
32	168	4	1	M	0.44810
33	168	1	2	S	0.55333
34	168	2	2	S	0.56757
35	168	3	2	S	0.56206
36	168	4	2	S	0.56099
37	168	1	1	S	0.55333

38	168	2	1	S	0.56757
39	168	3	1	S	0.56206
40	168	4	1	S	0.56099
41	168	1	2	F	0.45810
42	168	2	2	F	0.48466
43	168	3	2	F	0.47124
44	168	4	2	F	0.47133
45	168	1	1	F	0.47133
46	168	2	1	F	0.48466
47	168	3	1	F	0.47124
48	168	4	1	F	0.47575
49	176	1	2	M	2.37325
50	176	2	2	M	2.00316
51	176	3	2	M	2.19912
52	176	4	2	M	2.19184
53	176	1	2	S	1.10432
54	176	2	2	S	1.38752
55	176	3	2	S	1.25095
56	176	4	2	S	1.24760
57	176	1	2	F	1.21923
58	176	2	2	F	1.84674
59	176	3	2	F	1.67250
60	176	4	2	F	1.57949
61	185	1	2	M	3.38732
62	185	2	2	M	3.49804
63	185	3	2	M	3.38732
64	185	4	2	M	3.49804
65	185	1	1	M	1.93068
66	185	2	1	M	1.90740
67	185	3	1	M	1.93068
68	185	4	1	M	1.90740
69	185	1	2	S	1.41657
70	185	2	2	S	1.37778
71	185	3	2	S	1.41657
72	185	4	2	S	1.37778
73	185	1	1	S	1.05171
74	185	2	1	S	1.02630
75	185	3	1	S	1.05171

76	185	4	1	S	1.02630
77	185	1	2	F	3.78025
78	185	2	2	F	3.95644
79	185	3	2	F	3.78025
80	185	4	2	F	3.95644
81	185	1	1	F	1.54376
82	185	2	1	F	1.67455
83	185	3	1	F	1.54376
84	185	4	1	F	1.67455
85	196	1	2	M	6.59763
86	196	2	2	M	4.48241
87	196	3	2	M	5.95156
88	196	4	2	M	5.21500
89	196	1	1	M	5.05497
90	196	2	1	M	3.82129
91	196	3	1	M	2.32475
92	196	4	1	M	6.85365
93	196	1	2	S	9.63448
94	196	2	2	S	4.92300
95	196	3	2	S	6.65467
96	196	4	2	S	7.49728
97	196	1	1	S	4.63590
98	196	2	1	S	2.54400
99	196	3	1	S	2.42181
100	196	4	1	S	5.05650
101	196	1	2	F	4.82335
102	196	2	2	F	9.16699
103	196	3	2	F	7.51804
104	196	4	2	F	6.29618
105	196	1	1	F	5.74248
106	196	2	1	F	7.65420
107	196	3	1	F	9.30576
108	196	4	1	F	4.34633
109	211	1	2	M	12.39712
110	211	2	2	M	11.90800
111	211	3	2	M	13.41000
112	211	4	2	M	10.85442
113	211	1	1	M	9.95378

114	211	2	1	M	8.26690
115	211	3	1	M	10.37958
116	211	4	1	M	8.49705
117	211	1	2	S	12.27200
118	211	2	2	S	9.01893
119	211	3	2	S	8.94119
120	211	4	2	S	12.54775
121	211	1	1	S	7.42333
122	211	2	1	S	11.65094
123	211	3	1	S	7.39926
124	211	4	1	S	10.99033
125	211	1	2	F	16.07903
126	211	2	2	F	9.97832
127	211	3	2	F	13.92924
128	211	4	2	F	11.63735
129	211	1	1	F	15.29955
130	211	2	1	F	11.07560
131	211	3	1	F	13.02743
132	211	4	1	F	12.99200
133	225	1	2	M	18.97108
134	225	2	2	M	13.97785
135	225	3	2	M	20.12643
136	225	4	2	M	12.92471
137	225	1	1	M	11.86006
138	225	2	1	M	9.25771
139	225	3	1	M	8.91000
140	225	4	1	M	11.47636
141	225	1	2	S	20.81630
142	225	2	2	S	22.62614
143	225	3	2	S	26.43625
144	225	4	2	S	19.31603
145	225	1	1	S	16.46884
146	225	2	1	S	16.73960
147	225	3	1	S	17.17676
148	225	4	1	S	14.98377
149	225	1	2	F	21.22479
150	225	2	2	F	18.95059
151	225	3	2	F	18.06756

```
152 225     4     2    F 22.89504
153 225     1     1    F 17.94640
154 225     2     1    F 13.67034
155 225     3     1    F 16.94531
156 225     4     1    F 13.95996
157 243     1     2    M 22.25491
158 243     2     2    M 16.61565
159 243     3     2    M 21.82077
160 243     4     2    M 17.30287
161 243     1     1    M 12.59370
162 243     2     1    M 16.81818
163 243     3     1    M 15.77413
164 243     4     1    M 12.55910
165 243     1     2    S 27.04360
166 243     2     2    S 32.68473
167 243     3     2    S 24.24132
168 243     4     2    S 36.84177
169 243     1     1    S 22.58049
170 243     2     1    S 17.55427
171 243     3     1    S 24.20756
172 243     4     1    S 16.09398
173 243     1     2    F 36.68057
174 243     2     2    F 26.18346
175 243     3     2    F 31.59453
176 243     4     2    F 30.76946
177 243     1     1    F 22.20426
178 243     2     1    F 15.96902
179 243     3     1    F 24.52840
180 243     4     1    F 13.69994
181 263     1     2    M 21.43922
182 263     2     2    M 14.30769
183 263     3     2    M 18.69361
184 263     4     2    M 17.16525
185 263     1     1    M 12.24700
186 263     2     1    M 14.33091
187 263     3     1    M 14.93738
188 263     4     1    M 11.31081
189 263     1     2    S 46.63989
```

190	263	2	2	S	27.34436
191	263	3	2	S	21.64182
192	263	4	2	S	51.73975
193	263	1	1	S	36.63405
194	263	2	1	S	30.26200
195	263	3	1	S	45.29689
196	263	4	1	S	20.60917
197	263	1	2	F	36.54930
198	263	2	2	F	22.16076
199	263	3	2	F	30.65052
200	263	4	2	F	26.84545
201	263	1	1	F	26.59299
202	263	2	1	F	28.40977
203	263	3	1	F	24.73412
204	263	4	1	F	29.27921
205	283	1	2	S	40.03468
206	283	2	2	S	30.09828
207	283	3	2	S	28.16290
208	283	4	2	S	42.52825
209	283	1	1	S	30.06088
210	283	2	1	S	29.76598
211	283	3	1	S	38.43803
212	283	4	1	S	31.47936
213	283	1	2	F	38.22111
214	283	2	2	F	26.58728
215	283	3	2	F	38.43949
216	283	4	2	F	27.33577
217	283	1	1	F	27.03488
218	283	2	1	F	28.09636
219	283	3	1	F	29.34588
220	283	4	1	F	24.41846
221	303	1	2	S	34.43455
222	303	2	2	S	22.76971
223	303	3	2	S	26.45379
224	303	4	2	S	39.05722
225	303	1	1	S	23.20726
226	303	2	1	S	16.03402
227	303	3	1	S	29.30742

```
228 303     4     1    S 23.34675
229 303     1     2    F 37.65097
230 303     2     2    F 22.11074
231 303     3     2    F 29.16811
232 303     4     2    F 25.61200
233 303     1     1    F 26.22214
234 303     2     1    F 19.60814
235 303     3     1    F 30.39771
236 303     4     1    F 27.88253
```

The data represents `Yield` as harvested biomass for three crops: maize (`M`), fiber sorghum (`F`) and sweet sorghum (`S`).

Before starting with the model fit we need to manipulate the data by creating an index which describes the experimental unit (eu). We also delete the DOY 141 when crops where planted.

```
> sm$eu <- with(sm, factor(Block):factor(Input):factor(Crop))
> sm2 <- subset(sm, DOY != 141)
```

The next step is to create the `groupedData` which is a convenient structre to be used throughout the fitting process in nlme.

```
> smG <- groupedData(Yield ~ DOY | eu, data = sm2)
```

Originally, Danalatos et al. (2009) fitted the beta growth function as described by Yin et al. (2003). In nlraa we provide `selfStart` function `SSbgf` to improve the fitting process.

```
> fit.lis <- nlsList(Yield ~ SSbgf(DOY, w.max, t.e, t.m), data = smG)
```

There are three crops, two levels of agronomic input and four blocks which results in 24 possible combinations. We fitted the model to these 24 experimental units and obtained apparent convergence in only 10. This suggests that some modifications are needed.

```
> print(plot(fit.lis))
```

```
> print(plot(intervals(fit.lis)))
```

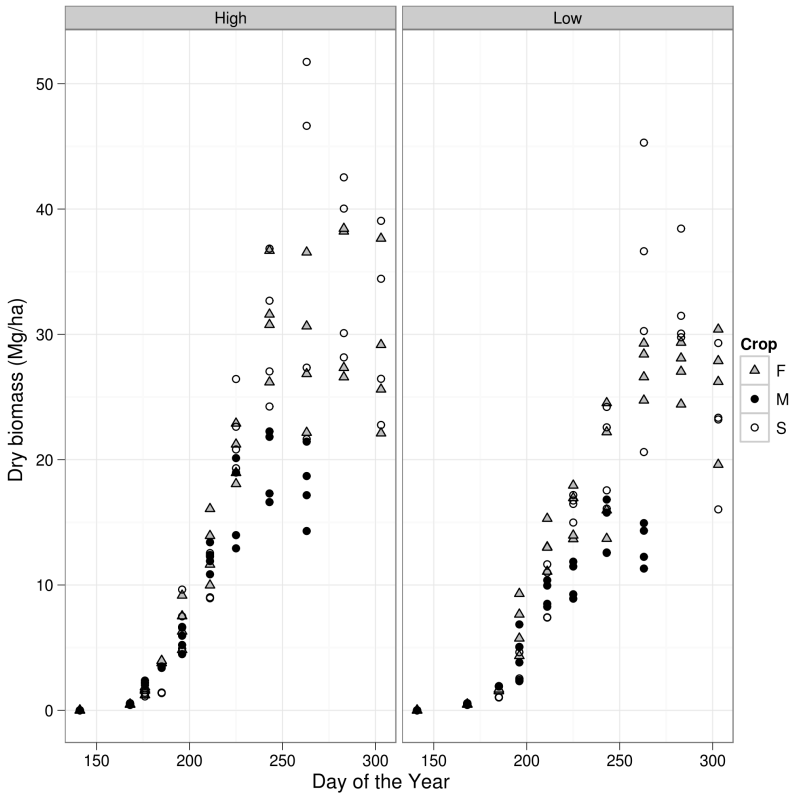

Figure 1: Biomass accumulation for three crops: maize (M), fiber sorghum (F) and sweet sorghum (S) collected in Greece in 2009. The two panels represent the level of agricultural input.

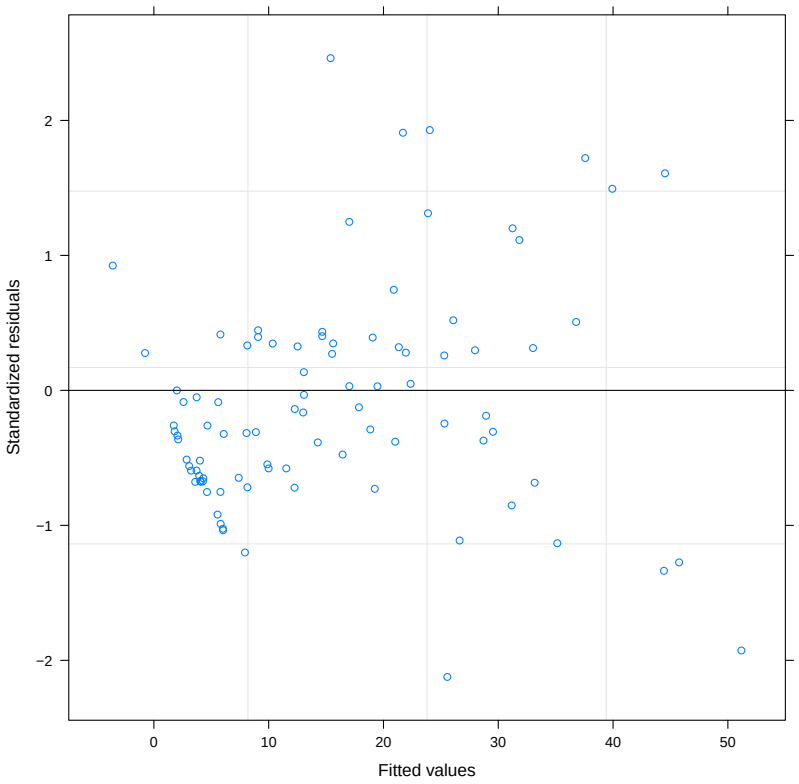

Figure 2: Residuals from individually fitting the beta growth function to all experimental units. There is evidence of bias for lower biomass values.

From the residuals plot we see some evidence of the inadequacy of the model. In particular the model over predicts at low values. We relax the convergence criteria to achieve a fitted model.

```
> fit.me <- nlme(fit.lis, control = list(minScale =1e-50, pnlsTol=0.01))

> print(plot(fit.me))


> print(plot(augPred(fit.me, level = 0:1)))
```

A modified beta growth function proposed by Yin et. al (2003) – included in the errata – allows for a delayed start of growth by modifying the t_b parameter.

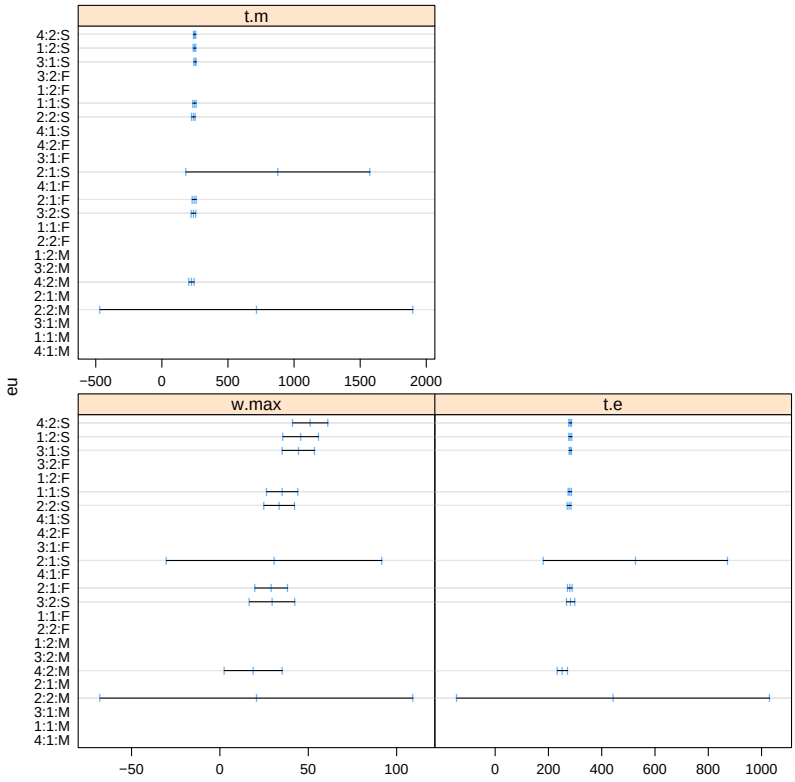

Figure 3: Confidence intervals from individually fitting the beta growth function to all experimental units.

$$y = w_b + (w_{max} - w_b)\left(1 + \frac{t_e - t}{t_e - t_m}\right)\left(\frac{t - t_b}{t_e - t_b}\right)^{\frac{t_e - t_b}{t_e - t_m}}$$

$$t_b < t_m < t_e$$

We include this as `bgf2` but not the selfStart version at this point. We also fix the w_b and the t_b parameters, so they are not part of the fitting process. There are good reasons for this: We know the initial biomass is minimal (seed weight) and we know the day of planting (it does not need to be optimized).

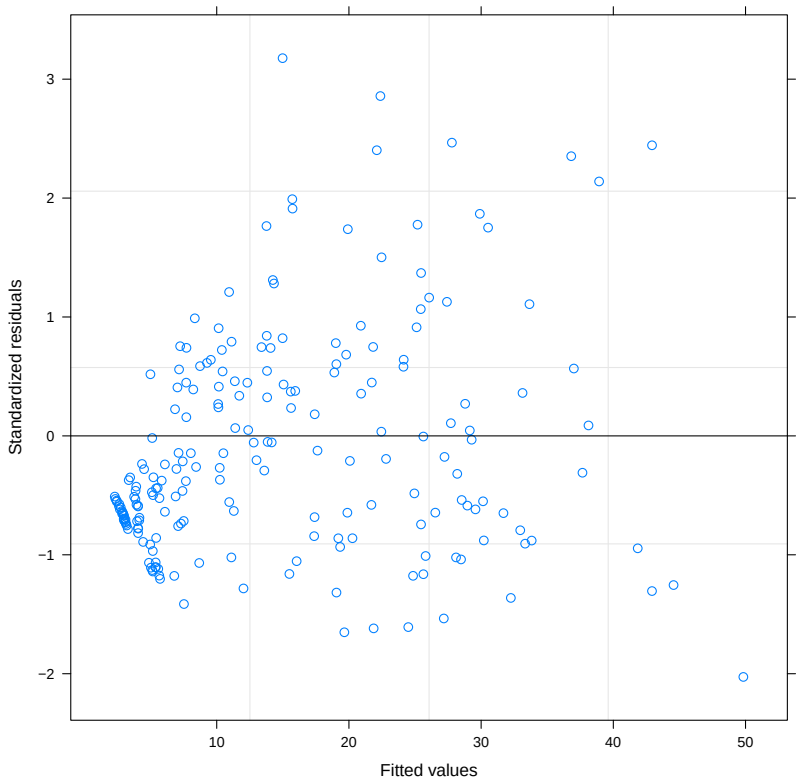

Figure 4: Residuals of the non-linear mixed model.

```
> fit.lis2 <- nlsList(Yield ~ bgf2(DOY, w.max, w.b = 0, t.e, t.m, t.b = 141),
+                     data = smG,
+                     start = c(w.max = 30, t.e=280, t.m=240))

> print(plot(fit.lis2))
```

Figure 6 shows a much lower bias at lower values.

We proceed to fit the non-linear mixed model and then we simplify the variance-covariance random effects structure.

```
> fit.me2 <- nlme(fit.lis2)
> fit2.me2 <- update(fit.me2, random = pdDiag(w.max + t.e + t.m ~ 1))
> anova(fit.me2, fit2.me2)
```

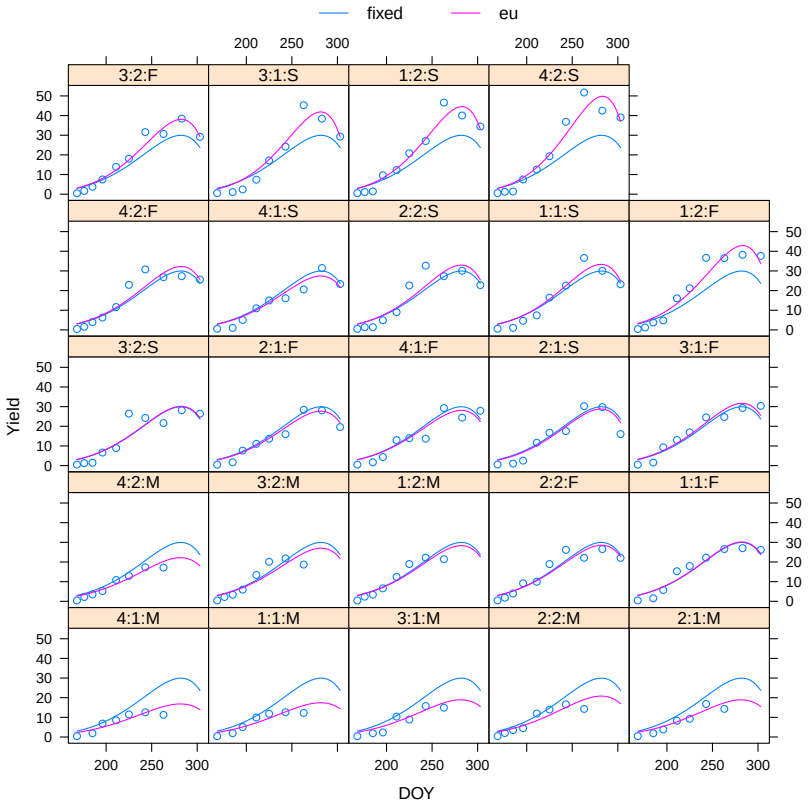

Figure 5: Fixed and experimental unit-level fits of the non-linear mixed model.

```
         Model df    AIC    BIC  logLik   Test L.Ratio p-value
fit.me2      1 10 1167.8 1201.3 -573.88
fit2.me2     2  7 1176.2 1199.7 -581.08 1 vs 2  14.401  0.0024
```

Some of the covariances might be significant, but we'll look at this later. We will next include the effects of Crop type and Input in the fixed part of the model. We want to know how the parameters are affected by the treatment effects.

```
> fe <- fixef(fit2.me2) ## Some starting values with visual help
> fit3.me2 <- update(fit2.me2, fixed = list(w.max + t.e + t.m ~ Crop),
+                    start = c(fe[1], -10, 20, fe[2], -40, 0, fe[3], -40, 0))
> ## We next include the Input
```

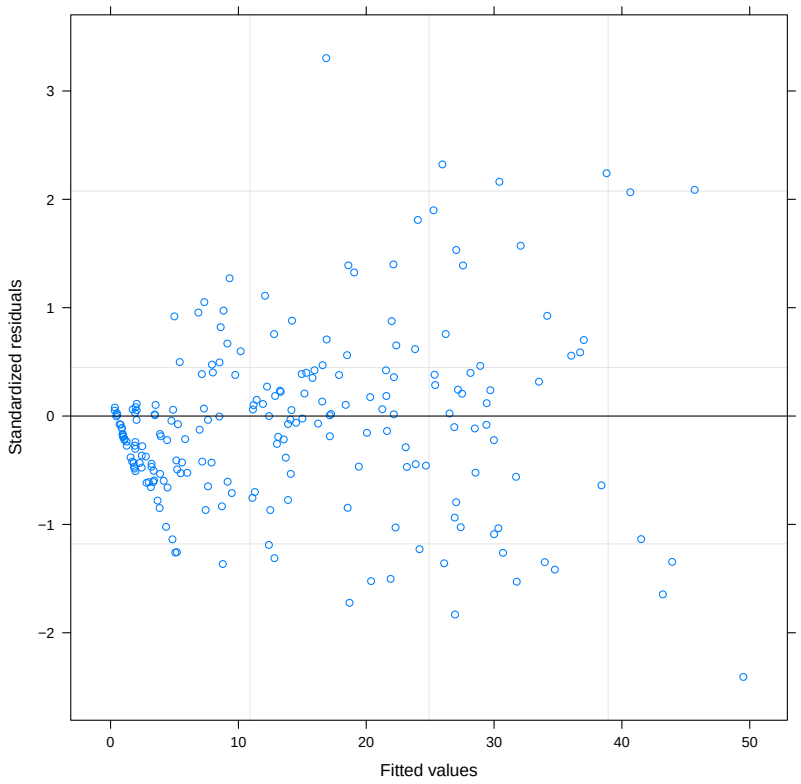

Figure 6: Residuals for the modified beta growth function for the individual fits.

```
> fe2 <- fixef(fit3.me2)
> fit4.me2 <- update(fit3.me2, fixed = list(w.max + t.e + t.m
+                                   ~ Crop + Input),
+                    start = c(fe2[1:3], 0, fe2[4:6], 0, fe2[7:9], 0))
> ## and the interaction
> fe3 <- fixef(fit4.me2)
> fit5.me2 <- update(fit4.me2,
+                     fixed = list(w.max + t.e + t.m
+                       ~ Crop + Input + Crop:Input),
+                    start = c(fe3[1:4], 0, 0,
+                              fe3[5:8], 0, 0,
+                              fe3[9:12], 0, 0))
```

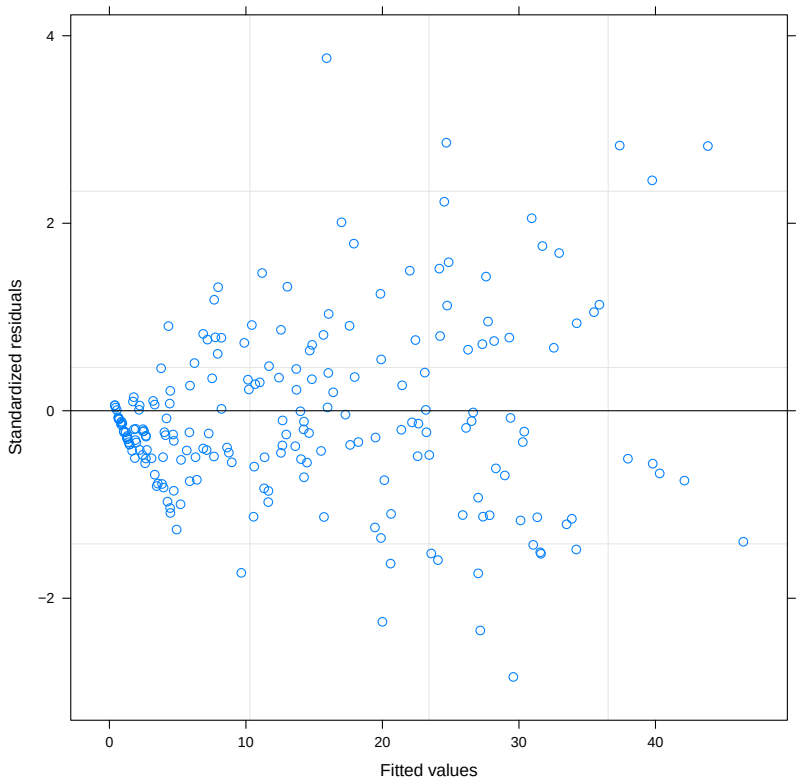

Figure 7: Residuals for the modified beta growth function for non-linear mixed model with the fixed effects of Crop and Input level.

The current model displays some evidence of unequal variance as shown in (Fig. 7). The amount of dispersion around zero is larger for low fitted values and the amount for large fitted values is larger.

```
> print(plot(fit5.me2))
```

We fit two models one where the variance depends on the Crop (since visually the crops are so different) and another one where it does not depend on the Crop.

```
> fit6.me2 <- update(fit5.me2,
+                    weights = varPower(form = ~ fitted(.) | Crop))
```

```
> fit7.me2 <- update(fit6.me2, weights = varPower(form = ~ fitted(.)))
> anova(fit6.me2, fit7.me2)

         Model df    AIC    BIC  logLik   Test L.Ratio p-value
fit6.me2     1 25 934.48 1018.4 -442.24
fit7.me2     2 23 940.50 1017.7 -447.25 1 vs 2  10.019  0.0067
```

Model `fit6.me2` is better according to the AIC criteria and the likelihood ratio test.

```
> fit6.me2

Nonlinear mixed-effects model fit by maximum likelihood
  Model: Yield ~ bgf2(DOY, w.max, w.b = 0, t.e, t.m, t.b = 141)
  Data: smG
  Log-likelihood: -442.24
  Fixed: list(w.max + t.e + t.m ~ Crop + Input + Crop:Input)
w.max.(Intercept)       w.max.CropM       w.max.CropS
        25.114128        -15.547062         -0.528663
      w.max.Input w.max.CropM:Input w.max.CropS:Input
         6.755663         -0.930456          2.536489
  t.e.(Intercept)         t.e.CropM         t.e.CropS
       281.724914        -32.063056         -2.321403
        t.e.Input   t.e.CropM:Input   t.e.CropS:Input
         -2.250049          1.432748          1.848762
  t.m.(Intercept)         t.m.CropM         t.m.CropS
       237.080755        -18.664416          3.572553
        t.m.Input   t.m.CropM:Input   t.m.CropS:Input
         -1.266612         -0.090739          0.664395

Random effects:
 Formula: list(w.max ~ 1, t.e ~ 1, t.m ~ 1)
 Level: eu
 Structure: Diagonal
        w.max.(Intercept) t.e.(Intercept) t.m.(Intercept)
StdDev:        2.2647e-09      0.00092412      2.2372e-08
        Residual
StdDev:   0.3479
```

```
Variance function:
 Structure: Power of variance covariate, different strata
 Formula: ~fitted(.) | Crop
 Parameter estimates:
      M       F       S
0.70278 0.85863 0.89604
Number of Observations: 212
Number of Groups: 24
```

Since random effects are almost zero. We remove them from the model and use the `gnls` function which is specifically written for models without random effects.

```
> ## Random effects are almost zero
> fit8.me2 <- gnls(Yield ~ bgf2(DOY, w.max, t.e, t.m, w.b=0, t.b=141),
+                  data = smG,
+                  params = list(w.max + t.e + t.m ~ Crop + Input
+                                                  + Crop:Input),
+                  weights = varPower(form = ~ fitted(.) | Crop),
+                  start = fixef(fit7.me2))
> anova(fit6.me2, fit8.me2)

         Model df    AIC    BIC  logLik   Test    L.Ratio
fit6.me2     1 25 934.48 1018.4 -442.24
fit8.me2     2 22 928.48 1002.3 -442.24 1 vs 2 0.00066557
         p-value
fit6.me2
fit8.me2       1
```

Model `fit8.me2` is better than `fit6.me2` according to AIC and BIC.

```
> anova(fit8.me2)

Denom. DF: 194
                  numDF F-value p-value
w.max.(Intercept)     1   11601  <.0001
w.max.Crop            2     902  <.0001
w.max.Input           1     441  <.0001
```

```
w.max.Crop:Input      2      34  <.0001
t.e.(Intercept)       1   23863  <.0001
t.e.Crop              2      90  <.0001
t.e.Input             1      33  <.0001
t.e.Crop:Input        2      59  <.0001
t.m.(Intercept)       1  127944  <.0001
t.m.Crop              2     139  <.0001
t.m.Input             1       1  0.4017
t.m.Crop:Input        2       0  0.9667
```

This shows that the Crop, Input and interaction are significant for all terms except for the `t.m` parameter.

Residuals look good with much less overprediction at lower values. The autocorrelation does not appear to be a concern (not shown).

```
> print(plot(fit8.me2))
```

We finalize the fitting exercise by plotting observed and predicted 8.

```
> smG$prds <- fitted(fit8.me2)
> doys <- 168:303
> ndat <- expand.grid(DOY=doys, Crop= unique(smG$Crop), Input=c(1,2))
> ndat$preds <- predict(fit8.me2, newdata = ndat)
> ndat2 <- ndat
> ndat2[ndat2$Crop == "M" & ndat2$DOY > 270,"preds"] <- NA
> ndat2 <- na.omit(ndat2)
```

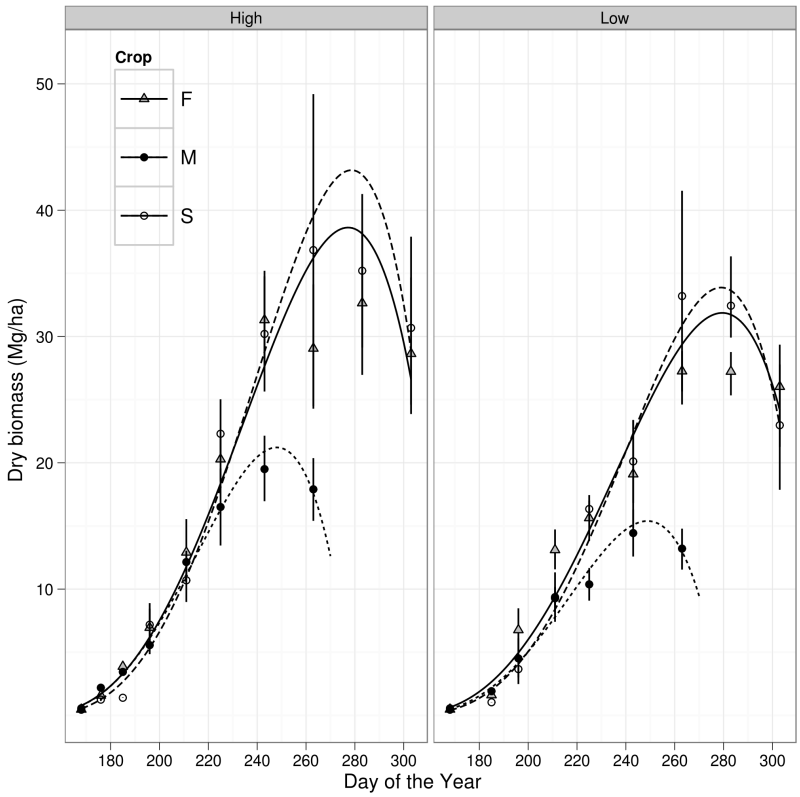

Figure 8: Observed data and fit for the final model.