```
> library(MASS)
> library(Matrix)
> # library(mvtnorm)
> library(nlme)
> library(car)
> library(gdata)
> library(ggplot2)
> library(gdata)
> library(lattice)
>
################################################################################################
> # Leitura do conjunto de dados
>
> dados <- read.xls("/home/jmsinger/Desktop/exemp126.xls", header=T, na="NA")
>
>
################################################################################################
>
> # Definição de variáveis para ajuste dos perfis individuais
> x <- 26:40
> # Mudança de unidade de tempo de semanas para meses
>
> xc <- (x - mean(x)) / 4.29
> xc2 <- xc^2
> xc3 <- xc^3
> np <- 3      # número de parâmetros
>
################################################################################################
> # Formatação dos dados para efeito de análise
>
> min.aux <- c
("t26","t27","t28","t29","t30","t31","t32","t33","t34","t35","t36","t37",
+              "t38","t39","t40")
>
> aig.pig <- reshape(dados, direction = "long", varying = 3:17, v.names =
"diametro",
+                    timevar = "semanas", time = as.factor(min.aux))
> aig.pig <- aig.pig[order(aig.pig$ind), ]
> aig.pig$semanas <- rep(x, dim(dados)[1], each = T)
> aig.pig$meses <- rep(xc, dim(dados)[1], each = T)
> # aig.pig$subject <- aga.sga$indf ???????
>
> aig.pig <- na.exclude(aig.pig)
> # Gráficos de perfis
>
> perfis1 <- xyplot(diametro ~ semanas | grupo, groups = ind, pch = 16,
+             par.settings = standard.theme(quartz, color = F),
+             scales = list(x = list(relation = 'same'),
+                           y = list(relation = 'same')),
+             type = 'l',
+             lty = 1, as.table = TRUE,
+             panel = function(x, y, col, ...) {
+             panel.xyplot(x, y, col = col, ...)
+             panel.average(x, y, fun = mean, horizontal = F,
+                           lwd = 2, col = 'black', ...)
+             },
+             ylab = "Diâmetro sistólico da aorta / peso (mm/kg)",
+             xlab = "Semanas pós-concepção",
+             na.action = na.omit, data = aig.pig)
> perfis1
> # print(perfis1, split = c(1,1,1,2), more = TRUE)
> # print(perfis2, split = c(1,2,1,2), more = FALSE)
>
> ######################################################################
> # Ajuste do modelo identificado por meio da análise descritiva
> vecd <- groupedData(diametro ~ semanas | ind, data = aig.pig)
```

```
> fit1 <- lme(diametro ~ I(as.numeric(grupo == 'AIG')) +
+                I(as.numeric(grupo == 'PIG')) +
+                I(as.numeric(I(grupo == 'AIG'))*meses) +
+                I(as.numeric(I(grupo == 'PIG'))*meses) +
+                I(as.numeric(I(grupo == 'PIG'))*(meses^2)) - 1,
+              na.action = na.omit,
+              data = vecd,
+              random = list(ind = pdBlocked(list(
+                       pdSymm(~as.numeric(I(grupo == 'AIG')) +
+                            I(as.numeric(I(grupo == 'AIG'))*meses) - 1),
+                       pdSymm(~as.numeric(I(grupo == 'PIG')) +
+                         I(as.numeric(I(grupo == 'PIG'))*meses) +
+                         I(as.numeric(I(grupo == 'PIG'))*(meses^2)) - 1)
+                         ) )),
+               control = list(MaxIter = 100000, niterEM = 100000,
+                            msMaxIter = 100000,
+                            tolerance = 0.0001, msTol = 0.0000001,
+                            msMaxEval = 1000000))
> summary(fit1)
Linear mixed-effects model fit by REML
 Data: vecd
       AIC      BIC    logLik
  707.5362 762.1116 -338.7681

Random effects:
 Composite Structure: Blocked

 Block 1: as.numeric(I(grupo == "AIG")), I(as.numeric(I(grupo == "AIG")) * meses)
 Formula: ~as.numeric(I(grupo == "AIG")) + I(as.numeric(I(grupo == "AIG")) *
meses) - 1 | ind
 Structure: General positive-definite
                                        StdDev    Corr
as.numeric(I(grupo == "AIG"))           0.7517863 a.(I="
I(as.numeric(I(grupo == "AIG")) * meses) 0.4370641 -0.706

 Block 2: as.numeric(I(grupo == "PIG")), I(as.numeric(I(grupo == "PIG")) * meses),
I(as.numeric(I(grupo == "PIG")) * (meses^2))
 Formula: ~as.numeric(I(grupo == "PIG")) + I(as.numeric(I(grupo == "PIG")) *
meses) + I(as.numeric(I(grupo == "PIG")) * (meses^2)) - 1 | ind
 Structure: General positive-definite
                                          StdDev    Corr
as.numeric(I(grupo == "PIG"))             0.9449863 a.(I=" I(="*m
I(as.numeric(I(grupo == "PIG")) * meses)  0.9624369  0.239
I(as.numeric(I(grupo == "PIG")) * (meses^2)) 0.3573257 -0.769 -0.805
Residual                                  0.5511466

Fixed effects: diametro ~ I(as.numeric(grupo == "AIG")) + I(as.numeric(grupo
==      "PIG")) + I(as.numeric(I(grupo == "AIG")) * meses) + I(as.numeric(I(grupo
==      "PIG")) * meses) + I(as.numeric(I(grupo == "PIG")) * (meses^2)) -      1
                                             Value Std.Error  DF   t-value p-
value
I(as.numeric(grupo == "AIG"))                 6.044780 0.1495821  59  40.41111
0.000
I(as.numeric(grupo == "PIG"))                 7.037226 0.1844723  59  38.14788
0.000
I(as.numeric(I(grupo == "AIG")) * meses)     -1.233244 0.1232110 223 -10.00921
0.000
I(as.numeric(I(grupo == "PIG")) * meses)     -1.358617 0.2033881 223  -6.67992
0.000
I(as.numeric(I(grupo == "PIG")) * (meses^2)) -0.326623 0.1239731 223  -2.63463
0.009
 Correlation:
                                          I(.(=="A I(.(=="P I(.(I(=="A*m I(.(I
(=="P*m
I(as.numeric(grupo == "PIG"))
0.000
```

```
I(as.numeric(I(grupo == "AIG")) * meses)     -0.462
0.000
I(as.numeric(I(grupo == "PIG")) * meses)      0.000    0.198
0.000
I(as.numeric(I(grupo == "PIG")) * (meses^2))  0.000   -0.509    0.000
-0.626

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-3.51368470 -0.44953253 -0.01431129  0.42166247  3.16616065

Number of Observations: 286
Number of Groups: 61
> round(summary(fit1)$tTable, digits = 2)
                                            Value Std.Error  DF t-value p-value
I(as.numeric(grupo == "AIG"))                6.04      0.15  59   40.41    0.00
I(as.numeric(grupo == "PIG"))                7.04      0.18  59   38.15    0.00
I(as.numeric(I(grupo == "AIG")) * meses)    -1.23      0.12 223  -10.01    0.00
I(as.numeric(I(grupo == "PIG")) * meses)    -1.36      0.20 223   -6.68    0.00
I(as.numeric(I(grupo == "PIG")) * (meses^2)) -0.33      0.12 223   -2.63    0.01
> getVarCov(fit1,'random effects')
Random effects variance covariance matrix
                                            as.numeric(I(grupo == "AIG")) I
(as.numeric(I(grupo == "AIG")) * meses)
as.numeric(I(grupo == "AIG"))
0.56518                                  -0.23208
I(as.numeric(I(grupo == "AIG")) * meses)
-0.23208                                  0.19103
as.numeric(I(grupo == "PIG"))
0.00000                                   0.00000
I(as.numeric(I(grupo == "PIG")) * meses)
0.00000                                   0.00000
I(as.numeric(I(grupo == "PIG")) * (meses^2))
0.00000                                   0.00000
                                              as.numeric(I(grupo == "PIG")) I
(as.numeric(I(grupo == "PIG")) * meses)
as.numeric(I(grupo == "AIG"))
0.00000                                   0.00000
I(as.numeric(I(grupo == "AIG")) * meses)
0.00000                                   0.00000
as.numeric(I(grupo == "PIG"))
0.89300                                   0.21724
I(as.numeric(I(grupo == "PIG")) * meses)
0.21724                                   0.92628
I(as.numeric(I(grupo == "PIG")) * (meses^2))
-0.25964                                 -0.27667
                                              I(as.numeric(I(grupo == "PIG")) *
(meses^2))
as.numeric(I(grupo == "AIG"))
0.00000
I(as.numeric(I(grupo == "AIG")) * meses)
0.00000
as.numeric(I(grupo == "PIG"))
-0.25964
I(as.numeric(I(grupo == "PIG")) * meses)
-0.27667
I(as.numeric(I(grupo == "PIG")) * (meses^2))
0.12768
  Standard Deviations: 0.75179 0.43706 0.94499 0.96244 0.35733
> getVarCov(fit1,"conditional", individuals = 1)
ind 1
Conditional variance covariance matrix
         1       2       3       4       5       6       7       8       9      10
1  0.30376 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
2  0.00000 0.30376 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
3  0.00000 0.00000 0.30376 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
```

```
4  0.00000 0.00000 0.00000 0.30376 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
5  0.00000 0.00000 0.00000 0.00000 0.30376 0.00000 0.00000 0.00000 0.00000 0.00000
6  0.00000 0.00000 0.00000 0.00000 0.00000 0.30376 0.00000 0.00000 0.00000 0.00000
7  0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.30376 0.00000 0.00000 0.00000
8  0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.30376 0.00000 0.00000
9  0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.30376 0.00000
10 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.30376
  Standard Deviations: 0.55115 0.55115 0.55115 0.55115 0.55115 0.55115 0.55115
0.55115 0.55115 0.55115
> round(AIC(fit1),digits = 1)
[1] 707.5
> round(BIC(fit1),digits = 1)
[1] 762.1
> round(logLik(fit1),digits = 1)
'log Lik.' -338.8 (df=15)
```