Genotype X Environment

Dr C. Ben, Prof. L. Gentzbittel

March 2022

As a preliminary to models for G × E and breeding concepts, it is useful to introduce the concepts of target population of genotypes (TPG) and target population of environments (TPE). The TPG and TPE define the set of genotypes and environments for which we want our inference and predictions to be valid and precise. The TPG contains all possible genotypes we hope to develop and grow the coming years. The TPE delineates the future growing conditions of the genotypes in the TPG. The TPE can be defined by geography, soil and meteorological conditions, management choices, and the incidence of biotic and abiotic stresses.

The statistical methodology is illustrated using a maize data set obtained from a series of drought and nitrogen stress trials from the maize breeding program at Centro Internacional de Mejoramiento de Maiz y Trigo (CIMMYT; the International Maize and Wheat Improvement Center; Ribaut et al., 1996, 1997). The data are released in Malosetti *et al.* (2013). Another smaller data set on the yield data of 1993 Ontario winter wheat performance trials, in which 18 genotypes (G1 to G18) were tested at nine locations (E1 to E9) will be used, released in Yan and Tinker (2006).

```
# libraries used for the analyses
library(tidyr)
library(dplyr)
library(ggplot2)
library(car)  ## for leveneTest()
library(scales)  ## to modify alpha of colours
library(GGally)  ## to plot pairwise correlations
library(gridExtra)  ## to combine graphics into one device.
# Constraints on effects for ANOVA
options(contrasts=c("contr.treatment", "contr.poly"), scipen = -1)
```

1 The data sets

A maize F2 population was generated by crossing a drought tolerant parent (P1) with a drought susceptible one (P2). Seeds harvested from each of 211 F2 plants formed F3 families, which were stored for further evaluation. The F3 families were evaluated in managed stress trials in 1992, 1994, and 1996.

- In the winter of 1992, a managed water stress trial was conducted in Mexico, including no stress (NS), intermediate stress (IS), and severe stress (SS).
- In the winter of 1994, a similar trial was conducted, but it only included the IS and SS treatments.
- In the summer of 1996, the families were tested in a nitrogen stress trial with two levels: low (LN) and high nitrogen (HN).
- An extra LN trial was conducted in the winter of the same year.

In total, the families were evaluated in eight different environments, each environment characterized by year, stress type and intensity, and management factors. Grain yield is recorded, as well as the minimum temperature during flowering (MINTF) and the amount of radiation during grain filling (RADG).

1.1 Single-step or two-step of analysis.

We do *not* know which experimental designs were used in each of 1992, 1994 and both 1996 experiments : the means by genotype and by environment are provided 'as is', not the raw data. These values come from a first stage of analysis – per environment – that compute 'adjusted means' per environment.

This example is thus typical of a so-called two-stage strategy for analyzing MET data. In the first stage, individual trials are analyzed with models including terms for design features and spatial variation. From these individual trial analyses, adjusted means and weights – usually reciprocals of the variances of the means – are carried forward

to the second stage, where a model is fitted to the genotype by environment means, using either no weights or weights estimated in the first stage.

- If a *fixed* linear model is used at the second stage,

$$\mathbf{Y} = \mathbf{Xb} + \mathbf{e} \qquad \text{with } e \underset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \mathbf{R} = \sigma^2\mathbf{I})$$

  the (unweighted) solution (Ordinary Least Squares) is

$$\mathbf{b} = ({}^t\mathbf{XX})^{-1}.{}^t\mathbf{XY}$$

  and the weighted solution (Weighted Least Squares) is

$$\mathbf{b} = ({}^t\mathbf{XWX})^{-1}.{}^t\mathbf{XWY}$$

  with $\mathbf{W}$ the matrix of weights. Here $\mathbf{R} = \sigma^2\mathbf{W}^{-1}$.

  WLS is a special case of Generalized Least Squares (GLS) where all the off-diagonal entries are 0. This situation arises when the variances of the observed values are unequal (i.e. heteroscedasticity is present), but where no correlations exist among the observed variances. The weight for unit j is proportional to the reciprocal of the variance of the response for unit j.

  The `gls()` function of package `nlme` allows to fit models with various forms of the variance-covariance matrix of the residuals $\mathbf{R}$, allowing to account for heteroscedasticity in fixed models. See below.

- If a *mixed* linear model is used at the second stage,

$$\mathbf{Y} = \mathbf{Xb} + \mathbf{Zu} + \mathbf{e}$$

  the solution is

$$\begin{pmatrix} {}^t\mathbf{XR}^{-1}\mathbf{X} & {}^t\mathbf{XR}^{-1}\mathbf{Z} \\ {}^t\mathbf{ZR}^{-1}\mathbf{X} & {}^t\mathbf{ZR}^{-1}\mathbf{Z} + \mathbf{G}^{-1} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{b}} \\ \hat{\boldsymbol{u}} \end{pmatrix} = \begin{pmatrix} {}^t\mathbf{XR}^{-1}\mathbf{y} \\ {}^t\mathbf{ZR}^{-1}\mathbf{y} \end{pmatrix}$$

  with :

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{e} \end{bmatrix} = \mathcal{N}(\mathbf{0}, \begin{bmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix})$$

  and $\mathbf{R}$ and $\mathbf{G}$ being estimated by Maximum Likelihood (ML) or Restricted Maximum Likelihood (REML).

The random vector $\mathbf{u}$ contains all random effects pertaining to genotypes and environments, if any. There are several strategies to define the weight matrix ($\mathbf{W}$ or $\mathbf{R}$ depending on the model), as described in Möhring & Piepho (2009) *Comparison of Weighting in Two-Stage Analysis of Plant Breeding Trials. Crop Sci. 49, 1977–1988.*

```
# read data :
METmaize <- read.table("./METmaize_data.csv", sep = ";", header = TRUE,
    dec = ",", stringsAsFactors = TRUE)
str(METmaize)
```

```
## 'data.frame':    1688 obs. of  6 variables:
##  $ Genotype: Factor w/ 211 levels "G001","G002",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ Environ : Factor w/ 8 levels "HN96b","IS92a",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ Group   : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ yield   : num  3.29 2.92 2.73 2.46 2.85 ...
##  $ MINTF   : num  9.7 9.7 9.7 9.7 9.7 9.7 9.7 9.7 9.7 9.7 ...
##  $ RADG    : num  18.2 18.2 18.2 18.2 18.2 18.2 18.2 18.2 18.2 18.2 ...
```

```
levels(METmaize$Environ)
```

```
## [1] "HN96b" "IS92a" "IS94a" "LN96a" "LN96b" "NS92a" "SS92a" "SS94a"
```

2 Standard methods

We will focus on two-stage analyses, starting with using a *fixed* model framework. As such, the variable to analyse is the vector of the means of each i accession in different j environments : μ_{ij}.

2.1 The additive model as the 'null model'

The additive model is

$$\mu_{ij} = \mu + G_i + E_j + \epsilon_{ij}$$

Here Genotypes G_i and environments E_j are fixed effects.

```
Additive <- aov(yield ~ Genotype + Environ, data = METmaize)
```

```
## x11()
par(mfrow = c(2, 2))
plot(Additive)
```

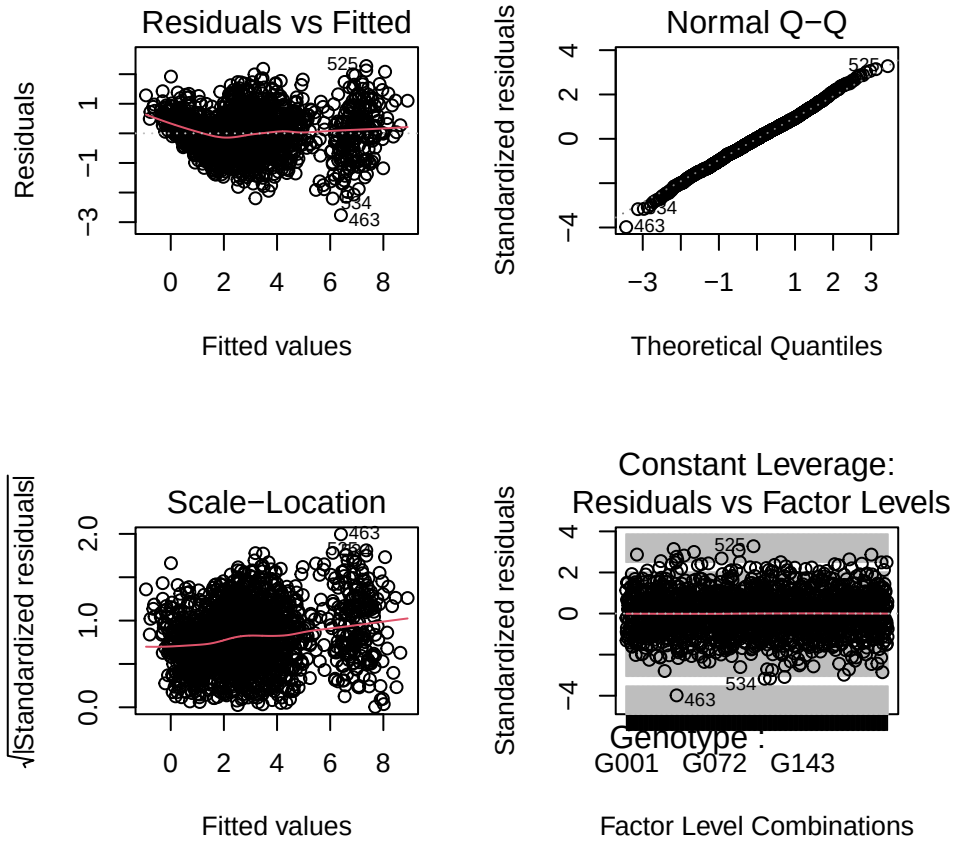


Figure 1: Graphical diagnostic for residuals of the additive model.

```
par(mfrow = c(1, 1))
```

Residuals are probably following a Normal distribution *but* variances are unlikely to be homogeneous.

```
shapiro.test(Additive$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Additive$residuals
## W = 0.99854, p-value = 0.1574
```

```
leveneTest(Additive$residuals, METmaize$Genotype)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value Pr(>F)
## group  210  0.9788 0.5708
##       1477
```

```
leveneTest(Additive$residuals, METmaize$Environ)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value    Pr(>F)
## group    7  13.828 < 2.2e-16 ***
##       1680
## ---
## Signif. codes:  0 '***' 1e-03 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This is a classical finding in MET analyses ! Well, let's say we decide to go on with the analysis ....

```
summary(Additive)
```

```
##               Df Sum Sq Mean Sq  F value Pr(>F)
## Genotype     210    614     2.9    5.288 <2e-16 ***
## Environ        7   5679   811.2 1466.473 <2e-16 ***
## Residuals   1470    813     0.6
## ---
## Signif. codes:  0 '***' 1e-03 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.2 Modeling heterogeneous variances among environments.

A solution to the above concern would be to analyse the data using a *weighted fixed* linear model to account for heterogeneous variances.

```
library(nlme)  ## to fit GLS model
AdditiveHetero <- gls(yield ~ Genotype + Environ, data = METmaize,
    weights = varIdent(form = ~1 | Environ))  ## different variances per stratum
# getting coefficient for estimated variances per environ.
AdditiveHetero$modelStruct$varStruct
```

```
## Variance function structure of class varIdent representing
##     SS92a     IS92a     NS92a     IS94a     SS94a     LN96a     LN96b     HN96b
## 1.0000000 1.0715726 1.6177107 1.0392855 1.0910871 0.7424591 0.8010471 1.2415500
```

```
summary(AdditiveHetero)$sigma
```

```
## [1] 0.6773897
```

the estimated SD per environ are :

```
summary(AdditiveHetero)$sigma * AdditiveHetero$modelStruct$varStruct
```

```
## Variance function structure of class varIdent representing
##     SS92a     IS92a     NS92a     IS94a     SS94a     LN96a     LN96b     HN96b
## 1.0000000 1.0479398 1.3851835 1.0264458 1.0608295 0.8173253 0.8604764 1.1578449
```

```
# overlaps of parameters ?
intervals(AdditiveHetero)$varStruct  # LN96a diff from IS92a
```

```
##           lower      est.     upper
## IS92a 0.9116378 1.0715726 1.2595659
## NS92a 1.3953117 1.6177107 1.8755579
## IS94a 0.8898589 1.0392855 1.2138041
## SS94a 0.9178519 1.0910871 1.2970187
## LN96a 0.6236603 0.7424591 0.8838876
## LN96b 0.6793291 0.8010471 0.9445738
## HN96b 1.0620773 1.2415500 1.4513505
## attr(,"label")
## [1] "Variance function:"
```

```
intervals(AdditiveHetero)$sigma
```

```
##     lower      est.     upper
## 0.6073866 0.6773897 0.7554609
## attr(,"label")
## [1] "Residual standard error:"
```

```
anova(AdditiveHetero)  ## no SS....
```

```
## Denom. DF: 1470
##             numDF   F-value p-value
## (Intercept)     1 20655.687  <.0001
## Genotype      210     4.489  <.0001
## Environ         7  1271.909  <.0001
```

```
# to compare with addtive w/o heterogeneous variances
AdditiveCste <- gls(yield ~ Genotype + Environ, data = METmaize)
```

Comparing the different models shows that accounting for heterogeneous variances among environments significantly improves the analysis.

```
anova(AdditiveCste, AdditiveHetero)
```

```
##                Model  df      AIC      BIC    logLik   Test  L.Ratio p-value
## AdditiveCste       1 219 4213.525 5372.696 -1887.763
## AdditiveHetero     2 226 4110.205 5306.427 -1829.102 1 vs 2 117.3204  <.0001
```

Looking at the residuals of the OLS and GLS additive model suggests that the heterogeneity of the variances between the environments was taken into account with the `gls` method: the residuals are better distributed over the surface, with no particular trend from left to right (Figure 2).

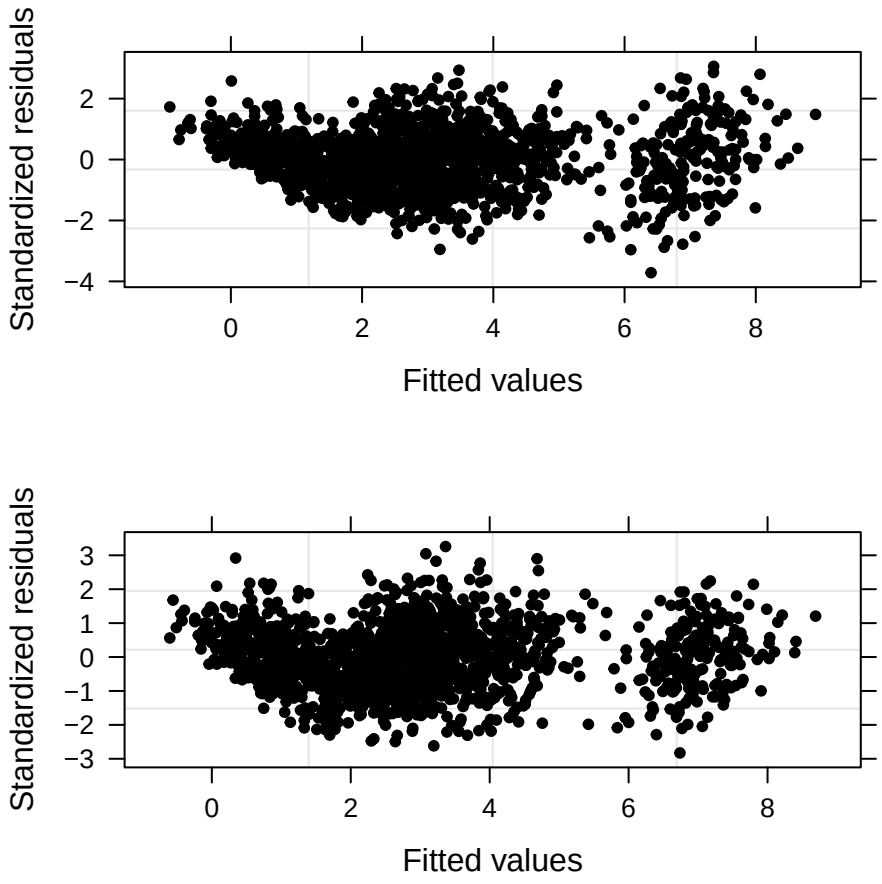


Figure 2: Graphical assessment of the residuals of OLS (up) and WLS (bottom) models for GxE data.

Several points developped in this part will be used when fitting Linear Mixed Models – LMM.

► Let's compute the predicted values and the interaction values given by the different models. Remember the (GE + residuals) are the residuals of these models.

```
Predit <- data.frame(Genotype = METmaize$Genotype, Environ = METmaize$Environ,
    Additive = predict(AdditiveCste), AddHetero = predict(AdditiveHetero))
```

```
InteractionEffects <- data.frame(Genotype = METmaize$Genotype,
    Environ = METmaize$Environ, Additive = residuals(AdditiveCste),
    AddHetero = residuals(AdditiveHetero))
```

Accounting for heterogeneous variances among environments only slightly modifies the predicted values of the G+E component, and has a more pronounced effect of the GE component (Figure 3). It would be better to use the predicted values of *GE accounting for heterogeneous variances among environments* for subsequent analyses (e.g. AMMI, GGE).

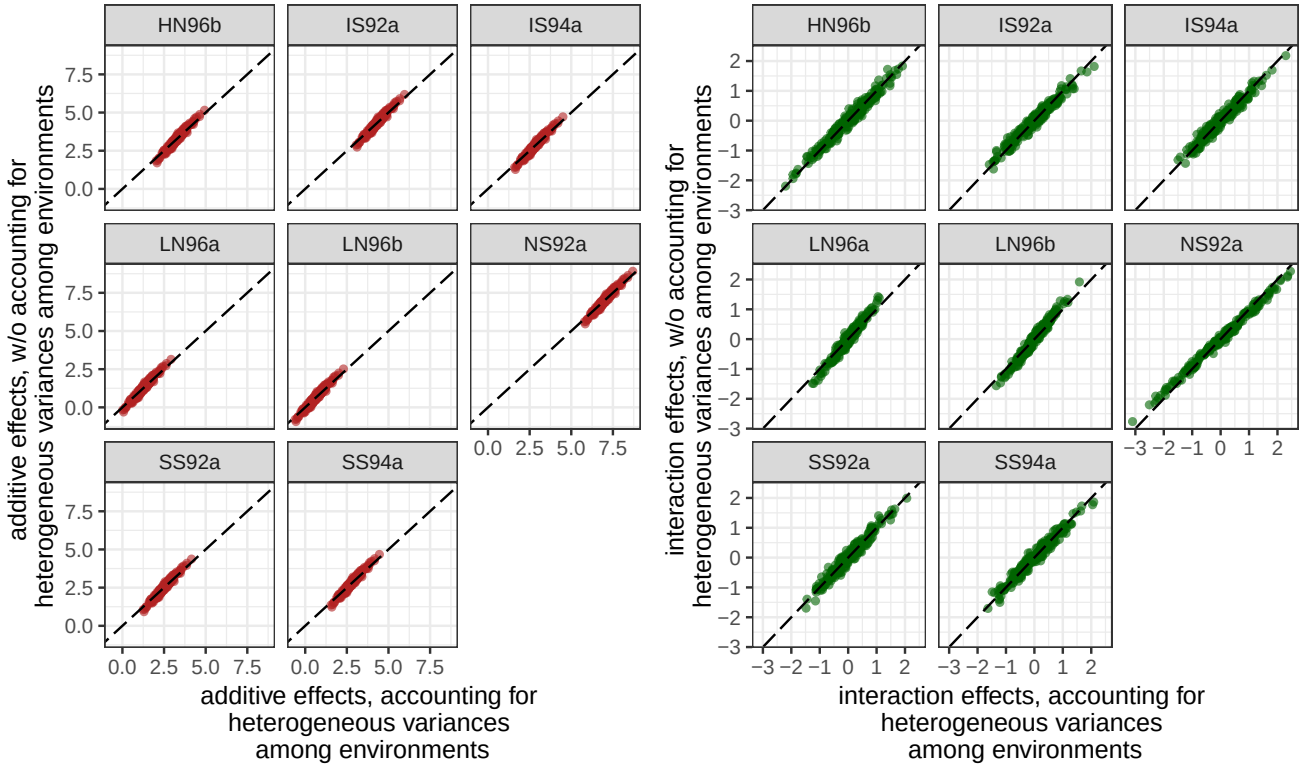


Figure 3: Predicted G+E (left) and GE (right) components, accounting or not for heterogeneous variances among environments.

2.3 Because we are working with a table of means, the model with interaction can NOT be fitted.

As a matter of fact, the model is

$$\mu_{ij} = \mu + G_i + E_j + (GE_{ij} + \epsilon_{ij})$$

```
Complete <- aov(yield ~ Genotype * Environ, data = METmaize)
summary(Complete)
```

```
##                     Df Sum Sq Mean Sq
## Genotype           210    614     2.9
## Environ              7   5679   811.2
## Genotype:Environ  1470    813     0.6
```

The interaction is summed with the residual, there is no MS to use to test the factor effects.

2.4 The Finlay and Wilkinson regression.

Also called ''joint regression"" or "regression on the mean". The Finlay and Wilkinson regression (1963) describes GE as a regression line on the environmental quality. In the absence of explicit environmental information, the biological quality of an environment can be reflected in the average performance of all genotypes in that environment. Good environments will have a high average genotypic performance, and bad environments will have a low average genotypic performance.

The GE part is described by genotype-specific regression slopes on the environmental quality, and the model can be written in the following equivalent ways:

$$\mu_{ij} = \mu + G_i + E_j + b_i E_j + \delta_{ij}$$

or

$$\mu_{ij} = G'_i + b'_i E_j + \delta_{ij}$$

by taking $\mu + G_i = G'_i$ and $E_j + b_i E_j = (1 + b_i)E_j = b'_i Ej$. This latter model is easier to interpret because it looks as a set of regression lines; each genotype has a linear reaction norm with intercept G' and slope b'. Genotypes are characterized in terms of:

- intercept (general performance, μ_i)
- slope (adaptability, sensitivity b'_i)
- deviations from regression (stability, var(ϵ) in each environment)

Note : it would have been better to work using the predicted values of GLS model than with the original raw data.

```
# environmental indexes, expressed as a deviation from the
# average :
Indexes <- METmaize %>%
    group_by(Environ) %>%
    summarise(Index = mean(yield))
Indexes$Index <- Indexes$Index - mean(METmaize$yield)  ## do not forget
Indexes
```

```
## # A tibble: 8 x 2
##   Environ  Index
##   <fct>    <dbl>
## 1 HN96b    0.193
## 2 IS92a    1.23
## 3 IS94a   -0.241
## 4 LN96a   -1.82
## 5 LN96b   -2.44
## 6 NS92a    3.95
## 7 SS92a   -0.588
## 8 SS94a   -0.285
```

Environments can be ranked on their Index : $Index > 0$: better than average, $E < 0$: worse than average

```
# shortcut to distribute index over the full table:
METmaize <- left_join(METmaize, Indexes, by = c(Environ = "Environ"))
head(METmaize)
```

```
##   Genotype Environ Group    yield MINTF RADG      Index
## 1     G001   SS92a     2 3.289333   9.7 18.2 -0.5878288
## 2     G002   SS92a     2 2.922667   9.7 18.2 -0.5878288
## 3     G003   SS92a     2 2.728667   9.7 18.2 -0.5878288
## 4     G004   SS92a     2 2.460667   9.7 18.2 -0.5878288
## 5     G005   SS92a     2 2.848667   9.7 18.2 -0.5878288
## 6     G006   SS92a     2 1.782667   9.7 18.2 -0.5878288
```

```
# Fitting FW regression:
JointReg <- lm(yield ~ Genotype + Environ + Genotype:Index, data = METmaize)
anova(JointReg)
```

```
## Analysis of Variance Table
##
## Response: yield
##                   Df Sum Sq Mean Sq   F value    Pr(>F)
## Genotype         210  614.3    2.93    6.3183 < 2.2e-16 ***
## Environ            7 5678.7  811.25 1752.3284 < 2.2e-16 ***
## Genotype:Index   210  229.9    1.09    2.3645 < 2.2e-16 ***
## Residuals       1260  583.3    0.46
## ---
## Signif. codes:  0 '***' 1e-03 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The `Genotype:Index` term is sometimes called `Heterogeneity of slopes`. In the regression on the mean model, GE is explained in terms of differential sensitivities to the improvement of the environment, with some genotypes (the ones with larger values of b_i) benefiting more than others from an increase in environmental quality.

To compare with the ANOVA of additive model :

```
summary(Additive)
```

```
##               Df Sum Sq Mean Sq  F value Pr(>F)
## Genotype     210    614     2.9    5.288 <2e-16 ***
## Environ        7   5679   811.2 1466.473 <2e-16 ***
## Residuals   1470    813     0.6
## ---
## Signif. codes:  0 '***' 1e-03 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The GE is "extracted" from the Residual of the Additive model. See df : $1260+210 = 1470$ and SS: $229.9+583.3 = 813$.

Let's plot some lines :

```
toto <- data.frame(METmaize, predicted = predict(JointReg))
x11()
toto %>%
    filter(Genotype %in% c("G025", "G045", "G016", "G012", "G008")) %>%
    ggplot(aes(x = Index, y = predicted, group = Genotype, colour = Genotype)) +
    geom_line(lwd = 1.5) + geom_abline(slope = 1, intercept = mean(METmaize$yield),
    lty = 2) + geom_vline(xintercept = 0, lty = 2) + theme(plot.background = element_rect(fill = "aliceb
    plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm"))  #top, right, bottom, left
```

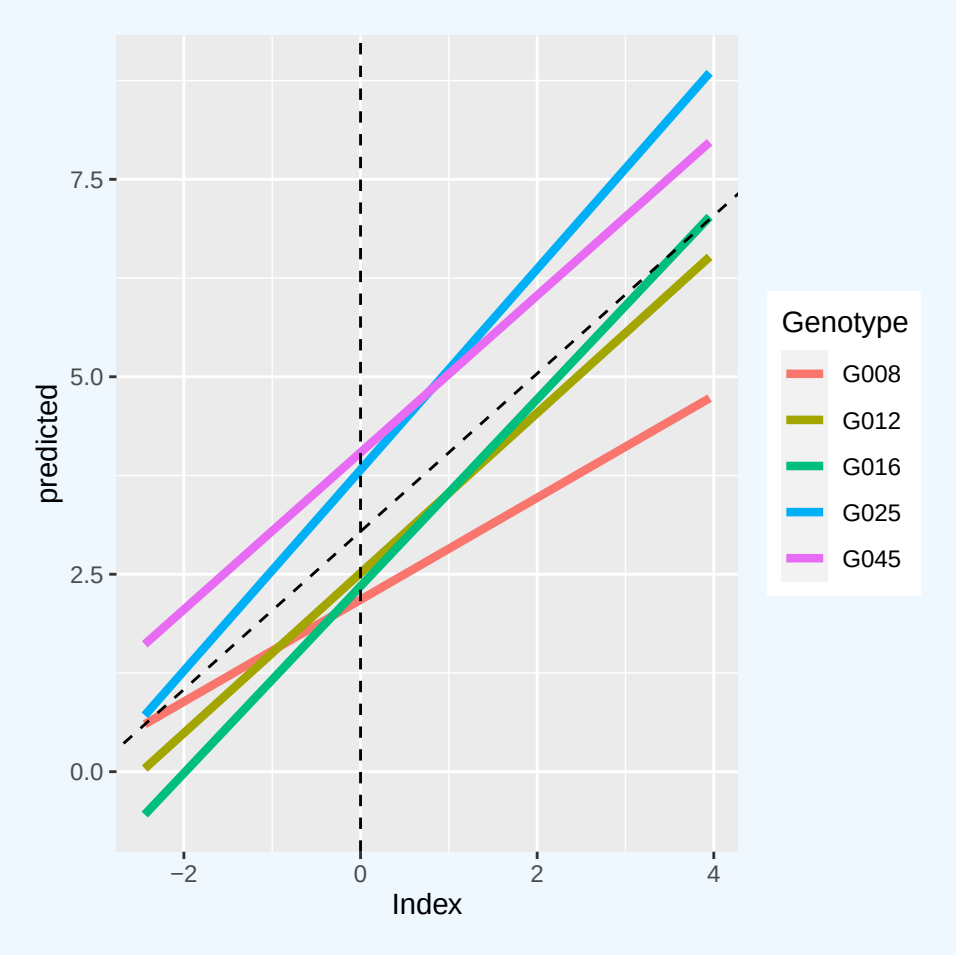


Figure 4: Finlay-Wilkinson regression.

Note that $b' > 1$ for genotypes with a higher than average sensitivity, and $b' < 1$ for genotypes that are less sensitive than average. Using the $\mu_{ij} = G'_i + b'_i E_j + \delta_{ij}$ form of the FW regreesion, we can compare some genotypes:

```
coef(JointReg)["(Intercept)"] + coef(JointReg)["GenotypeG025"]
```

```
## (Intercept)
##    4.022517
```

```
coef(JointReg)["GenotypeG025:Index"] + 1
```

```
## GenotypeG025:Index
##           1.229274
```

```
coef(JointReg)["(Intercept)"] + coef(JointReg)["GenotypeG045"]
```

```
## (Intercept)
##    4.242933
```

```
coef(JointReg)["GenotypeG045:Index"] + 1
```

```
## GenotypeG045:Index
##          0.9511155
```

```
coef(JointReg)["(Intercept)"] + coef(JointReg)["GenotypeG008"]
```

```
## (Intercept)
##    2.378933
```

```
coef(JointReg)["GenotypeG008:Index"] + 1
```

```
## GenotypeG008:Index
##          0.6027311
```

The model can be used to predict the performance of genotypes in environments that were not present in the MET, as long as the environment for which predictions are required can reasonably be placed within the range of environments used in the original MET.

The Finlay-Wilkinson procedure can be suboptimal for at least four reasons: (1) in the first step environmental means are typically estimated without considering genetic-by-environment interactions, (2) in the second step uncertainty about the environmental means is ignored, (3) estimation is performed regarding lines and environment as fixed effects, and (4) the procedure does not incorporate genetic (either pedigree-derived or marker-derived) relationships.

3 Bilinear models

Before getting into the details of biliner models, we will explore the concept of 'extracting' information from a matrix based on Singular Value Decomposition. The example will be based on *reconstructing* a picture. **Script**

3.1 the AMMI decomposition of GE

The AMMI model is

$$\mu_{ij} = \mu + G_i + E_j + \sum_{k=1}^{r}(G_{ik} \cdot E_{jk}) + \delta_{ij}$$

where the GE is now explained by k multiplicative terms ($k = 1 \dots r$), each multiplicative term formed by the product of a genotypic sensitivity G_{ik} (genotypic score) and a hypothetical environmental characterization E_{jk} (environmental score).

The genotypic and environmental score come from the SVD of the two-way matrix of the interaction terms, with the genotypes as rows and enviroments as coulumns. The SVD of a rectangular matrix $\mathbf{A}_{m\times n}$ is:

$$\mathbf{A} = \mathbf{U}\Lambda\ {}^{t}\mathbf{V}$$

with $\mathbf{U}_{m\times m}$ the *left-singular* vectors, $\Lambda_{m\times n}$ the matrix of the *singular values* and $\mathbf{V}_{n\times n}$ the *right-singular* vectors. We usually force $\mathbf{U} \cdot^{t} \mathbf{U} = \mathbf{V} \cdot^{t} \mathbf{V} = \mathbf{I}$.

```
# Getting the table of interaction values
Additive2 <- aov(yield ~ Genotype + Environ + Genotype:Environ,
    data = METmaize)
modelTables <- model.tables(Additive2, type = "effects", cterms = "Genotype:Environ")
```

```
interaction.effects <- modelTables$tables$"Genotype:Environ"  ## to get the table of interactions
head(interaction.effects)
```

```
##          Environ
## Genotype       HN96b       IS92a      IS94a      LN96a        LN96b      NS92a
##     G001  0.78392077 -0.14914400 -0.9129661 -0.6203130 -0.486474171  1.0447328
##     G002 -0.71549589  0.41543934  1.0262839 -0.8330630  0.140775829  0.4319828
##     G003  0.46717077  0.13943934 -0.6443828 -0.2103964 -0.003224171  0.5613161
##     G004 -0.04949589  0.32343934 -0.5143828  0.4062703  0.273442496 -0.9826839
##     G005 -0.01566256  0.67260600  0.1314506 -0.2065630  1.333942496 -1.1168506
##     G006 -0.74582923 -0.05956066 -0.4047161  0.6966036 -0.109557504  0.8816494
##          Environ
## Genotype      SS92a      SS94a
##     G001  0.4744121 -0.1341683
##     G002  0.2749954 -0.7409183
##     G003  0.1503288 -0.4602517
##     G004  0.9056621 -0.3622517
##     G005  0.3074954 -1.1064183
##     G006 -0.4220046  0.1634150
```

```
## SVD let's use the 8 eigenvectors
PC <- 8
decomposition <- svd(interaction.effects, nu = PC, nv = PC)
if (PC > 1) {
    D <- diag(decomposition$d[1:PC])
} else {
    D <- decomposition$d[1:PC]  ## because of diag()
}
D
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]    [,6]     [,7]          [,8]
## [1,] 15.56605  0.00000  0.00000 0.000000 0.000000 0.00000 0.000000 0.000000e+00
## [2,]  0.00000 13.14559  0.00000 0.000000 0.000000 0.00000 0.000000 0.000000e+00
## [3,]  0.00000  0.00000 11.37249 0.000000 0.000000 0.00000 0.000000 0.000000e+00
## [4,]  0.00000  0.00000  0.00000 9.455393 0.000000 0.00000 0.000000 0.000000e+00
## [5,]  0.00000  0.00000  0.00000 0.000000 8.913677 0.00000 0.000000 0.000000e+00
## [6,]  0.00000  0.00000  0.00000 0.000000 0.000000 8.48159 0.000000 0.000000e+00
## [7,]  0.00000  0.00000  0.00000 0.000000 0.000000 0.00000 5.287942 0.000000e+00
## [8,]  0.00000  0.00000  0.00000 0.000000 0.000000 0.00000 0.000000 2.163311e-13
```

We can illustrate the full reconstruction using the 8 eigen vectors, by forming $\mathbf{u} \cdot \mathbf{D} \cdot^t \mathbf{v}$.

```
interaction.effects[1:7, 1:5]
```

```
##          Environ
## Genotype       HN96b       IS92a      IS94a      LN96a        LN96b
##     G001  0.78392077 -0.14914400 -0.9129661 -0.6203130 -0.486474171
##     G002 -0.71549589  0.41543934  1.0262839 -0.8330630  0.140775829
##     G003  0.46717077  0.13943934 -0.6443828 -0.2103964 -0.003224171
##     G004 -0.04949589  0.32343934 -0.5143828  0.4062703  0.273442496
##     G005 -0.01566256  0.67260600  0.1314506 -0.2065630  1.333942496
##     G006 -0.74582923 -0.05956066 -0.4047161  0.6966036 -0.109557504
##     G007  0.50617077  0.13710600 -0.6753828  0.9952703 -0.397557504
```

```
reconstruction <- decomposition$u %*% D %*% t(decomposition$v)  ## be careful, use t(v)
reconstruction[1:7, 1:5]
```

```
##             [,1]        [,2]       [,3]       [,4]         [,5]
## [1,]  0.78392077 -0.14914400 -0.9129661 -0.6203130 -0.486474171
## [2,] -0.71549589  0.41543934  1.0262839 -0.8330630  0.140775829
## [3,]  0.46717077  0.13943934 -0.6443828 -0.2103964 -0.003224171
## [4,] -0.04949589  0.32343934 -0.5143828  0.4062703  0.273442496
## [5,] -0.01566256  0.67260600  0.1314506 -0.2065630  1.333942496
## [6,] -0.74582923 -0.05956066 -0.4047161  0.6966036 -0.109557504
```

```
## [7,]  0.50617077  0.13710600 -0.6753828  0.9952703 -0.397557504
```

```
## if needed, check that uDv does NOT give the correct
## results.
## let's use 2 eigenvectors to approximate the GE matrix
PC <- 2
decomposition <- svd(interaction.effects, nu = PC, nv = PC)
str(decomposition)
```

```
## List of 3
##  $ d: num [1:8] 15.57 13.15 11.37 9.46 8.91 ...
##  $ u: num [1:211, 1:2] 0.0645 0.0519 0.0318 -0.0657 -0.0747 ...
##  $ v: num [1:8, 1:2] -0.195 0.153 0.013 -0.252 -0.358 ...
```

```
if (PC > 1) {
    D <- diag(decomposition$d[1:PC])
} else {
    D <- decomposition$d[1:PC]  ## because of diag()
}
D
```

```
##          [,1]     [,2]
## [1,] 15.56605  0.00000
## [2,]  0.00000 13.14559
```

```
# Show the approximation using 2 eigen vectors
interaction.effects[1:7, 1:5]
```

```
##         Environ
## Genotype       HN96b       IS92a      IS94a      LN96a        LN96b
##     G001  0.78392077 -0.14914400 -0.9129661 -0.6203130 -0.486474171
##     G002 -0.71549589  0.41543934  1.0262839 -0.8330630  0.140775829
##     G003  0.46717077  0.13943934 -0.6443828 -0.2103964 -0.003224171
##     G004 -0.04949589  0.32343934 -0.5143828  0.4062703  0.273442496
##     G005 -0.01566256  0.67260600  0.1314506 -0.2065630  1.333942496
##     G006 -0.74582923 -0.05956066 -0.4047161  0.6966036 -0.109557504
##     G007  0.50617077  0.13710600 -0.6753828  0.9952703 -0.397557504
```

```
reconstruction <- decomposition$u %*% D %*% t(decomposition$v)
reconstruction[1:7, 1:5]
```

```
##             [,1]        [,2]        [,3]        [,4]        [,5]
## [1,]  0.42385647 -0.11641911 -0.30089975 -0.14210595 -0.24765254
## [2,] -0.85238393  0.42709738  0.36218950 -0.32863847 -0.41492671
## [3,]  0.35419419 -0.12076669 -0.22188645 -0.04398778 -0.09586033
## [4,] -0.09847781 -0.02677818  0.13766951  0.20439531  0.31219059
## [5,] -0.03083537 -0.06583526  0.11542798  0.24691669  0.36954852
## [6,] -0.09417611  0.09317396 -0.01787211 -0.18272410 -0.26340033
## [7,]  0.55734195 -0.23960699 -0.28674269  0.08862376  0.08369444
```

```
# percentage of information using 2 eigen vectors
sum(decomposition$d[1:2])/sum(decomposition$d)  ## not really great
```

```
## [1] 0.397543
```

Rewriting

$$GE = \sum_{k=1}^{r} u_{ik}\lambda_k \, {}^t v_{jk} = \sum_{k=1}^{r} (u_{ik}\sqrt{\lambda_k}) \cdot ({}^t v_{jk}\sqrt{\lambda_k}) = \sum_{i=1}^{r} G_{ik}E_{jk}$$

with r the rank of the GE matrix (minimum of the number of genotypes or of the number of environments), gives a *multiplicative* decomposition of the GE matrix. G_{ik} is the genotypic sensitivity (genotypic score), and E_{jk} is a hypothetical environmental characterization (environmental score) for each of the 'principal components'.

```
f <- 0.5  ## A classical partitioning for AMMI (see below GGE), called symmetrical partitioning
G <- decomposition$u %*% D^(f)
E <- decomposition$v %*% D^(1 - f)
```

The below code is provided 'as is'. It allows to understand the computation of the ANOVA table of the AMMI decomposition.

```
Ecolnumb <- c(1:PC)  ## number of retained PCs. Required for df computations
Ecolnames <- paste("PC", Ecolnumb, sep = "")
dimnames(E) <- list(levels(METmaize$Environ), Ecolnames)
dimnames(G) <- list(levels(METmaize$Genotype), Ecolnames)
## Significance of PCs
var.num <- length(levels(METmaize$Genotype))
envir.num <- length(levels(METmaize$Environ))
interaction.SS <- t(as.vector(interaction.effects)) %*% as.vector(interaction.effects)  # by definition
## singular values are sqrt or eigenvalues.  eigenvalues
## are the variance explained by a PC axis. They are thus
## SS
(PC.SS <- (decomposition$d[1:PC]^2))
```

```
## [1] 242.3020 172.8066
```

```
(PC.DF <- var.num + envir.num - 1 - 2 * Ecolnumb)  ## loosely explain in Gollob (1968)
```

```
## [1] 216 214
```

```
## A statistical model which combines features of factor
## analytic and analysis of variance techniques.
## Psychometrika 33, 73:115.
## https://doi.org/10.1007/BF02289676
## extraction of PC.SS from interaction.SS
residual.SS <- interaction.SS - sum(PC.SS)
residual.DF <- ((var.num - 1) * (envir.num - 1)) - sum(PC.DF)
PC.SS[PC + 1] <- residual.SS  # add a pc+1 line to contain that SS
PC.DF[PC + 1] <- residual.DF  # add a pc+1 line to contain that DF
MS <- PC.SS/PC.DF
F <- MS/(deviance(Additive)/Additive$df.residual)  ## the GxE MS of initial additive model is used to co
Fbis <- MS/MS[PC + 1]  ## The 'residual' MS of current analysis is used to compute F values
proba <- pf(F, PC.DF, Additive$df.residual, lower.tail = FALSE)
probab <- pf(Fbis, PC.DF, PC.DF[PC + 1], lower.tail = FALSE)
(percSS <- PC.SS/interaction.SS)
```

```
## [1] 0.2979613 0.2125020 0.4895368
```

```
rowlabels <- c(Ecolnames, "Residuals")
(AMMI.anova <- data.frame(Effect = rowlabels, d.f. = PC.DF, SS = PC.SS,
    MS = MS, F = F, Prob. = proba))
```

```
##      Effect d.f.       SS        MS         F        Prob.
## 1       PC1  216 242.3020 1.1217687 2.0277919 3.298073e-14
## 2       PC2  214 172.8066 0.8075073 1.4597098 5.812181e-05
## 3 Residuals 1040 398.0912 0.3827800 0.6919414 1.000000e+00
```

▶ If only one PC is kept, the method is called AMMI1 (Figure 5).

The AMMI1 biplot shows contemporarily main effects (genotypes and environments average yields) and interaction, as PC1 scores. To read this biplot, it is necessary to remember that genotypes and environments on the right side of the graph shows yield levels above the average. Besides, genotypes and environments laying close to the x-axis (PC 1 score close to 0) did not interact with each other, while data with positive/negative score on y-axis interacted positively with environments characterised by a score of same sign.

▶ If two PCs are kept, the method is called AMMI2 (Figure 6). These biplots facilitate the exploration of relationships between genotypes and/or environments.

```
## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
```

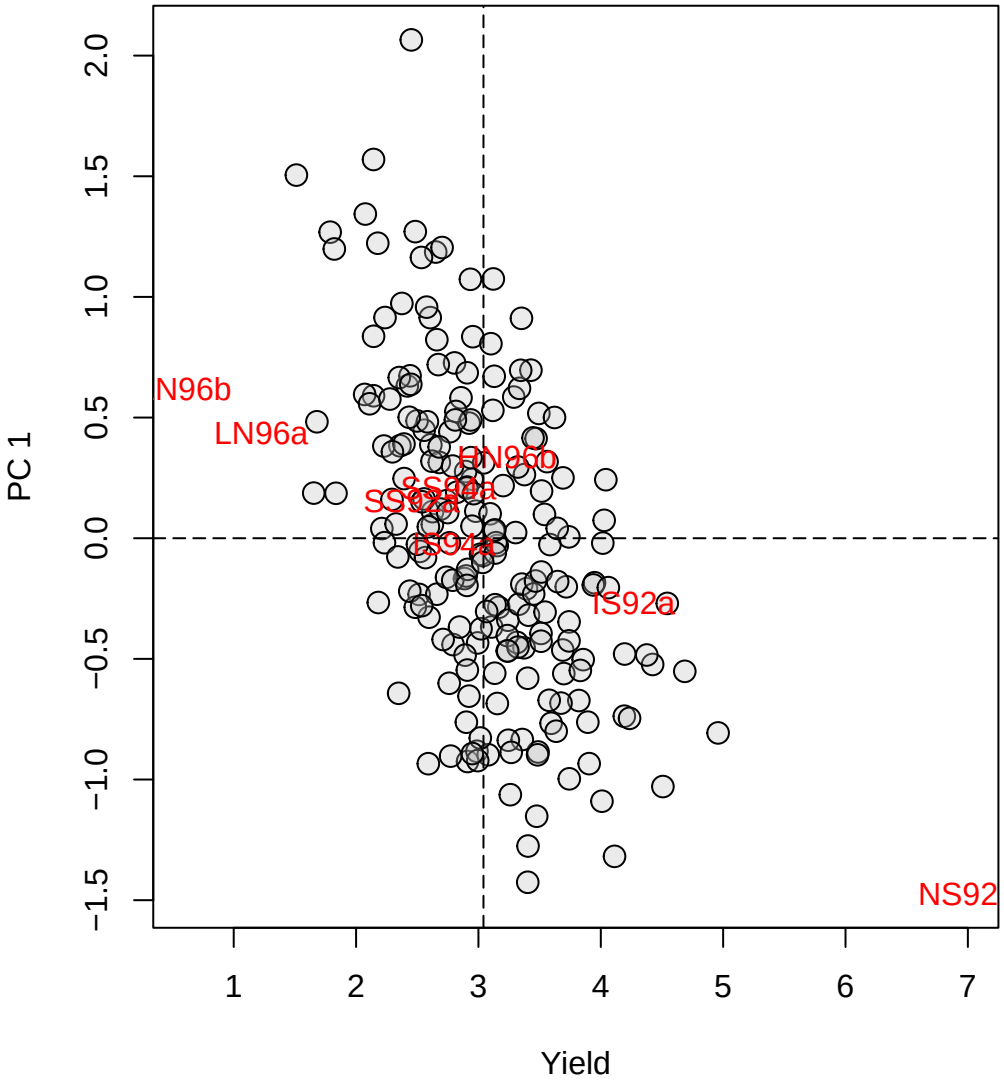


Figure 5: AMMI1 biplot.

```
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
##
## [[7]]
## NULL
##
## [[8]]
## NULL
```

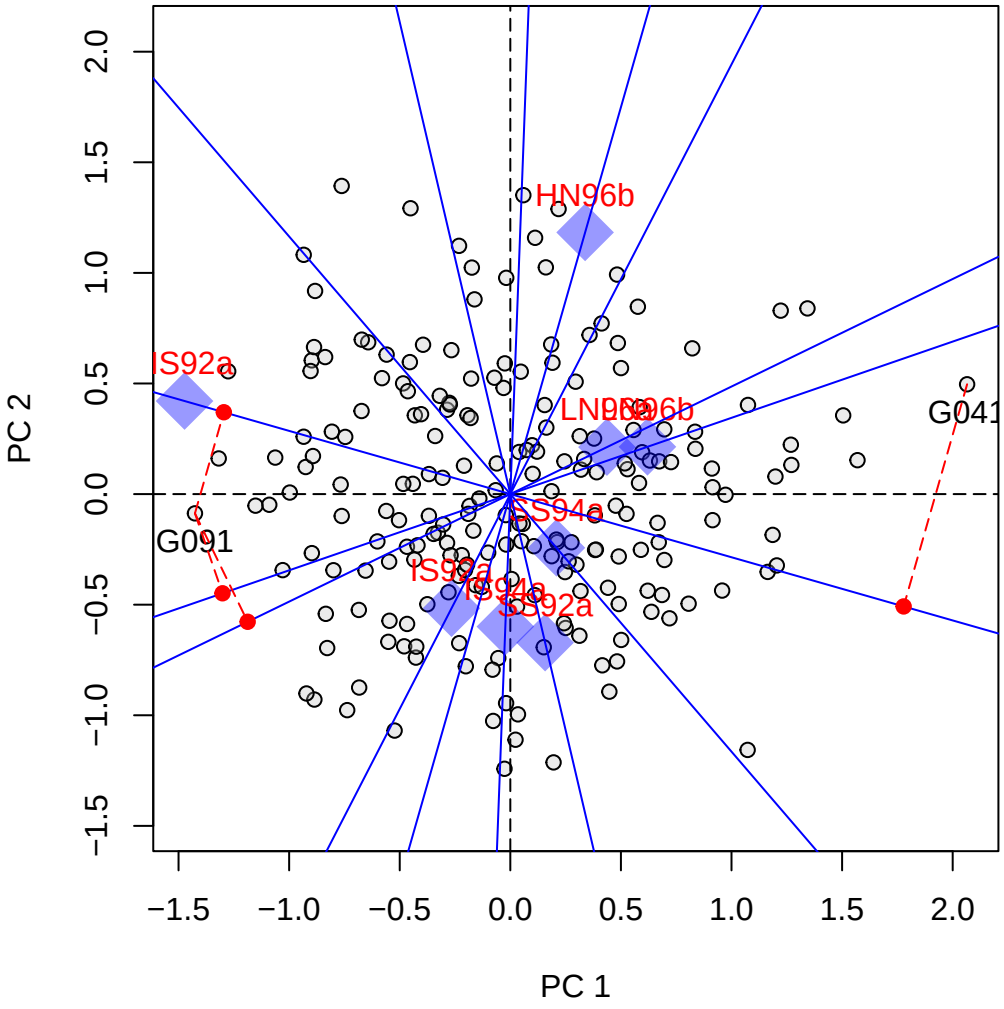


Figure 6: AMMI2 biplot.

- Genotypes/environments that are alike tend to cluster together.
- The angle between environmental axes is related to the correlation between the environments. An acute angle indicates positive correlation (e.g., between LN96a and LN96b), a right angle indicates no correlation (e.g., between HN96b and NS92a), and an obtuse angle indicates negative correlation (e.g., NS92a and LN96a).
- The projection of a genotype onto an environmental axis reflects the performance of that genotype in that environment (for GEI). For example, genotype G091 projects on the NS92a axis above the origin, indicating

a positive interaction with that environment i.e., the relative performance (GEI part) of G091 in NS92a is above the average of all genotypes in NS92a. Conversely, genotype G041 (on the right hand side of the plot) projects below the origin on the same axis, which points to a negative interaction with environment NS92a (i.e., G041 performs worse than average).

- Following a similar procedure it is possible to conclude that while genotype G091 showed positive adaptation to environment NS92a, it is not well adapted to environments LN96a and LN96b (the projection of G091 on the LN96a and LN96b axes falls below the origin). Biplots are thus useful tools to investigate patterns in GEI, because they can help to identify interesting genotypes that are adapted to particular environments, and to classify environments in groups.

The `agricolae` package has a `AMMI` function. The `plantbreeding` package has also a `ammi.full` function. **Both require to have data with replicates per environment , typically 'blocks within locations'**.

3.2 The GGE decomposition of GE

As a matter of fact the AMMI method is a particular case of a more general method aiming to decompose the information in a matrix using SVD. Given a genotype by environment two-way table $\mathbf{P}$ of m genotypes and n environments, and using the classical two-way ANOVA model $\mu_{ij} = \mu + \alpha_i + \beta_j + \gamma_{ij}$, with α_i the effect of the genotype, β_j the effect of the environment and γ_{ij} the interaction effect, we would write :

$$P_{ij} = \mu_{ij} = \mu + \alpha_i + \beta_j + \gamma_{ij} \quad (1)$$
$$P_{ij} = \mu_{ij} - \mu = \alpha_i + \beta_j + \gamma_{ij} \quad (2)$$
$$P_{ij} = \mu_{ij} - \mu - \alpha_i = \beta_j + \gamma_{ij} \quad (3)$$
$$P_{ij} = \mu_{ij} - \mu - \beta_j = \alpha_i + \gamma_{ij} \quad (4)$$
$$P_{ij} = \mu_{ij} - \mu - \alpha_i - \beta_j = \gamma_{ij} \quad (5)$$

The AMMI method corresponds to the SVD decomposition of the GE-matrix (Eq. 5). If one is interested in genotype evaluation, Eq. 4 is also appropriate, as it contains both G and GE, which can be considered simultaneously. Biplots based on Eq. 4 are referred to as "GGE biplots". Other models above are know as completely multiplicative model (COMM - Eq. 2), and shifted multiplicative model (SHMM - Eq. 3).

A second point to consider is the so-called 'partitioning of the singular value' into the genotype (ξ^*_{ik}) and environment (η^*_{jk}) scores before a biplot can be constructed to approximate the two-way table:

$$P_{ij} = \sum_{k=1}^{r} \xi^*_{ik}\eta^*_{jk} = \sum_{k=1}^{r} \left(\xi_{ik}\lambda_k^f\right)\left(\eta_{jk}\lambda_k^{1-f}\right)$$

Note that the partitionning for AMMI is classicaly $f = 0.5$, called symmetrical partitioning (see above example).

• When $f = 0$, the singular values are entirely partitioned into the column (here environment) eigenvectors, referred to column-metric preserving.

• When $f = 1$, the singular values are entirely partitioned into the row eigenvectors, which is referred to as row-metric preserving. It is, therefore, appropriate for visualizing the similarity/dissimilarity among row factors (here genotypes).

A third point to consider is the 'scaling' of the data. The GGE biplot model (Eq. 4) can be more generally presented as:

$$P_{ij} = (\mu_{ij} - \mu - \beta_j)/s_j = (\alpha_i + \gamma_{ij})/s_j$$

where s_j is a scaling factor. When s_j is the standard error for environment j, any heterogeneity among the environments will (supposedly) be removed.

Getting the two-way table ready for the diverse analyses is as simple as getting the two-table of the residuals of each model. Example for GGE :

```
# the Pij matrix is the residual of :
ModelPrGGE <- aov(yield ~ Environ, data = METmaize)
Resids <- cbind(METmaize, prGGE = ModelPrGGE$residuals)
t(Resids$prGGE) %*% Resids$prGGE  ## compare to sum of Genotype and G:E SS of Additive2
```

```
##          [,1]
## [1,] 1427.467
```

```r
# here is the table to subject to SVD for GGE analysis
# (long to wide) :
GGEready <- Resids %>%
    select(Genotype, Environ, prGGE) %>%
    pivot_wider(names_from = Environ, values_from = prGGE)
head(GGEready)
```

```
## # A tibble: 6 x 9
##   Genotype    SS92a  IS92a  NS92a  IS94a   SS94a   LN96a  LN96b   HN96b
##   <fct>       <dbl>  <dbl>  <dbl>  <dbl>   <dbl>   <dbl>  <dbl>   <dbl>
## 1 G001      0.836    0.212  1.41  -0.551  0.227  -0.259  -0.125  1.15
## 2 G002      0.469    0.610  0.626  1.22  -0.547  -0.639   0.335 -0.521
## 3 G003      0.275    0.264  0.686 -0.519 -0.335  -0.0854  0.122  0.592
## 4 G004      0.00729 -0.575 -1.88  -1.41  -1.26   -0.492  -0.625 -0.948
## 5 G005      0.395    0.760 -1.03   0.219 -1.02   -0.119   1.42   0.0721
## 6 G006     -0.671   -0.308  0.633 -0.653 -0.0853  0.448  -0.358 -0.995
```

► Using a simpler dataset for ease of interpretation, let's try a GGE biplot analysis aiming to compare genotypes of 18 winter wheat genotypes assessed at 9 places in Canada-Ontario (in ton.ha^{-1}).

The `gge` and `GGEBiplots`(based on `gge`) R packages may be useful. The `gge` package allows for principal component analysis (SVD) of a data table that can contain missing value, using the Non-linear Iterative Partial Least Squares (NIPALS) algorithm.

```r
METwheat <- read.table("METwheat_data.csv", sep = ";", dec = ".",
    header = TRUE, stringsAsFactors = TRUE)  ##  achtung ! two-way table
head(METwheat)
```

```
##   Genotype   E1   E2   E3   E4   E5   E6   E7   E8   E9
## 1       G1 4.46 4.15 2.85 3.08 5.94 4.45 4.35 4.04 2.67
## 2       G2 4.42 4.77 2.91 3.51 5.70 5.15 4.96 4.39 2.94
## 3       G3 4.67 4.58 3.10 3.46 6.07 5.03 4.73 3.90 2.62
## 4       G4 4.73 4.75 3.38 3.90 6.22 5.34 4.23 4.89 3.45
## 5       G5 4.39 4.60 3.51 3.85 5.77 5.42 5.15 4.10 2.83
## 6       G6 5.18 4.48 2.99 3.77 6.58 5.05 3.99 4.27 2.78
```

```r
# wide to long
METwheatLong <- METwheat %>%
    pivot_longer(c(E1, E2, E3, E4, E5, E6, E7, E8, E9), names_to = "Environ",
        values_to = "yield")
## GGE analysis
library(gge)
m1 <- gge(METwheatLong, yield ~ Genotype * Environ, scale = FALSE)
```

Figure 7 shows a "which-won-where" graph: A polygon is first drawn on genotypes that are furthest from the biplot origin so that all other genotypes are contained within the polygon. Then perpendicular lines to each side of the polygon are drawn, starting from the biplot origin.

The interpretations are as follows:

1. Genotypes located on the vertices of the polygon performed either the best or the poorest in one or more environments.
2. The perpendicular lines are equality lines between adjacent genotypes on the polygon, which facilitate visual comparison of them. For example, the equality line between G18 and G8 indicates that G18 was better in E7 and E5, whereas G8 was better in the other environments. The equality line between G18 and G7 indicates that G18 was better than G7 in all environments. Note that G3 and G1 are located on the line that connects G18 and G7. This means that the rank G18 > G3 > G1 > G7 was true in all environments.
3. The equality lines divide the biplot into sectors, and the winning genotype for each sector is the one located on the respective vertex. In this example, the nine environments fall into *two sectors*. G18 was the winner in environments E7 and E5, and G8 was the winner for the other environments. This pattern suggests that the target environment may consist of *two* different *mega-environments* and that different cultivars should be selected and deployed for each.

For the "which-won-where" graph, the *environment-focused partitioning* is preferred because it correctly shows the relationships among environments.

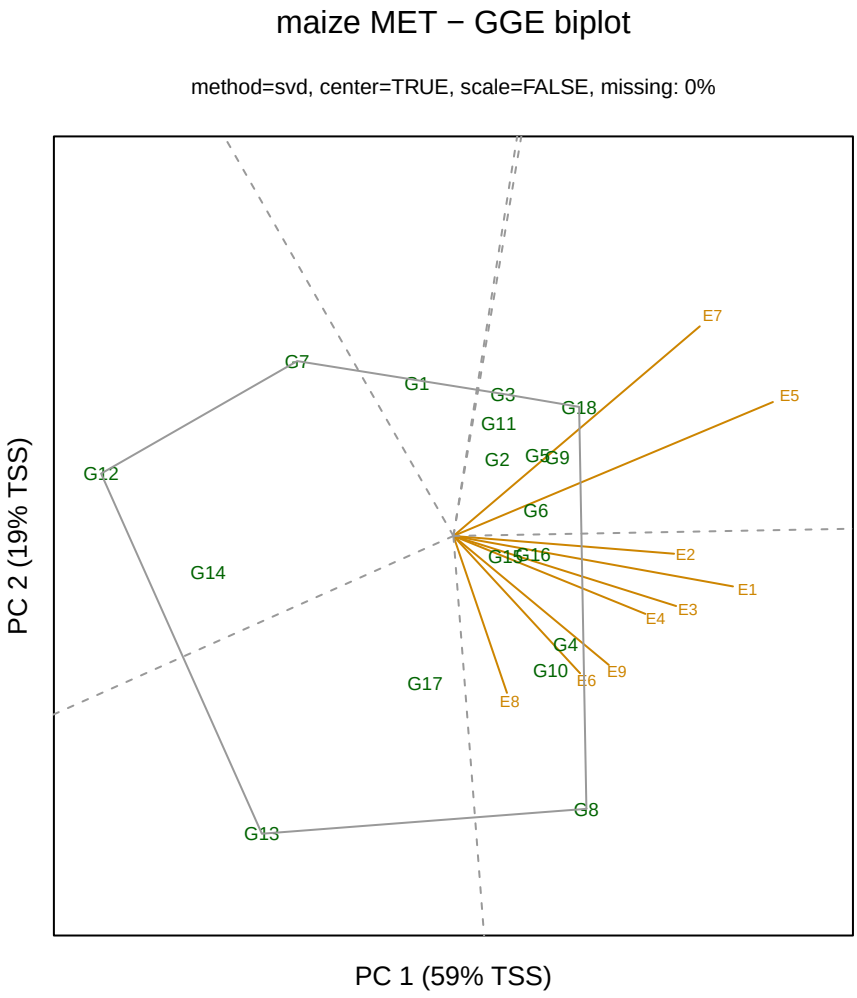


Figure 7: GGE biplot: 'Which-Won-Where' of the winter wheat MET dataset, 'gge' package.

▶ `GGEBiplots` can fit all models (AMMI, GGE, etc..) and allows for more flexibility in partionning. See package documentation. However, the quality of graphs is not really high (Figure 8). Going back to the maize MET dataset to visualize a GGE biplot analysis with *genotype-focused partitioning* and no scaling of environments :

```
library(GGEBiplots)
# need to re-organize raw data as a two-way table : from long to wide
GGEready2 <- data.frame(METmaize %>% select(Genotype, Environ, yield) %>%
                          pivot_wider(names_from = Environ, values_from = yield))
rownames(GGEready2) <- GGEready2$Genotype
GGEready2 <- GGEready2[, -1]
head(GGEready2)
```

```
##         SS92a    IS92a    NS92a    IS94a    SS94a     LN96a       LN96b
## G001 3.289333 4.480000 8.400000 2.248667 2.984000 0.9666667  0.47333333
## G002 2.922667 4.877333 7.620000 4.020667 2.210000 0.5866667  0.93333333
## G003 2.728667 4.532000 7.680000 2.280667 2.421333 1.1400000  0.72000000
## G004 2.460667 3.692667 5.112667 1.387333 1.496000 0.7333333 -0.02666667
## G005 2.848667 5.028000 5.964667 3.019333 1.738000 1.1066667  2.02000000
## G006 1.782667 3.959333 7.626667 2.146667 2.671333 1.6733333  0.24000000
##         HN96b
## G001 4.380000
## G002 2.713333
## G003 3.826667
## G004 2.286667
## G005 3.306667
## G006 2.240000
```

```
# a GGE biplot with genotype-metric preserving partitioning and no scaling
m2 <- GGEModel(GGEready2, centering = "tester", # GGE
               scaling = "none",
               SVP = "row") # genotype-preserving partitioning
```

- Typically, GGE analysis allows to address the following questions:

1. Can the target environment be divided into meaningful mega-environments so that some of the GE can be exploited or avoided? Multi-year data are essential to address this question.
2. What are the best test environments (representative and discriminating)?
3. What are the superior genotypes (both high and stable performance within a mega-environment)?

4 Factorial regression

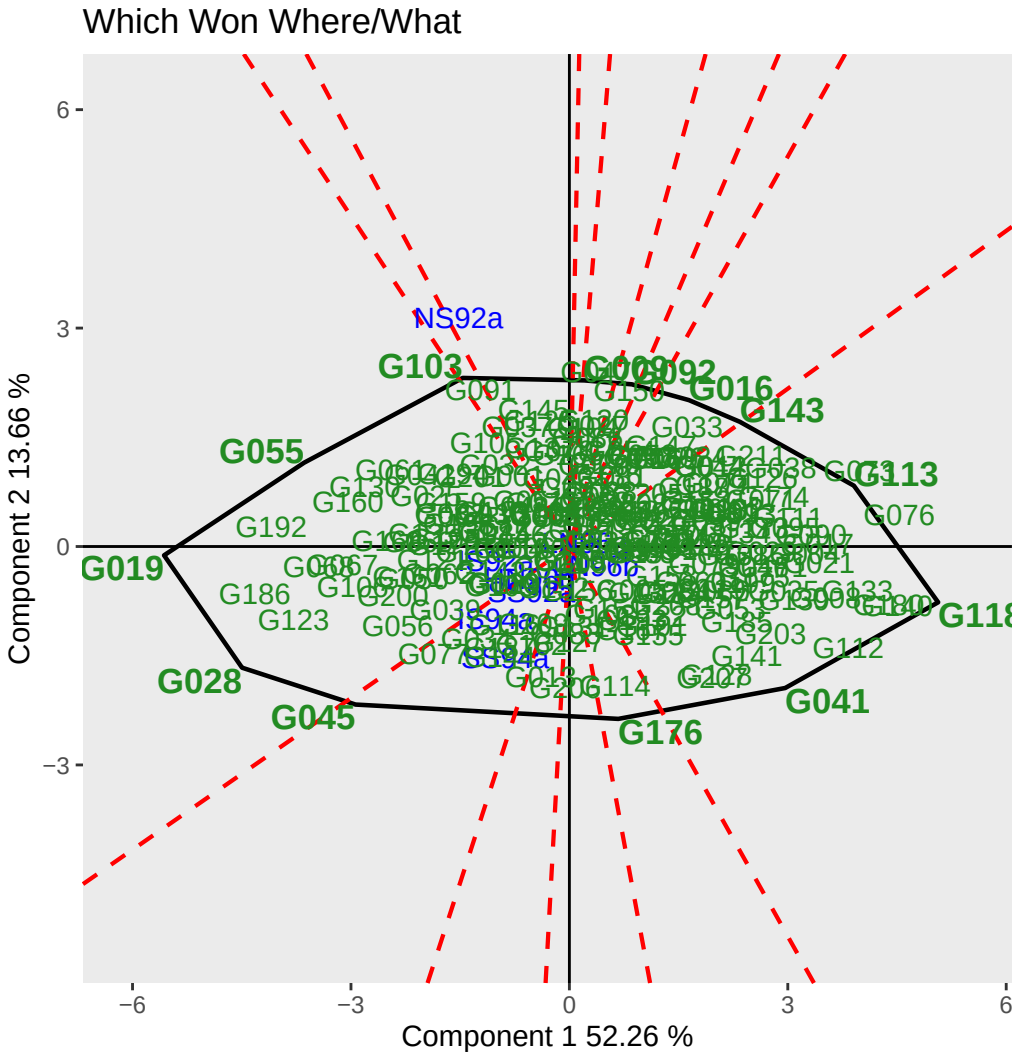
If we do have explicit information about the environment, the information can be used directly in the model by including it in the form of explanatory variables. GE is then described as differential genotypic sensitivity to explicit environmental factors such as temperature, precipitation, water availability etc. Such models are known as factorial regression models.

Formally, they are simply linear models where the G × E is partitioned into several G × covariate interactions.

The maize MET data set has two covariates that will be used to predict genotypic means :

```
FactReg <- lm(yield ~ Genotype + Environ + Genotype:MINTF + Genotype:RADG,
    data = METmaize)
anova(FactReg)
```

```
## Analysis of Variance Table
##
## Response: yield
##                   Df Sum Sq Mean Sq   F value    Pr(>F)
## Genotype         210  614.3    2.93    5.9440 < 2.2e-16 ***
## Environ            7 5678.7  811.25 1648.5124 < 2.2e-16 ***
## Genotype:MINTF   210  172.3    0.82    1.6675 1.949e-07 ***
## Genotype:RADG    210  124.2    0.59    1.2015   0.03797 *
## Residuals       1050  516.7    0.49
## ---
## Signif. codes:  0 '***' 1e-03 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



GGE Biplot showing components 1 and 2 explaining 65.92% of the total variation using Row Metric Preserving SVP and Tester−Centered G+GE with no scaling

Figure 8: GGE biplot: 'Which-Won-Where' of the maize MET dataset, 'GGEBiplots' package.

Here, the GE is explained by differential genotypic sensitivities to the minimum temperature during flowering and to the amount of radiation during grain filling.

► You may be confronted to the situation that there are many, possibly correlated, predictor variables, and relatively few samples/sites. This a situation that is common especially in so-called 'enviro-typing' analyses, where attempts are made to identify and characterise the TPE.

Multivariate regression methods like Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR or PLS) enjoy currently a large popularity in some companies. The main reason is that they have been designed to handle these situations. The `pls` R package implements Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR). The user interface is modelled after the traditional formula interface, as exemplified by `lm`.

5 Mixed models

The models considered so far focus on modeling the mean response. We are switching to the framework of so-called *mixed models*. Mixed models are a powerful alternative, as they easily handle missing data (i.e., not all combinations of G and E explored). Mixed models can be used to model GE in terms of heterogeneity of variance and covariance. (Smith, A.B., Cullis, B.R., Thompson, R., 2005. The analysis of crop cultivar breeding and evaluation trials: an overview of current mixed model approaches. The Journal of Agricultural Science 143, 449.)

As a matter of fact, the correlations of the phenotypic values in the different sites is a quantity of interest. Let's explore the correlations among environments at first (Figure 9).

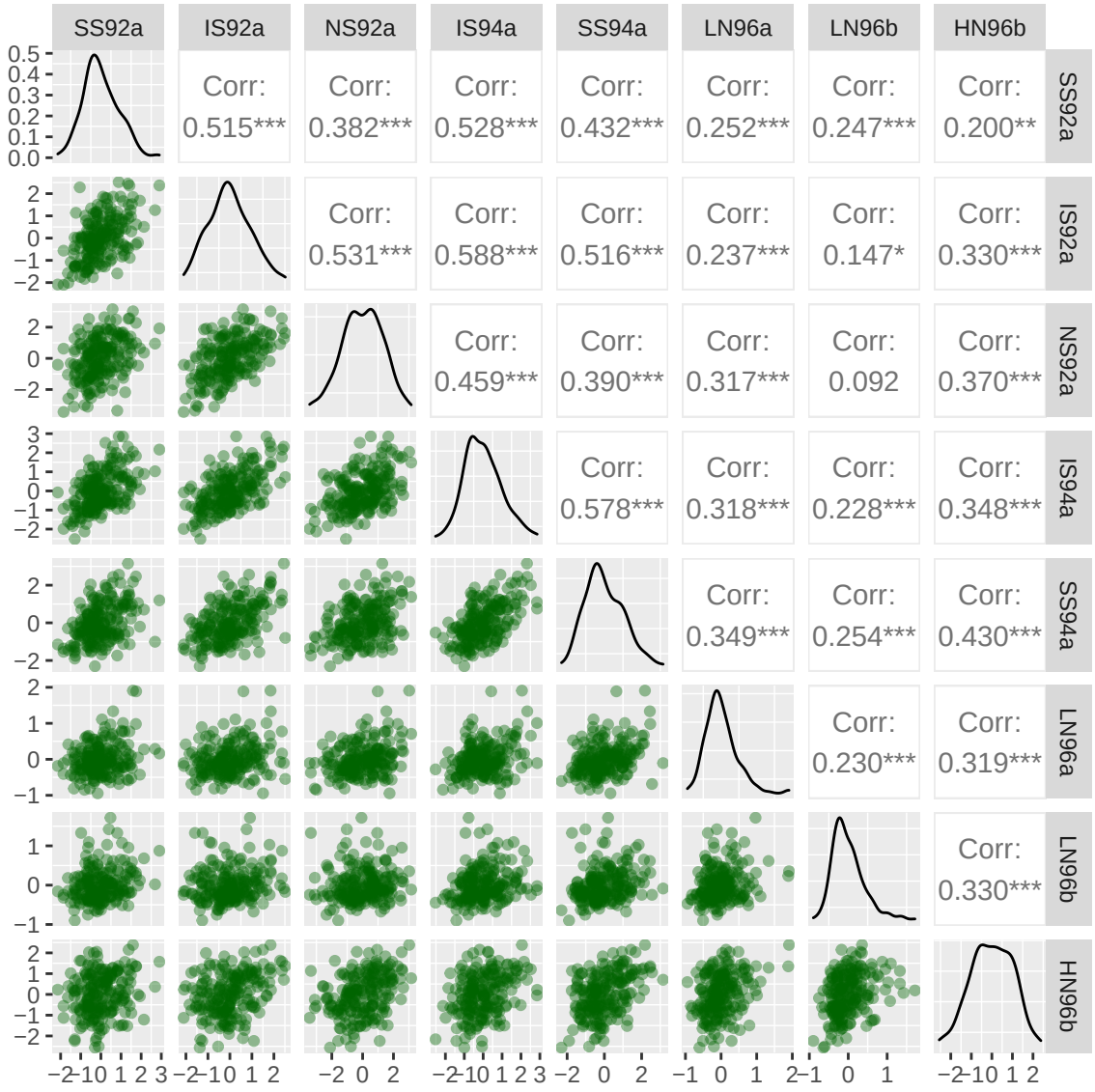


Figure 9: Phenotypic correlations for yield, for 211 maize lines assessed in eight environments.

As with all mixed models, key is the assumed covariance structure (the correlations being only the covariance scaled by the variances of each environment) among the genotypes and/or environments. As shown previously, the *mixed* linear model is,

$$\mathbf{Y} = \mathbf{Xb} + \mathbf{Zu} + \mathbf{e}$$

the solution is

$$\begin{pmatrix} {}^t\mathbf{X}\mathbf{R}^{-1}\mathbf{X} & {}^t\mathbf{X}\mathbf{R}^{-1}\mathbf{Z} \\ {}^t\mathbf{Z}\mathbf{R}^{-1}\mathbf{X} & {}^t\mathbf{Z}\mathbf{R}^{-1}\mathbf{Z} + \mathbf{G}^{-1} \end{pmatrix} \begin{pmatrix} \hat{b} \\ \hat{u} \end{pmatrix} = \begin{pmatrix} {}^t\mathbf{X}\mathbf{R}^{-1}\mathbf{y} \\ {}^t\mathbf{Z}\mathbf{R}^{-1}\mathbf{y} \end{pmatrix}$$

with :

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{e} \end{bmatrix} = \mathcal{N}(\mathbf{0}, \begin{bmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix})$$

and $\mathbf{R}$ and $\mathbf{G}$ being estimated by Maximum Likelihood (ML) or Restricted Maximum Likelihood (REML).

The random vector $\mathbf{u}$ contains all random effects pertaining to genotypes and environments, if any.

Most important to consider is the variance-covariance matrix of the response vector $\mathbf{y}$:

$$\text{Var}(\mathbf{y}) = \mathbf{Z}\mathbf{G}{}^t\mathbf{Z} + \mathbf{R}$$

5.1 Fitting LMM using `lme4` : the compound symetry model.

`lme4` allows to fit model with *crossed random effects* **but** does not allow to set particular structure to the $\mathbf{G}$ and $\mathbf{R}$ matrices. It is thus not possible to account for heterogeneous variances among environments in the residuals, or to account for the genetic relationships among genotypes for example.

Crossed random effects shall be understood as random effects *crossed with each others*, for example a MET where Genotypes and Environments are considered as random. Another example is a MET with several measurements of the Genotypes in each Environments (typically, values of a RCBD in each environment) where Genotypes and Blocks are considered as random and Environments as fixed, the GxE being also a random effect. Here the Block and Genotype effects are crossed with each other, and Genotype effects and GxE effects are also random crossed effects.

Let's fit a LMM with environments considered as fixed effects and genotypes as random effects. As such, this model do not really need to be fitted using `lme4` because there is no crossed random effects with each other, but it is way to understand the process. We will get a value for the genetic variance σ^2_g.

It is always usefull to compute the size of the different matrices of the design:

$$\mathbf{Y}_{(1688,1)} = \mathbf{X}_{(1688,8)}\mathbf{b}_{(8,1)} + \mathbf{Z}_{(1688,211)}\mathbf{u}_{(211,1)} + \mathbf{e}_{(1688,1)}$$

and $\mathbf{G}_{(211,211)}$, $\mathbf{R}_{(1688,1688)}$

```
library(lme4)
METmaizeTri <- METmaize[order(METmaize$Genotype, METmaize$Environ),
    ]
head(METmaizeTri, 12)
```

```
##      Genotype Environ Group     yield MINTF  RADG      Index
## 1478     G001   HN96b     2 4.3800000 22.55 22.79  0.1933292
## 212      G001   IS92a     2 4.4800000 10.70 18.93  1.2263940
## 634      G001   IS94a     2 2.2486667 11.38 28.73 -0.2411172
## 1056     G001   LN96a     1 0.9666667 15.72 18.19 -1.8157703
## 1267     G001   LN96b     1 0.4733333 22.58 22.94 -2.4429425
## 423      G001   NS92a     3 8.4000000 11.58 17.28  3.9525172
## 1        G001   SS92a     2 3.2893333  9.70 18.20 -0.5878288
## 845      G001   SS94a     2 2.9840000 11.48 28.14 -0.2845817
## 1479     G002   HN96b     2 2.7133333 22.55 22.79  0.1933292
## 213      G002   IS92a     2 4.8773333 10.70 18.93  1.2263940
## 635      G002   IS94a     2 4.0206667 11.38 28.73 -0.2411172
## 1057     G002   LN96a     1 0.5866667 15.72 18.19 -1.8157703
```

```
## fit the LMM:
lmm1 <- lmer(yield ~ Environ + (1 | Genotype), data = METmaizeTri)
lmm1
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: yield ~ Environ + (1 | Genotype)
##    Data: METmaizeTri
## REML criterion at convergence: 4165.546
## Random effects:
##  Groups   Name        Std.Dev.
##  Genotype (Intercept) 0.5445
##  Residual             0.7438
## Number of obs: 1688, groups:  Genotype, 211
## Fixed Effects:
##  (Intercept)  EnvironIS92a  EnvironIS94a  EnvironLN96a  EnvironLN96b
##       3.2345        1.0331       -0.4344       -2.0091       -2.6363
## EnvironNS92a  EnvironSS92a  EnvironSS94a
##       3.7592       -0.7812       -0.4779
```

```
VarCorr(lmm1)  ## get the variance components (as sd ....)
```

```
##  Groups   Name        Std.Dev.
##  Genotype (Intercept) 0.54451
##  Residual             0.74377
```

With this model, we are restricted to the situation where $\mathbf{G} = \sigma^2_g \mathbf{I} = 0.297 \cdot \mathbf{I}$ and $\mathbf{R} = \sigma^2_e \mathbf{I} = 0.554 \cdot \mathbf{I}$.

An important consequence of including genotypes as random is that *automatically* genetic covariances and correlations between performances in different environments *are imposed*, that depend on the structure of $\mathbf{G}$ and $\mathbf{R}$. Let's compute $\text{Var}(\mathbf{y})$ to see the consequences in that particular model :

```
Z <- lme4::getME(lmm1, "Z")
dim(Z)  ## as expected
```

```
## [1] 1688  211
```

```
Vy <- Z %*% (0.297 * diag(211)) %*% t(Z) + 0.554 * diag(1688)
Vy[1:12, 1:12]
```

```
## 12 x 12 sparse Matrix of class "dgCMatrix"
##    [[ suppressing 12 column names '1478', '212', '634' ... ]]
##
## 1478 0.851 0.297 0.297 0.297 0.297 0.297 0.297 0.297 .     .     .     .
## 212  0.297 0.851 0.297 0.297 0.297 0.297 0.297 0.297 .     .     .     .
## 634  0.297 0.297 0.851 0.297 0.297 0.297 0.297 0.297 .     .     .     .
## 1056 0.297 0.297 0.297 0.851 0.297 0.297 0.297 0.297 .     .     .     .
## 1267 0.297 0.297 0.297 0.297 0.851 0.297 0.297 0.297 .     .     .     .
## 423  0.297 0.297 0.297 0.297 0.297 0.851 0.297 0.297 .     .     .     .
## 1    0.297 0.297 0.297 0.297 0.297 0.297 0.851 0.297 .     .     .     .
## 845  0.297 0.297 0.297 0.297 0.297 0.297 0.297 0.851 .     .     .     .
## 1479 .     .     .     .     .     .     .     .     0.851 0.297 0.297 0.297
## 213  .     .     .     .     .     .     .     .     0.297 0.851 0.297 0.297
## 635  .     .     .     .     .     .     .     .     0.297 0.297 0.851 0.297
## 1057 .     .     .     .     .     .     .     .     0.297 0.297 0.297 0.851
```

Following $P = G + E$, the total variance for individual phenotypic observations in a particular environment j, σ^2_j, is the sum of two sources of variation: $\sigma^2_j = \sigma^2_G + \sigma^2_e$. From the model, the covariance between observations for a particular genotype in environments j and j' is $\sigma_{jj'} = \sigma^2_G$. For observations on different genotypes $\sigma_{jj'} = 0$.

So, with this model, similarities (or covariation, and therefore correlation) between observations made on the same genotype in different environments are assumed to be identical and positive, but covariation between observations on different genotypes (regardless whether the observation is done in the same or in different environments) is assumed to be zero. This model is referred as the *compound symmetry model*.

The model imposes a constant correlation between environments, with the correlation between any pair of environments j and j', being equal to:

$$\mathrm{r}_{(j,j')} = \frac{\sigma_{jj'}}{\sqrt{\sigma_j^2}\sqrt{\sigma_{j'}^2}} = \frac{\sigma_G^2}{\sqrt{\sigma_G^2+\sigma_e^2}\sqrt{\sigma_G^2+\sigma_e^2}} = \frac{\sigma_G^2}{\sigma_G^2+\sigma_e^2} = H^2$$

This is the genetic correlation between any two environments.

5.2 Fitting LMM with heterogeneous environmental variances

The *compound symmetry model* assumes a constant genetic variance and correlation between pairs of environments. For METs, the assumption of constant genetic variance and genetic correlation across environments is unrealistic. In the presence of GE, a more realistic model would allow the total genetic variance to change from environment to environment, which will in turn, cause heterogeneous genetic correlations between environments.

Because we are considering the environment as a fixed effect here, we will model the heterogeneity of environments in the **R** matrix (see the example of –fixed– GLS with heterogeneous variances for environments). Because `lme4` does not allow to impose structures on the **R** matrix, we will use the `nlme::lme()` function.

```
# baseline : compound symmetry model with nlme::lme
lmm1b <- lme(yield ~ Environ, data = METmaizeTri, random = ~1 |
    Genotype)
lmm1b
```

```
## Linear mixed-effects model fit by REML
##   Data: METmaizeTri
##   Log-restricted-likelihood: -2082.773
##   Fixed: yield ~ Environ
##  (Intercept) EnvironIS92a EnvironIS94a EnvironLN96a EnvironLN96b EnvironNS92a
##    3.2345340    1.0330648   -0.4344464   -2.0090995   -2.6362717    3.7591880
## EnvironSS92a EnvironSS94a
##   -0.7811580   -0.4779109
##
## Random effects:
##  Formula: ~1 | Genotype
##         (Intercept)  Residual
## StdDev:   0.5445051 0.7437723
##
## Number of Observations: 1688
## Number of Groups: 211
```

```
# model with heterogeneous variances among environ.
lmm2 <- lme(yield ~ Environ, data = METmaizeTri, random = ~1 |
    Genotype, weights = varIdent(form = ~1 | Environ)  ## for R matrix
)
lmm2
```

```
## Linear mixed-effects model fit by REML
##   Data: METmaizeTri
##   Log-restricted-likelihood: -1962.991
##   Fixed: yield ~ Environ
##  (Intercept) EnvironIS92a EnvironIS94a EnvironLN96a EnvironLN96b EnvironNS92a
##    3.2345340    1.0330648   -0.4344464   -2.0090995   -2.6362717    3.7591880
## EnvironSS92a EnvironSS94a
##   -0.7811580   -0.4779109
##
## Random effects:
##  Formula: ~1 | Genotype
##         (Intercept)  Residual
## StdDev:   0.3534791 0.8722701
##
## Variance function:
##  Structure: Different standard deviations per stratum
##  Formula: ~1 | Environ
```

```
##  Parameter estimates:
##     HN96b     IS92a     IS94a     LN96a     LN96b     NS92a     SS92a     SS94a
## 1.0000000 0.9535480 0.9395624 0.4204533 0.4472504 1.3557808 0.8508820 0.9620678
## Number of Observations: 1688
## Number of Groups: 211
```

```
VarCorr(lmm2)
```

```
## Genotype = pdLogChol(1)
##             Variance  StdDev
## (Intercept) 0.1249475 0.3534791
## Residual    0.7608552 0.8722701
```

```
lmm2$modelStruct$varStruct
```

```
## Variance function structure of class varIdent representing
##     HN96b     IS92a     IS94a     LN96a     LN96b     NS92a     SS92a     SS94a
## 1.0000000 0.9535480 0.9395624 0.4204533 0.4472504 1.3557808 0.8508820 0.9620678
```

```
anova(lmm1b, lmm2)
```

```
##       Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## lmm1b     1 10 4185.546 4239.812 -2082.773
## lmm2      2 17 3959.981 4052.232 -1962.991 1 vs 2 239.5651  <.0001
```

Model `lmm2` is significantly better, even it has more parameters. Let's compute $\text{Var}(\mathbf{y})$:

```
G <- 0.125 * diag(211)
# to compute R
D <- lmm2$sigma^2 * diag(c(1, 0.954, 0.94, 0.42, 0.447, 1.356,
    0.851, 0.962))
dimnames(D) <- list(x = c("HN96b", "IS92a", "IS94a", "LN96a",
    "LN96b", "NS92a", "SS92a", "SS94a"), y = c("HN96b", "IS92a",
    "IS94a", "LN96a", "LN96b", "NS92a", "SS92a", "SS94a"))
D
```

```
##        y
## x           HN96b     IS92a     IS94a     LN96a     LN96b   NS92a     SS92a
##   HN96b 0.7608552 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##   IS92a 0.0000000 0.7258559 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##   IS94a 0.0000000 0.0000000 0.7152039 0.0000000 0.0000000 0.00000 0.0000000
##   LN96a 0.0000000 0.0000000 0.0000000 0.3195592 0.0000000 0.00000 0.0000000
##   LN96b 0.0000000 0.0000000 0.0000000 0.0000000 0.3401023 0.00000 0.0000000
##   NS92a 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 1.03172 0.0000000
##   SS92a 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.6474878
##   SS94a 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##        y
## x           SS94a
##   HN96b 0.0000000
##   IS92a 0.0000000
##   IS94a 0.0000000
##   LN96a 0.0000000
##   LN96b 0.0000000
##   NS92a 0.0000000
##   SS92a 0.0000000
##   SS94a 0.7319427
```

```
R <- kronecker(diag(211), D)
```

The Kronecker product is written $\mathbf{I}_{211} \otimes \mathbf{D}$.

```
dim(R)
```

```
## [1] 1688 1688
```

```
R[1:12, 1:12]
```

```
##            [,1]      [,2]      [,3]      [,4]      [,5]     [,6]      [,7]
```

```
##  [1,] 0.7608552 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##  [2,] 0.0000000 0.7258559 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##  [3,] 0.0000000 0.0000000 0.7152039 0.0000000 0.0000000 0.00000 0.0000000
##  [4,] 0.0000000 0.0000000 0.0000000 0.3195592 0.0000000 0.00000 0.0000000
##  [5,] 0.0000000 0.0000000 0.0000000 0.0000000 0.3401023 0.00000 0.0000000
##  [6,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 1.03172 0.0000000
##  [7,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.6474878
##  [8,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##  [9,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
## [10,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
## [11,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
## [12,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##            [,8]      [,9]     [,10]     [,11]     [,12]
##  [1,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [2,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [3,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [4,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [5,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [6,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [7,] 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000
##  [8,] 0.7319427 0.0000000 0.0000000 0.0000000 0.0000000
##  [9,] 0.0000000 0.7608552 0.0000000 0.0000000 0.0000000
## [10,] 0.0000000 0.0000000 0.7258559 0.0000000 0.0000000
## [11,] 0.0000000 0.0000000 0.0000000 0.7152039 0.0000000
## [12,] 0.0000000 0.0000000 0.0000000 0.0000000 0.3195592
```

```
Vy <- Z %*% G %*% t(Z) + R
Vy[1:12, 1:12]
```

```
##           1478       212       634      1056      1267     423         1
## 1478 0.8858552 0.1250000 0.1250000 0.1250000 0.1250000 0.12500 0.1250000
## 212  0.1250000 0.8508559 0.1250000 0.1250000 0.1250000 0.12500 0.1250000
## 634  0.1250000 0.1250000 0.8402039 0.1250000 0.1250000 0.12500 0.1250000
## 1056 0.1250000 0.1250000 0.1250000 0.4445592 0.1250000 0.12500 0.1250000
## 1267 0.1250000 0.1250000 0.1250000 0.1250000 0.4651023 0.12500 0.1250000
## 423  0.1250000 0.1250000 0.1250000 0.1250000 0.1250000 1.15672 0.1250000
## 1    0.1250000 0.1250000 0.1250000 0.1250000 0.1250000 0.12500 0.7724878
## 845  0.1250000 0.1250000 0.1250000 0.1250000 0.1250000 0.12500 0.1250000
## 1479 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
## 213  0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
## 635  0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
## 1057 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.00000 0.0000000
##            845      1479       213       635      1057
## 1478 0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 212  0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 634  0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 1056 0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 1267 0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 423  0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 1    0.1250000 0.0000000 0.0000000 0.0000000 0.0000000
## 845  0.8569427 0.0000000 0.0000000 0.0000000 0.0000000
## 1479 0.0000000 0.8858552 0.1250000 0.1250000 0.1250000
## 213  0.0000000 0.1250000 0.8508559 0.1250000 0.1250000
## 635  0.0000000 0.1250000 0.1250000 0.8402039 0.1250000
## 1057 0.0000000 0.1250000 0.1250000 0.1250000 0.4445592
```

There is still a single genetic variance component for genotypes, and therefore, a *constant genetic covariance* between environments. However, the variance for the term ϵ_{ij} that includes GE and error, is assumed to depend on the environment. Instead of two variance components, there are now nine, one corresponding to the variance component for genotypes ($\sigma^2_g = 0.125$), and eight corresponding to a form of GE for each of the eight environments. The heterogeneity of variance for ϵ_{ij} reflects that in some environments there is a larger variation (e.g. in environment NS92a, which is the high-yielding) than in other environments (e.g., in environments LN96a and LN96b which are low-yielding).

The heterogeneity of variance leads to heterogeneous genetic correlations between environments. For example, the correlation between environments NS92a and SS92a is:

$$r_{6,7} = \frac{\sigma_G^2}{\sqrt{\sigma_G^2 + \sigma_6^2}\sqrt{\sigma_G^2 + \sigma_7^2}} = \frac{0.125}{\sqrt{0.125 + 1.03}\sqrt{0.125 + 0.647}} = 0.132$$

and between environments 1 and 2 is:

$$r_{1,2} = \frac{\sigma_G^2}{\sqrt{\sigma_G^2 + \sigma_1^2}\sqrt{\sigma_G^2 + \sigma_2^2}} = \frac{0.125}{\sqrt{0.125 + 0.761}\sqrt{0.125 + 0.726}} = 0.144$$

In conclusion, this model accommodates heterogeneity of variance between environments and, with it, allows for heterogeneous correlations between environments, which can be desirable when analyzing environments that strongly differ (e.g., with strong stress and without stress).

5.3 Fitting a stability variance model

If we have data from RCBD in each environments, a good candidate model for data analyses is the following linear model:

$$y_{ijk} = \mu + \gamma_{kj} + g_i + e_j + ge_{ij} + \epsilon_{ijk}$$

where y_{ijk} is yield (or other trait) for the k-th block, i-th genotype and j-th environment, μ is the intercept, γ_{kj} is the effect of the k-th block in the j-th environment, g_i is the effect of the i-th genotype, e_j is the effect of the j-th environment, ge_{ij} is the interaction effect of the i-th genotype and j-th environment, while ϵ_{ijk} is the residual random term, with variance σ^2.

The block effect, the environment effect and the 'genotype x environment' interaction are usually regarded as random. Therefore, they are assumed as normally distributed, with means equal to 0 and variances respectively equal to σ_γ^2, σ_e^2 and σ_{ge}^2.

Let's concentrate on σ_{ge}^2. It is clear that this value is a measure of instability: if it is high, genotypes may respond differently to different environments. In this way, each genotype can be favored in some specific environments and disfavored in some others. Shukla (1974) has suggested that we should allow σ_{ge}^2 assume a different value for each genotype and use these components as a measure of stability (stability variances). According to Shukla, a genotype is considered stable when its stability variance is lower than σ^2. That stability variances can be obtained within the mixed model framework.

As we have to model the variance-covariance of random effects, we need to use the `lme` function in the `nlme` package. The problem is that random effects are crossed and they are not easily coded with this package, but is is *still* possible.

▶ The main advantage of `nlme` relative to `lme4` is a user interface for fitting models with structure in the residuals (various forms of heteroscedasticity and autocorrelation) and in the random-effects covariance matrices (e.g., compound symmetric models).