**The environmental variance model**

Fitting mixed models has become very common in biology and recent developments involve the manipulation of the variance-covariance matrix for random effects and residuals. To the best of my knowledge, within the frame of frequentist methods, the only freeware solution in R should be based on the ‘nlme’ package, as the ‘lmer’ package does not easily permit such manipulations. The ‘nlme’ package is fully described in Pinheiro and Bates (2000). Of course, the ‘asreml’ package can be used, but, unfortunately, this is not freeware.

Coding mixed models in ‘nlme’ is not always easy, especially when we have crossed random effects, which is very common with agricultural experiments. I have been struggling with this issue very often in the last years and I thought it might be useful to publish a few examples in this blog, to save collegues from a few headaches. Please, note that I have already published other posts dealing with the use of the ‘lme()’ function in the ‘nlme’ package.

The first example in this series relates to a randomised complete block design with three replicates, comparing winter wheat genotypes. The experiment was repeated in seven years in the same location. The dataset (‘WinterWheat’) is available in the ‘aomisc’ package, which is the companion package for this blog and it is available on gitHub. Information on how to download and install the ‘aomisc’ package are given. Please, note that this dataset shows the data for eight genotypes, but the model that we want to fit requires that the number of environments is higher than the number of genotypes. Therefore, we have to make a subset, at the beginning, removing a couple of genotypes.

The first code snippet loads the ‘aomisc’ package and other necessary packages. Afterwards, it loads the ‘WinterWheat’ dataset, subsets it and turns the ‘Genotype’, ‘Year’ and ‘Block’ variables into factors.

library(plyr)

library(nlme)

library(aomisc)

data(WinterWheat)

WinterWheat <- WinterWheat[WinterWheat$Genotype != "SIMETO" & WinterWheat$Genotype != "SOLEX",]

WinterWheat$Genotype <- factor(WinterWheat$Genotype)

WinterWheat$Year <- factor(WinterWheat$Year)

WinterWheat$Block <- factor(WinterWheat$Block)

head(WinterWheat, 10)

## Plot Block Genotype Yield Year

## 1 2 1 COLOSSEO 6.73 1996

## 2 1 1 CRESO 6.02 1996

## 3 50 1 DUILIO 6.06 1996

## 4 49 1 GRAZIA 6.24 1996

## 5 63 1 IRIDE 6.23 1996

## 6 32 1 SANCARLO 5.45 1996

## 9 110 2 COLOSSEO 6.96 1996

## 10 137 2 CRESO 5.34 1996

## 11 91 2 DUILIO 5.57 1996

## 12 138 2 GRAZIA 6.09 1996

Dealing with the above dataset, a good candidate model for data analyses is the so-called ‘environmental variance model’. This model is often used in stability analyses for multi-environment experiments and I will closely follow the coding proposed in Piepho (1999):

\[y\_{ijk} = \mu + g\_i + r\_{jk} + h\_{ij} + \varepsilon\_{ijk}\]

where \(y\_{ijk}\) is yield (or other trait) for the \(k\)-th block, \(i\)-th genotype and \(j\)-th environment, \(\mu\) is the intercept, \(g\_i\) is the effect for the i-th genotype, \(r\_{jk}\) is the effect for the \(k\)-th block in the \(j\)-th environment, \(h\_{ij}\) is a random deviation from the expected yield for the \(i\)-th genotype in the \(j\)-th environment and \(\varepsilon\_{ijk}\) is the residual variability of yield between plots, within each environment and block.

We usually assume that \(r\_{jk}\) and \(\varepsilon\_{ijk}\) are independent and normally distributed, with variances equal to \(\sigma^2\_r\) and \(\sigma^2\_e\), respectively. Such an assumption may be questioned, but we will not do it now, for the sake of simplicity.

Let’s concentrate on \(h\_{ij}\), which we will assume as normally distributed with variance-covariance matrix equal to \(\Omega\). In particular, it is reasonable to expect that the genotypes will have different variances across environments (heteroscedasticity), which can be interpreted as static stability measures (‘environmental variances’; hence the name ‘environmental variance model’). Furthermore, it is reasonable that, if an environment is good for one genotype, it may also be good for other genotypes, so that yields in each environment are correlated, although the correlations can be different for each couple of genotypes. To reflect our expectations, the \(\Omega\) matrix needs to be totally unstructured, with the only constraint that it is positive definite.

Piepho (1999) has shown how the above model can be coded by using SAS and I translated his code into R.

EnvVarMod <- lme(Yield ~ Genotype,

random = list(Year = pdSymm(~Genotype - 1),

Year = pdIdent(~Block - 1)),

control = list(opt = "optim", maxIter = 100),

data=WinterWheat)

VarCorr(EnvVarMod)

## Variance StdDev Corr

## Year = pdSymm(Genotype - 1)

## GenotypeCOLOSSEO 0.48876512 0.6991174 GCOLOS GCRESO GDUILI

## GenotypeCRESO 0.70949309 0.8423141 0.969

## GenotypeDUILIO 2.37438440 1.5409038 0.840 0.840

## GenotypeGRAZIA 1.18078525 1.0866394 0.844 0.763 0.942

## GenotypeIRIDE 1.23555204 1.1115539 0.857 0.885 0.970

## GenotypeSANCARLO 0.93335518 0.9661031 0.928 0.941 0.962

## Year = pdIdent(Block - 1)

## Block1 0.02748257 0.1657787

## Block2 0.02748257 0.1657787

## Block3 0.02748257 0.1657787

## Residual 0.12990355 0.3604214

##

## Year =

## GenotypeCOLOSSEO GGRAZI GIRIDE

## GenotypeCRESO

## GenotypeDUILIO

## GenotypeGRAZIA

## GenotypeIRIDE 0.896

## GenotypeSANCARLO 0.884 0.942

## Year =

## Block1

## Block2

## Block3

## Residual

I coded the random effects as a list, by using the ‘Year’ as the nesting factor (Galecki and Burzykowski, 2013). In order to specify a totally unstructured variance-covariance matrix for the genotypes within years, I used the ‘pdMat’ construct ‘pdSymm()’. This model is rather complex and may take long to converge.

The environmental variances are retrieved by the following code:

envVar <- as.numeric ( VarCorr(EnvVarMod)[2:7,1] )

envVar

## [1] 0.4887651 0.7094931 2.3743844 1.1807853 1.2355520 0.9333552

while the correlations are given by:

VarCorr(EnvVarMod)[2:7,3:7]

## Corr

## GenotypeCOLOSSEO "GCOLOS" "GCRESO" "GDUILI" "GGRAZI" "GIRIDE"

## GenotypeCRESO "0.969" "" "" "" ""

## GenotypeDUILIO "0.840" "0.840" "" "" ""

## GenotypeGRAZIA "0.844" "0.763" "0.942" "" ""

## GenotypeIRIDE "0.857" "0.885" "0.970" "0.896" ""

## GenotypeSANCARLO "0.928" "0.941" "0.962" "0.884" "0.942"

**Unweighted two-stage fitting**

In his original paper, Piepho (1999) also gave SAS code to analyse the means of the ‘genotype x environment’ combinations. Indeed, agronomists and plant breeders often adopt a two-steps fitting procedure: in the first step, the means across blocks are calculated for all genotypes in all environments. In the second step, these means are used to fit an environmental variance model. This two-step process is less demanding in terms of computer resources and it is correct whenever the experiments are equireplicated, with no missing ‘genotype x environment’ combinations. Furthermore, we need to be able to assume similar variances within all experiments.

I would also like to give an example of this two-step analysis method. In the first step, we can use the ‘ddply()’ function in the package ‘plyr’:

#First step

WinterWheatM <- ddply(WinterWheat, c("Genotype", "Year"),

function(df) c(Yield = mean(df$Yield)) )

Once we have retrieved the means for genotypes in all years, we can fit the following model:

\[y\_{ij} = \mu + g\_i + a\_{ij}\]

where \(y\_{ij}\) is the mean yield for the \(i\)-th genotype in the \(j\)-th environment and \(a\_{ij}\) is the residual term, which includes the genotype x environment random interaction, the block x environment random interaction and the residual error term.

In this model we have only one random effect (\(a\_{ij}\)) and, therefore, this is a fixed linear model. However, we need to model the variance-covariance matrix of residuals (\(R\)), by adopting a totally unstructured form. Please, note that, when working with raw data, we have modelled \(\Omega\), i.e. the variance-covariance matrix for the random effects. I have used the ‘gls()’ function, together with the ‘weights’ and ‘correlation’ arguments. See the code below.

#Second step

envVarModM <- gls(Yield ~ Genotype,

data = WinterWheatM,

weights = varIdent(form=~1|Genotype),

correlation = corSymm(form=~1|Year))

summary(envVarModM)

## Generalized least squares fit by REML

## Model: Yield ~ Genotype

## Data: WinterWheatM

## AIC BIC logLik

## 80.6022 123.3572 -13.3011

##

## Correlation Structure: General

## Formula: ~1 | Year

## Parameter estimate(s):

## Correlation:

## 1 2 3 4 5

## 2 0.947

## 3 0.809 0.815

## 4 0.816 0.736 0.921

## 5 0.817 0.866 0.952 0.869

## 6 0.888 0.925 0.949 0.856 0.907

## Variance function:

## Structure: Different standard deviations per stratum

## Formula: ~1 | Genotype

## Parameter estimates:

## COLOSSEO CRESO DUILIO GRAZIA IRIDE SANCARLO

## 1.000000 1.189653 2.143713 1.528848 1.560620 1.356423

##

## Coefficients:

## Value Std.Error t-value p-value

## (Intercept) 6.413333 0.2742314 23.386574 0.0000

## GenotypeCRESO -0.439524 0.1107463 -3.968746 0.0003

## GenotypeDUILIO 0.178571 0.3999797 0.446451 0.6579

## GenotypeGRAZIA -0.330952 0.2518270 -1.314205 0.1971

## GenotypeIRIDE 0.281905 0.2580726 1.092347 0.2819

## GenotypeSANCARLO -0.192857 0.1802547 -1.069915 0.2918

##

## Correlation:

## (Intr) GCRESO GDUILI GGRAZI GIRIDE

## GenotypeCRESO 0.312

## GenotypeDUILIO 0.503 0.371

## GenotypeGRAZIA 0.269 -0.095 0.774

## GenotypeIRIDE 0.292 0.545 0.857 0.638

## GenotypeSANCARLO 0.310 0.612 0.856 0.537 0.713

##

## Standardized residuals:

## Min Q1 Med Q3 Max

## -2.0949678 -0.5680656 0.1735444 0.7599596 1.3395000

##

## Residual standard error: 0.7255481

## Degrees of freedom: 42 total; 36 residual

The variance-covariance matrix for residuals can be obtained using the ‘getVarCov()’ function in the ‘nlme’ package, although I had to discover that there is a small buglet there, which causes problems in some instances (such as here). Please, [see this link](https://www.jepusto.com/bug-in-nlme-getvarcov/); I have included the correct code in the ‘getVarCov.gls()’ function in the ‘aomisc’ package, that is the companion package for this blog.

R <- getVarCov.gls(envVarModM)

R

## Marginal variance covariance matrix

## [,1] [,2] [,3] [,4] [,5] [,6]

## [1,] 0.52642 0.59280 0.91285 0.65647 0.67116 0.63376

## [2,] 0.59280 0.74503 1.09440 0.70422 0.84652 0.78560

## [3,] 0.91285 1.09440 2.41920 1.58850 1.67700 1.45230

## [4,] 0.65647 0.70422 1.58850 1.23040 1.09160 0.93442

## [5,] 0.67116 0.84652 1.67700 1.09160 1.28210 1.01070

## [6,] 0.63376 0.78560 1.45230 0.93442 1.01070 0.96855

## Standard Deviations: 0.72555 0.86315 1.5554 1.1093 1.1323 0.98415

As the design is perfectly balanced, the diagonal elements of the above matrix correspond to the variances of genotypes across environments:

tapply(WinterWheatM$Yield, WinterWheatM$Genotype, var)

## COLOSSEO CRESO DUILIO GRAZIA IRIDE SANCARLO

## 0.5264185 0.7450275 2.4191624 1.2304397 1.2821143 0.9685497

which can also be retreived by the ‘stability’ package:

library(stability)

## Registered S3 methods overwritten by 'lme4':

## method from

## cooks.distance.influence.merMod car

## influence.merMod car

## dfbeta.influence.merMod car

## dfbetas.influence.merMod car

envVarStab <-

stab\_measures(

.data = WinterWheatM,

.y = Yield,

.gen = Genotype,

.env = Year

)

envVarStab$StabMeasures

## # A tibble: 6 x 7

## Genotype Mean GenSS Var CV Ecov ShuklaVar

##

## 1 COLOSSEO 6.41 3.16 0.526 11.3 1.25 0.258

## 2 CRESO 5.97 4.47 0.745 14.4 1.01 0.198

## 3 DUILIO 6.59 14.5 2.42 23.6 2.31 0.522

## 4 GRAZIA 6.08 7.38 1.23 18.2 1.05 0.208

## 5 IRIDE 6.70 7.69 1.28 16.9 0.614 0.0989

## 6 SANCARLO 6.22 5.81 0.969 15.8 0.320 0.0254

Strictly speaking, those variances are not the environmental variances, as they also contain the within-experiment and within block random variability, which needs to be separately estimated during the first step.