Orange tree growth: A demonstration and evaluation of nonlinear mixed-effects models in R and ADMB

Arni Magnusson
Mark Maunder

30 January 2013

Contents

1 Introduction

This document serves two purposes:

- Demonstrate how simple nonlinear mixed-effects models can be fitted in R and AD Model Builder
- Evaluate the estimation performance of models implemented in R and AD Model Builder

2 Methods

2.1 Data

The data describe the growth of orange trees (Table 1, Figure 1). The trunk circumference of 5 trees is measured at 7 different ages, giving a total of 35 datapoints. These data were used as example data by Pinheiro and Bates (2000, Ch. 8). The `Orange` data object is among the core datasets that come with R.

Table 1. Growth of orange trees. Trunk circumference at breast height of 5 trees measured at 7 different ages.

Age	Circumference (mm)				
(days)	Tree 1	Tree 2	Tree 3	Tree 4	Tree 5
118	30	33	30	32	30
484	58	69	51	62	49
664	87	111	75	112	81
1004	115	156	108	167	125
1231	120	172	115	179	142
1372	142	203	139	209	174
1582	145	203	140	214	177

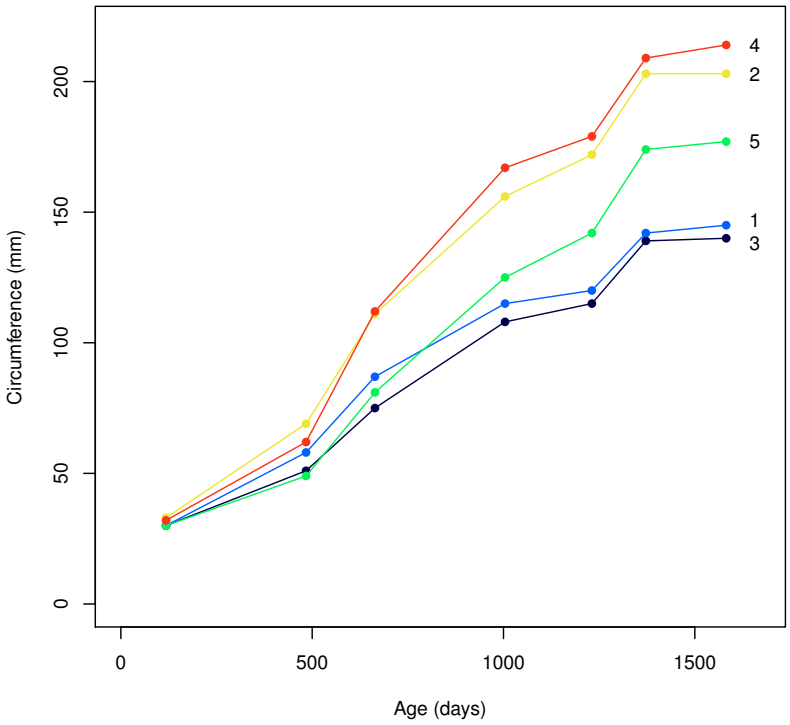

Figure 1. Growth of orange trees. Trunk circumference at breast height of 5 trees measured at 7 different ages. Tree numbers are shown on the right.

2.2 Model

Pinheiro and Bates (2000, pp. 356,360) used the following mixed-effects logistic model to analyze the orange tree growth data,

$$y_{ij} = \frac{\phi_1 + b_i}{1 + \exp[-(x_{ij} - \phi_2)/\phi_3]} + \epsilon_{ij}, \qquad \epsilon_{ij} \sim N(0, \sigma^2)$$

where y_{ij} is the circumference of tree i at time j, x is the age in days, ϕ_1 is the asymptote (maximum circumference), ϕ_2 is the inflection point (age when trees have reached half maximum circumference), and ϕ_3 is a scale parameter (days it takes to grow from 50% to 73% of maximum circumference[1]).

The asymptote varies between trees as a random effect:

$$b_i \sim N(0, \sigma_b^2)$$

The traces in Figure 1 can be used to get sensible starting values for the three parameters and set the initial asymptote $\phi_1 = 200$, the inflection point $\phi_2 = 800$, and the scale parameter $\phi_3 = 400$.

Symmetric confidence limits around the ϕ regression coefficients are constructed by multiplying the estimated standard error with the normal quantile z:

$$\mathrm{CI}_{\hat{\phi}_i} = \hat{\phi}_i \pm z_{\alpha/2}\, \widehat{\mathrm{SE}}_{\hat{\phi}_i}$$

Asymmetric confidence limits around the σ and σ_b standard deviations are based on the standard error of the log-transformed parameters:

$$\mathrm{CI}_{\hat{\sigma}} = \exp\Big(\log \hat{\sigma} \pm z_{\alpha/2}\, \widehat{\mathrm{SE}}_{\log \hat{\sigma}}\Big)$$

R

Implementing the model in R is easy after loading the 'nlme' package:

```
fm <- nlme(circumference ~ phi1/(1+exp(-(age-phi2)/phi3)),
           fixed=phi1+phi2+phi3 ~ 1, random=phi1 ~ 1|Tree,
           data=Orange, start=c(phi1=200,phi2=800,phi3=400))
```

There is also a "self-starting" `SSlogis` function in R, specifically for fitting logistic models, but the above is a basic general approach for any nonlinear mixed-effects model.

[1] The 73% comes from $1/[1+\exp(-1)]$.

ADMB

In ADMB, the data and model are in two different text files, and the initial parameter values are in a third text file.

The data are in a file called `ora.dat`,

```
# Number of trees (M)
5

# Number of ages (n)
7

# Age (x, in days)
118 484 664 1004 1231 1372 1582

# Circumference (y, in mm)
30 58  87 115 120 142 145
33 69 111 156 172 203 203
30 51  75 108 115 139 140
32 62 112 167 179 209 214
30 49  81 125 142 174 177
```

the model code is in a file called `ora.tpl`,

```
DATA_SECTION
  init_int M
  init_int n
  init_vector x(1,n)
  init_matrix y(1,M,1,n)

PARAMETER_SECTION
  init_number phi1
  init_number phi2
  init_number phi3
  init_number logSigma
  init_number logSigmaB(3)         // estimated in phase 3
  random_effects_vector b(1,M,2)   // estimated in phase 2
  sdreport_number sigma
  sdreport_number sigmaB
  matrix yfit(1,M,1,n)
  number RSS
  objective_function_value f

PROCEDURE_SECTION
  sigma = exp(logSigma);
  sigmaB = exp(logSigmaB);
  for (int i=1; i<=M; i++)
  {
    for (int j=1; j<=n; j++)
    {
      yfit(i,j) = (phi1 + b(i)) / (1.0 + exp(-(x(j)-phi2)/phi3));
    }
  }
  RSS = sum(square(y-yfit));
  f = 0.5*M*n*log(2.0*M_PI) + M*n*logSigma + RSS/(2.0*square(sigma));
  f += 0.5*M*log(2.0*M_PI) + M*logSigmaB + sum(square(b))/(2.0*square(sigmaB));
```

and the initial parameter values are in a file called `ora.pin`:

```
# phi1
200

# phi2
800

# phi3
400

# logSigma
1

# logSigmaB
0

# b (1,M)
0 0 0 0 0
```

The ADMB implementation is a simplistic one, not taking advantage of efficiency improvements such as separable functions and estimating unscaled random effects (Skaug and Fournier 2006, Fournier et al. 2012).

The model is compiled with the shell command

```
admb -r ora
```

and then run:

```
ora
```

2.3 Simulations

10 000 datasets are generated (Table 2) and the R and ADMB model implementations are evaluated in terms of computational speed, convergence, bias, and coverage probability.

Table 2. Parameter values used to simulate datasets for the second part of this study.

Parameter	Value
ϕ_1	191.05
ϕ_2	722.54
ϕ_3	344.15
σ	7.85
σ_b	31.48

3 Results

3.1 Model fit to original data

R

The R command

```
summary(fm)
```

summarizes the model fit,

```
Nonlinear mixed-effects model fit by maximum likelihood
  Model: circumference ~ phi1/(1 + exp(-(age - phi2)/phi3))
 Data: Orange
       AIC      BIC    logLik
  273.1693 280.9461 -131.5847

Random effects:
 Formula: phi1 ~ 1 | Tree
            phi1 Residual
StdDev: 31.48254 7.846255

Fixed effects: phi1 + phi2 + phi3 ~ 1
        Value Std.Error DF  t-value p-value
phi1 191.0455  16.15380 28 11.82666       0
phi2 722.5357  35.14849 28 20.55666       0
phi3 344.1529  27.14659 28 12.67757       0
 Correlation:
     phi1  phi2
phi2 0.375
phi3 0.354 0.755

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max
-1.9147537 -0.5351318  0.1436489  0.7309596  1.6615020

Number of Observations: 35
Number of Groups: 5
```

and

```
ranef(fm)
```

shows the random effects:

```
        phi1
3 -37.000971
1 -29.405325
5  -5.178094
2  31.564769
4  40.019621
```

ADMB

The ADMB executable produces several output files.

The negative log-likelihood and parameter estimates are in a file called `ora.par`,

```
# Number of parameters = 5  Objective function value = 131.572  Maximum gradient compon...

# phi1:
192.053262266
# phi2:
727.906256812
# phi3:
348.073016216
# logSigma:
2.05962317252
# logSigmaB:
3.45462142798
# b:
 -29.5622333483 31.7278488346 -37.1935580846 40.2245465263 -5.19720349330
```

and standard errors and correlations are in a file called `ora.cor`:

```
The logarithm of the determinant of the hessian = -11.7684
index   name        value      std.dev       1       2       3       4       5       6...
    1   phi1       1.9205e+02 1.5658e+01   1.0000
    2   phi2       7.2791e+02 3.5249e+01   0.3937  1.0000
    3   phi3       3.4807e+02 2.7080e+01   0.3732  0.7747  1.0000
    4   logSigma   2.0596e+00 1.2910e-01   0.0002  0.0001  0.0010  1.0000
    5   logSigmaB  3.4548e+00 3.2425e-01   0.0414  0.0987  0.0913 -0.0079  1.0000
    6   b         -2.9562e+01 1.4739e+01   0.8957  0.0671  0.0601 -0.0104  0.0323     ...
    7   b          3.1728e+01 1.4742e+01   0.8391 -0.0677 -0.0642  0.0111 -0.0344     ...
    8   b         -3.7194e+01 1.4759e+01   0.9007  0.0808  0.0749 -0.0130  0.0403     ...
    9   b          4.0225e+01 1.4767e+01   0.8301 -0.0864 -0.0796  0.0141 -0.0435     ...
   10   b         -5.1972e+00 1.4700e+01   0.8734  0.0064  0.0089 -0.0018  0.0053     ...
   11   sigma      7.8430e+00 1.0125e+00   0.0002  0.0001  0.0010  1.0000 -0.0079     ...
   12   sigmaB     3.1653e+01 1.0264e+01   0.0414  0.0987  0.0913 -0.0079  1.0000     ...
```

Alternatively, the point estimates and standard errors (without correlations) can be found in a file called `ora.std`.

In summary, the estimates are quite similar between R and ADMB (Table 3, Figure 2).

Table 3. Estimated parameters, random effects, and negative log likelihood of the model, as implemented in R and ADMB.

	R		ADMB	
	Value	95% CI	Value	95% CI
ϕ_1	191.05	(159.41, 222.69)	192.05	(161.36, 222.74)
ϕ_2	722.54	(653.69, 791.38)	727.91	(658.82, 797.00)
ϕ_3	344.15	(290.98, 397.32)	348.07	(294.99, 401.15)
σ	7.85	(6.09, 10.11)	7.84	(6.09, 10.10)
σ_b	31.48	(16.68, 59.43)	31.65	(16.76, 59.76)
b_1	−29.41		−29.56	
b_2	31.56		31.73	
b_3	−37.00		−37.19	
b_4	40.02		40.23	
b_5	−5.18		−5.20	
$-\log L$	131.58		131.57	

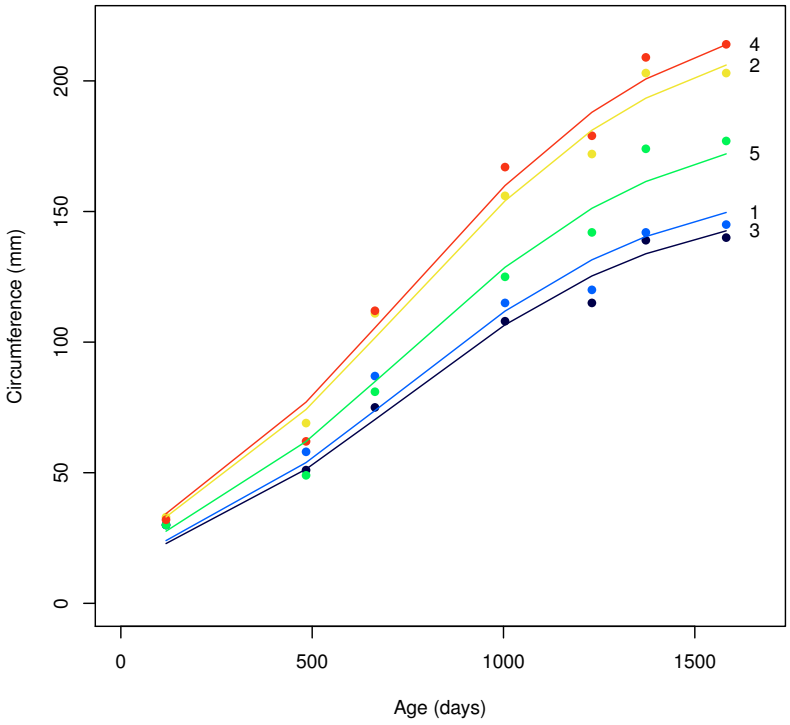

Figure 2. Model fit to the data. The fitted values from R and ADMB are indistinguishable in this figure. Tree numbers are shown on the right.

3.2 Performance with simulated data

Speed

Fitting 10 000 models took 7:37 minutes in R (0.05 sec/run) and 56:37 minutes in ADMB (0.34 sec/run) on an old laptop computer, so the model runs seven times faster in R than in ADMB. If a similar model was to be used in a computationally intensive simulation study, it would be worthwhile to take advantage of efficiency improvements such as separable functions and estimating unscaled random effects in ADMB (Skaug and Fournier 2006, Fournier et al. 2012).

Convergence

Out of 10 000 simulated datasets, 13 model runs did not converge, both in R and ADMB. In R, non-convergence was identified using the `intervals` function, where 13 models returned either an error or an upper σ_b confidence limit greater than 10^{10}. Likewise, non-convergence in ADMB was identified from the standard error of $\log \hat{\sigma}_b$ being either `NA` or greater than 10^{10}. This occurred in the same set of 13 simulated datasets.

To circumvent the problem of non-convergence, 13 additional simulated datasets were generated. The subsequent analysis of bias and coverage probability is therefore based on 10 000 converged simulations.

Bias

In both R and ADMB, the ϕ parameter estimates are unbiased, but σ and σ_b are underestimated with a relative bias of around -0.04 and -0.19 (Table 4, Figure 3). In terms of bias, the difference between R and ADMB is negligible.

Table 4. Median of 10 000 parameter estimates compared to the true parameter values used from the operating model. The relative bias is calculated as median$((\hat{\theta}-\theta)/\theta)$.

		R		ADMB	
	True	Median	Bias	Median	Bias
ϕ_1	191.05	190.78	0.00	191.59	0.00
ϕ_2	722.54	718.62	−0.01	723.31	0.00
ϕ_3	344.15	340.72	−0.01	344.04	0.00
σ	7.85	7.51	−0.04	7.51	−0.04
σ_b	31.48	25.52	−0.19	25.66	−0.18

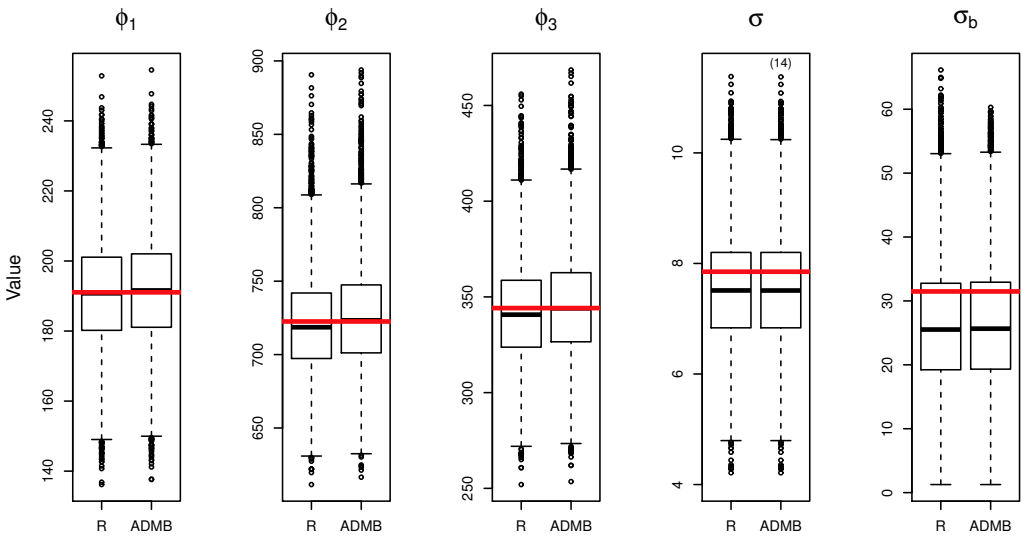

Figure 3. Distribution of point estimates (Tukey boxplots) compared to the true parameter value (red line) of each parameter. Fourteen σ estimates are outside the y-axis limits in the case of ADMB.

Coverage probability

Analysis of 10 000 confidence intervals the 90% confidence level shows that both R and ADMB generate confidence intervals that cover the true parameter less than 90% of the time (Table 5). The performance is poor for σ_b and ϕ_1, with coverage probability around 78% and 82%, but considerably better for the other parameters.

Table 5. Coverage probability of 90% confidence intervals generated using R and ADMB. Ideally, the coverage probability at this confidence level should be 0.900 for every parameter.

	R	ADMB
ϕ_1	0.824	0.813
ϕ_2	0.884	0.879
ϕ_3	0.886	0.877
σ	0.861	0.860
σ_b	0.783	0.787

Analysis of confidence levels ranging from 0 to 99% shows the same trends (Figure 4). Both R and ADMB generate confidence intervals that are too narrow, especially for σ_b and ϕ_1. Overall, R and ADMB show similar performance in terms of coverage probability: R performs slightly better for the ϕ parameters, but ADMB performs slightly better for the problematic σ_b parameter.

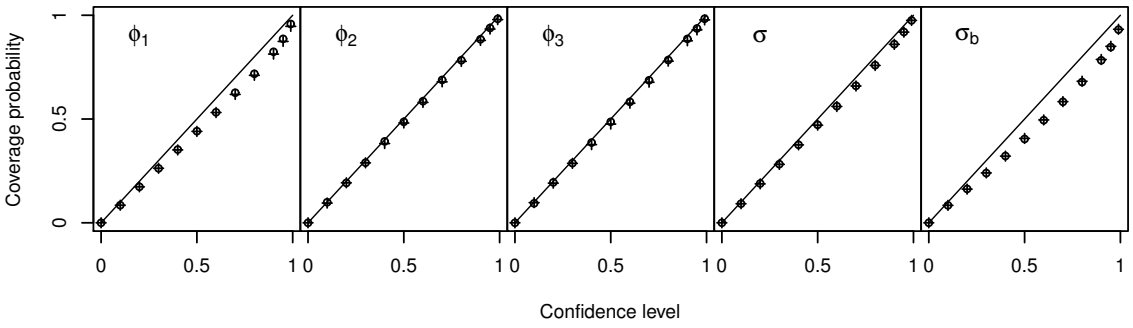

Figure 4. Coverage probability for confidence intervals evaluated at several confidence levels (0, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 99%). Each panel shows the performance of R (circle) and ADMB (cross) for a given parameter.

4 Discussion

We have demonstrated how simple nonlinear mixed-effects models can be fitted in R and AD Model Builder.

For this basic example, R and ADMB show similar estimation performance on the whole. To fit a mixed-effects logistic growth model to the orange tree data, it is easier and faster to use the 'nlme' package in R, yielding similar results as ADMB. One possible reason to use ADMB for analyzing this dataset might be to explore other modelling options (statistical assumptions and methods) that are not provided by the `nlme` function in R.

5 References

Fournier, D.A., H.J. Skaug, J. Ancheta, J. Ianelli, A. Magnusson, M.N. Maunder, A. Nielsen, and J. Sibert. 2012. AD Model Builder: Using automatic differentiation for statistical inference of highly parameterized complex nonlinear models. Optim. Methods Softw. 27:233–249.

Pinheiro, J.C. and D.M. Bates. 2000. Mixed-effects models in S and S-Plus. New York: Springer.

Skaug, H.J. and D.A. Fournier. 2006. Automatic approximation of the marginal likelihood in non-Gaussian hierarchical models. Comput. Stat. Data Anal. 51:699–709.